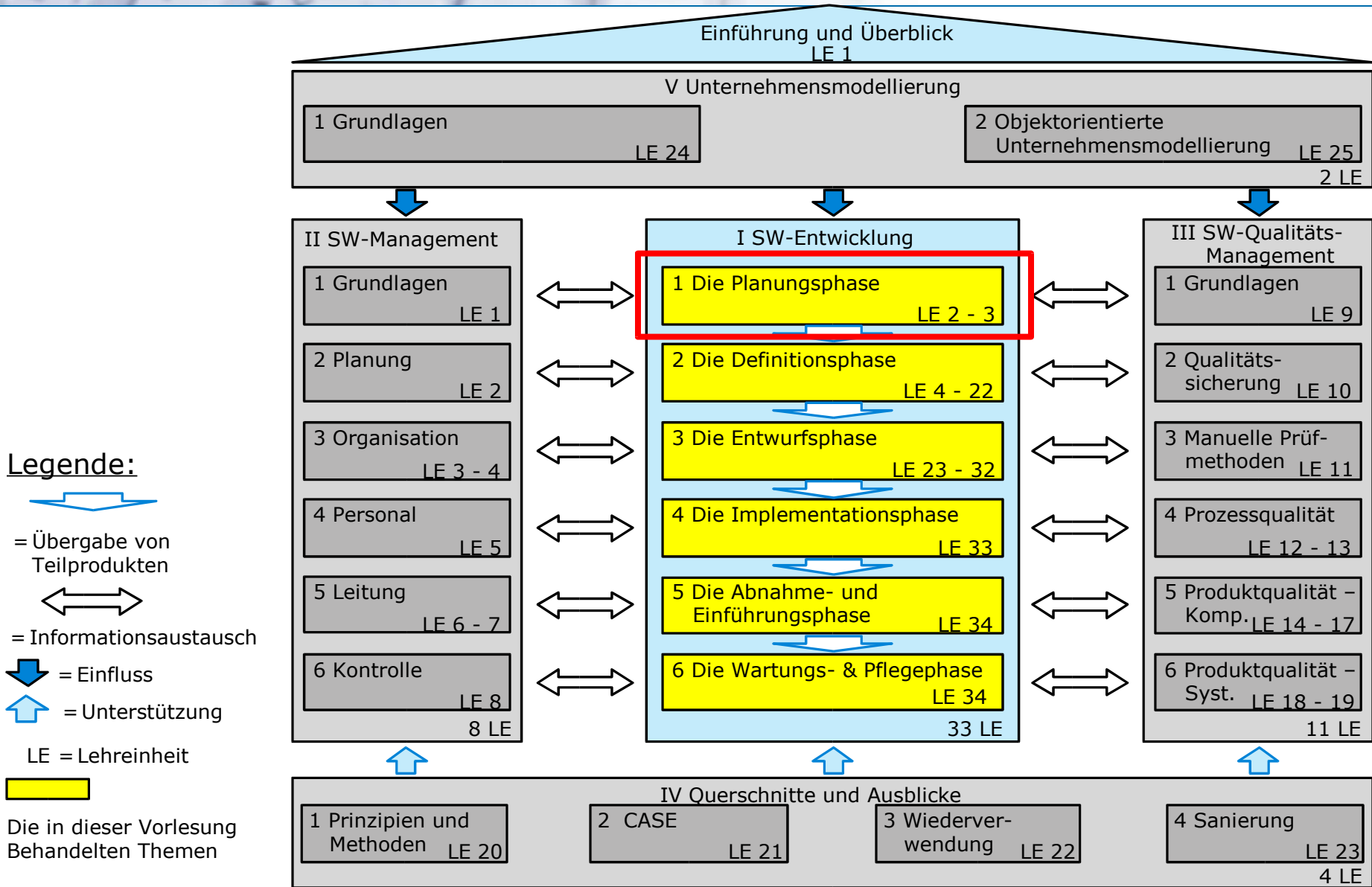


Vorlesung Softwaretechnik - Planungsphase -

Prof. Dr. Klaus-Peter Fährnich

WS 2005/2006



Überblick LE 2

LE 2: Lastenheft und Glossar

- Lernziele
- Prozessmodelle
- Die Planungsphase – Einführung/Überblick
- Das Lastenheft
- Das Glossar
- CASE-Werkzeuge
- Zusammenfassung

LE 3: Aufwandschätzmethoden

Lernziele

1. Erarbeiten einer Definition für Prozessmodelle;
2. Das evolutionäre und das inkrementelle Prozessmodell;
3. Funktion eines Lastenhefts und eines Glossars;
4. Vorgehensweise beim Planungsprozess;
5. Erstellung eines Lastenheftes und eines Glossars für vorgegebene Aufgabenstellungen;
6. Umwelt-Identifikation eines Produkts durch Akteure und Geschäftsprozesse sowie durch Schnittstellen und Datenflüsse;
7. Einsetzen von CASE-Werkzeuge für die Erstellung von Durchführbarkeitsstudien.

Definitionen

Prozess-Modell (Vorgehensmodell)

- Allgemeiner Entwicklungsplan, der das generelle Vorgehen beim Entwickeln eines Software-Produkts festlegt
- Festlegung, welche Aktivitäten in welcher Reihenfolge von welchen Personen erledigt und welche Ergebnisse (Artefakte) dabei entstehen und wie diese überprüft werden

Rolle

- Beschreibt die notwendigen Erfahrungen, Kenntnisse und Fähigkeiten, über die ein Mitarbeiter verfügen muss, um eine bestimmte Aktivität durchzuführen

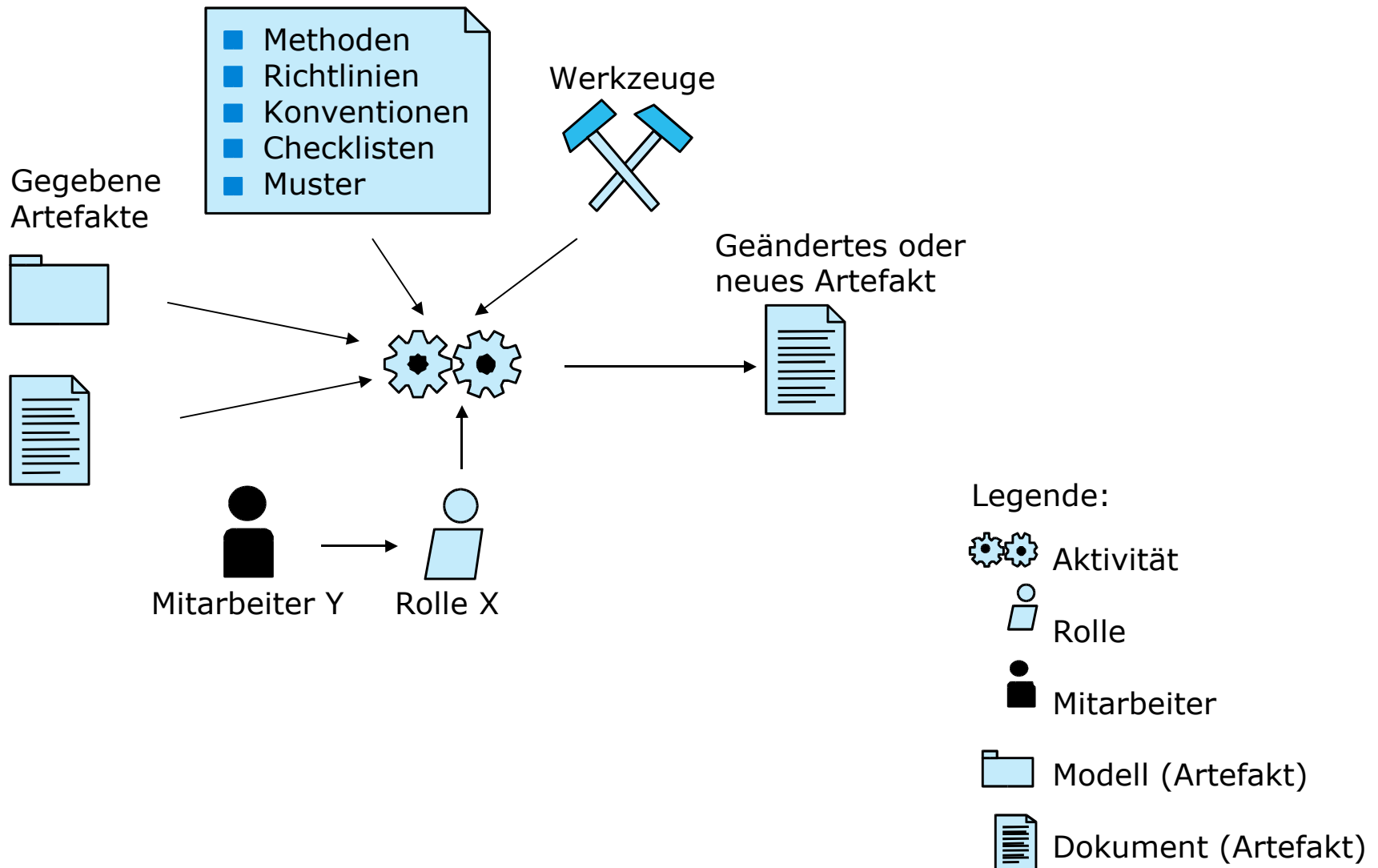
Artefakt

- Ein greifbares Stück Information, das durch Mitarbeiter erzeugt, geändert und benutzt wird, wenn sie Aktivitäten ausführen
- Kann ein Modell, ein Modellelement oder ein Dokument sein
- Beispiele: Dokument, z. B. Lastenheft
Modell, z. B. Objektorientiertes Analysemodell
Quellcode, z. B. C++ -Programm

