

# Software Projektkalkulation

## „Zur Schätzung der Aufwände für Entwicklungs- Test-, Reengineering- und Wartungsprojekte“

Eine Ringvorlesung für die TU Dresden

am 15.12.2014

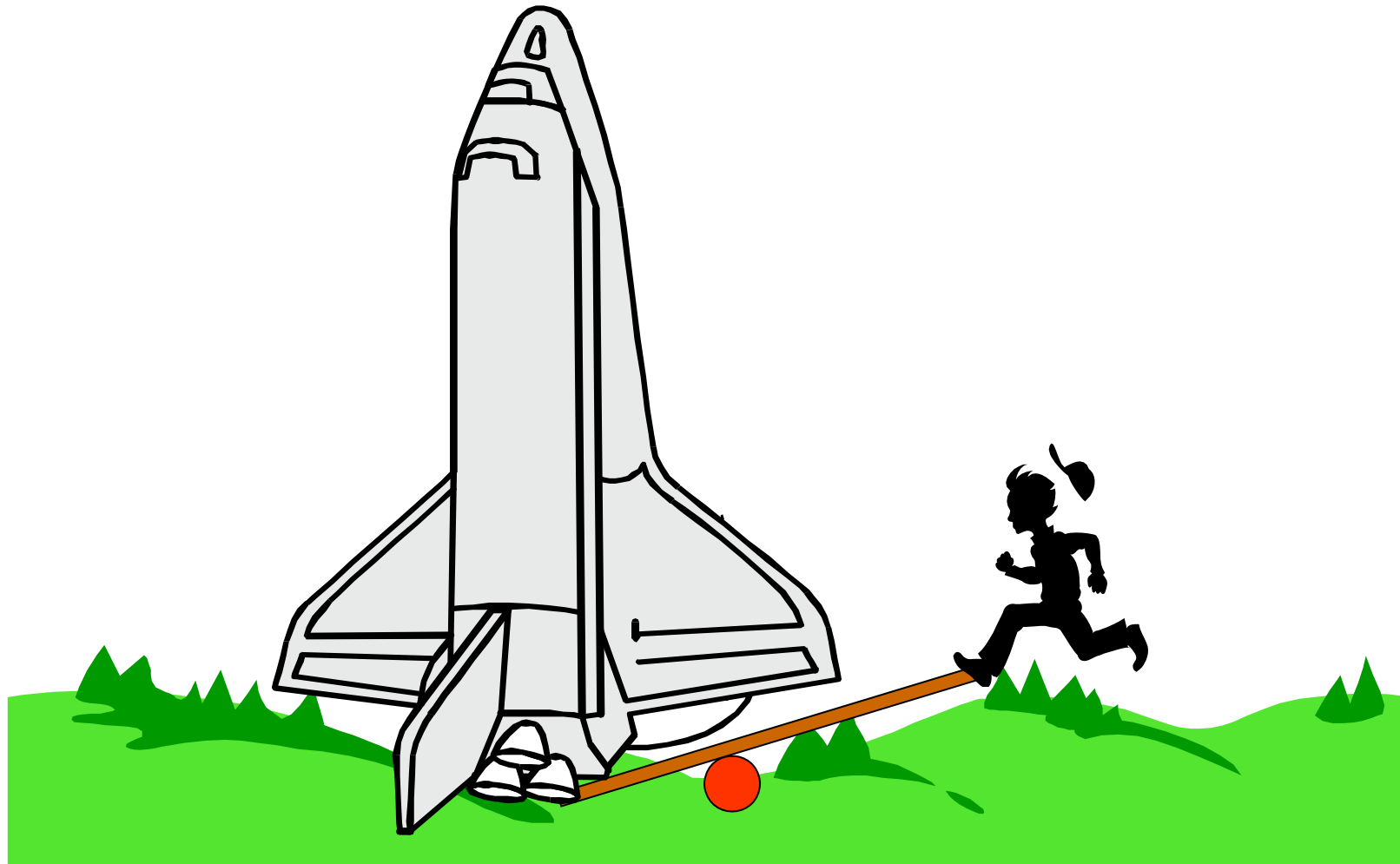
von Harry M. Sneed

Softwaremigrationsberater für das Burgenländische  
Landesamt für Datenverarbeitung und Lehrbeauftragter  
für Software Engineering an den Universitäten  
Regensburg, Szeged & Dresden sowie an den  
Fachhochschulen Wien & Hagenberg

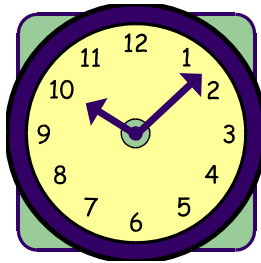
# Einführung in die Schätzproblematik

- Das Verhältnis vom Zweck und Mittel
- Das Wesen eines Projektes
- Das Teufelsquadrat
- Softwaregröße
- Softwarekomplexität
- Software Qualität
- Cone of Uncertainty

**Passen die Mittel zum Zweck?**



# Ein Software Projekt



Die Uhr  
läuft

Projektziel  
(6. Stunde?)

Felsen

Hier hinten  
wartet das  
nächste  
Ziel

Gletscher

3. Meilenstein  
(4. Stunde)

Steinschlag

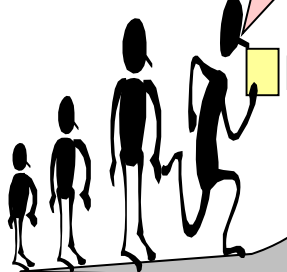
2. Meilenstein  
(3. Stunde)

Mir  
nach!

Plan

Berg der tausend Gefahren

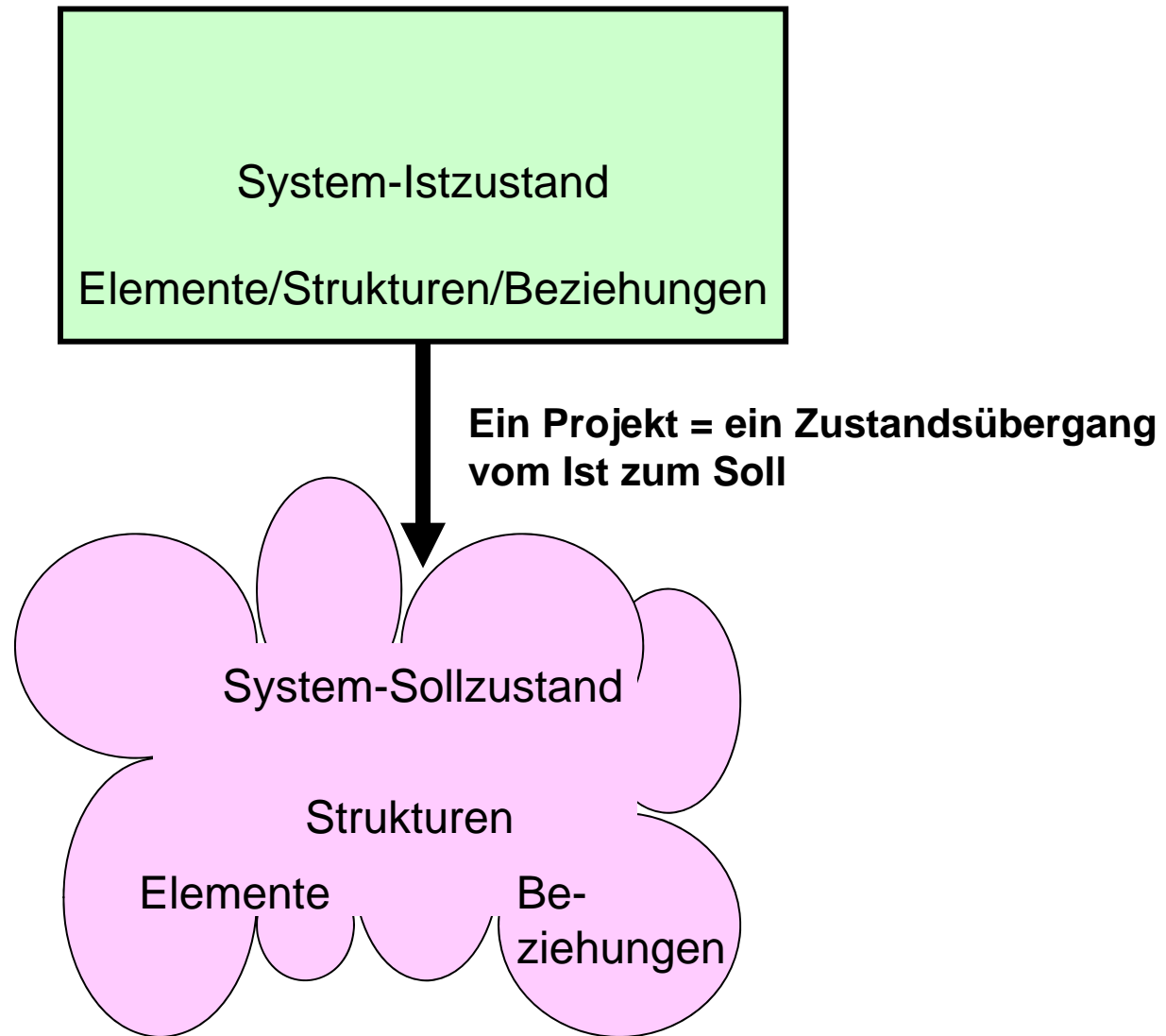
1. Meilenstein  
(1. Stunde)



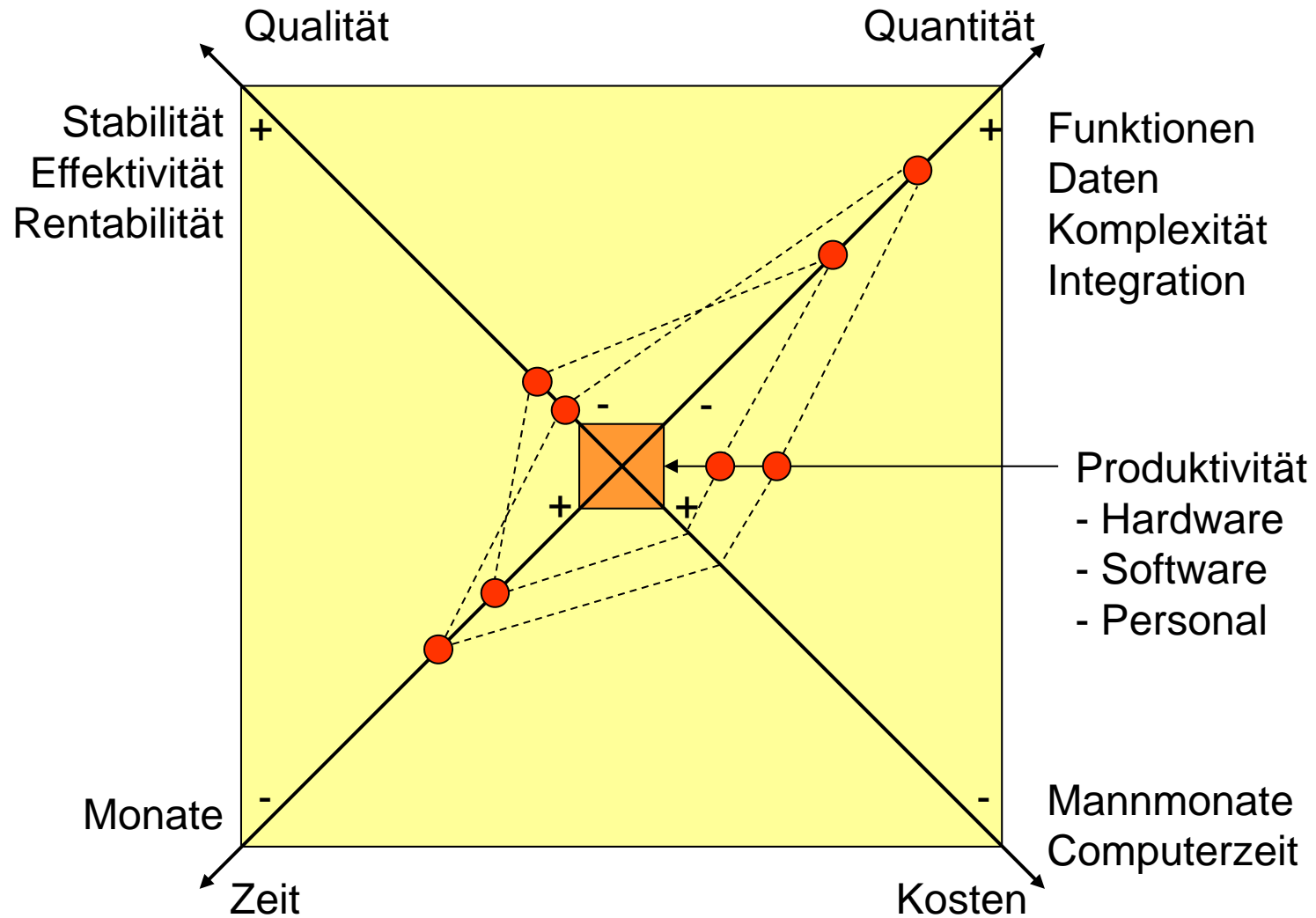
Projektteam

heißt ein hohes Ziel erreichen mit beschränkter Zeit und Kapazität

# Ein Projekt als Zustandsübergang



# Das Teufelsquadrat



## Bilanzierung von Quantität, Qualität, Zeit und Kosten

### Vorwärtskalkulation

- **Quantity = 200 FPs**
- **Quality = 1.2 or 20% higher**
- **Productivity = 20 FPs per PM**

$$\text{Effort} = 200 \times 1.2 = 240 / 20 = 12 \text{ PMs}$$

$$\text{Persons} = 2$$

$$\text{Time} = 12 / (2^{**} 0.8) = 6.9 \text{ months}$$

$$\text{Output} = 200 / (6.9 \times 2) = 14.5 \text{ FP per PM}$$

$$\text{Persons} = 3$$

$$\text{Time} = 12 / (3^{**} 0.9) = 5.0 \text{ months}$$

$$\text{Output} = 200 / (5 \times 3) = 13 \text{ FPs per PM}$$

### Rückwärtskalkulation

- **Quality = 1.2 or 20% higher**
- **Effort = 12 PMs**
- **Productivity = 20 FPs per PM**

