

Vorlesung "Software-Engineering"

Prof. Ralf Möller, TUHH, Arbeitsbereich STS

■ Vorige Vorlesung

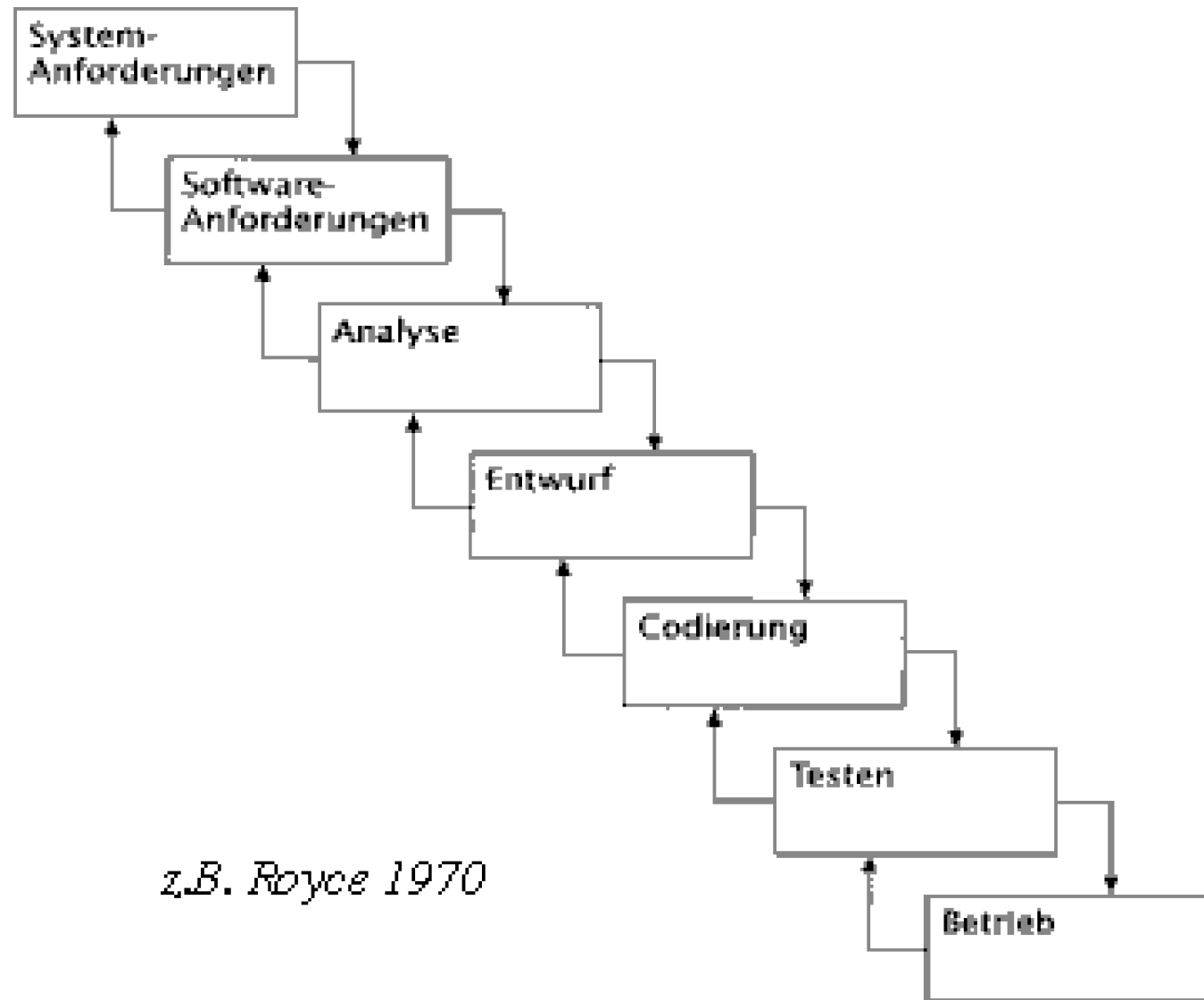
- Qualitätsmerkmale
- Produkte und Leistungen
- Projektphasen und Vorgehensmodelle

■ Heute

- Fortsetzung Projektphasen und Vorgehensmodelle
- Lastenheft
- Beginn: Verfahren zur Aufwandsschätzung

Vorgehensmodelle (Wdhlg.)