Artefakt-Muster

- Legt die Struktur, den Inhalt und das Layout eines Artefakts fest
- Kann durch Richtlinien oder implizit durch CASE-Werkzeuge erfolgen

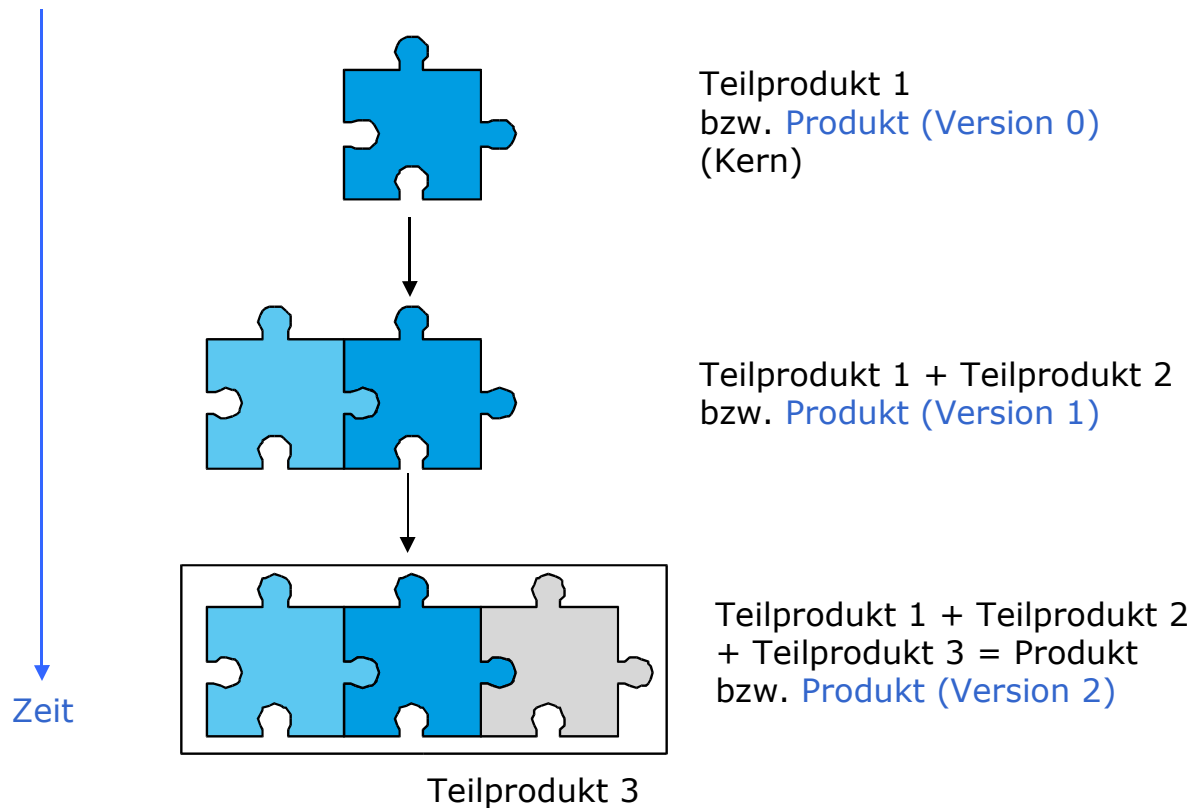
Werkzeuge der Softwaretechnik



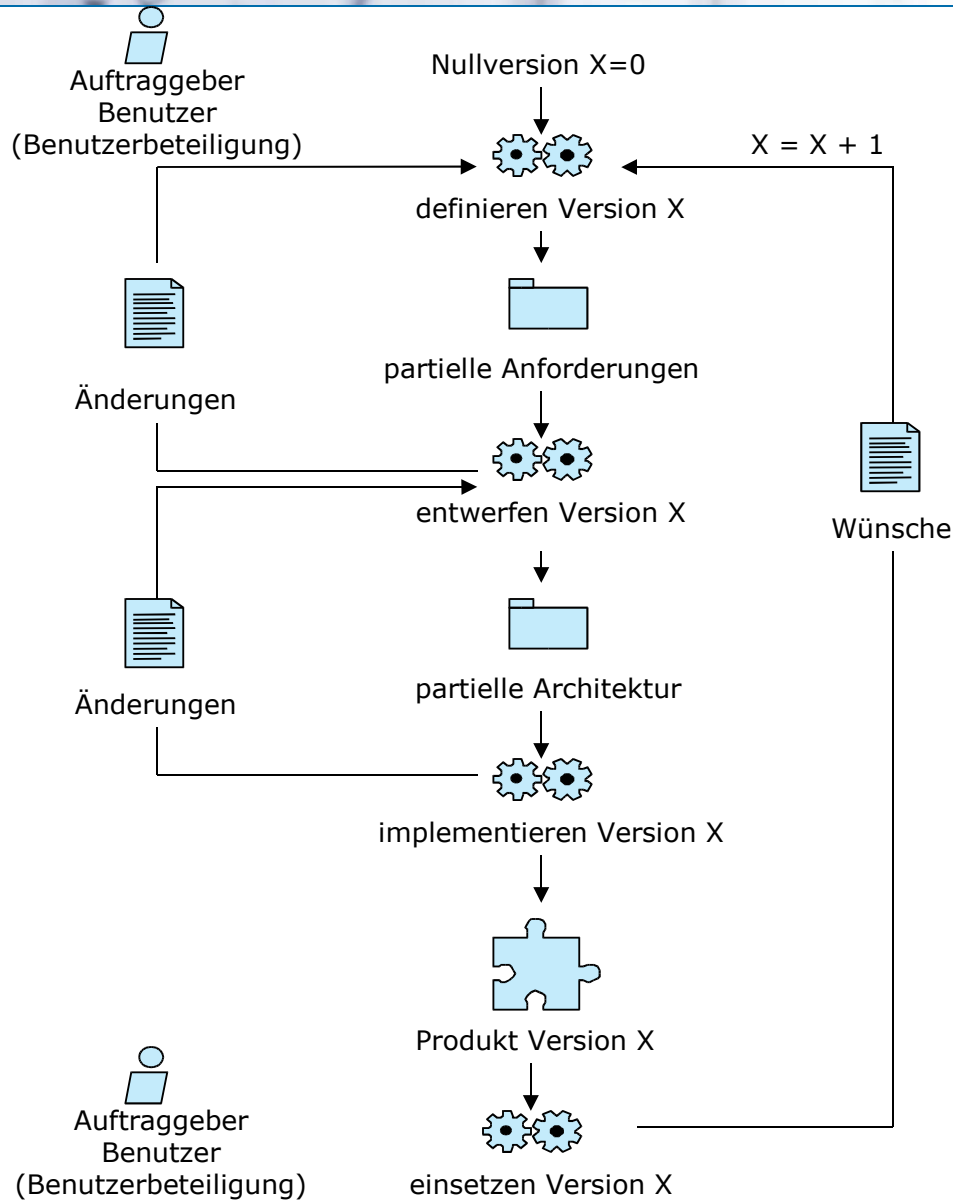
Ausbaustufen eines Produkts

Jedes 1/2 Jahr ein lauffähiges Teilprodukt

- Produkt entsteht in Ausbaustufen



Das evolutionäre Modell



Das evolutionäre Modell

Vorteile

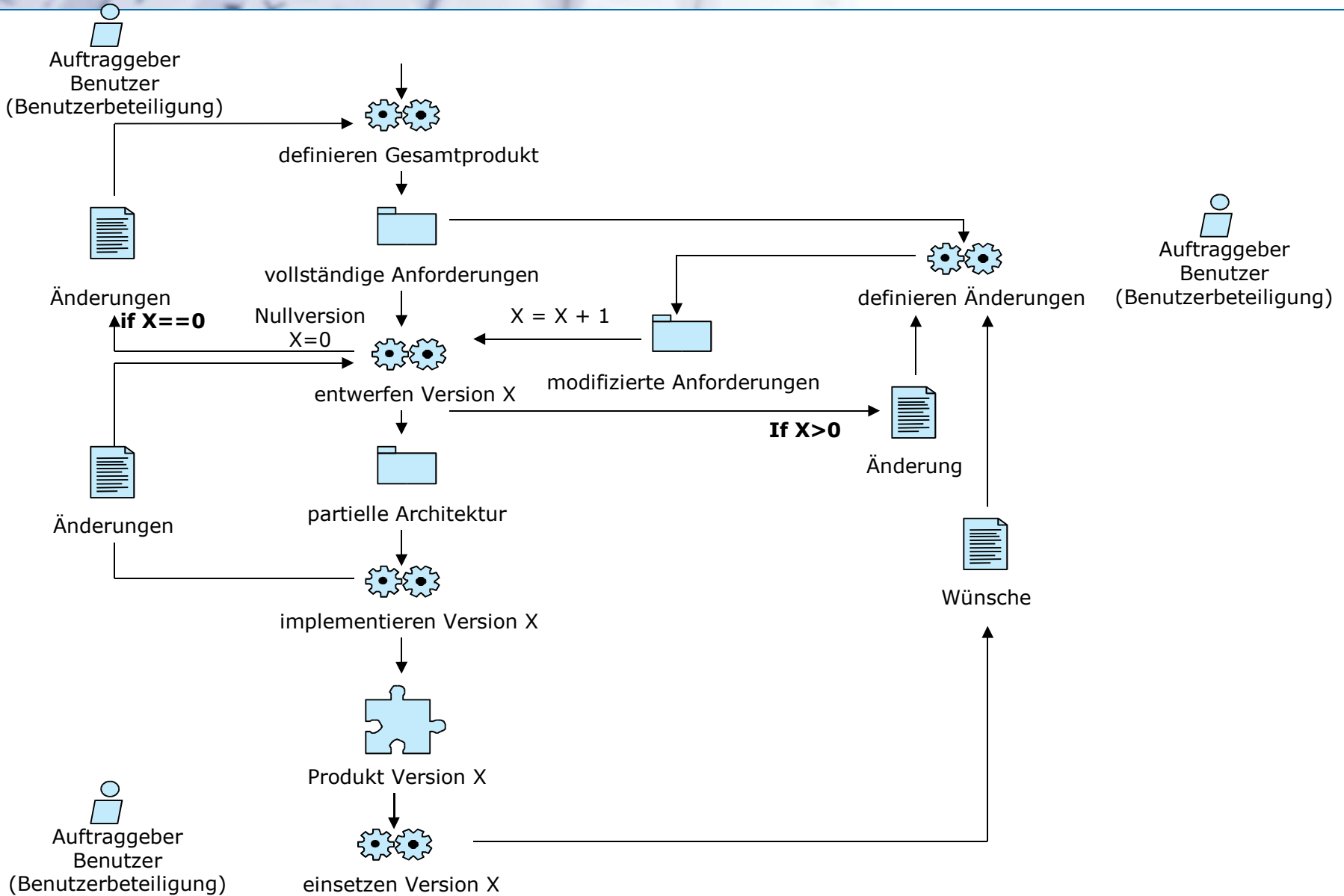
- Auftraggeber erhält in kurzen Zeitabständen lauffähige Produkte
- Frühzeitiger Einsatz einer eingeschränkten Produktversion erlaubt das Studieren der Auswirkungen des Produkteinsatzes auf die Arbeitsabläufe und diese Erfahrungen in die nächste Produktversion einzubringen
- Entwicklung ist nicht nur auf einen einzigen Endabgabetermin ausgerichtet sondern es gibt einsatzfähige Zwischenergebnisse