$$\text{Quantity} = 12 \times 20 \times (2 - 1.2) = 192$$

$$\text{Quality} = 12 / (200 / 20) = 1.2 \text{ or } 20\% \text{ higher}$$

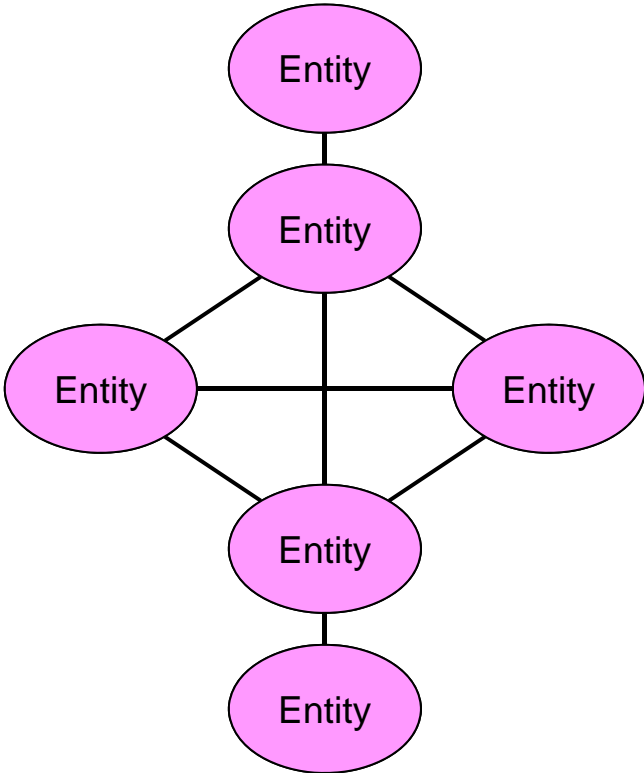
**If Quality = 0.8 or 20% lower then**

$$\text{Effort} = 200 \times 0.8 = 160 / 20 = 8 \text{ PMs}$$

$$\text{Time} = 8 / (2^{**} 0.8) = 5 \text{ months}$$

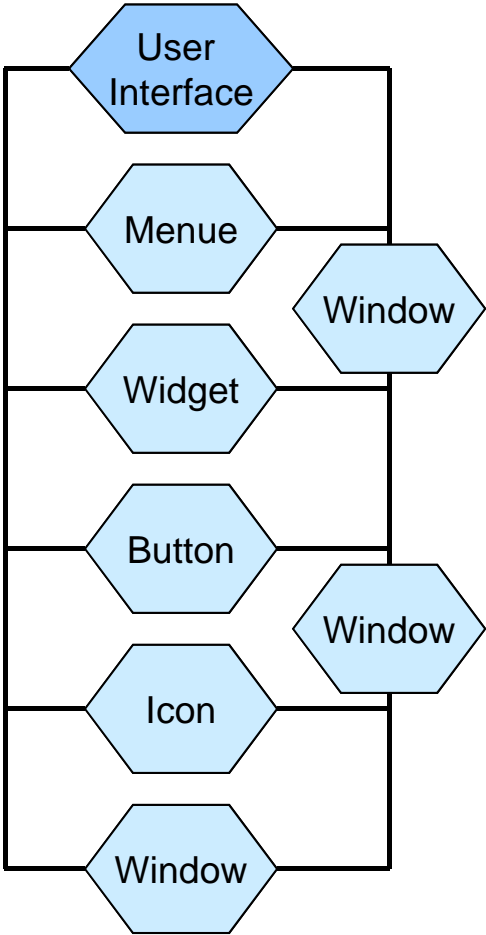
# Softwarekomplexitätsmessung

**Data Complexity**



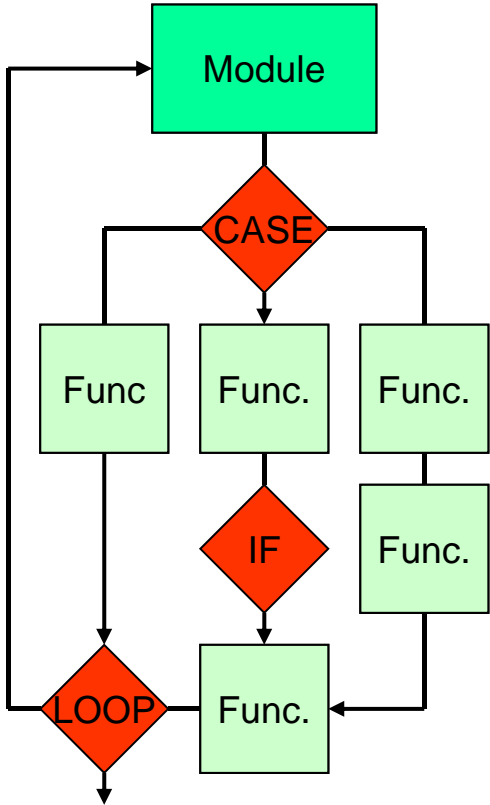
1 - Entities  
Relations

**Communication Complexity**



1 - interface  
Widgets

**Process Complexity**



1 - Nodes  
Edges



## Wirkung der Komplexität auf die Schätzung

- **Complexity = 1 – (Entities / Relationships)**
- **Number of Modules/Classes = 40**
- **Number of Relationships between Modules = 100**
- **Complexity of Software Architecture = 1 – (40 / 100) = 0.6**
- **Adjusting System Size by Complexity**
- **System Size = 100K Statements**
- **Complexity Factor = Complexity / Median Complexity**
- **Complexity Factor = 0.6 / 0.5 = 1.2**
- **Adjusted System Size = 100K x 1.2 = 120K Statements**
- **Cost of Migration = Statements x €2,-**
- **Cost of Migration based on raw size = 100K x €2,- = €200.000**
- **Cost of Migration based on adjusted size = 120K x €2,- = €240.000**

## Softwarequalitätsmessung

Characteristic	Fulfillment Degree	Weight	Factor
Reliability	0.98	22	1.1
Efficiency	0.90	16	1.1
Usability	0.85	14	1.0
Security	0.99	24	1.2
Portability	0.75	8	0.9
Testability	0.90	10	1.1
Maintainability	0.60	4	0.8

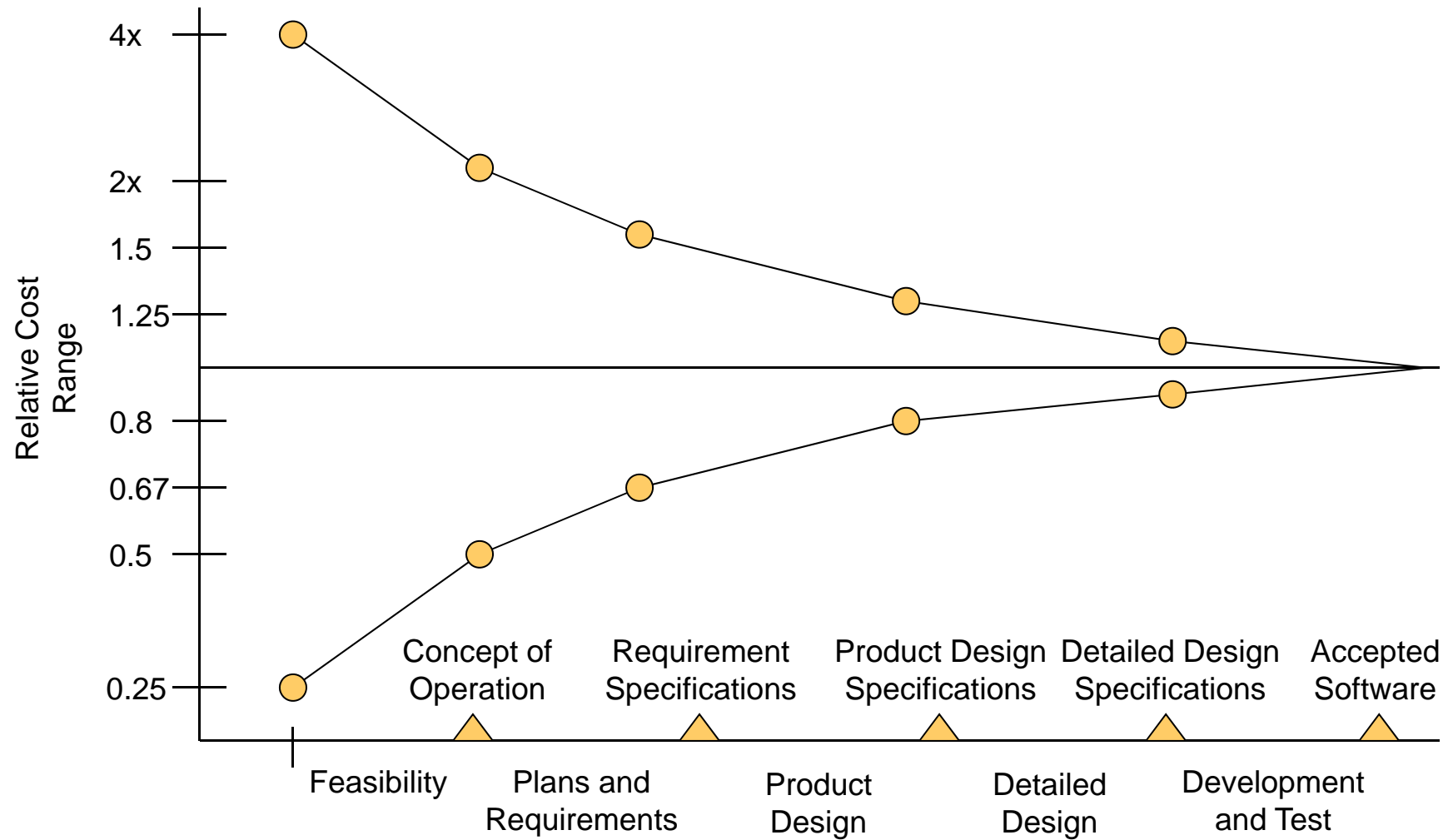
**Overall Quality Factor = 1.15**

## Wirkung der Qualität auf die Schätzung

- **Quality = Product of all Quality Factors**
- **Quality Factor = Reliability \* Efficiency \* Usability \* Security  
\* Portability \* Testability \* Maintainability**
- **Quality Factor = 1.1 \* 1.1 \* 1.0 \* 1.2 \* 0.9 \* 1.1 \* 0.8 = 1.15**
- **System Size = 100K Statements**
- **Complexity Factor = Complexity / Median Quality**
- **Complexity Factor = 0.6 / 0.5 = 1.2**
- **Complexity Adjusted System Size = 100K x 1.2 = 120K Statements**
- **Quality Adjusted System Size = 120K x 1.15 = 138K**
- **Cost of Migration = Statements x €2,-**
- **Cost of Migration based on raw size = 100K x €2,- = €200.000**
- **Cost of Migration based on complexity = 120K x €2,- = €240.000**
- **Cost of Migration based on adjusted size = 138K x €2,- = €276.000**

# Cone of Uncertainty

Eine Schätzung muss eine Bandbreite möglicher Kosten vorgeben



# Softwareproduktivität

- Produktivitätsmessung
- Produktivitätsunterschiede
- Produktivitätsanalysen
- Der unbekannte Faktor

# Software Produktivitätsmaße

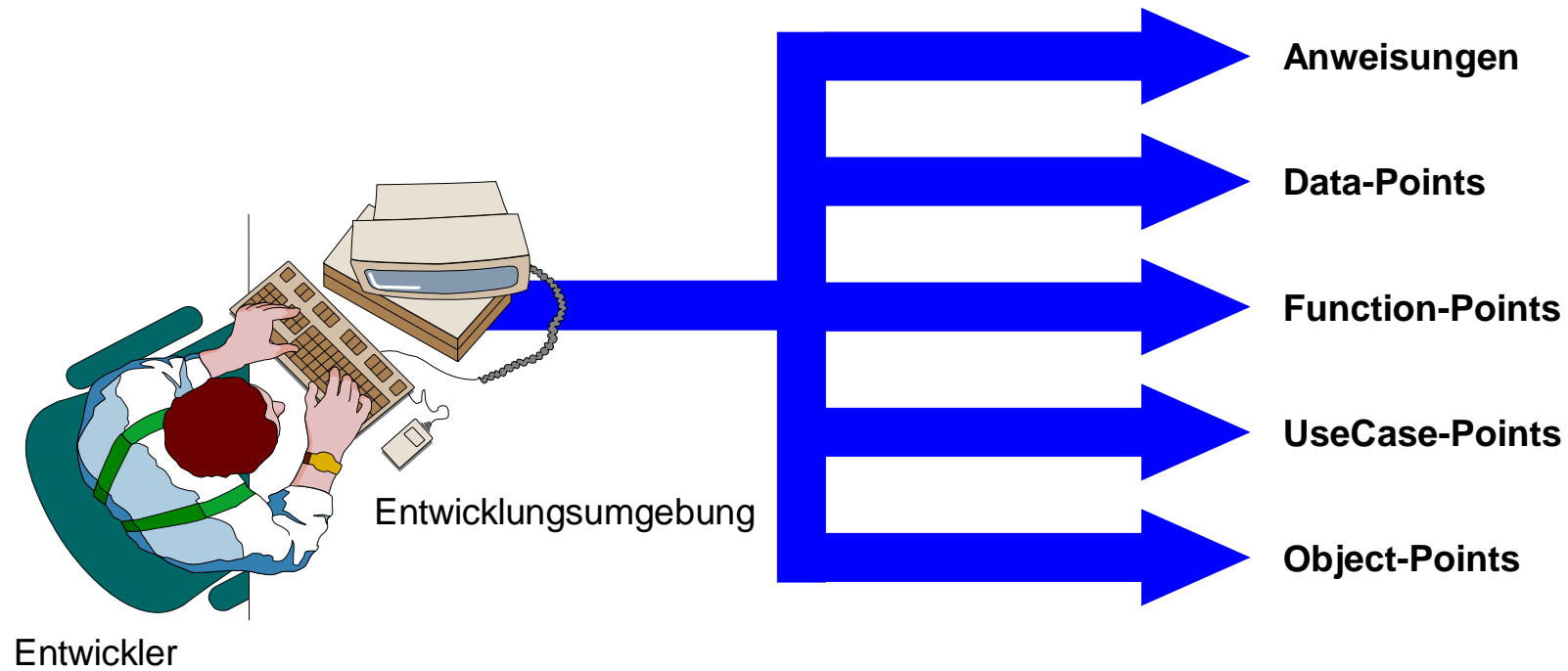


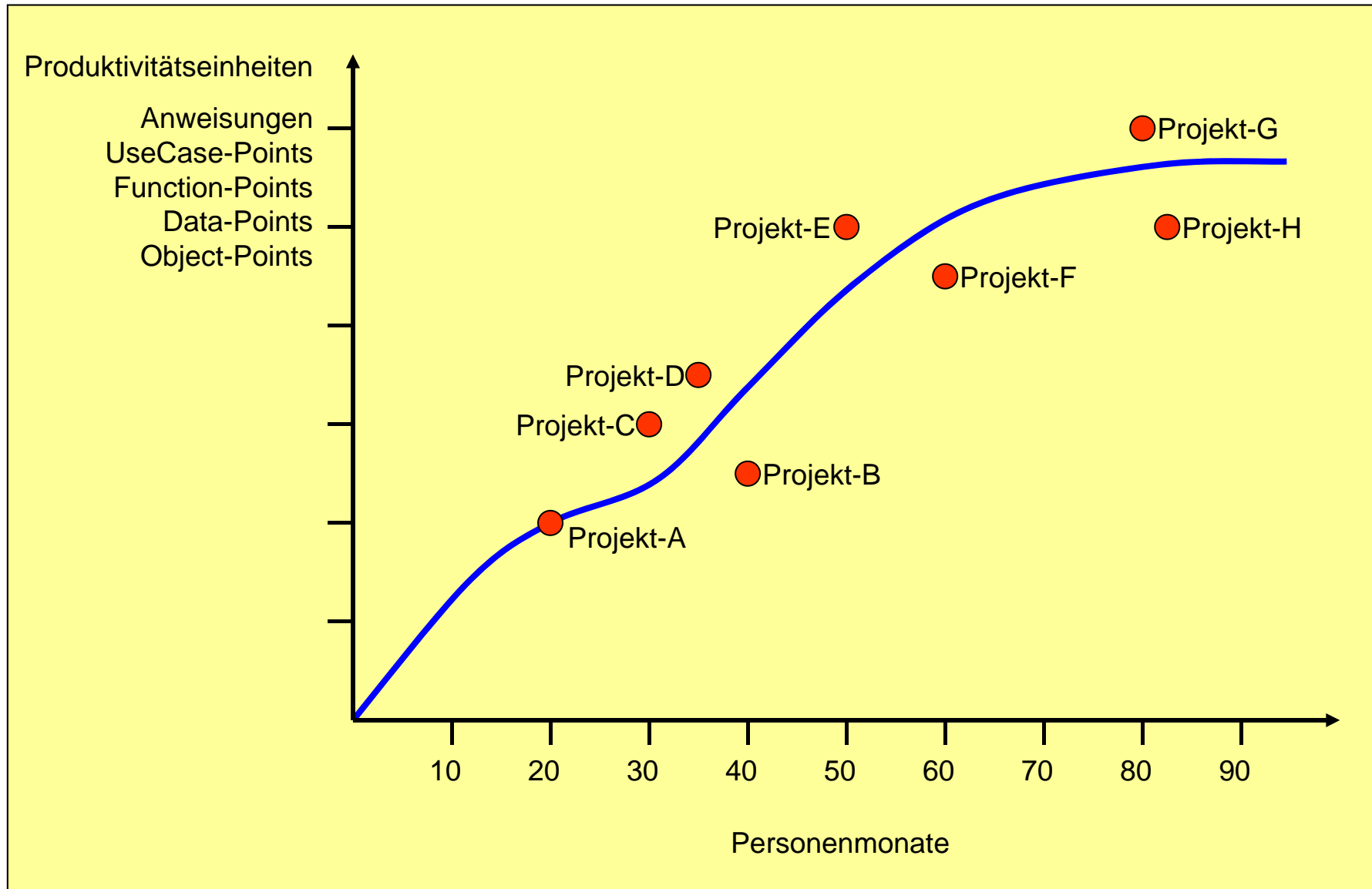
Abb. 2.1: Alternative Produktivitätseinheiten

## Produktivitätsmessung im GEOS Projekt

DATE: 27.04.2001							FIRM: SDS	
SYSTEM: GEOS		EFFORT IN PERSONDAYS					PAGE: 06	
Type	1996	1997	1998	1999	2000	Total	Percent	
SPEC	4,169	9,238	4,228	4,667	3,935	26,237	22 %	
CODE	5,591	8,780	6,797	6,007	6,800	33,975	28 %	
TEST	4,563	4,071	6,973	8,169	5,956	29,732	24 %	
DOCU	1,540	1,213	2,169	1,593	1,074	7,589	6 %	
MAIN	1,284	0	2,114	7,289	14,030	24,717	20 %	
ALL	17,147	23,302	22,281	27,725	31,795	122,250	100 %	

## Berechnung einer Produktivitätskurve

Produktivität wird anhand der Erfahrung bisheriger Projekte ermittelt





## Erstellung einer Produktivitätstabelle

	Object-Points	Personenmonate	Kosten (Euro)
	100	1,2	12.000
	200	2,5	25.000
	300	3,7	27.000
	400	5,0	50.000
	500	6,2	62.000
	600	7,5	75.000
	700	8,7	87.000
	800	10,0	100.000
	900	11,2	112.000
	1000	12,5	125.00
	1100	14,0	140.000
	1200	15,5	155.000
	1300	17,0	170.000
	1400	18,5	185.000
	1500	20,0	200.000
	1600	21,5	215.000
	1700	23,0	230.000
	1800	24,5	245.000
	1900	26,0	260.000
	2000	27,5	275.000

840  
Object-Points

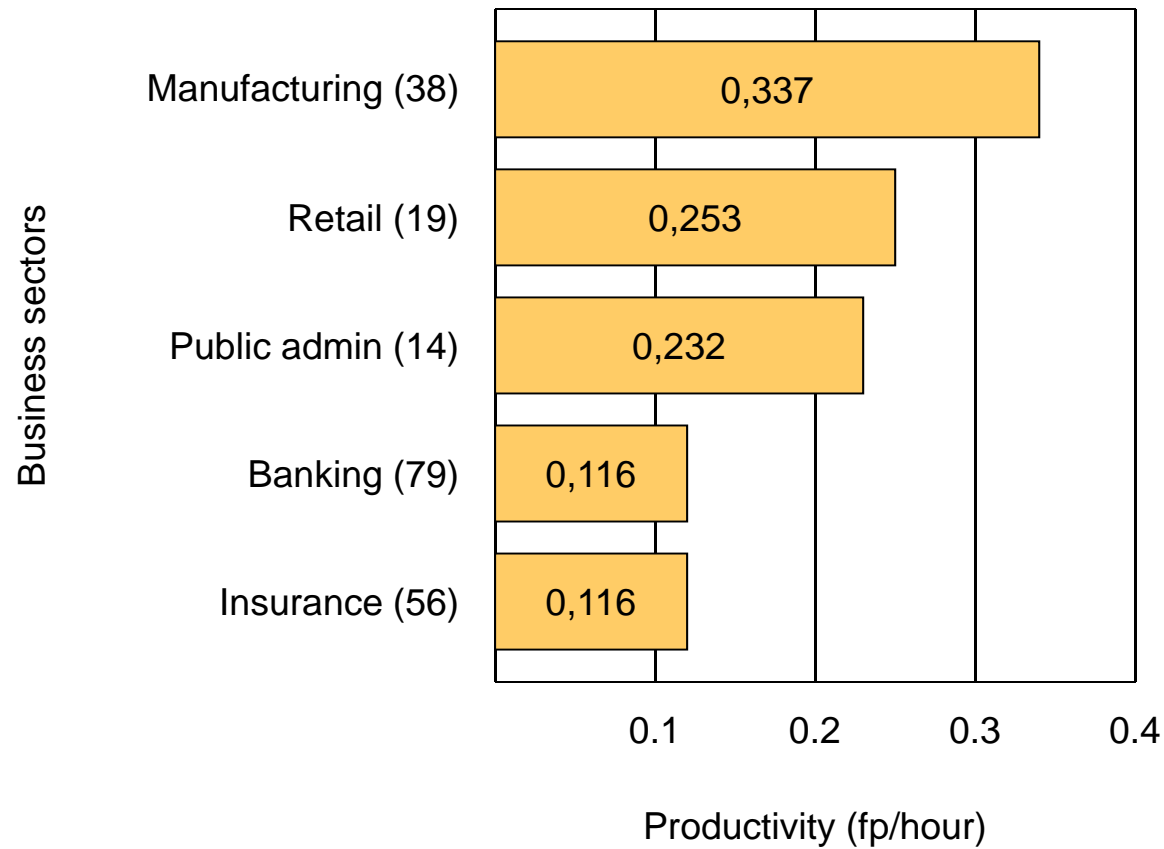
→

105,000  
Euros

←

# Produktivitätsunterschiede in Europa

( ) = Anzahl Projekte untersucht



# Produktivitätsanalyse

DATE: 27.04.2001

FIRM: SDS

SYSTEM: DRIVE

EFFORT IN PERSONDAYS

PAGE: 06

Type	1997	1998	1999	2000	Total	Percent
SPEC	1,006	1,104	3,785	3,836	9,731	28 %
CODE	905	3,617	3,116	4,115	11,753	33 %
TEST	276	1,060	1,262	3,196	5,794	17 %
DOCU	9	840	660	640	2,149	6 %
MAIN	0	1,398	2,489	1,809	5,696	16 %
ALL	2,196	8,019	11,312	13,596	35,123	100 %

## SOFTMESS PRODUCTIVITY REPORT

DATE: 27.04.2001

FIRM: SDS

SYSTEM: DRIVE

PRODUCTIVITY RATES

PAGE: 07

(Person Days per Unit)

ohne Wartung

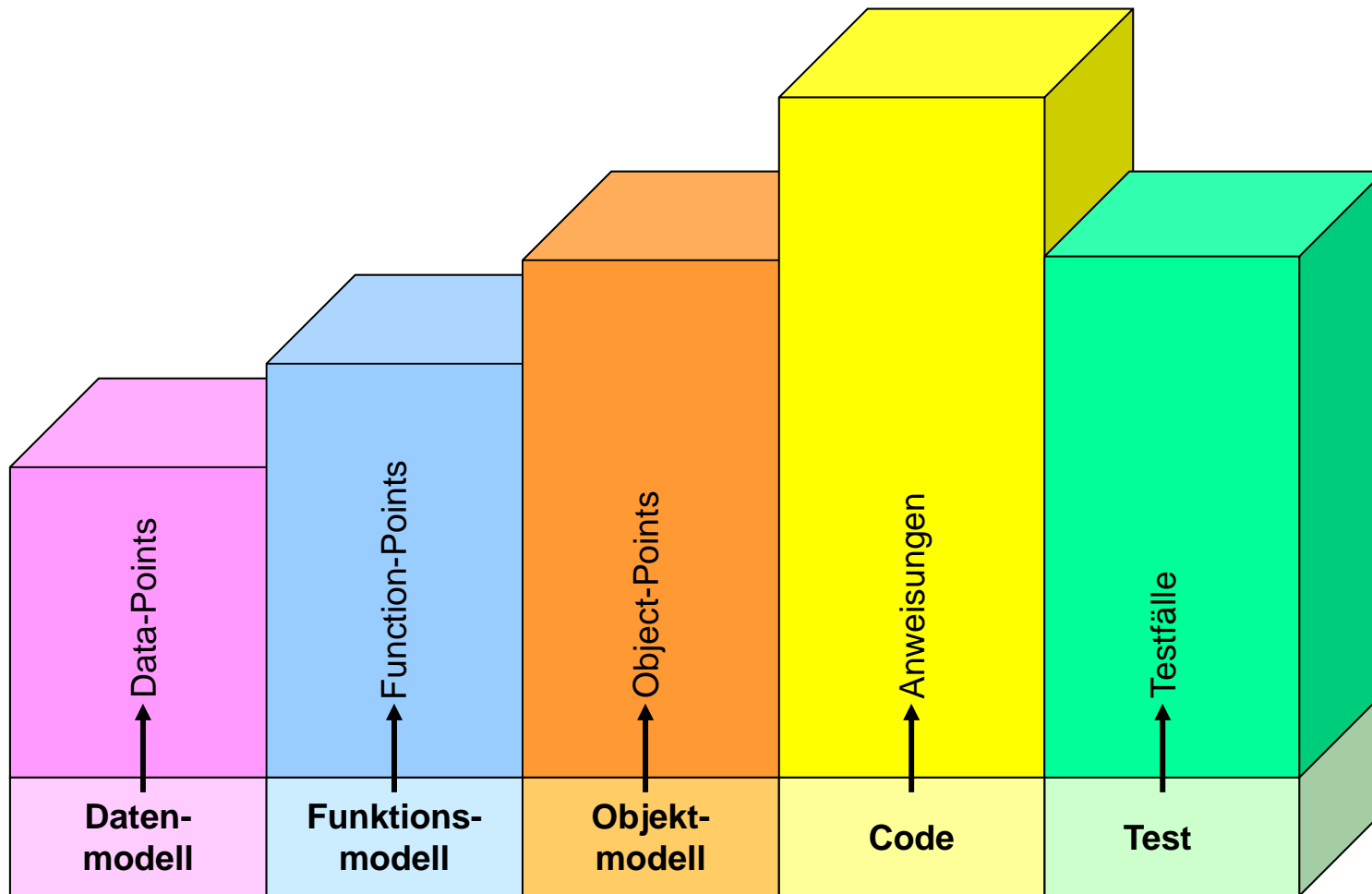
Type	Members	Lines	Funcs	Obj-Pts	Dat-Pts	Func-Pts
SPEC	1.94	0.02	1.98	0.05	0.06	0.12
CODE	2.86	0.02	0.51	0.06	0.06	0.30
TEST	1.41	0.01	0.25	0.03	0.03	0.15
ALL	3.22	0.07	1.29	0.15	0.15	0.76

Productivity = 14 per PD, 0.8 per PD 7 per PD 7 per PD 1.3 per PD

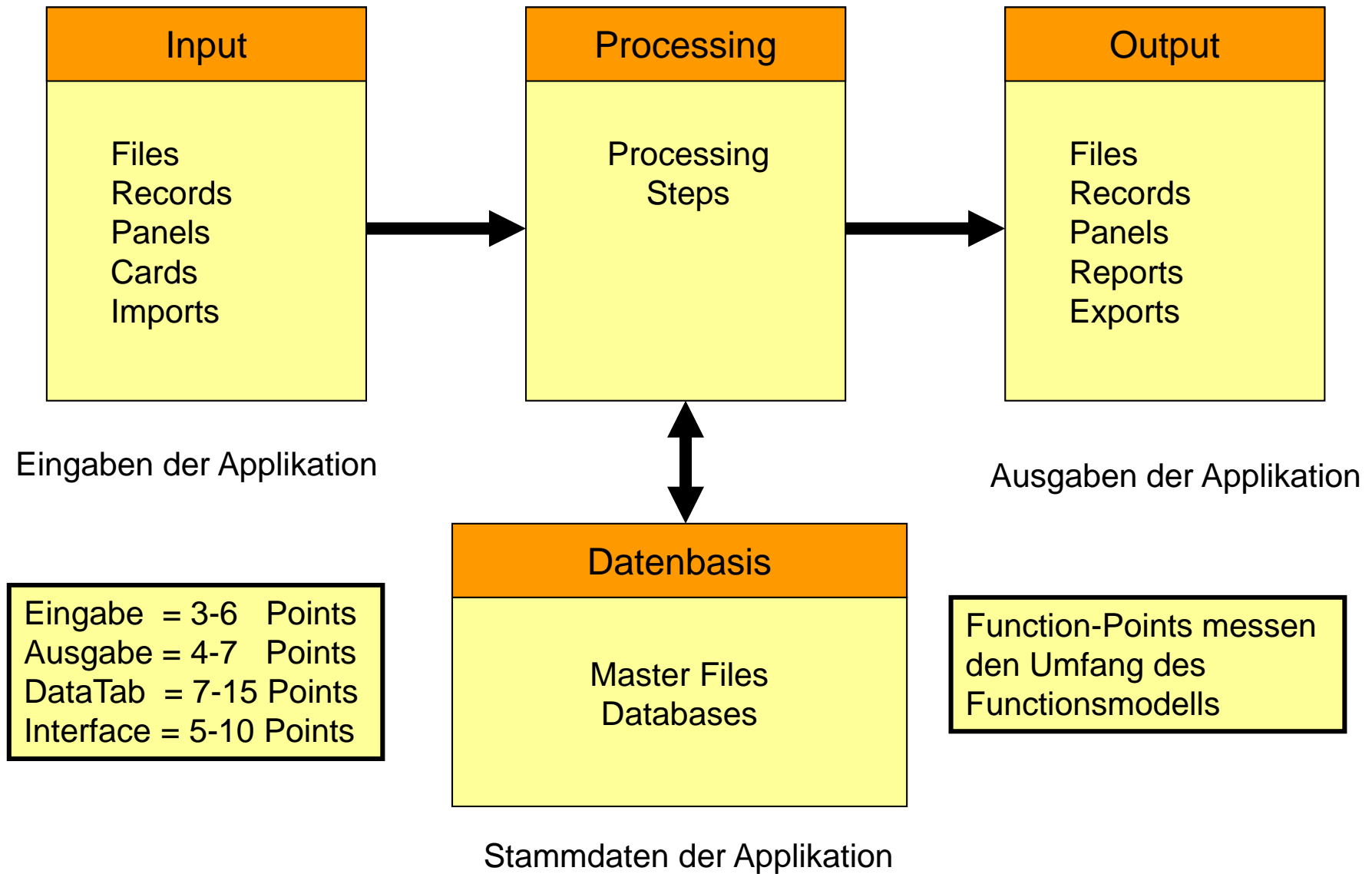
# Softwaregrößenmessung

- Function-Points
- Data-Points
- Object-Points
- Use Case Points
- Constructive Cost Model (COCOMO)
- Story-Points
- Testfälle bzw. Test-Points

# Software-Quantitätsmaße



# HIPO – der Ursprung der Function-Points



# IFPUG-4 Function-Point Zählung

Function Type	Functional Complexity	Complexity Totals	Function Type Totals
ILF	_____ Low	x 7 = _____	_____
	_____ Average	x 10 = _____	
	_____ High	x 15 = _____	
EIF	_____ Low	x 5 = _____	_____
	_____ Average	x 7 = _____	
	_____ High	x 10 = _____	
EI	_____ Low	x 3 = _____	_____
	_____ Average	x 4 = _____	
	_____ High	x 6 = _____	
EO	_____ Low	x 4 = _____	_____
	_____ Average	x 5 = _____	
	_____ High	x 7 = _____	
EQ	_____ Low	x 3 = _____	_____
	_____ Average	x 4 = _____	
	_____ High	x 6 = _____	
<b>Unadjusted Functional Point Count</b>			_____