Nachteile

- Systemarchitektur muss in nachfolgenden Versionen möglicherweise überarbeitet werden, weil bei der Nullversion Kernanforderungen übersehen wurden
- Nullversion ist möglicherweise nicht flexibel genug um sich an ungeplante Evolutionspfade anzupassen.

Diese Nachteile werden beim **inkrementellen Modell** vermieden.

Das inkrementelle Modell



Planungsphase: Einführung

Voruntersuchung bzw. **Durchführbarkeitsuntersuchung**

- Zeigen der fachlichen, ökonomischen und personelle Durchführbarkeit
- Am Ende der Planungsphase steht die Entscheidung über die weitere Vorgehensweise

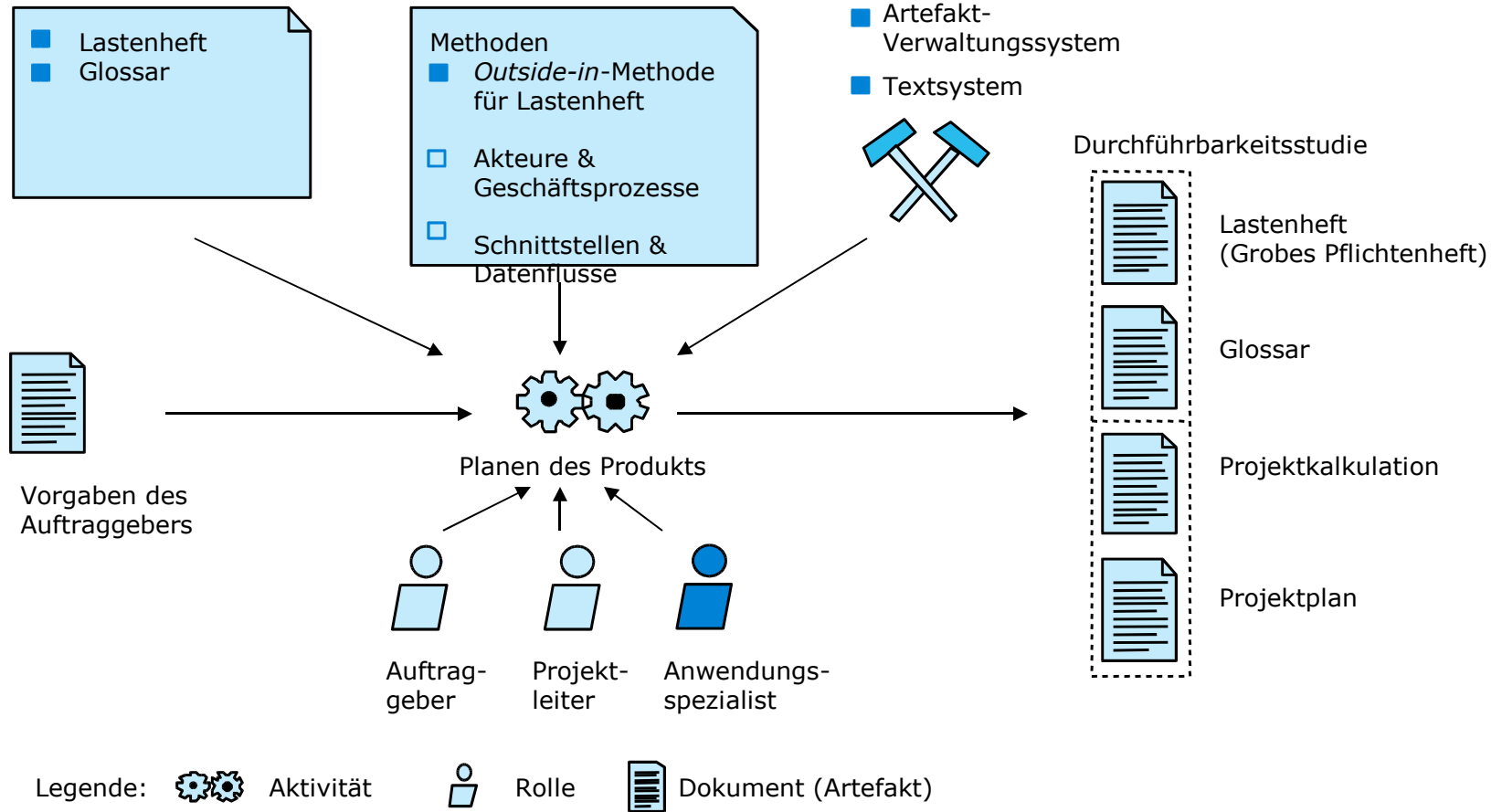
Aktivität **Planen des Produktes** beinhaltet :

- Auswählen des Produktes
 - Trendstudien
 - Marktanalysen
 - Forschungsergebnisse
 - Kundenanfragen
 - Vorentwicklungen.
- Voruntersuchung des Produkts
 - gezielte Ist-Aufnahme, wenn bereits Vorgängerprodukt vorhanden; anschl. Ist-Analyse
 - Festlegen der Hauptanforderungen
 - Festlegen der Hauptfunktionen, -daten und -leistungen
 - Festlegen der wichtigsten Aspekte der Benutzungsschnittstelle
 - Festlegen der wichtigsten Qualitätsmerkmale

Planungsphase: Einführung

- Durchführbarkeitsuntersuchung
 - Prüfen der fachlichen Durchführbarkeit
 - softwaretechnische Realisierbarkeit
 - Verfügbarkeit Entwicklungs- und Zielmaschinen
 - Prüfen alternativer Lösungsvorschläge
 - Prüfen der personellen Durchführbarkeit
 - Verfügbarkeit qualifizierter Fachkräfte für die Entwicklung
 - Prüfen der Risiken
 - Aufwands- und Terminschätzung
 - Wirtschaftlichkeitsrechnung.
- Ergebnisse dieser Tätigkeiten:
 - Durchführbarkeitsstudie (feasibility study)
 - Lastenheft (grobes Pflichtenheft)
 - Projektkalkulation
 - Projektplan

Überblick Planungsphase



Aktivitäten/Rollenzuordnung

- **Auftraggeber:** Vorgabe der Anforderungen und Abnahme der erstellten Artefakte
- **Projektleiter:** Planung, Steuerung und Kontrolle des Projekts
- **Anwendungsspezialist:** Fachgerechte Beschreibung der Anforderungen des Auftraggebers

Aktivitäten	Auftraggeber	Projektleiter	Anwendungsspezialist
Lastenheft erstellen	m	m	v
Glossar erstellen	m	m	v
Projektkalkulation erstellen	m	v	m (Aufwandschätzung)
Projektplan erstellen	m	v	m

Legende :

m=Mitwirkend

v=Verantwortlich

Gliederungsschema eines Lastenheftes

- **Zielbestimmung:** Hier wird beschrieben, welche Ziele die durch den Einsatz des Produktes erreicht werden sollen.
- **Produkteinsatz:** Es wird festgelegt für welche Anwendungsbereiche und für welche Zielgruppen das Produkt vorgesehen ist.
- **Produktübersicht:** Gibt einen (meist grafischen) Überblick über die Produktumgebung, z. B. durch ein Umweltdiagramm.
- **Produktfunktionen:** Hauptfunktionen des Produkts aus Auftraggeber-sicht sind auf oberster Abstraktionsebene zu beschreiben. Funktionalität kann mit Hilfe von Akteuren und Geschäftsprozessen oder Schnittstellen und Datenflüssen systematisch ermittelt werden.
- **Produktdaten:** Die langfristig zu speichernden Hauptdaten und deren voraussichtlicher Umfang (Mengengerüst) sind aus Benutzersicht auszuführen (/LDnn/).
- **Produktleistungen:** Werden Leistungsanforderungen bzgl. Zeit oder Genauigkeit gestellt, dann werden sie hier aufgeführt und mit /LLnn/ markiert.
- **Qualitätsanforderungen:** Die wichtigsten Qualitätsanforderungen und die jeweils geforderte Qualitätsstufe sind hier auszuführen.
(Benutzbarkeit, Effizienz,)
- **Ergänzungen:** Hier werden Ergänzungen oder spezielle Anforderungen beschrieben.