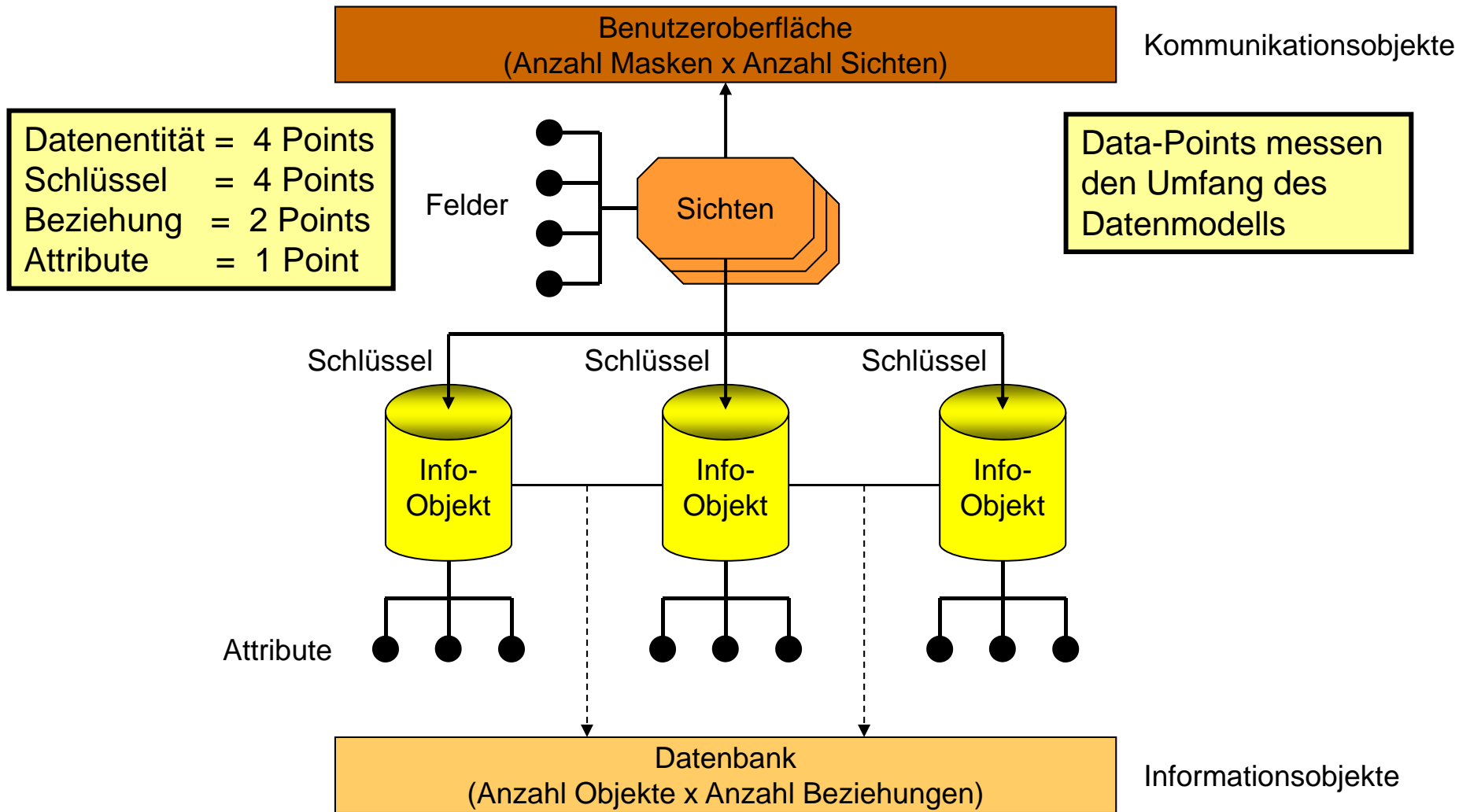
# IFPUG-4 Justierungsfaktoren

General System Characteristics (GSC)	Degree of Influence (DI) 0-5
1. Data Communication	_____
2. Distributed Processing	_____
3. Performance	_____
4. Heavily Used Configuration	_____
5. Transaction Rates	_____
6. On-Line Data Entry	_____
7. Design for End User Efficiency	_____
8. On-Line Update	_____
9. Complex Processing	_____
10. Usable in Other Applications	_____
11. Installation Ease	_____
12. Operational Ease	_____
13. Multiple Sites	_____
14. Facilitate Change	_____
Total Degree of Influences (TDI)	_____
Value Adjustment Factor (VAF) _____	= ( _____ * 0.01) + 0.65
VAF	= (TDI * 0.01) + 0.65



# Data-Point Zählung

## Größenmessung des Datenmodells



Ableitung der Data-Points aus dem Datenmodell

## Zählung der Information Data-Points

$$\text{Lieferanten} \quad \frac{50.6}{\text{Data-Points}} = \frac{30}{\text{Attribute}} + \frac{(3 \times 4)}{\text{Keys } 12} + \frac{4}{\text{Integr.}} \times \frac{1.1}{\text{Nutzen}}$$

$$\text{Artikel} \quad \frac{44}{\text{Data-Points}} = \frac{20}{\text{Attribute}} + \frac{(4 \times 4)}{\text{Keys } 16} + \frac{4}{\text{Integr.}} \times \frac{1.1}{\text{Nutzen}}$$

$$\text{Bestellposten} \quad \frac{27.5}{\text{Data-Points}} = \frac{15}{\text{Attribute}} + \frac{(2 \times 4)}{\text{Keys } 8} + \frac{2}{\text{Integr.}} \times \frac{1.1}{\text{Nutzen}}$$

$$\text{Kunden} \quad \frac{58}{\text{Data-Points}} = \frac{40}{\text{Attribute}} + \frac{(4 \times 4)}{\text{Keys } 16} + \frac{2}{\text{Integr.}} \times \frac{1.0}{\text{Nutzen}}$$

$$\text{Rückstellposten} \quad \frac{27.5}{\text{Data-Points}} = \frac{15}{\text{Attribute}} + \frac{(2 \times 4)}{\text{Keys } 8} + \frac{2}{\text{Integr.}} \times \frac{1.1}{\text{Nutzen}}$$

$$\text{Info-Objekte} \quad \frac{208}{\text{Data-Points}} = \frac{120}{\text{Attribute}} + \frac{(15 \times 4)}{\text{Keys}} + \frac{\quad}{\text{Integr.}} \times \frac{\quad}{\text{Nutzen}}$$

## Zählung der Kommunikation Data-Points

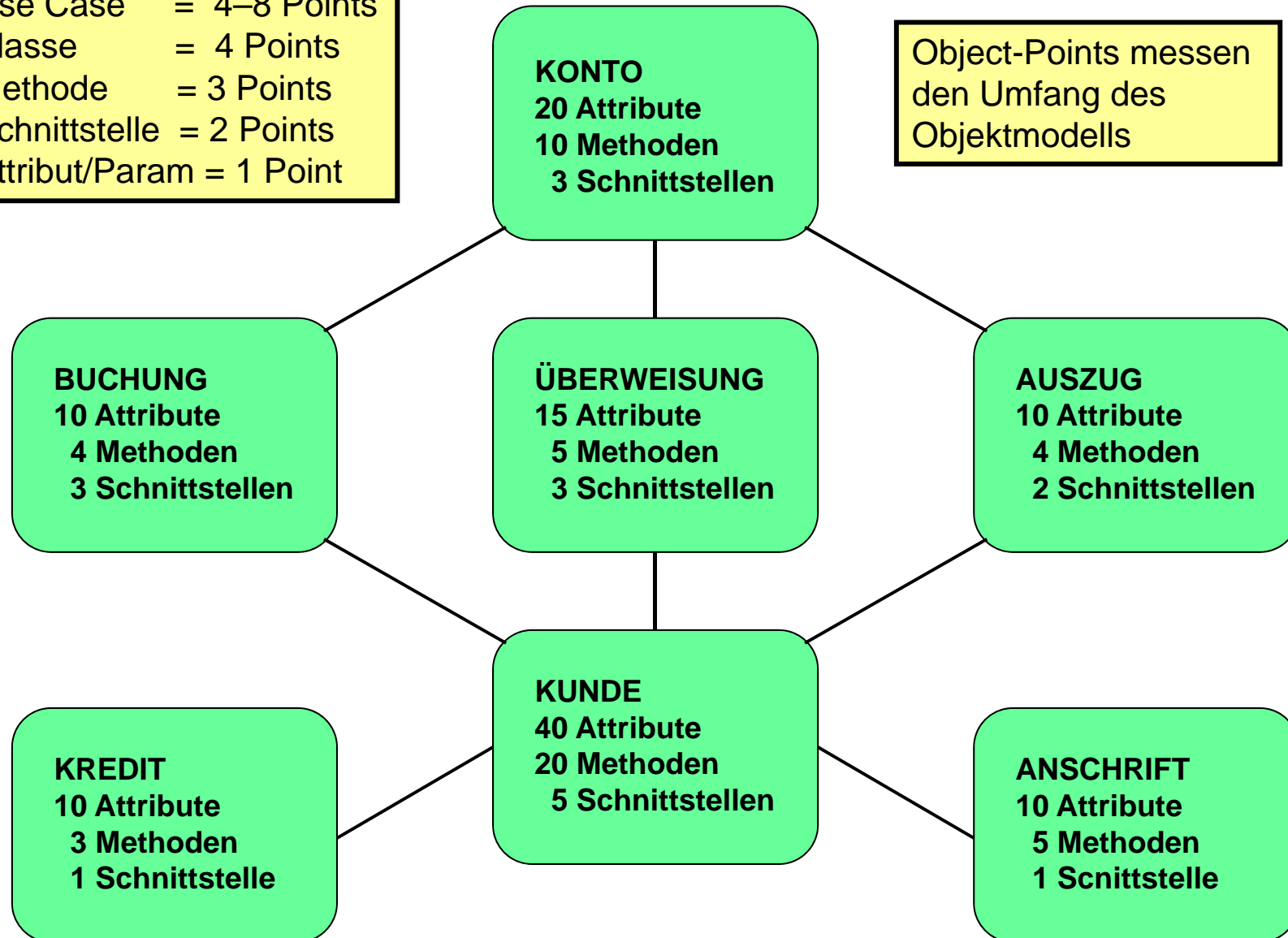
Artikelbewegungen	$\frac{30,8}{\text{Data-Points}}$	=	$\frac{22}{\text{Felder}}$	+	$\frac{(1 \times 4)}{\text{Views}}$	+	$\frac{2}{\text{Komp}}$	x	$\frac{1,1}{\text{Nutzen}}$
Lieferantenbewegungen	$\frac{41,8}{\text{Data-Points}}$	=	$\frac{32}{\text{Felder}}$	+	$\frac{(1 \times 4)}{\text{Views}}$	+	$\frac{2}{\text{Komp}}$	x	$\frac{1,1}{\text{Nutzen}}$
Artikelprotokoll	$\frac{22}{\text{Data-Points}}$	=	$\frac{16}{\text{Felder}}$	+	$\frac{(1 \times 4)}{\text{Views}}$	+	$\frac{2}{\text{Komp}}$	x	$\frac{1,0}{\text{Nutzen}}$
Lieferantenprotokoll	$\frac{26}{\text{Data-Points}}$	=	$\frac{20}{\text{Felder}}$	+	$\frac{(1 \times 4)}{\text{Views}}$	+	$\frac{2}{\text{Komp}}$	x	$\frac{1,0}{\text{Nutzen}}$
Lagerbewegung	$\frac{41,8}{\text{Data-Points}}$	=	$\frac{10}{\text{Felder}}$	+	$\frac{(5 \times 4)}{\text{Views}}$	+	$\frac{8}{\text{Komp}}$	x	$\frac{1,1}{\text{Nutzen}}$
Bestellungen	$\frac{24}{\text{Data-Points}}$	=	$\frac{12}{\text{Felder}}$	+	$\frac{(2 \times 4)}{\text{Views}}$	+	$\frac{4}{\text{Komp}}$	x	$\frac{1,0}{\text{Nutzen}}$
Aufträge	$\frac{28}{\text{Data-Points}}$	=	$\frac{16}{\text{Felder}}$	+	$\frac{(2 \times 4)}{\text{Views}}$	+	$\frac{4}{\text{Komp}}$	x	$\frac{1,0}{\text{Nutzen}}$
Komm.-Objekte	$\frac{214,4}{\text{Data-Points}}$	=	$\frac{128}{\text{Felder}}$	+	$\frac{(13 \times 4)}{\text{Views}}$	+	$\frac{24}{\text{Komp}}$	x	$\frac{\quad}{\text{Nutzen}}$



# Object-Point Zählung

Use Case = 4–8 Points  
Klasse = 4 Points  
Methode = 3 Points  
Schnittstelle = 2 Points  
Attribut/Param = 1 Point

Object-Points messen  
den Umfang des  
Objektmodells



## Object-Point Einflussfaktoren

Einflußfaktor	Erfüllungsgrad				
	kein	niedrig	teils	hoch	voll
Technische Unterstützung	0	1	2	3	4
Graphische Benutzeroberfläche	0	1	2	3	4
Gruppenware	0	1	2	3	4
Projektvernetzung	0	1	2	3	4
Prozeßreife	0	1	2	3	4
OO-Methoden	0	1	2	3	4
OO-Werkzeuge	0	1	2	3	4
OO-Sprachen	0	1	2	3	4
OO-Testautomation	0	1	2	3	4
Object Repository	0	1	2	3	4
<b>SUMME</b>	0	10	20	30	40

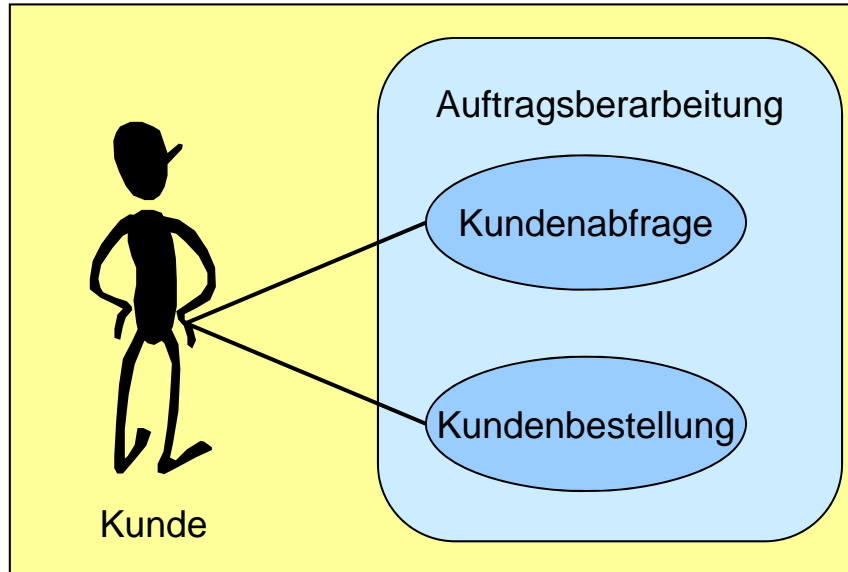
$$\text{Einflußfaktor} = 1 - (\text{Summe} / 100)$$

## UseCase-Point Methode

### Zählung von UseCase-Points

Typ	Akteur Schnittstellenart	Gewicht
Einfach	Programm Schnittstelle (API)	1
Mittel	Nachrichtenschnittstelle (XML/WSDL)	2
Komplex	Menschliche Bedienung (GUI)	3

Gewichtung der Systemakteure



Use Case Diagramm

1. Kunde selektiert den Menueintrag „Bestellung“ .
2. System zeigt Bestellungsmaske mit Artikel an.
3. Kunde wählt bis zu 10 Artikel aus und gibt die gewünschte Menge ein.
4. System prüft ob die Artikelmenge auf Lager ausreichend ist.
5. Wenn ausreichend System zieht bestellte Menge von Artikelmenge ab und erstellt Versandposten und Rechnungsposten.
6. Wenn nicht ausreichend erstellt System einen Lieferposten.

### Use Case Spezifikation

Typ	Anzahl der Prozessschritte	Gewicht
Einfach	3 oder weniger Schritte	5
Mittel	4 bis 7 Schritte	10
Komplex	Mehr als 7 Schritte	15

Gewichtung der Anwendungsfälle

Use Case-Points messen den Umfang des Anwendungsmodells

## Klassifizierung der Systemakteure

Actor Type	Description	Weight
Simple	Application with defined API	1
Average	Another System interacting through a communication protocol, like TCP/IP or FTP	2
Complex	A user interaction through a graphic interface (GUI or WEB)	3



## Klassifizierung der Anwendungsfälle

Use Case Type	No. of Steps	Weight
Simple	$\leq 3$ Steps	5
Average	4 – 7 Steps	10
Complex	$> 7$ Steps	15

## Ermittlung der Projekteinflussfaktoren

Factor	Description	Value	Weight	Factor
F1	Familiarity with RUP	0-5	1.5	0-7.5
F2	Application experience	0-5	0.5	0-2.5
F3	Object-oriented experience	0-5	1	0-5
F4	Lead analyst capability	0-5	0.5	0-2.5
F5	Team motivation	0-5	1	0-5
F6	Stable requirements	0-5	2	0-10
F7	Part-time workers	0-5	-1	-(0-5)
F8	Programming language difficulty	0-5	-1	-(0-5)
				<b>0-70</b>

## COCOMO-II Methode

Das COCOMO-II Modell benutzt alternative Größen um den Aufwand zu schätzen. Ausschlaggebend sind außer der Größe der Systemtyp, die Skalierung, die Projekteinflüsse und die angestrebte Qualität.

$$\text{Aufwand} = \text{Typ} \times \left[ \frac{\text{Größe}}{\text{Produktivität}} \right]^{\text{SE}} \times (\text{EF} \times \text{QF})$$

Aufwand: = Personenmonate

Typ := Systemtyp [ 0,5 für Standalone  
1 für Integrated  
2 für Distributed  
4 für Embedded Realtime]

Size := Kilo Source Anweisungen,  
Function-Points, Data-Points, Object-Points oder  
UseCase-Points

Prod := Anweisungen, Function-Points, Data-Points, Object-  
Points, oder UseCase-Points pro Personenmonat

SE := Skalierungsexponent (0,91:1,23)      SE = Mittelwert von  
Wiederverwendungsgrad,  
EF := Einflußfaktor (0,70:1,40)      Umgebungsreife,  
QF := Qualitätsfaktor (0,5:1,5)      Zielarchitektur,  
Teamkohäsion und  
Prozessreife.