Funktionen eines Lastenheftes

- **Aufgabe:** Enthält die Zusammenfassung aller fachlichen Basisanforderungen, die das zu entwickelnde Software-Produkt aus Sicht des Auftraggebers erfüllen muss. Dient als Grundlage für das Angebot.
 - **Adressaten:** Auftraggeber sowie Auftragnehmer
 - **Inhalt:** Bewusste Konzentration auf die fundamentalen Eigenschaften des Produktes. Beschreibung des „Was“ nicht des „Wie“
 - **Form:** Vorgegebenes, standardisiertes, grobes Gliederungs-schema (Lastenheftmuster) mit vorgegebenen Inhalten
 - **Sprache:** Beschreibung auf angepasstem Abstraktionsniveau in verbaler und grafischer Form. Die einzelnen Anforderungen werden nummeriert.
 - **Didaktik:** Gliederungsschema so aufgebaut dass das Lastenheft gut lesbar ist
 - **Zeitpunkt:** Ist erstes Dokument, das die Anforderungen an ein neues Produkt beschreibt.
 - **Umfang:** Sollte auf wenige Seiten beschränkt werden

Akteure

Beispiel: Seminarorganisation

- **Beschreibung eines Akteurs** (Person, die an einem Geschehen aktiv oder unmittelbar beteiligt ist):
 - Kundensachbearbeiter: Wickelt alle Vorgänge mit Endkunden (Privatkunden) und Firmenkunden ab.
- **Umweltdiagramm:**

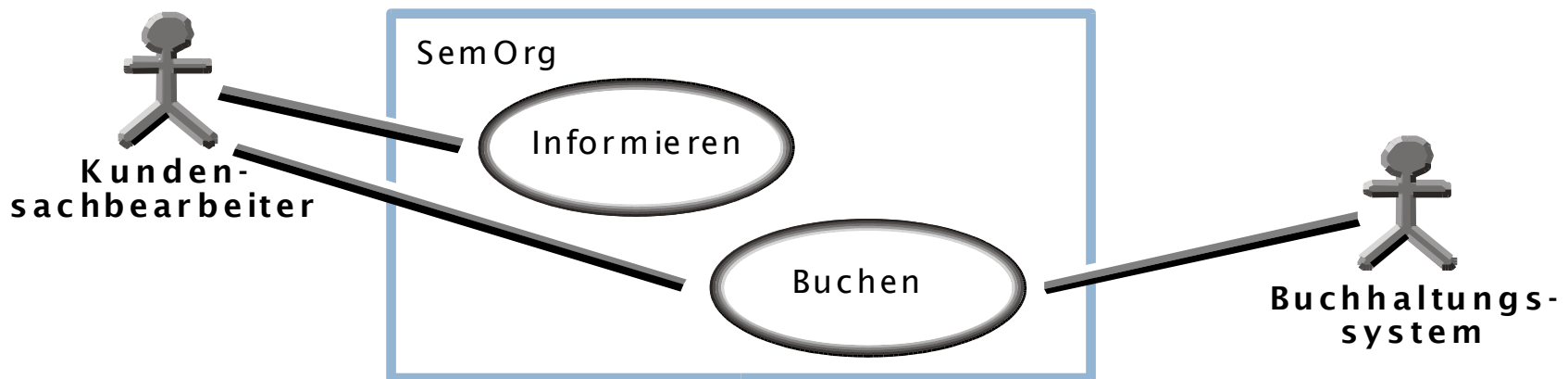


Geschäftsprozesse

- Geschäftsprozess: (use case) besteht aus mehreren zusammenhängenden Aufgaben, die von einem Akteur durchgeführt werden.

Beispiel: Seminarorganisation

- **Geschäftsprozess:**
 - „Informieren: Von Anfrage bis Auskunft“
 - Akteure: Kundensachbearbeiter
 - Beschreibung: Ein Interessent wünscht eine Auskunft über Seminare und Veranstaltungstermine oder möchte einen Seminarkatalog zugesandt bekommen



Beispiel eines Lastenheftes (1)

Lastenheft Seminarorganisation (Autor: Balzert)

1. Zielbestimmung

Die Firma Teachware soll durch das Produkt in die Lage versetzt werden, die von ihr veranstalteten Seminare rechnerunterstützt zu verwalten.

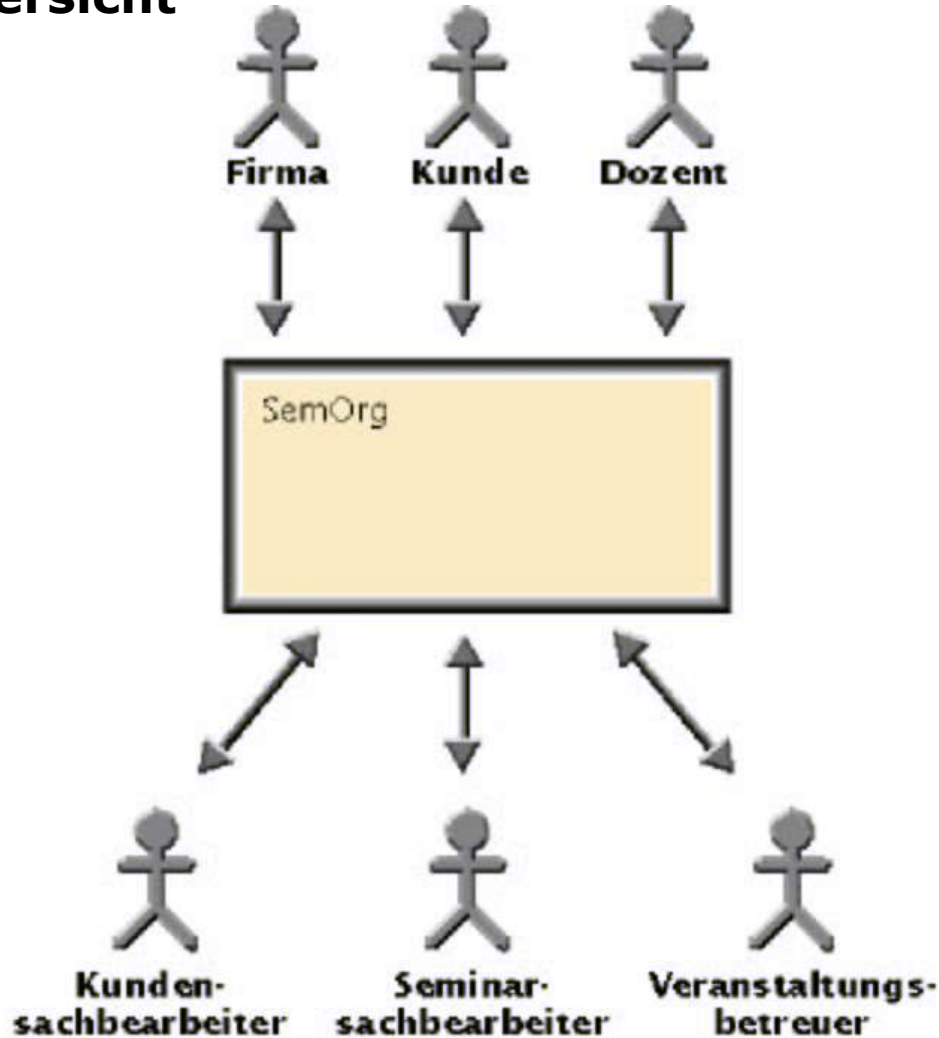
2. Produkteinsatz

Das Produkt dient zur Kunden- und Seminarverwaltung der Firma Teachware. Außerdem sollen verschiedene Anfragen beantwortet werden können. Zielgruppe des Produktes sind die Mitarbeiter der Firma Teachware.

Kunden und Firmen können sich über Seminare und Veranstaltungen informieren und selbst über das Internet Buchungen durchführen.

Beispiel eines Lastenheftes (2)

3. Produktübersicht



Beispiel eines Lastenheftes (3)

4. Produktfunktionen

/LF10/ Geschäftsprozess: Informieren: Von Anfrage bis Auskunft

Akteur: Kundensachbearbeiter, Kunde, Firma

Beschreibung: Ein Interessent wünscht eine Auskunft über Seminare und Veranstaltungen oder möchte einen Seminarkatalog zugesandt bekommen.

/LF20/ Geschäftsprozess: Buchen: Von Anmeldung bis Buchung

Akteur: Kundensachbearbeiter, Kunde, Firma

Beschreibung: Ein (Privat-)Kunde meldet sich oder eine Firma meldet Mitarbeiter zu einer Veranstaltung an.

/LF30/ Geschäftsprozess: Veranstaltung durchführen: Von Teilnahme bis Beurteilung

Akteur: Veranstaltungsbetreuer, Dozent(en), Teilnehmer

Beschreibung: Ein Teilnehmer nimmt an der gebuchten Veranstaltung teil und beurteilt sie am Schluss der Veranstaltung.

/LF40/ Geschäftsprozess: Seminarentwicklung: Von Idee zu neuem Seminar

Akteur: Seminarsachbearbeiter

Beschreibung: Aufgrund von Kunden-, Firmen- und Dozenten Anregungen werden neue Seminare konzipiert und erfasst bzw. vorhandene Seminare weiterentwickelt oder gestrichen.