## COCOMO-II Systemtypen

Die COCOMO Methode zielt auf die Kalkulation vom Zeit und Aufwand aufgrund:

- des Systemtyps,
- der Systemgröße (LoCs|Anweisungen|Function-Points|Object-Points)
- der Produktionsbedingungen (Skalierungsexponente),
- der Projekteinflüsse (Produktivitätsfaktoren)
- der Produktqualität.

### Systemtypen:

- 0,5 = Standalone PC-Anwendung oder Host Batchanwendung
- 1 = Integrierte Anwendung oder Host Online-Anwendung
- 2 = Verteilte Anwendung bzw. Client/Server- oder Internet-System
- 4 = Embedded Realtime System

## COCOMO-II Skalierungsexponenten

→ Unjustierte PM = A x  $\frac{(\text{Size})^{\text{SE}}}{\text{Prod}}$

SE = 0,91:1,23

SE = Median ( $\Sigma$  Exponentenfaktoren)

Exponentenfaktoren	sehr niedrig	niedrig	mittel	hoch	sehr hoch
Wiederverwendung	0%	25%	50%	75%	100%
Entwicklungsumgebung	völlig unerprobt	teils erprobt	voll erprobt	teils beherrscht	voll beherrscht
Zielarchitektur	völlig unbekannt	teils bekannt	gemischt	teils unbekannt	völlig bekannt
Teamkohäsion	keine	etwas	mittelmäßig	gut	voll
Prozeßreife	Chaos	definiert	wiederholbar	gesteuert	optimiert
Exponent	1,23	1,10	1,00	0,96	0,91

## COCOMO-II Einflussfaktoren

Faktor	sehr niedrig	niedrig	mittel	hoch	sehr hoch
Komplexität	>0,8	0,6:0,6	0,4:0,6	0,2:0,4	<0,2
Datenmenge	>100 Mill	50:100 Mill	10:50 Mill	1:10	<1 Mill
Dokumentationsgrad	0%	25%	50%	75%	100%
Wiederverwendungsgrad	0%	<20%	>20%	>40%	>50%
Werkzeugausstattung	keine	teils	mittel	gut	voll
Hardwareausstattung	schlecht	befriedigend	mäßig	gut	spitze
Vernetzungsgrad	keine	teils	mittel	gut	voll
Support	keine	teils	mittel	gut	spitze
Personalerfahrung	keine	gering	mäßig	gut	spitze
Personalkenntnisse	keine	gering	mäßig	gut	spitze
Personalfähigkeiten	keine	gering	mäßig	gut	spitze
Personalführung	keine	gering	mäßig	gut	spitze
Prozeßnormierung	keine	skizziert	praktiziert	erprobt	bewährt
Konzeptnormierung	keine	skizziert	praktiziert	erprobt	bewährt
Programmnormierung	keine	skizziert	praktiziert	erprobt	bewährt
Testnormierung	keine	skizziert	praktiziert	erprobt	bewährt
Projektverteilung	mehrere Länder	mehrere Städte	eine Stadt	ein Gelände	ein Raum
Projektunterstützung	keine	teils	mäßig	gut	voll
Kundenbeteiligung	keine	teils	mäßig	gut	voll
PM-System	keine	manuell	teils automatis.	automatisiert	optimiert
Faktor	1,40	1,20	1,00	0,85	0,70

## COCOMO-II Rechenbeispiel

<p>Lagerhaltung = Integriertes System = 1,0</p> <p>Size in Anweisungen = 4850 = 5 Kilo</p> <p>Mittlere Produktivität = 500 Anw. pro PM = 0,5 Kilo</p> <p>SE = Wiederverwendung = sehr niedrig = 1,23</p> <p style="padding-left: 20px;">Entwicklungsumgebung = voll erprobt = 1,00</p> <p style="padding-left: 20px;">Zielarchitektur = völlig bekannt = 0,91</p> <p style="padding-left: 20px;">Teamkohäsion = gut = 0,95</p> <p style="padding-left: 20px;">Prozeßreife = definiert = 1,10</p> <p style="padding-left: 20px;">Skalierung = 1,17</p> <p>Aufwand = <math>1 \times \{ [(5 / 0,5)^{1,17}] \times [1,10 \times 1,16] \}</math></p> <p>Aufwand = <math>1 \times 15 \times (1,27)</math></p> <p>Aufwand = 19 PM</p>	<p>Mittelwert = 1,0</p> <table style="width: 100%; border-collapse: collapse;"> <tr><td>EF = Komplexität</td><td>= mittel</td><td>= 1,00</td></tr> <tr><td>Datenmenge</td><td>= niedrig</td><td>= 0,85</td></tr> <tr><td>Dokumentation</td><td>= niedrig</td><td>= 1,20</td></tr> <tr><td>Wiederverwendung</td><td>= keine</td><td>= 1,40</td></tr> <tr><td>Werkzeuge</td><td>= mittel</td><td>= 1,00</td></tr> <tr><td>Hardware</td><td>= gut</td><td>= 0,85</td></tr> <tr><td>Vernetzung</td><td>= gut</td><td>= 0,85</td></tr> <tr><td>Support</td><td>= mittel</td><td>= 1,00</td></tr> <tr><td>Erfahrung</td><td>= gut</td><td>= 0,85</td></tr> <tr><td>Kenntnisse</td><td>= mäßig</td><td>= 1,00</td></tr> <tr><td>Fähigkeiten</td><td>= gut</td><td>= 0,85</td></tr> <tr><td>Führung</td><td>= gering</td><td>= 1,20</td></tr> <tr><td>Prozeßnorm.</td><td>= praktiz.</td><td>= 1,00</td></tr> <tr><td>Konzeptnorm.</td><td>= erprobt</td><td>= 0,85</td></tr> <tr><td>Programmnorm.</td><td>= erprobt</td><td>= 0,85</td></tr> <tr><td>Testnormierung</td><td>= skizziert</td><td>= 1,20</td></tr> <tr><td>Projektverteilung</td><td>= 1 Geb.</td><td>= 0,85</td></tr> <tr><td>Projektunterstütz.</td><td>= mäßig</td><td>= 1,00</td></tr> <tr><td>Kundenbeteil.</td><td>= keine</td><td>= 1,40</td></tr> <tr><td>QM-System</td><td>= automat.</td><td>= 1,20</td></tr> </table> <table style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr> <td style="border-top: 1px solid black; border-bottom: 1px solid black;">Einflußfaktor</td> <td style="border-top: 1px solid black; border-bottom: 1px solid black; text-align: right;">= 1,10</td> </tr> <tr> <td style="border-bottom: 1px solid black;">Qualitätsfaktor</td> <td style="border-bottom: 1px solid black; text-align: right;">= 1,16</td> </tr> </table>	EF = Komplexität	= mittel	= 1,00	Datenmenge	= niedrig	= 0,85	Dokumentation	= niedrig	= 1,20	Wiederverwendung	= keine	= 1,40	Werkzeuge	= mittel	= 1,00	Hardware	= gut	= 0,85	Vernetzung	= gut	= 0,85	Support	= mittel	= 1,00	Erfahrung	= gut	= 0,85	Kenntnisse	= mäßig	= 1,00	Fähigkeiten	= gut	= 0,85	Führung	= gering	= 1,20	Prozeßnorm.	= praktiz.	= 1,00	Konzeptnorm.	= erprobt	= 0,85	Programmnorm.	= erprobt	= 0,85	Testnormierung	= skizziert	= 1,20	Projektverteilung	= 1 Geb.	= 0,85	Projektunterstütz.	= mäßig	= 1,00	Kundenbeteil.	= keine	= 1,40	QM-System	= automat.	= 1,20	Einflußfaktor	= 1,10	Qualitätsfaktor	= 1,16
EF = Komplexität	= mittel	= 1,00																																																															
Datenmenge	= niedrig	= 0,85																																																															
Dokumentation	= niedrig	= 1,20																																																															
Wiederverwendung	= keine	= 1,40																																																															
Werkzeuge	= mittel	= 1,00																																																															
Hardware	= gut	= 0,85																																																															
Vernetzung	= gut	= 0,85																																																															
Support	= mittel	= 1,00																																																															
Erfahrung	= gut	= 0,85																																																															
Kenntnisse	= mäßig	= 1,00																																																															
Fähigkeiten	= gut	= 0,85																																																															
Führung	= gering	= 1,20																																																															
Prozeßnorm.	= praktiz.	= 1,00																																																															
Konzeptnorm.	= erprobt	= 0,85																																																															
Programmnorm.	= erprobt	= 0,85																																																															
Testnormierung	= skizziert	= 1,20																																																															
Projektverteilung	= 1 Geb.	= 0,85																																																															
Projektunterstütz.	= mäßig	= 1,00																																																															
Kundenbeteil.	= keine	= 1,40																																																															
QM-System	= automat.	= 1,20																																																															
Einflußfaktor	= 1,10																																																																
Qualitätsfaktor	= 1,16																																																																

# The TestCost Testing Cost Model

- Zähle die Anzahl Testfälle
- Ermittle die Testpunkte durch die Gewichtung der Testfälle
- Justiere die Testpunkte durch Testbarkeit
- Ermittle die bisherige Testproduktivität in Testfälle bzw. Testpunkte pro PT
- Kalkuliere den Testaufwand



**Für die Kalkulation von  
Testkosten sind drei Maße  
ausschlaggebend:**

- die Zahl der erforderlichen Testfälle
- die Testbarkeit des Testobjektes
- die bisherige Testproduktivität

## Ermittlung der Anzahl Testfälle

- **Blackbox-Testfälle** = {UseCases x Schritte x Bedingungen }  
+ {Oberflächen x Objekte x Zustände }  
+ {DB-Tabellen x Tuples x Ausprägungen }
- **Greybox-Testfälle** = {Schnittstellen x Parameter x Werte }
- **Whitebox-Testfälle** = {Methoden x Methodenaufrufe }  
+ {Objekt x Objektzustände }  
» | { Anzahl Ablaufpfade }

## Testaufwandskalkulation

- **Modularität** =  $1 - \frac{\{\text{Module} \times \text{Modulschnittstellen} \times \text{Modulbeziehungen}\}}{\text{Modulanweisungen}}$
- **Datenbanknutzung** =  $1 - \frac{\{\text{Testobjekte}\}}{\text{Testobjekte} + \text{Datenbanktabellen}}$
- **Schnittstellenbreite** =  $1 - \frac{\{\text{Testobjekte} \times (\text{Sollparameter}/\text{Sollschnittstellen})\}}{\text{Testobjekte} \times (\text{Istparameter}/\text{Istschnittstellen})}$
- **Pfadkomplexität** =  $\frac{\text{Zweige}}{\text{Anweisungen}} \times \frac{\text{Prädikate}}{\text{Parameter}}$
- **Testaufwand** =  $\text{Systemtype} \times \frac{\{\text{Testfälle}\}}{\text{Testproduktivität}}^{**} \text{SE} \times \text{Testbarkeitsfaktor}$

Wobei ST = Systemtypfaktor (0,5:4)

und SE = Skalierungsexponent (0,91:1,23)

# Sachsen Web Portal Anforderungsanalyse

```

TEXT ANALYSIS REQUIREMENT METRICS
TXT-TYPE: German                DATE: 29.01.05
SYSTEM:  SACHSEN                PAGE:      1
  
```

```

      C O N C E P T   Q U A N T I T Y   M E T R I C S

Number of Text Documents analyzed      =====>      4
Number of Text Lines analyzed          =====>     331
Number of Text Titles analyzed         =====>     157
Number of Text Segments analyzed       =====>     155
Number of Sentences contained in texts =====>    1754
Number of Bullet Points contained in text=====>     157
Number of Words used in texts          =====>    17824
Number of Nouns used in texts          =====>     5050
Number of Verbs used in texts          =====>     2346
Number of Conditions used in texts     =====>     138
Number of Requirements specified        =====>      0
Number of User Interfaces specified     =====>     165
Number of User Reports specified        =====>      5
Number of Objects specified             =====>     743
Number of Objects referenced            =====>    3296
Number of Predicates specified          =====>     283
Number of numeric Constants specified   =====>     437
Number of Literals specified            =====>     174
Number of States identified              =====>     258
Number of Actions identified            =====>     536
Number of Rules identified              =====>     173
Number of System Actors specified       =====>      12
Number of Use Cases specified           =====>     167
Number of Use Case Triggers specified   =====>      16
Number of Use Case Paths specified      =====>      60
Number of Use Case Steps specified      =====>     279
Number of UseCase Preconditions specified=====>      32
Number of UseCase Postconditions specified=====>      31
Number of UseCase Exceptions specified  =====>      30
Number of UseCase Relations specified   =====>      0
Number of UseCase Attributes specified  =====>     342
Number of Test Cases extracted          =====>     946
  
```

## Sachsen Web Portal Testaufwandsschätzung

C O N C E P T	S I Z E	M E T R I C S	
Number of Function-Points		=====>	4588
Number of Data-Points		=====>	9313
Number of Object-Points		=====>	11091
Number of Use Case Points		=====>	847
Number of Test Case Points		=====>	1793

**Test Case Points = 1783 / 4 TestPoints per Tag = 446 PTs**

**Damalige Testaufwandsschätzung**

## Testkennzahlen aus dem Sachsen Web Portal Projekt

- ➔ **Egovernment Web Site mit drei Applikationen**
- ➔ **1844 Testfälle wurden spezifiziert**
- ➔ **1495 Testfälle wurden ausgeführt = 81% Deckungsrate**
- ➔ **176 Testfälle deckten Fehler auf = 12% Trefferrate**
- ➔ **450 Fehler wurden durch den Test entdeckt = 18 per KDSI**
- ➔ **56 Fehler wurden nach der Freigabe entdeckt = 11% von allen**
- ➔ **Systemtest entdeckte 89% aller bisher berichteten Fehler**
- ➔ **Systemtest dauerte 4 Monate von Jänner bis Mai 2005**
- ➔ **392 Testertage wurden gebraucht**
  - = 3.8 Testfälle pro Testertag**
  - = 1.1 Fehler gefunden pro Testertag**