/LF50/ Geschäftsprozess: Dozentenakquirierung: Von Auswahl bis Verpflichtung

Akteur: Seminarsachbearbeiter

Beschreibung: Für eine neue oder vorhandene Seminare neue Dozenten suchen und als freie Mitarbeiter verpflichten.

Beispiel eines Lastenheftes (4)

/LF60/ Geschäftsprozess: Veranstaltungsplanung: Von Terminierung bis Reservierung

Akteur: Seminarsachbearbeiter

Beschreibung: Für alle Veranstaltungen Termine festlegen, Hotels auswählen und Räume reservieren.

/LF70/ Teilnehmerliste pro Veranstaltung

/LF80/ Teilnehmerurkunde für jeden Veranstaltungsteilnehmer

/LF90/ Anfragen der folgenden Art sollen möglich sein:

Wann findet das nächste Seminar X statt?

Welche Mitarbeiter der Firma Y haben das Seminar X besucht?

5. Produktdaten

/LD10/ Kundendaten (max. 50.000)

/LD20/ Firmendaten (max. 10.000), wenn Kunde zu einer Firma gehört.

/LD30/ Veranstaltungsdaten (max. 10.000)

/LD40/ Seminartypdaten (max. 10.000)

/LD50/ Dozentendaten (max. 5.000)

6. Produktleistungen

/LL10/ Die Funktion /LF90/ darf nicht länger als 15 Sekunden Reaktionszeit benötigen.

/LL20/ Alle Reaktionszeiten auf Benutzeraktionen müssen unter 2 Sekunden liegen (außer Funktion /LF90/).

Beispiel eines Lastenheftes (5)

7. Qualitätsanforderungen

Produktqualität	sehr gut	gut	normal	nicht relevant
Funktionalität		X		
Zuverlässigkeit			X	
Benutzbarkeit		X		
Effizienz		X		
Änderbarkeit			X	
Übertragbarkeit			X	

8. Ergänzungen

5 Prozent aller Kunden sind erfahrungsgemäß im Zahlungsverzug.

Glossar

- **Definiert** und erläutert **Begriffe**, um eine einheitliche Terminologie sicherzustellen.

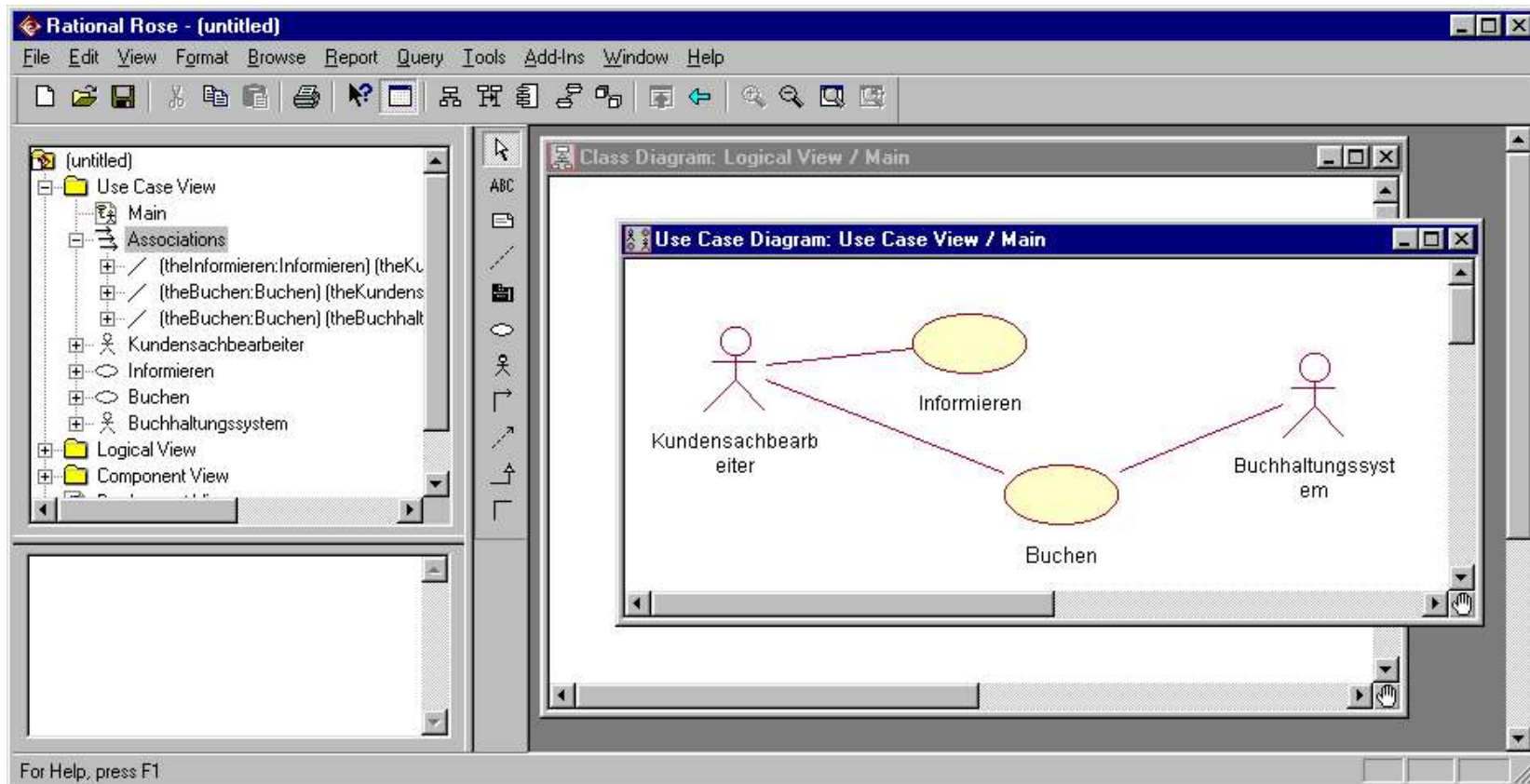
Beispiel:

Kundensachbearbeiter: Verantwortlich für die Kommunikation mit →Kunden und →Firmen einschließlich der Auskunftserteilung und Buchung.

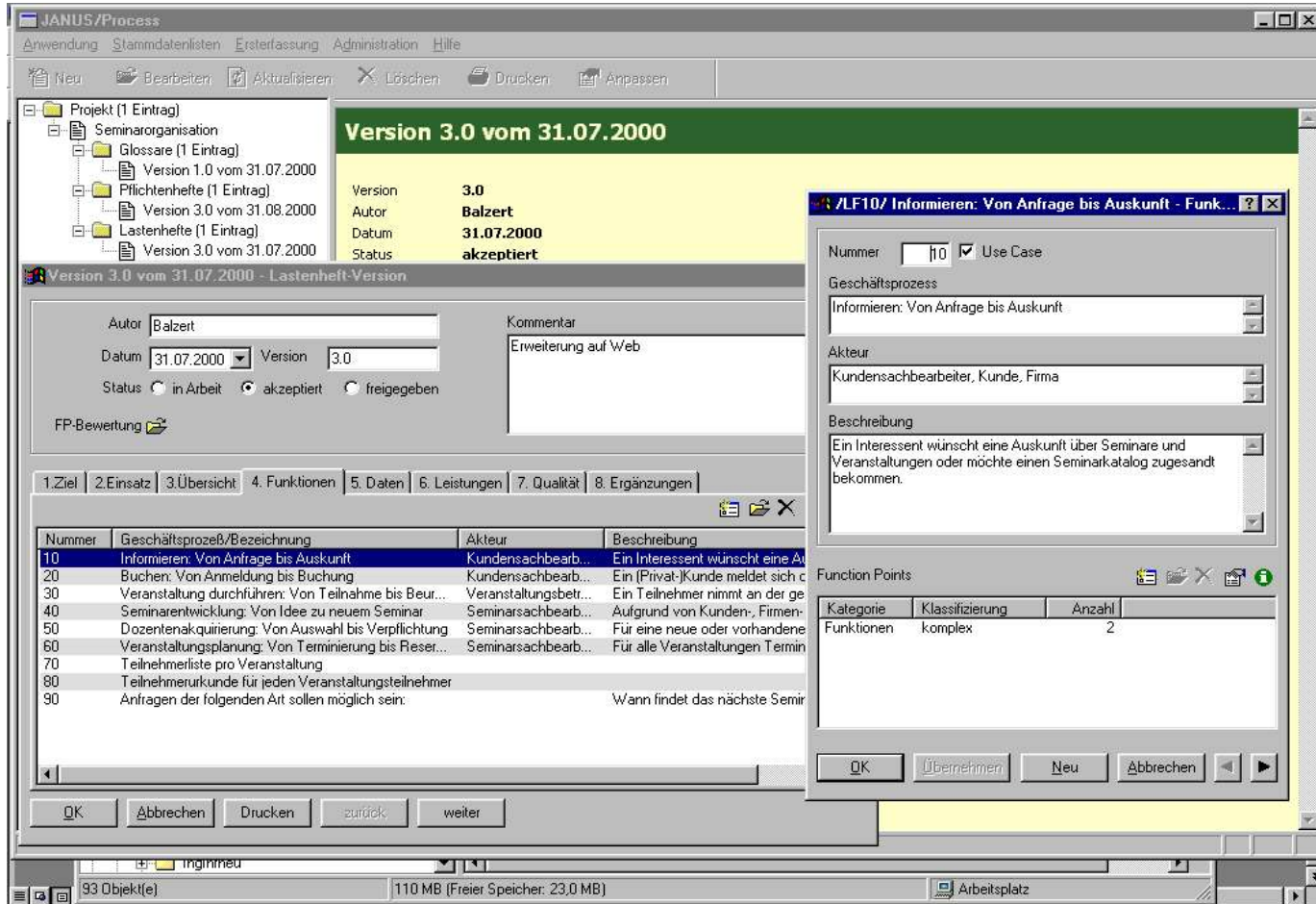
- Wichtig ist, dass die in der jeweiligen Branche **üblichen Begriffe** verwendet werden, die insbesondere auch für den Produkt-Benutzer verständlich sind.
- Die **Glossarbegriffe** werden sowohl für die Benutzungsoberfläche als auch für die Online-Hilfe und das Benutzerhandbuch verwendet.