# The MainCost Maintenance Cost Model

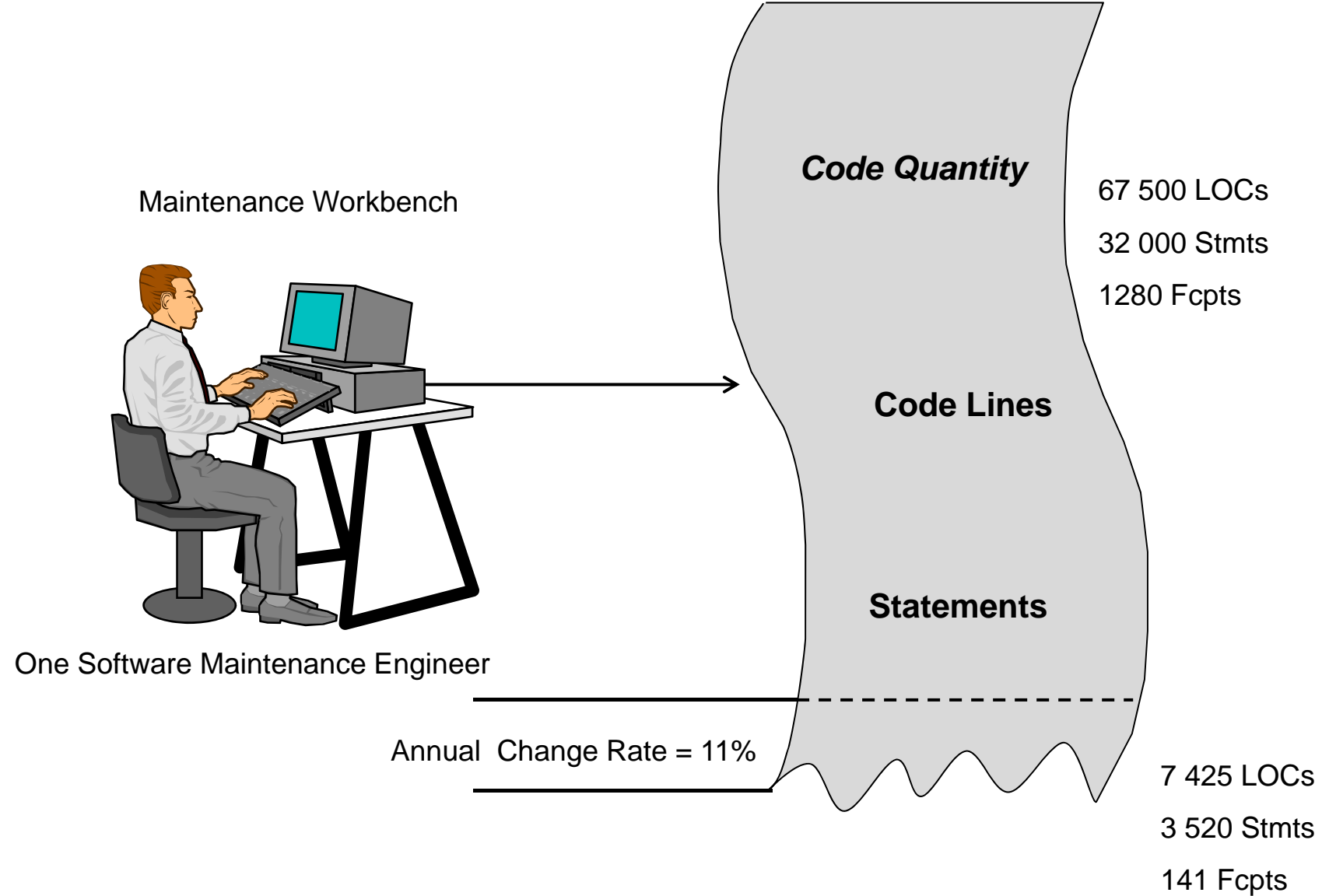
- Messe Größe, Komplexität & Qualität des bestehenden Produktes
- Hochrechnen der Änderungsrate
- Verfolge die Fehlerrate & Fehlerdichte
- Messe die Lücke zwischen Soll und Ist
- Vergleiche Lückengröße mit der Reengineering-Productivität

# Software-Wartungsaufwandsschätzung

- Verhältnis Wartungspersonal zur Codegröße
- Maintenance Cost Model
- Messung der Codegröße
- Justierung der Codegröße durch Komplexität und Qualität
- Hochrechnung der Fehleranzahl
- Schätzung der Fehlerbehebungskosten
- Schätzung der Weiterentwicklungskosten
- Kalkulation von Reengineering Projekten



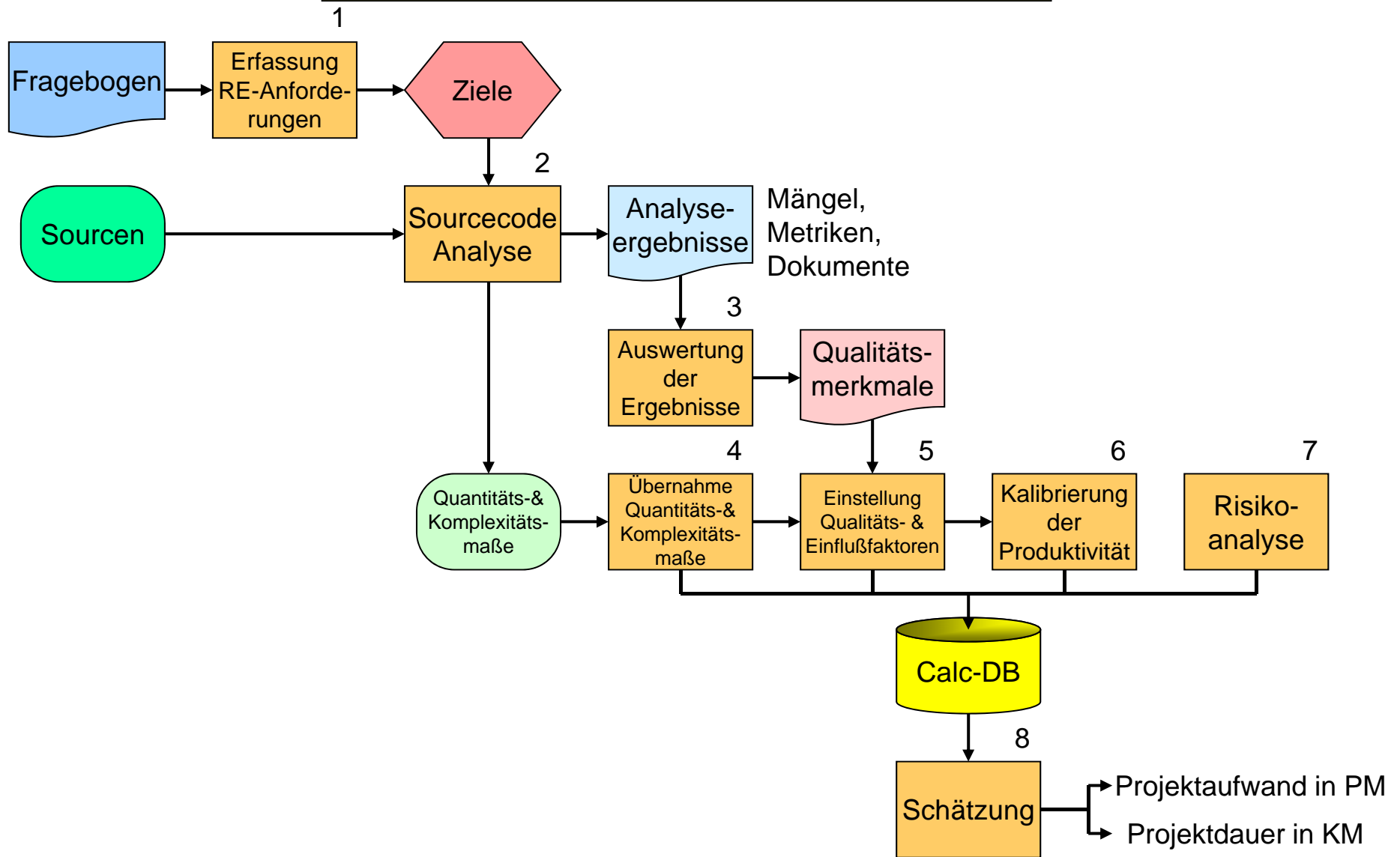
# Verhältnis Wartungspersonal zum Code



## Ratio of Maintenance Personnel to Code Quantity

according to Study of Large User Organisations in Switzerland (1997)

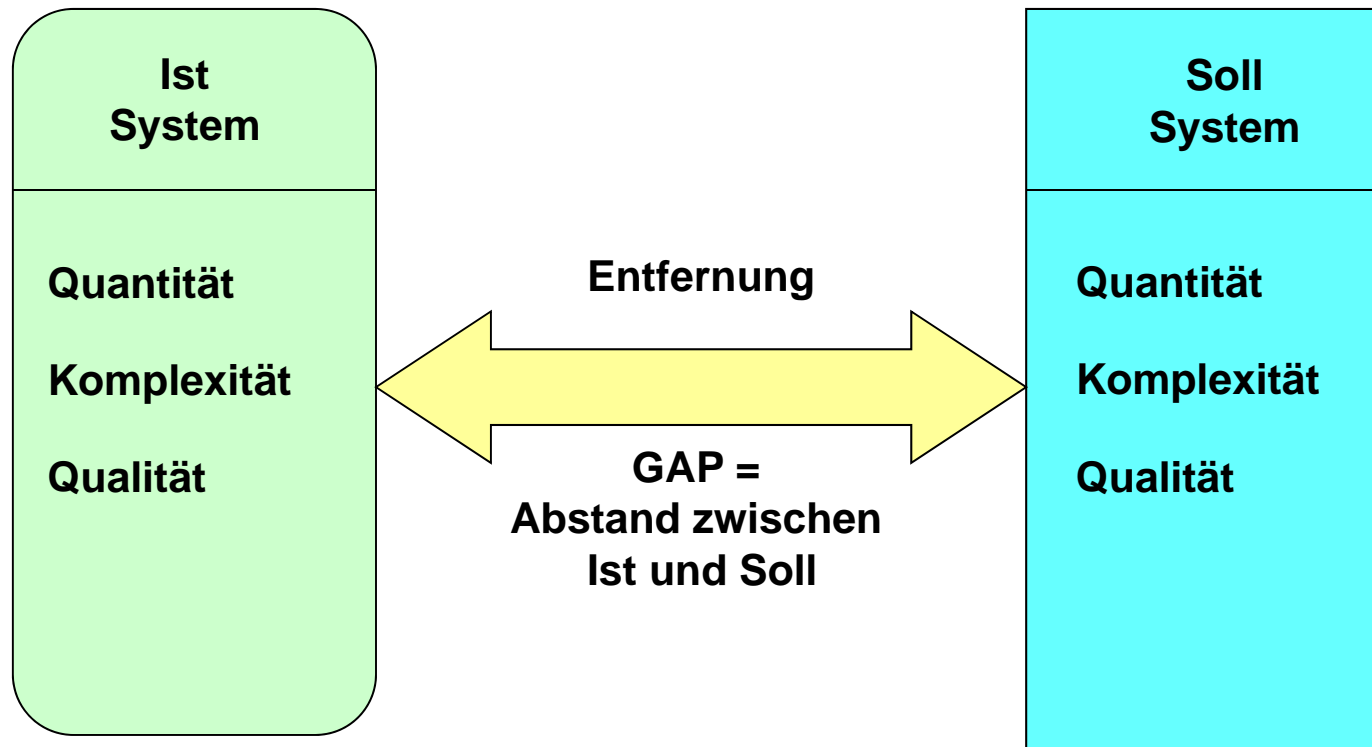
# Schätzung der Reengineeringkosten



## Schritte zu einer Reengineering-Kostenschätzung

- **1) Analyze the reengineering requirements (quantity, complexity & quality change required)**
- **2) Analyze the Source of the existing System**
- **3) Transform the results of the Source Analysis over into the Metric Database**
- **4) Compute the adjusted system size as raw size \* complexity \* quality**
- **5) Define the Degree of Quality improvement**
- **6) Calibrate the productivity data = Past Effort / PDs**
- **7) Analyze Risks and compute Risk Factor**
- **8) Calculate Effort & Time required**

# Gap Analyse



## Messung der Abstandsgröße

Adjusted\_current\_size = [current\_system\_size x current\_comp\_Factor x  
(6937) (5000) (1.11)

current\_qual\_Factor]  
(1.25)

Current\_comp\_Factor = Median\_complexity / 0.5 eg. 0.55/0.5 = 1.11

Current\_qual\_Factor = 0.5 / Median\_quality eg. 0.5/0.40 = 1.25

Adjusted\_future\_size = [Future\_system\_size x Future\_comp\_Factor x  
(3187) (4000) (0.96)

Future\_qual\_Factor]  
(0.83)

Future\_comp\_Factor = Median\_complexity / 0.5 eg. 0.48/0.5 = 0.96

Future\_qual\_Factor = 0.5 / Median\_quality eg. 0.5/0.6 = 0.83

Adjusted\_GAP = Adjusted\_current\_size - Adjusted\_future\_size  
(3750) (6937) (3187)

Adjusted\_GAP = size of reengineering task

## Kalkulation von Reengineering-Kosten

$$\textit{Effort} = \left[ \frac{\textit{Adjusted\_GAP\_size} \times \textit{Quality\_Factor}}{\textit{Productivity}} \right] \times \textit{Risk\_Factor}$$

$$\textit{Effort} = \left[ \frac{3750 \times 1.2}{705} \right] \times 1.58$$

$$\textit{Effort} = 6.38 \times 1.58 = 10 \textit{PersonMont hs}$$

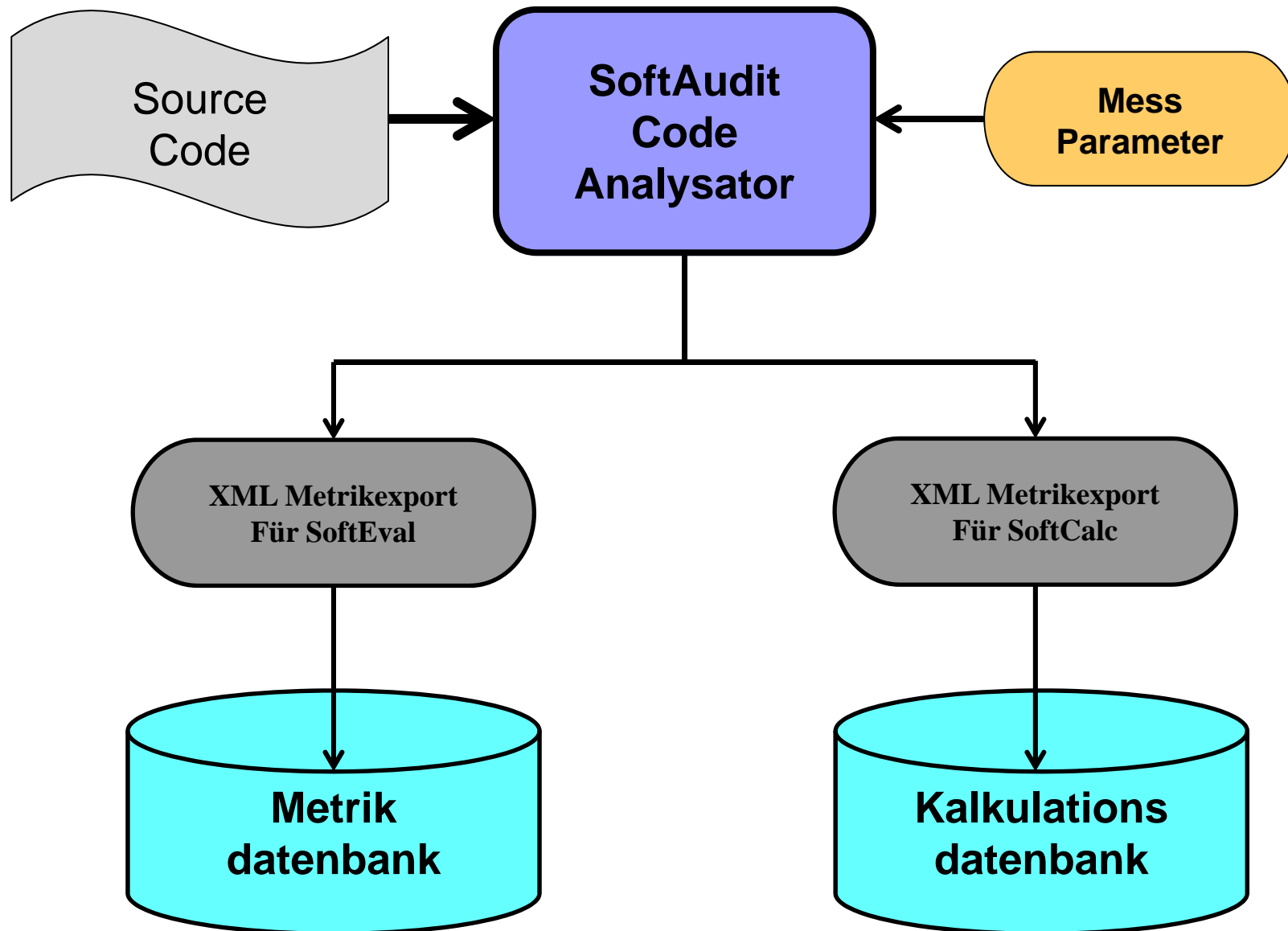
$$\textit{Time} = 1.5(\textit{Effort})^{0.35}$$

$$\textit{Time} = 1.5(10)^{0.35} = 3.35 \textit{CalenderMonths}$$

## Mengengerüst eines IBM Mainframe Systems

- 387 VA-COBOL Sourcen mit 6.822.599 LoCs
- 2.397 PL/I Sourcen mit 1.881.271 LoCs
- 324 BMS Masken mit 16,945 LoCs
- 387 DB-2-SQL Tabellen mit 21.903 LoCs
- 303 MVS-JCL Prozeduren mit 540,502 LoCs.
- Insgesamt 3.798 Source Dateien mit 9.283.180 LoCs.
- Die Analyse dieser Codemenge dauert ein Tag

# Automatische Messung des IBM Mainframe Systems





Estimation method: COCOMO-2  Project: Burgenland

- Import files
- CENTITY(IPAR) - C:\HARRY\PROJECTS\MEASUREM
  - CENTITY(IPAR) - C:\HARRY\PROJECTS\MEASUREM
  - CENTITY(IPAR) - C:\HARRY\PROJECTS\MEASUREM
  - CENTITY(IPAR) - C:\HARRY\PROJECTS\MEASUREM
  - CENTITY(IPAR) - C:\HARRY\PROJECTS\MEASUREM

Product/Project data | Object Table | Interface Table | Component Table

Project related data  
 Project Name: Burgenland  
 Product name: PersonalabrechnungBatch  
 Product type: Integrated  
 Project type: Migration  
 Project technology: Relational  
 Short product name: IPAR  
 Maximum effort (PMS): 200  
 Maximum time (Months): 24  
 Requirement Change Rate: 0,1  
 Overhead Rate: 0,2  
 Monthly Personnel Cost: 6000

Product related data  
 Maximum metrics from BWD calculation:  
 Nr KLocs: 689  
 Nr KStmts: 459  
 Nr Funct Points: 4081  
 Nr Data Points: 43125  
 Nr Objt Points: 54375  
 Nr UseCase Points: 1632  
 Nr Test Points: 18857  
 Probable Nr Errors: 3213

Measured or FWD calculated metrics:  
 Nr KLocs: 555  
 Nr KStmts: 219  
 Nr Funct Points: 2187  
 Nr Data Points: 9957  
 Nr Objt Points:   
 Nr UseCase Points:   
 Nr Test Points:   
 Probable Nr Errors: 1534

Metrics from test:  
 Nr Components:   
 Nr Statements:   
 Nr Procedures:   
 Nr Procs Tested:   
 Nr Files:   
 Nr Data:   
 Nr Data Tested:   
 Actual Nr Errors:   
 Planned values: Quality Rate 0,75 Complexity Rate 0,5 Annual Change Rate 0,1 Coverage Rate 0,8  
 Actual values: Quality Rate   
 Complexity Rate   
 Annual Change Rate   
 Coverage Rate

Calculation results | COCOMO-2 Exponent Table | Productivity Table

**System quantities:**

UseCases	
Objects:	174
Interfaces:	940
Components:	2320
Test Cases:	

**Effort estimates**

Undjusted Effort:	111,0 PMs
<input checked="" type="checkbox"/> Influence Factor	1,00
Influence adjusted Effort	111,0 PMs
<input checked="" type="checkbox"/> Resource Factor	1,00
Resource adjusted Effort	111,0 PMs
<input checked="" type="checkbox"/> Risk Factor	1,00
Risk adjusted Effort:	111,0 PMs
<input checked="" type="checkbox"/> Use Overhead Factor	1,20
Final Effort:	133,2 PMs
COCOMO-II Unadjusted	111,0 PMs
Exponent Factor:	1,00
Exponent adjusted Effort	111,0 PMs

Influence Name	Influence Type	Influence Weight	Median Influence	Planned Influence	Actual Influence	Influence Factor
Degree of Reuse	COM2	1	3	Medium	Medium	1
Devel. Environment	COM2	1	3	Medium	Medium	1
Process Maturity	COM2	1	3	Medium	Medium	1
Target Architecture	COM2	1	3	Medium	Medium	1
Team Cohesion	COM2	1	3	Medium	Medium	1

# Aufwandsschätzung einer Bankapplikation

- FIDIS Bankapplikation
- Automatische Messung des Fachkonzeptes
- Projektion der Spezifikation auf die Implementierung
- Automatische Messung des Source-Codes
- Schlüsse aus der Schätzung

# Function-Point Schätzung auf Basis des Systementwurfs

	KMS			PMS		GEA	KOS		ZAS				FIS	Zentrale Systeme				Total
	KMS	PVBE	MMS	PMS	PMSI	GEA	KOS	ITM	ZAS	ARPL	AUF	TRM	FIS	KOND	ARS	LAW	STS	
Batch Proc.	0	0	0	0	0	0	546	0	117	39	0	117	195	0	0	0	78	1092
Dialoge Proc.	2418	182	0	78	572	1300	1274	78	234	494	104	104	494	1066	286	546	1300	10530
Service Proc.	1354	414	0	1587	161	2553	4554	138	460	1058	184	391	2047	552	1265	92	739	17549
Data Objects	570	150	0	120	0	250	790	10	30	100	50	120	550	110	500	0	50	3400
Reports	0	117	0	0	0	39	0	0	39	0	0	0	195	0	0	0	0	390
Anz. Tabellen	915	120	0	300	0	390	465	30	0	195	225	90	195	630	360	300	1350	5565
<b>Total</b>	<b>5260</b>	<b>983</b>	<b>0</b>	<b>2085</b>	<b>733</b>	<b>4532</b>	<b>7629</b>	<b>256</b>	<b>880</b>	<b>1886</b>	<b>563</b>	<b>822</b>	<b>3676</b>	<b>2358</b>	<b>2411</b>	<b>938</b>	<b>3514</b>	<b>38526</b>
Person Months	263	49,2	0	104,3	36,7	226,6	381,5	12,8	44,0	94,3	28,2	41,1	183,8	117,9	120,6	46,9	175,7	1926,3

Subsystem	FPs	PM's	PY's
KMS	6243	312	26
PMS	2818	141	12
GEA	4532	227	19
KOS	7885	394	33
ZAS	4151	208	17
FIS	3676	184	15
Zentrale Systeme	9221	461	38
<b>Gesamt</b>	<b>38526</b>	<b>1926</b>	<b>161</b>

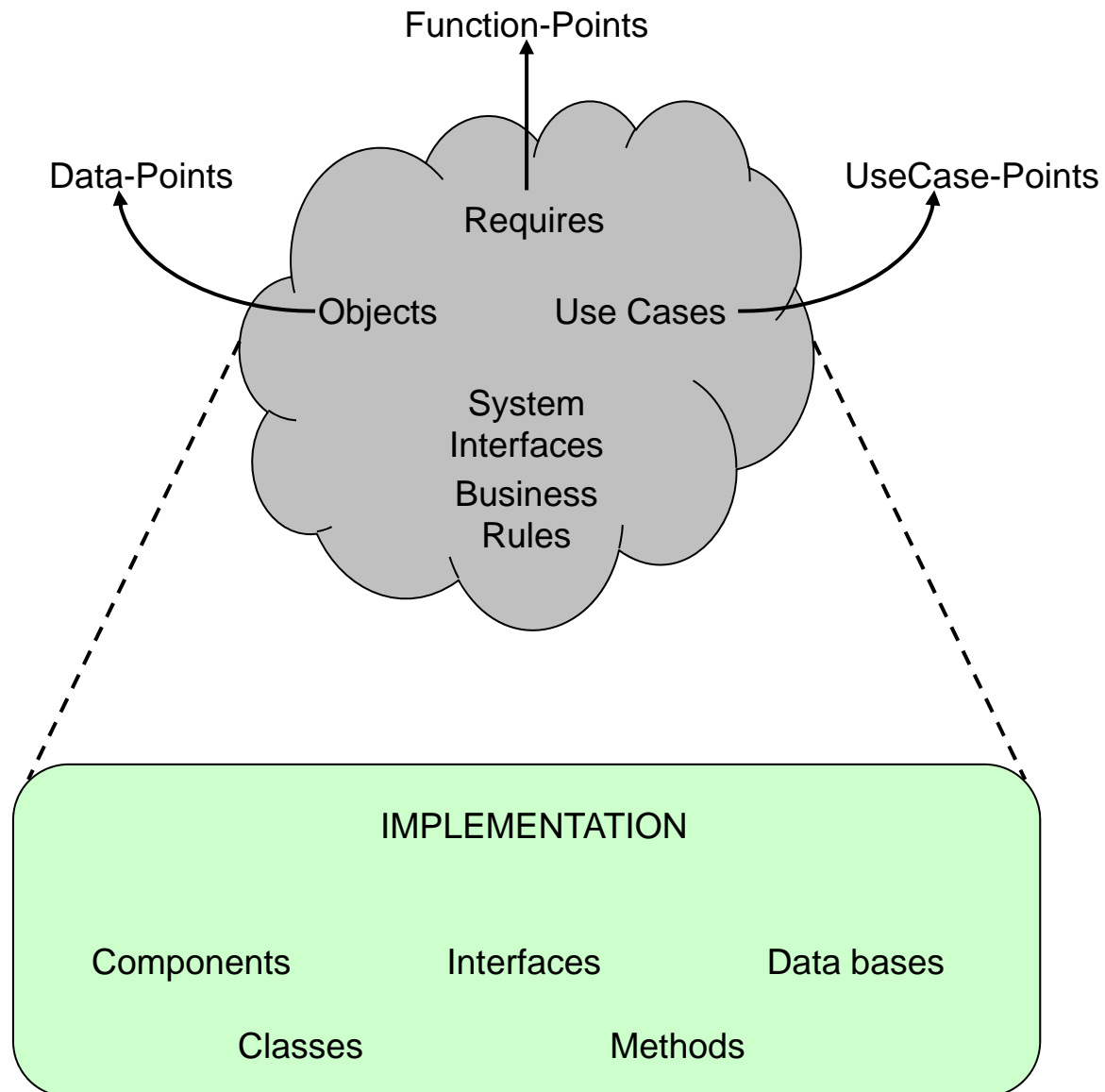
Estimated Development  
Costs of Drive = 1926 PM

Actual Development  
Costs of Drive = 1756 PM

Type	Berechnung	Faktor
Batch Process	(3 Inputs x 6)+(3 Outputs x 7)	39
Dialog Process	(2 Inputs x 6)+(2 Outputs x 7)	26
Service Process	(2 Inputs x 4)+(3 Outputs x 5)	23
Data Objects	(x 10)	10
Tables	(x 15)	15
Reports (Müller)	wie Batch	39

Person month (PM) = 20 PDs  
Person Year (PY) = 11 PMs

# Projektion der Anforderungen auf die Implementierung



# Automatisierte Anforderungsmessung

```
-----+-----+
          C M F A N A L   C O N C E P T   M E T R I C   R E P O R T
LANGUGAGE: Objekt                                DATE: 25.03.99
PROCESS:d:\fidis\sts\FIDIS                       PAGE:      1
-----+-----+
```

```
          C O N C E P T   Q U A N T I T Y   M E T R I C S
Number of Concept Members analyzed             =====>    3171
Number of Concept Lines ina all                =====> 2470490
Number of Structure Concept Lines             =====>   253135
Number of Text Lines                          =====> 2217355
```

```
          S T R U C T U R A L   Q U A N T I T Y   M E T R I C S
Number of Processes analyzed                  =====>    1305
Number of Online Processes analyzed           =====>     379
Number of Object Frame Processes analyzed     =====>     52
Number of Service Processes analyzed          =====>     41
Number of Batch Processes analyzed            =====>      3
Number of Structures analyzed                 =====>     97
Number of Data Structures included             =====>   1323
Number of Text Members referenced             =====>  21182
Number of Functions referenced                =====>   3840
Number of Maps sepcified                     =====>   1014
Number of Reports specified                   =====>     30
Number of Views specified                     =====>   1845
Number of Keys specified                      =====>   3690
Number of Parameters specified                =====>   1705
Number of Rules specified                     =====>    462
Number of Actions specified                   =====>   1322
Number of Objects specified                   =====>   2271
Number of Controls specified                  =====>   5543
Number of Procedures specified                 =====>    958
Number of Atributes specified                 =====>  11498
Number of Map Fields specified                =====>  10757
```



# Automatisierte Anforderungsmessung



D I A L O G Q U A N T I T Y M E T R I C S		
Number of Dialog Menues	=====>	56
Number of Dialog Editbars	=====>	56
Number of Dialog Folders	=====>	56
Number of Dialog Events	=====>	319
T E X T Q U A N T I T Y M E T R I C S		
Number of Sentences contained in texts	=====>	20608
Number of Conditions contained in texts	=====>	4916
Number of Nouns contained in texts	=====>	115764
Number of Constants contained in texts	=====>	26005
S Y S T E M S I Z E M E T R I C S		
Number of Function-Points	=====>	52258
Number of Data-Points	=====>	178222
Number of Object-Points	=====>	193991
C O N C E P T Q U A L I T Y M E T R I C S		
Number of Changes to the Concept	=====>	3878
Number of Major Rule Violations	=====>	20
Number of Medium Rule Violations	=====>	1913
Number of Minor Rule Violations	=====>	14291
Number of Rule Violations in Total	=====>	16224
Degree of Conformance	=====>	0.935
Degree of Complexity	=====>	0.263
Quality Rating	=====>	0.928

# Automatisierte Source-Code Messung

S O U R C E  
LANGUAGE: C++ DATE: 10.08.01  
SYSTEM: DRIVE PAGE: 1

## Q U A N T I T Y M E T R I C S

C O D E	Q U A N T I T Y	M E T R I C S
Number of Source Members analyzed	=====>	4463
Number of Source Lines in all	=====>	1480242
<b>Number of Genuine Code Lines</b>	<b>=====&gt;</b>	<b>820807</b>
Number of Comment Lines	=====>	413753
Number of Major Rule Violations	=====>	68753
Number of Medium Rule Violations	=====>	26802
Number of Minor Rule Violations	=====>	107379

## S T R U C T U R A L Q U A N T I T Y M E T R I C S

Number of Modules	=====>	947
Number of Includes	=====>	16992
Number of Classes declared	=====>	1747
Number of Classes inherited	=====>	1718
Number of Methods declared	=====>	13410
Number of Methods inherited	=====>	8865
Number of Procedures declared	=====>	16749
Number of Interfaces declared	=====>	26343
<b>Number of Object-Points</b>	<b>=====&gt;</b>	<b>230809</b>

## D A T A Q U A N T I T Y M E T R I C S

Number of Panels processed	=====>	624
Number of Reports produced	=====>	54
Number of Files declared	=====>	1106
Number of Data Bases accessed	=====>	526
Number of Data Views selected	=====>	2024
Number of Data Structures	=====>	5891
Number of Defined Definitions	=====>	25810
Number of Data Variables declared	=====>	87646
Number of Data Variables inherited	=====>	21161
Number of Data-Points	=====>	226559

## Source Code Metrik

### PROCEDURAL QUANTITY METRICS

<b>Number of Statements</b>	<b>=====&gt;</b>	<b>513443</b>
Number of Input Operations	=====>	2442
Number of Output Operations	=====>	5763
Number of File & Database Accesses	=====>	4612
Number of Function References	=====>	174136
Number of Foreign Functions referenced	=====>	97284
Number of Macro References	=====>	58754
Number of Macros referenced	=====>	11709
Number of If Statements	=====>	53398
Number of Switch Statements	=====>	3760
Number of Case Statements	=====>	20871
Number of Loop Statements	=====>	8008
Number of GOTO Branches	=====>	0
Number of Return statements	=====>	38968
Number of Control Flow Branches	=====>	182651
Number of all Control Statements	=====>	93186
Number of Literals in Statements	=====>	77308
Number of Nesting Levels (Maximum)	=====>	14
Number of Test Cases (Minimum)	=====>	116546
Number of different Statement Types	=====>	260101
Number of Assertions made	=====>	192
<b>Number of Function-Points</b>	<b>=====&gt;</b>	<b>48485</b>