Rational Rose

- Erstellen von Geschäftsprozessdiagrammen mit Rational Rose



- Verwaltung eines Lastenhefts mit JANUS/Process



Zusammenfassung

Prozessmodelle –

legen fest, in welcher Abfolge welche **Rollen** welche **Aktivitäten** der Software-Entwicklung durchführen sollen, um definierte **Artefakte** entsprechend vorgegebener **Artefakt-Muster** zu erstellen oder zu ändern. Mehrere Aktivitäten werden in der Regel zusammengefasst.

Beim **evolutionären** und **inkrementellen Modell** werden zunächst gut verstandene Teile des Software-Produkts entwickelt und anschließend schrittweise ausgebaut. Während beim **evolutionären Modell** zunächst nur die **Kernanforderungen** aufgestellt werden werden beim **inkrementellen Modell** zu Beginn **alle Anforderungen** an das Gesamtprodukt zusammengestellt.

Planungsphase –

ist die erste Phase einer Software-Entwicklung. In ihr wird geprüft, ob ein Produkt „durchführbar“ ist. Die fachlichen Anforderungen werden im **Lastenheft** festgehalten. Im **Glossar** werden die verwendeten Begriffe definiert. Die Ergebnisse der Planungsphase werden in einer **Durchführbarkeitsstudie** (feasibility study) zusammengefasst. Sie enthält eine Empfehlung, ob die Entwicklung fortgeführt oder abgebrochen werden soll.

Überblick LE 3

LE 2: Lastenheft und Glossar

LE 3: Aufwandschätzmethoden

- Einflussfaktoren
- Begriffe LOC und MM
- Die Relationsmethode
- Die Multiplikatormethode
- Methode der parametrischen Gleichungen
- Die Prozentsatzmethode
- Die Function Point Methode
- Zusammenfassung

Lernziele

1. Begriffe LOC und MM;
2. Das „Teufelsquadrat“ definieren lernen;
3. Verschiedene Methoden der Aufwandschätzung erklären können;
4. Konzept der Function Point Methode;
5. Faustregeln;
6. Anwenden der „Übungszählregeln“ für die Function Point-Methode auf gegebene kommerzielle Problemstellungen;
7. CASE-Werkzeuge einsetzen können.

Einflussfaktoren der Aufwandschätzung

Quantität

- Größe
 - Maß »Anzahl Programmzeilen« (LOC= **L**ines **O**f **C**ode)
 - lineare oder überproportionale Beziehung zwischen LOC und dem Aufwand
- Umfang
 - Problem der exakten Definition von Funktions- und Datenumfang
 - Maß unabhängig von einer Programmiersprache
- Komplexität
 - qualitative Maße »leicht«, »mittel« und »schwer«
 - Abbildung auf Zahlenreihe (z.B. Noten zwischen 1 und 6.)

Qualität

- Je höher die Qualitätsanforderungen, desto größer ist der Aufwand.
- Es gibt nicht die Qualität, sondern es gibt verschiedene Qualitätsmerkmale.
- Jedem Qualitätsmerkmal lassen sich Kennzahlen zuordnen.

Einflussfaktoren der Aufwandschätzung

Entwicklungsdauer

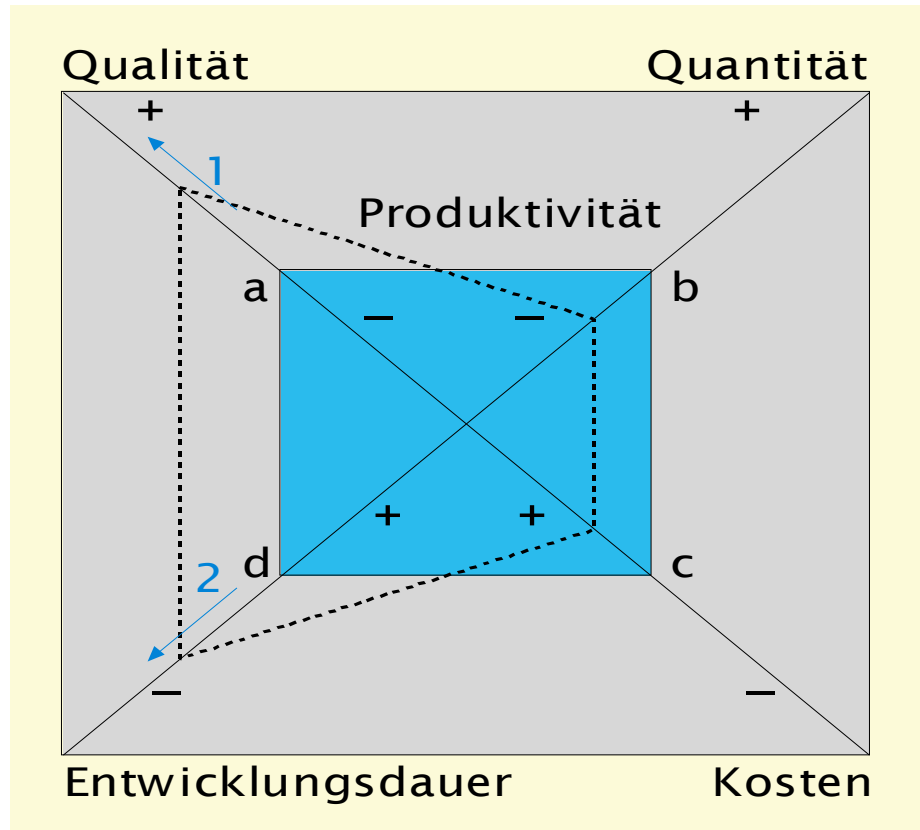
- Soll die Zeit verkürzt werden, dann werden mehr Mitarbeiter benötigt.
- Mehr Mitarbeiter erhöhen den Kommunikationsaufwand innerhalb des Entwicklungsteams.
- Der höhere Kommunikationsanteil reduziert die Produktivität.
- Kann die Entwicklungsdauer verlängert werden, dann werden weniger Mitarbeiter benötigt.
- Berechnung der optimalen Entwicklungsdauer, wenn der Aufwand in Mitarbeitermonaten bekannt ist:
Optimale Entwicklungsdauer = $2,5 * (\text{Aufwand in MM})^s$ [Monate]
mit $s = 0,38$ für Stapel-Systeme
 $s = 0,35$ für Dialog-Systeme
 $s = 0,32$ für Echtzeit-Systeme.

Produktivität

- Wird von vielen verschiedenen Faktoren beeinflusst.
- Die Lernfähigkeit und Motivation der Mitarbeiter sind entscheidend.

Einflussfaktoren der Aufwandschätzung

- Das Teufelsquadrat



Relationsmethode

- **Definition:** Das zu schätzende Produkt wird direkt mit ähnlichen Entwicklungen verglichen, Aufwandsanpassung erfolgt im Rahmen einer formalisierten Vorgehensweise, für die Aufwandsanpassung stehen Faktorenlisten und Richtlinien zur Verfügung.
- **Beispiel:** Relationsmethode
Faktoren: Ein neues Produkt soll in PL/1 realisiert werden. Das Entwicklungsteam hat im Durchschnitt 3 Jahre Programmiererfahrung. Es ist eine indexsequentielle Dateiorganisation zu verwenden.
- Zum Vergleich: Entwicklung die im Assembler programmiert wurde (-40), eine sequenzielle Dateiorganisation verwendete (+40), von einem Team mit 5 Jahren Programmiererfahrung erstellt wurde (+20).
Es ergibt sich ein Mehraufwand von 20 Punkten.

Multiplikatormethode

Das zu entwickelnde System wird soweit in Teilprodukte zerlegt, bis jedem Teilprodukt ein bereits feststehender Aufwand zugeordnet werden kann (z. B. LOC). Der Aufwand pro Teilprodukt wird meist durch Analyse vorhandener Produkte ermittelt. Oft werden auch die Teilprodukte bestimmten Kategorien zugeordnet wie Steuerprogramme, E/A-Programme, Datenverwaltungsroutinen, Algorithmen, usw. Die Anzahl der Teilprodukte, die einer Kategorie zugeordnet sind, wird mit dem Aufwand dieser Kategorie multipliziert. Die erhaltenen Werte für eine Kategorie werden dann addiert, um den Gesamtaufwand zu erhalten. Auch „Aufwand-pro-Einheit-Methode“ genannt.

Beispiel : Die Aufteilung eines Produkts in Teilprodukte hat folgendes ergeben

Kategorie	Teil- produkte	Summe LOC	Aufwands- faktor	LOC bewertet
Steuerprog.	1*500 LOC	500	1,8	900
E/A-Prog.	1*700+2*500	1700	1,5	2550
Datenverwaltg	1*800+2*250	1300	1,0	1300
Algorithmen	1*300+5*100	800	2,0	1600
Summe				6350.

Methode der parametrischen Gleichungen

- Durch Korrelationsanalysen wird ermittelt, welche Faktoren welchen wertmäßigen Einfluss auf den Gesamtaufwand haben
- Solche Analysen müssen mit einer großen Anzahl von abgeschlossenen Entwicklungen und einer Vielzahl von Faktoren durchgeführt werden
- Die Faktoren, die die höchste Korrelation besitzen, werden zu einer Gleichung zusammengefasst
- Der zu jedem Faktor gehörende Koeffizient repräsentiert die Stärke des Einflusses auf den Gesamtaufwand.
- Der Aufwandsfaktor repräsentiert den Einfluss des jeweiligen Faktors auf den Gesamtaufwand
- Als Gleichung würde sich ergeben:

$$LOC_{\text{bewertet}} = LOC_{\text{Steuerprogramme}} * 1,8 + LOC_{\text{E/A-Programme}} * 1,5 + LOC_{\text{Datenverwaltung}} * 1 + LOC_{\text{Algorithmen}} * 2$$

Bewertung

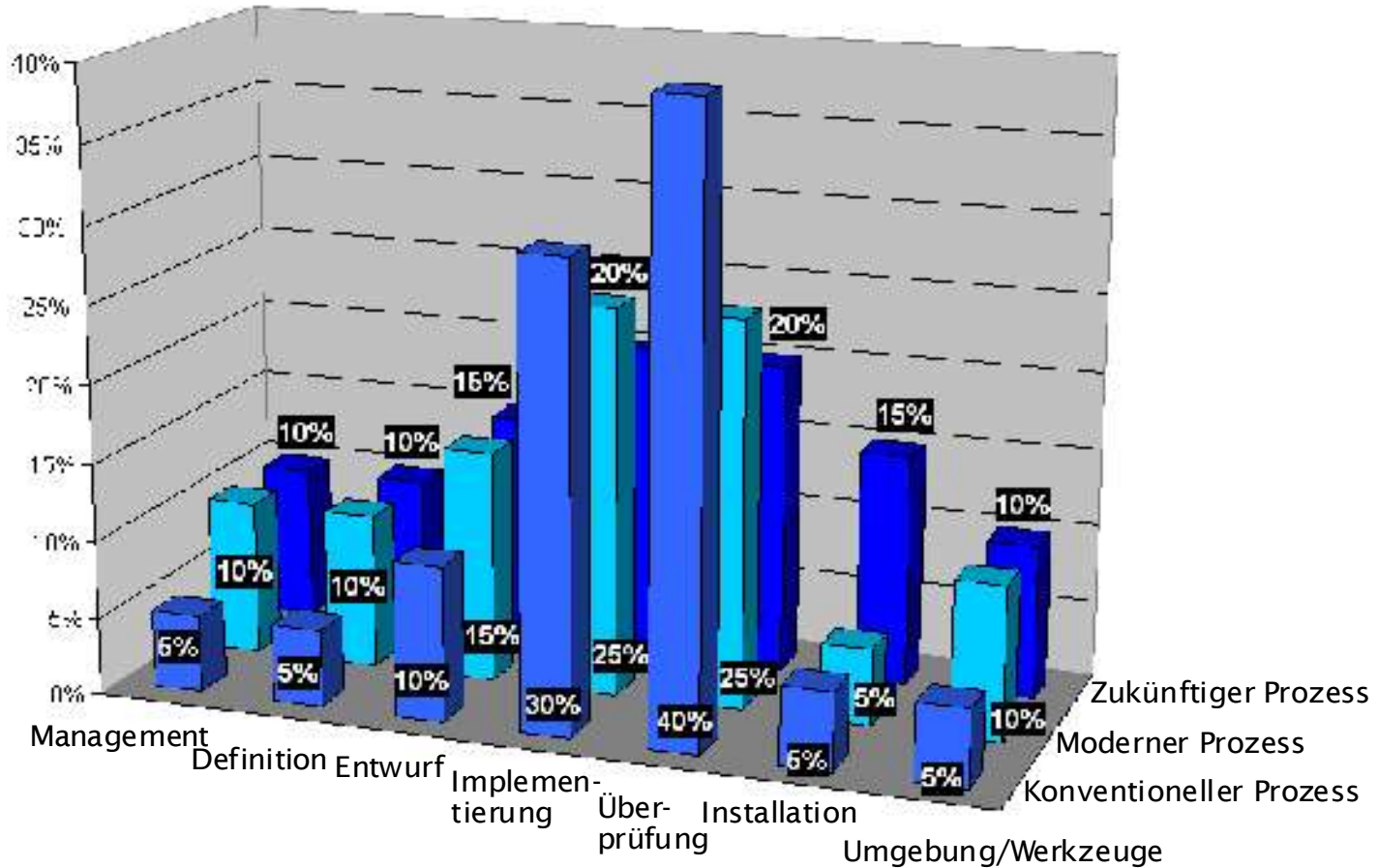
- Es ist eine umfangreiche, empirische Datensammlung und -auswertung erforderlich, um die zu berücksichtigenden Faktoren unternehmensspezifisch zu bewerten
- Die Koeffizienten müssen permanent überprüft werden, um den technischen Fortschritt zu berücksichtigen.

Prozentsatzmethode

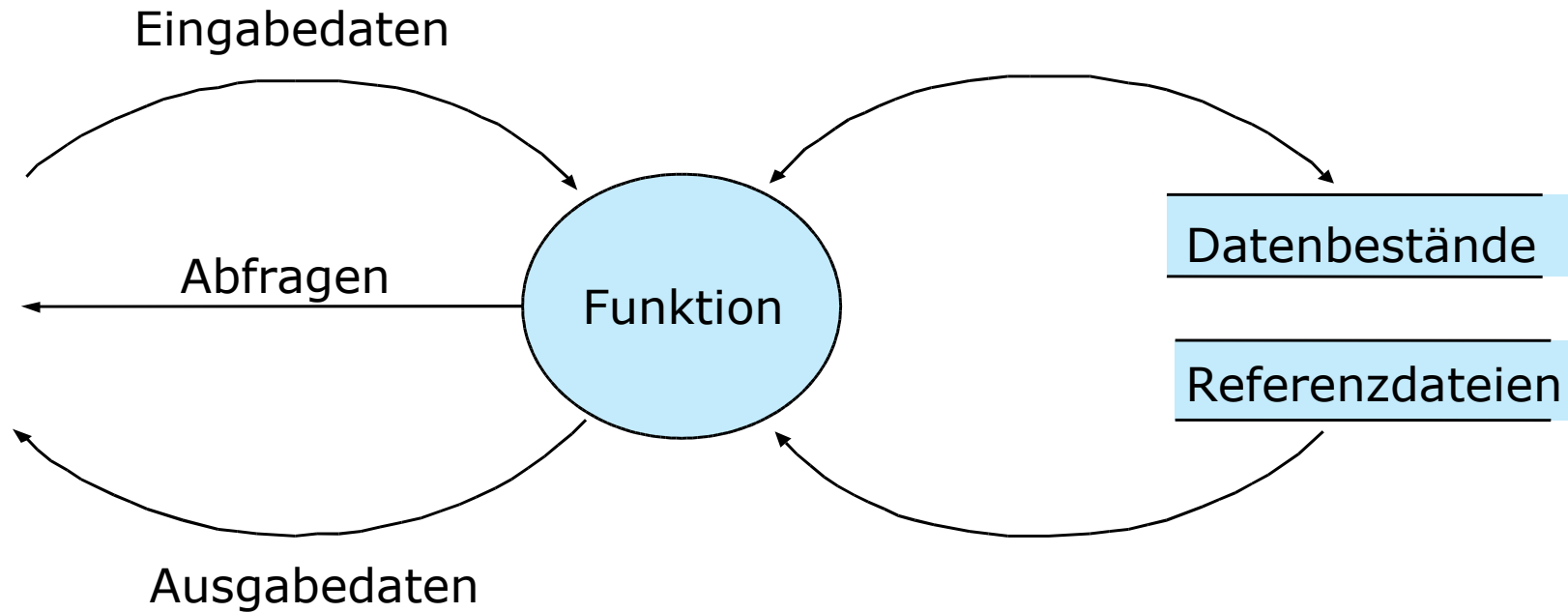
- Aus **abgeschlossenen Entwicklungen**
 - Wie hat sich der Aufwand auf die einzelnen Entwicklungsphasen verteilt
- Bei **neuen Entwicklungen**
 - Entweder man schließt eine Phase zunächst vollständig ab und ermittelt aus dem Ist-Aufwand den Soll-Aufwand für die restlichen Phasen
 - Oder man führt eine detaillierte Schätzung einer Phase durch und schließt hieraus dann auf den Gesamtaufwand
- Frühzeitig einsetzbar, wenn Aufwand für mindestens eine Phase durch den Einsatz einer anderen Methode bestimmt wurde.