# Aufwandsschätzung einer Auftragsbearbeitung

- Anwendungsgeschichte
- Akteure und Anwendungsfälle
- Story-Points
- Automatisierte UML Messung
- PC-Calc Schätzung
- SoftCalc Schätzung

## Eine Anwendungsgeschichte

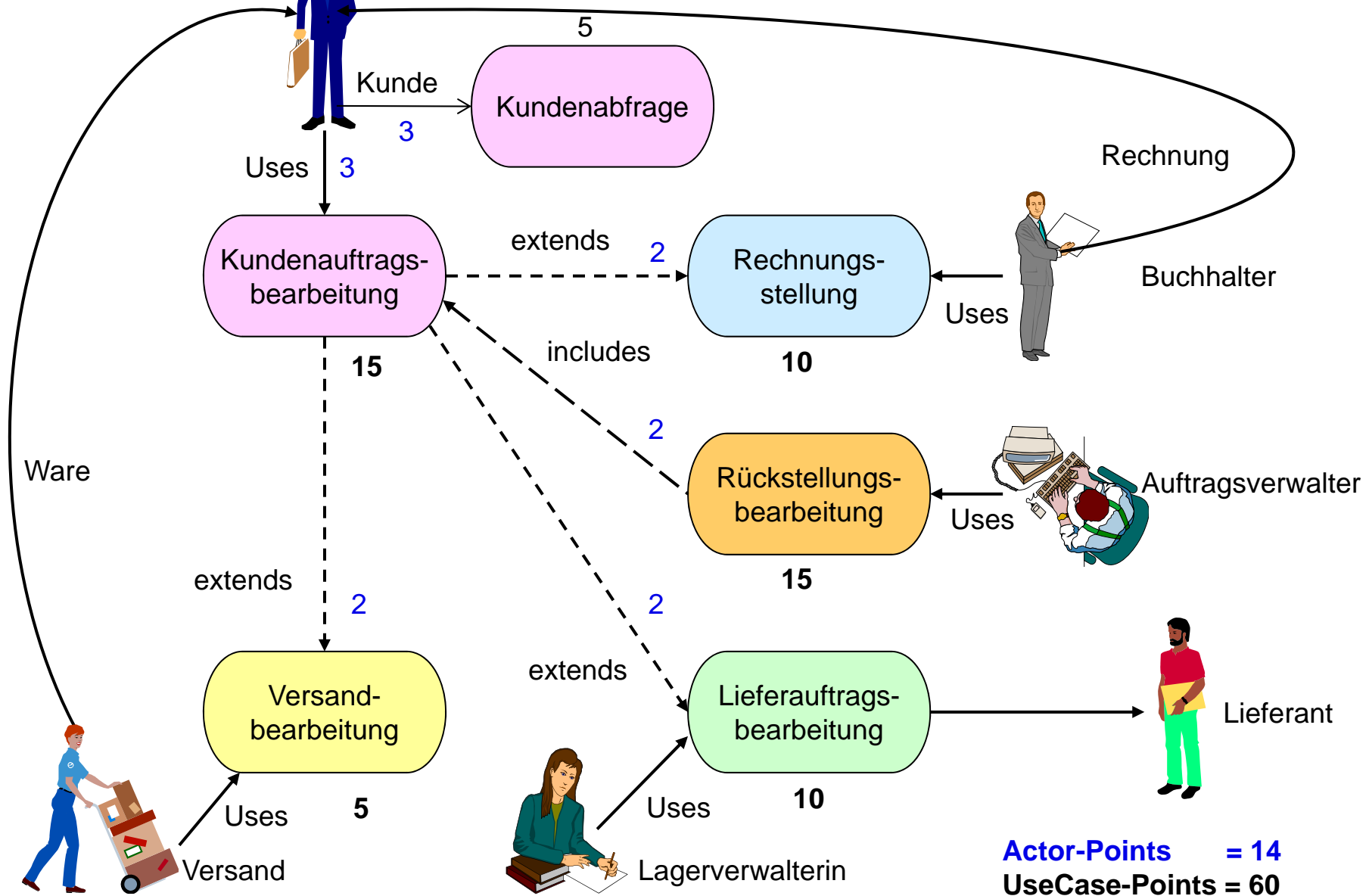
Das Auftragsbearbeitungssystem bearbeitet die Aufträge, die von den Kunden kommen. Sie werden entweder erfüllt, abgelehnt oder zurückgestellt. Wenn ein Auftrag eintrifft, wird geprüft ob der Kunde bekannt ist und ob seine Bonität in Ordnung ist. Falls der Kunde unbekannt oder seine Bonität nicht in Ordnung ist, wird der Auftrag abgelehnt.

Jeder Kundenauftrag kann bis zu 10 Bestellposten enthalten. Jeder Bestellposten hat die Nummer, der Name und die Bestellmenge des gewünschten Artikels. Erst, wird geprüft ob der gewünschte Artikel auf Lager ist. Wenn nicht, wird diese Bestellung abgelehnt. Als nächstes wird geprüft ob die Artikelmenge auf Lager ausreichend ist, um die Bestellung zu erfüllen. Wenn nicht, wird aus dem Bestellposten eine Rückstellung erzeugt. Falls, der Artikel vorhanden und die Menge ausreichend ist, wird die bestellte Menge abgezogen, die Auslieferung angewiesen und ein Rechnungsposten erstellt.

Immer wenn die Artikelmenge auf Lager reduziert wird, soll geprüft werden, ob die Menge noch über die Mindestmenge ist. Wenn nicht, ist eine Lieferbestellung für den im Moment billigsten Lieferant automatisch zu erzeugen. Ein Lieferauftrag wird aus den Lieferbestellungen eines Lieferanten gebildet und am Ende jeder Woche an die betroffenen Lieferanten versendet.

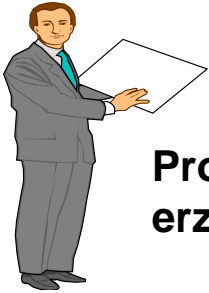
Die Rechnungsposten sind am Ende jeder Woche zu sammeln und nach Kunde zu ordnen. Jeder Kunde bekommt eine Rechnung für die von ihm in der Woche bestellten Artikel. Der Stückpreis der einzelnen Artikel wird mit der Menge multipliziert, um den Rechnungsbetrag zu ermitteln. Dazu kommt noch der im Moment gültigen Mehrwertsteuersatz. Rechnungen sind innerhalb von 30 Tage zu begleichen.

# Akteure & Anwendungsfälle

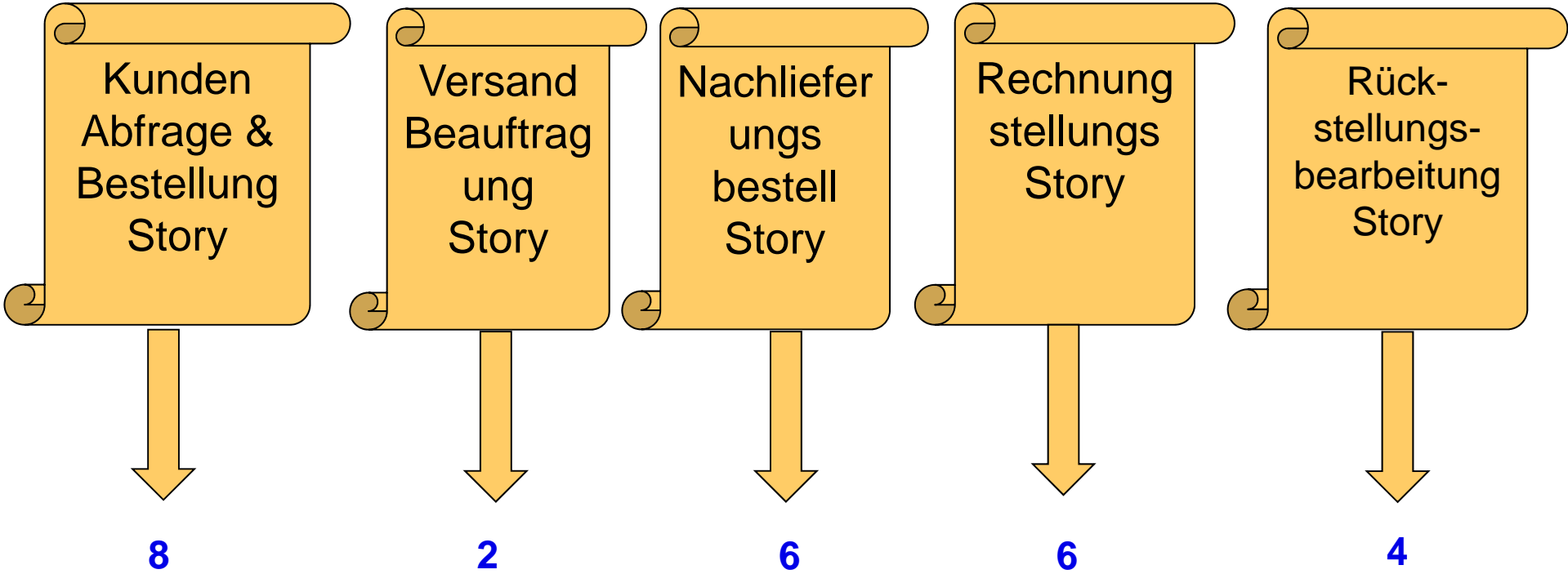


**Actor-Points = 14**  
**UseCase-Points = 60**

# Story-Point Zählung



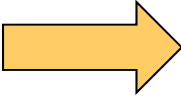
Product Owner erzählt Stories



26 Story-Points +



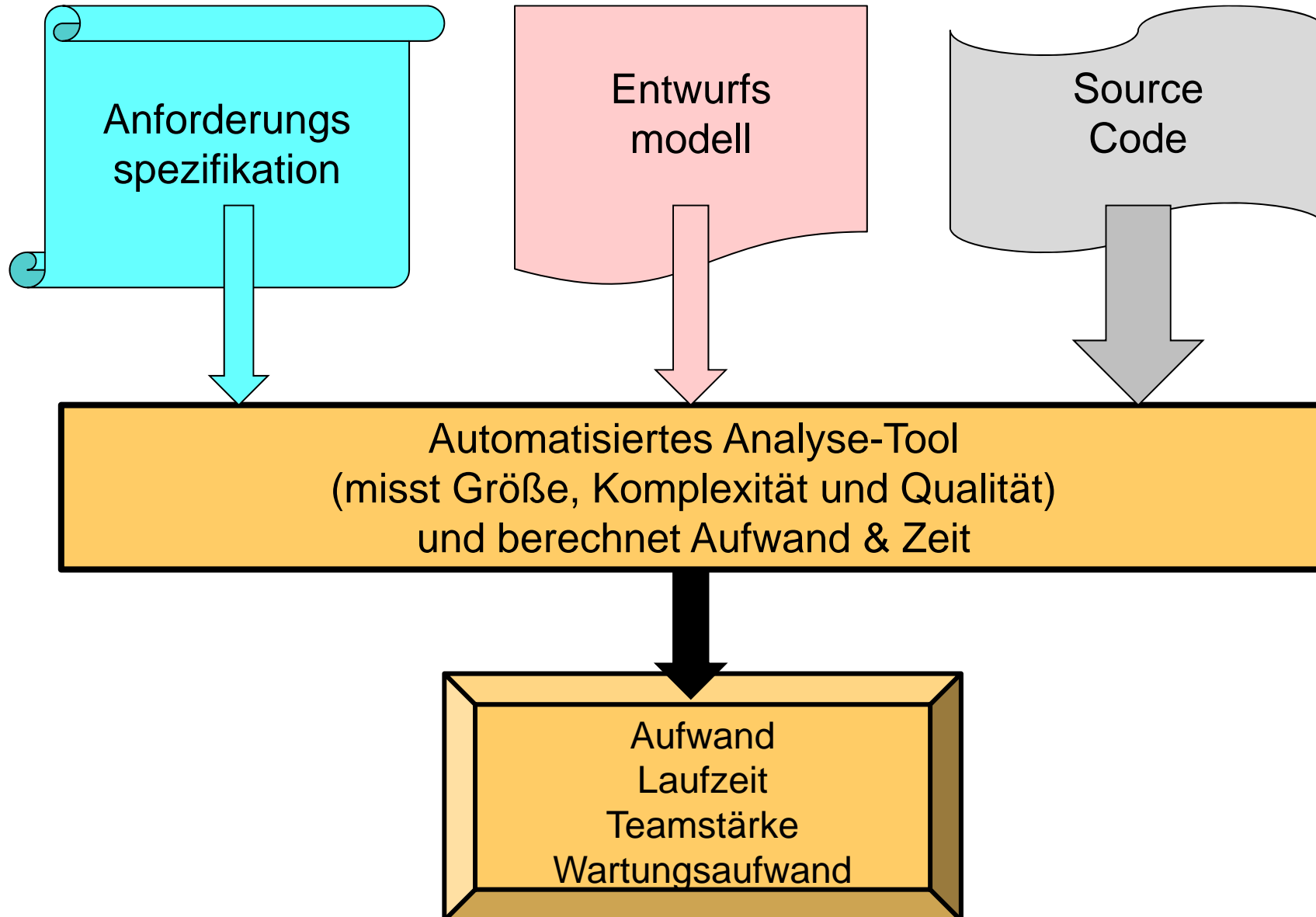
Tester



8 Test-Points = 34 Story-Points

/ 6 Points per PM = 5,5 PMs

## Automatisierte Aufwandsschätzung



## Automatisierte Aufwandschätzung mit PC-Calc

Kalkulationsergebnisse / F u n c t i o n - P o i n t - M e t h o d e

Einzelparameter:

Overhead: 0,15    Änderungsrate: 0,12    geforderte Laufzeit:    8 Monate

Ergebnisse:

Aufwand in Mannmonaten.....9,26	FPT unjustiert.....219
Computer-Stunden.....0,00	FPT Qual.-justiert.....216
Kosten.....92561,30	FPT justiert.....205

Qualitäts-Rate.....0,68	Anzahl Datenobjekte.....26
Qualitäts-Faktor.....0,98	Anzahl Schnittstellen.....42
Einflussfaktor.....0,95	Anzahl Prozesse.....10

Minimale Projektdauer.....5,8 Monate  
 optimale Besetzung.....2 Personen  
 geschätzter Wartungsaufwand.....1,2 Mannmonate pro Jahr

## Automatisierte Aufwandschätzung mit SoftCalc

<p><b><u>System quantities:</u></b></p> <p>UseCases: 6            Objects: 12            Interfaces: 12            Components:            Test Cases:</p>	<p><b><u>Effort estimates</u></b></p> <p>Undjusted Effort: 5,8 PMs  <input checked="" type="checkbox"/> Influence Factor 1,07            Influence adjusted Effort: 6,3 PMs  <input checked="" type="checkbox"/> Resource Factor 1,00            Resource adjusted Effort: 6,3 PMs  <input checked="" type="checkbox"/> Risk Factor 1,36            Risk adjusted Effort: 8,5 PMs  <input checked="" type="checkbox"/> Use Overhead Factor 0,10            Final Effort: 9,4 PMs</p>
<p><b><u>Size measurement</u></b></p> <p>Unadjusted Size: 155,00            Complexity Factor:            Complexity adjusted Size:            Quality Factor: 1,00            Quality adjusted Size: 155,00            Final adjusted Size: 170,50</p>	<p><b><u>Project estimates</u></b></p> <p>Minimum Effort: 8,5 PMs            Minimum Time: 5,1 Months            Minimum Cost: 42.570            Optimal Staff: 1,7 Prs.            Annual Maint. Effort: 1,0 PMs</p>

# Zusammenfassung

- Um Projekte zu kalkulieren, muß man die Produktivität der Mitarbeiter, die Größe und Komplexität des Vorhabens und die Projektbedingungen kennen.
- Produktivität wird anhand bisheriger Projekte ermittelt.
- Man muß die Projekttypen unterscheiden und je nach Projekttyp den richtigen Größenmaß auswählen.
- Man soll mit mindestens drei Schätzmethoden schätzen.
- Man muß den geschätzten Aufwand immer mit den Erfahrungswerten vergleichen.
- Man sollte die Schätzparameter immer wieder kalibrieren, denn Schätzen ist ein Lernprozeß, der nie zu Ende geht.