Prozentsatzmethode

- Aufwandsverteilung für verschiedene Prozessmodelle



Function Point-Methode



Function Point-Methode

- Berechnungsformular zur Ermittlung der bewerteten Function Points

Kategorie	Anzahl	Klassifizierung	Gewichtung	Zeilensumme
Funktionen		einfach	x 7	=
		mittel	x 10	=
		komplex	x 15	=
Eingaben		einfach	x 4	=
		mittel	x 5	=
		komplex	x 7	=
Ausgaben		einfach	x 4	=
		mittel	x 5	=
		komplex	x 7	=
Datenbestände		einfach	x 5	=
		mittel	x 7	=
		komplex	x 10	=
Referenzdaten		einfach	x 5	=
		mittel	x 7	=
		komplex	x 10	=
Summe			E 1	=
Einflussfaktoren (ändern den Function-Point-Wert um $\pm 30\%$)		Produktleistungen (0..6)		=
		Qualitätsanforderungen (0..6)		=
		Architektur (0..6)		=
		GUI-Anforderungen (0..6)		=
		Werkzeugeinsatz (0..6)		=
		Nicht-funktionale Anforderungen (0..6)		=
		Erfahrungen (0..6)		=
		Schnittstellen (0..6)		=
		Prozessreife (0..6)		=
		Algorithmische Komplexität (0..6)		=
Summe der 10 Einflüsse		E 2		=
Faktor Einflussbewertung = $E2 / 100 + 0,7$		E 3		=
Bewertete Function Points $E1 * E3$				=

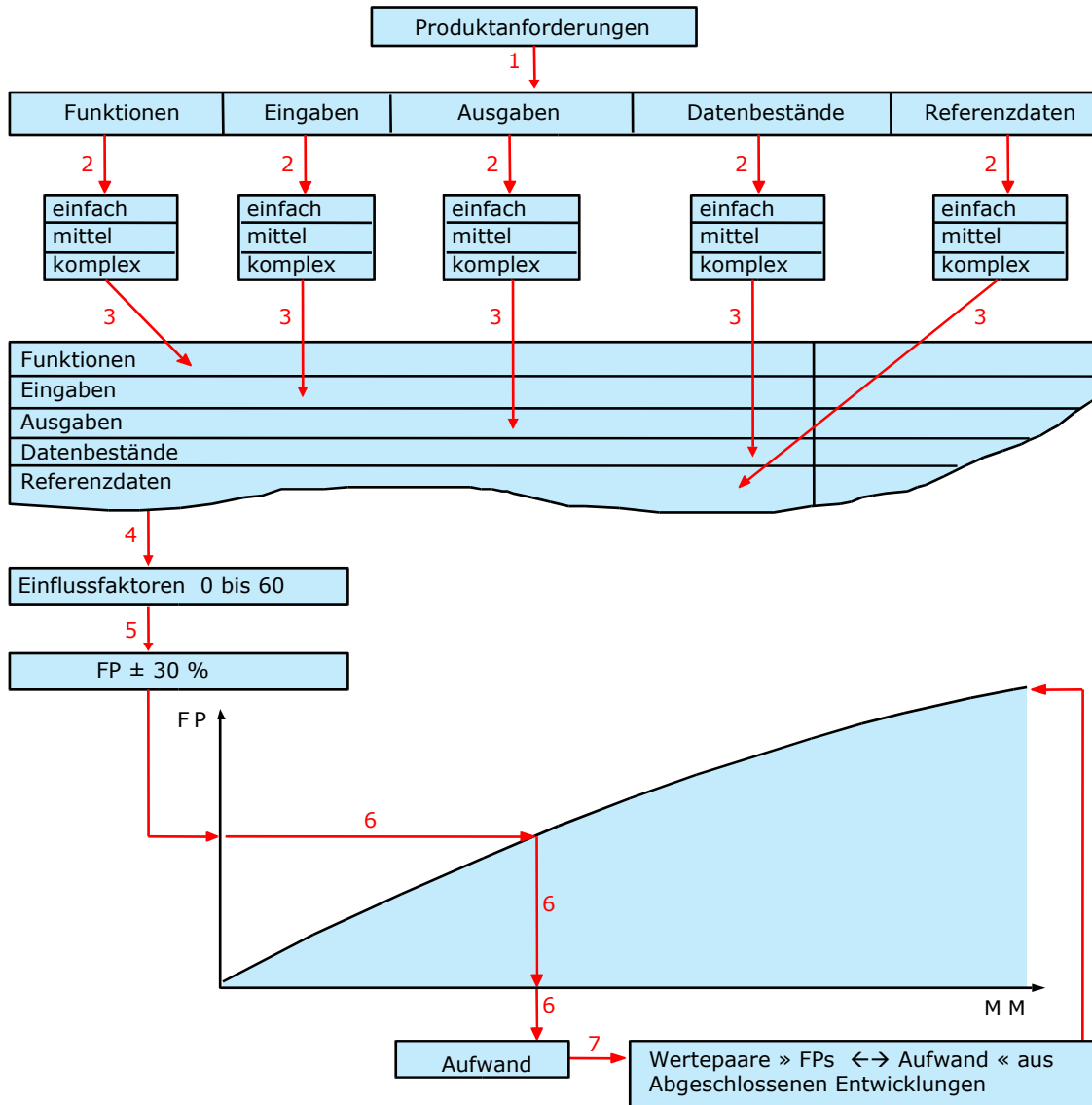
Quelle: IBM 85, S. 12

Function Point-Methode

- IBM-Tabelle

<i>Function P.</i>	IBM-MM	<i>Function P.</i>	IBM-MM	<i>Function P.</i>	IBM-MM
50	5	700	52	1700	142
100	8	750	56	1800	153
150	11	800	60	1900	164
200	14	850	64	2000	175
250	17	900	68	2100	188
300	20	950	72	2200	201
350	24	1000	76	2300	215
400	28	1100	85	2400	230
450	32	1200	94	2500	245
500	36	1300	103	2600	263
550	40	1400	112	2700	284
600	44	1500	122	2800	307
650	48	1600	132	2900	341.

Vorgehensweise bei der Function-Point-Methode



1. Schritt: Jede Anforderung einer Kategorie zuordnen

2. Schritt: Klassifizierung

- Jede Anforderung in eine der Klassen einfach, mittel oder komplex einordnen.

3. Schritt: Eintrag in Berechnungsformular

4. Schritt: Bewertung der Einflussfaktoren

5. Schritt: Berechnung der bewerteten Function Points

6. Schritt: Ablesen des Aufwands in MM

- Voraussetzung: Empirische Ermittlung der Zuordnung $FP \leftrightarrow MM$

7. Schritt: Aktualisierung der empirischen Daten.

Vorteile der Function Point-Methode

- Ausgangspunkt sind Produktanforderungen, nicht LOC;
- Anpassbar an verschiedene Anwendungsbereiche (Änderung der Kategorien);
- Anpassbar an neue Techniken (Änderung der Einflussfaktoren und der Einflussbewertung);
- Anpassbar an unternehmensspezifische Verhältnisse (Änderung der Einflussfaktoren, der Einflussbewertung und der Gewichtungsfaktoren);
- Verfeinerung der Schätzung entsprechend dem Entwicklungsfortschritt (iterative Methode);
- Erste Schätzung bereits zu einem frühen Zeitpunkt möglich (Planungsphase);
- Festgelegte methodische Schritte;
- Leicht erlernbar;
- Benötigt nur einen geringen Zeitaufwand;
- Gute Transparenz;
- Gute Schätzgenauigkeit;
- Werkzeugunterstützungen verfügbar.

Nachteile der Function Point-Methode

- Es kann nur der Gesamtaufwand geschätzt werden (Umrechnung auf einzelne Phasen muss mit der Prozentsatzmethode erfolgen);
- In der Originalform von Albrecht personalintensiv und nicht automatisierbar;
- Zu stark funktionsbezogen;
- Qualitäts-Anforderungen werden nicht berücksichtigt;
- Mischung von Projekt- und Produkteigenschaften bei den Einflussfaktoren;
- Ursprüngliche Einflussfaktoren heute überholt;
- Neigt zur Unterschätzung, da Anforderungen oft lückenhaft;
- Methodische Mängel.

Zusammenfassung Function Point-Methode

- Bei dieser Methode sind die Produktanforderungen aus Auftraggebersicht Ausgangspunkt für die Schätzung. In sieben Schritten werden aus den Produktanforderungen und dem Schwierigkeitsgrad des Produkts die Function Points abgeleitet.
- Besonderer Vorteil der Methode ist der methodische Rahmen, der eine Anpassung an die jeweilige Unternehmenssituation erlaubt.
- Da die ursprünglichen Kategorien und Einflussfaktoren der Function Point Methode heute nicht mehr zutreffen sind firmenspezifische Anpassungen unabdingbar.

Zusammenfassung

Um die ökonomische Durchführbarkeit eines Projektes zu prüfen, wird der Zeitaufwand mit einer Schätzmethode ermittelt. Der Aufwand wird oft in **LOC** gemessen. Der Arbeitsaufwand eines Mitarbeiters wird oft in Mitarbeitermonate (**MM**) (auch Personenmonate genannt) angegeben. Eine Aufwandschätzung kann m.H. verschiedener Basismethoden (Relationsmethode, Multiplikatormethode, ... , Prozentsatzmethode) erfolgen.

Untersuchungen haben folgende **Faustregeln** eingeführt :

- Eine durchschnittliche Software-Entwicklung liefert ungefähr **350 LOC** (ohne Kommentare) pro Ingenieurmonat (Aufwand von der Definition bis zur Implementierung).
- Entwicklung die in erster Linie vorhandene Software wiederverwenden benötigen ungefähr $\frac{1}{4}$ der Zeit und der Ressourcen von Neuentwicklungen.
- Der Aufwand verteilt sich folgendermaßen:
 - Management 5 – 10%
 - Definition 5 – 10%
 - Entwurf 10 – 15%
 - Implementierung 20 – 30%
 - Überprüfung 20 – 40%
 - Installation 5 – 15%
 - Umgebung/Werkzeuge 5 – 10%

Literatur

- Balzert H., Lehrbuch der Softwaretechnik, 2. Auflage, Spektrum Akademischer Verlag, Heidelberg, 2000.
-
- ANSI/IEEE Std. 729-1983, IEEE Standard Glossary of Software Engineering Terminology, IEEE Inc., New York, 1983.
 - Balzert H., Lehrbuch Grundlagen der Informatik, Spektrum Akademischer Verlag, Heidelberg, 1999.
 - Böhm B.W., Software Engineering in IEEE Transactions on Computers S.1226-1241, Dezember 1976.
 - Böhm B.W., Improving Software Productivity in Computer S. 43-57, Sept. 1987.
 - Ludewig J., Softwaretechnik in Stuttgart – ein konstruktiver Informatikstudiengang in Informatik-Spektrum, Februar 1999.
 - Schneider H.-J., Lexikon der Informatik und Datenverarbeitung. Oldenburg Verlag, München, 1986.