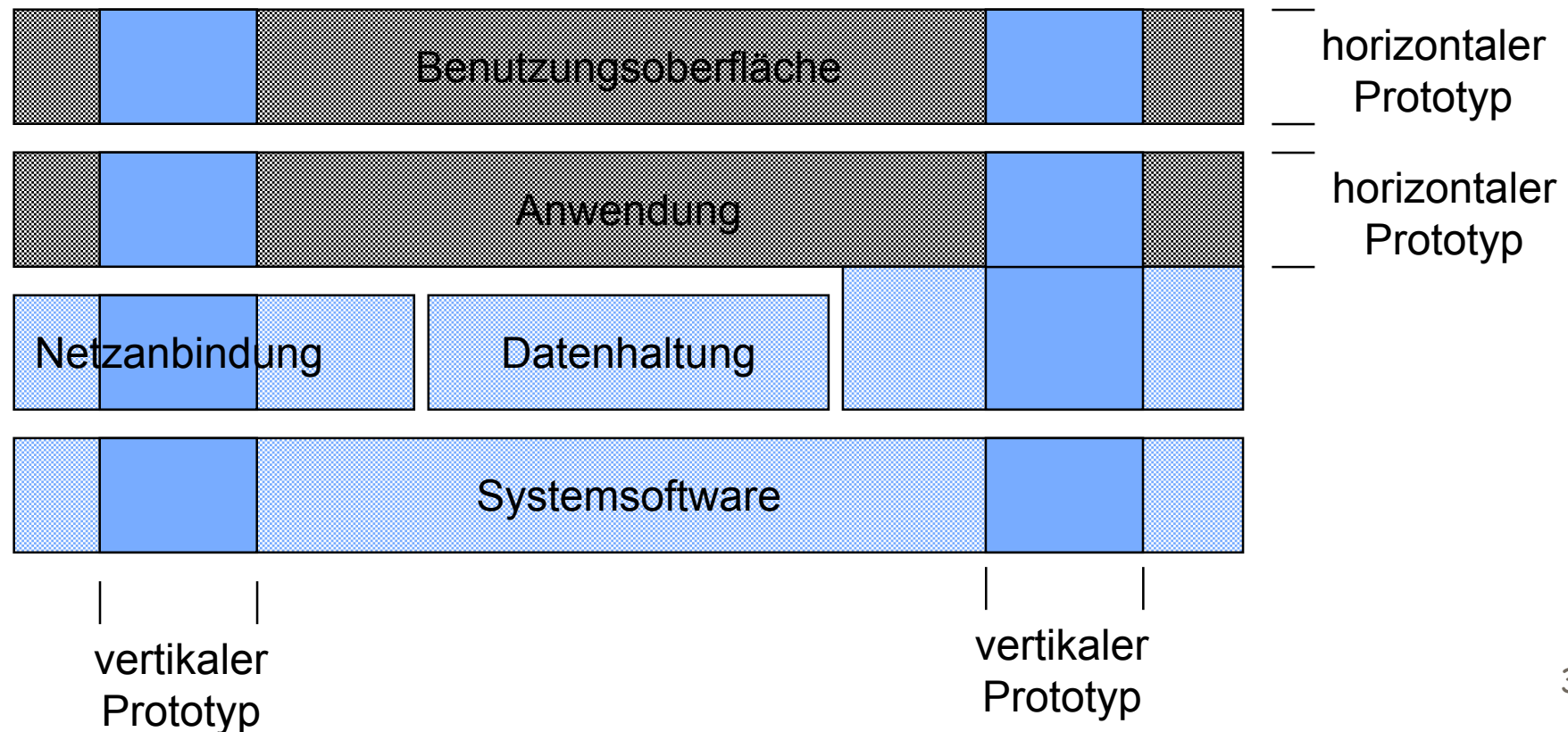
■ Wasserfall-Modell



z.B. Royce 1970

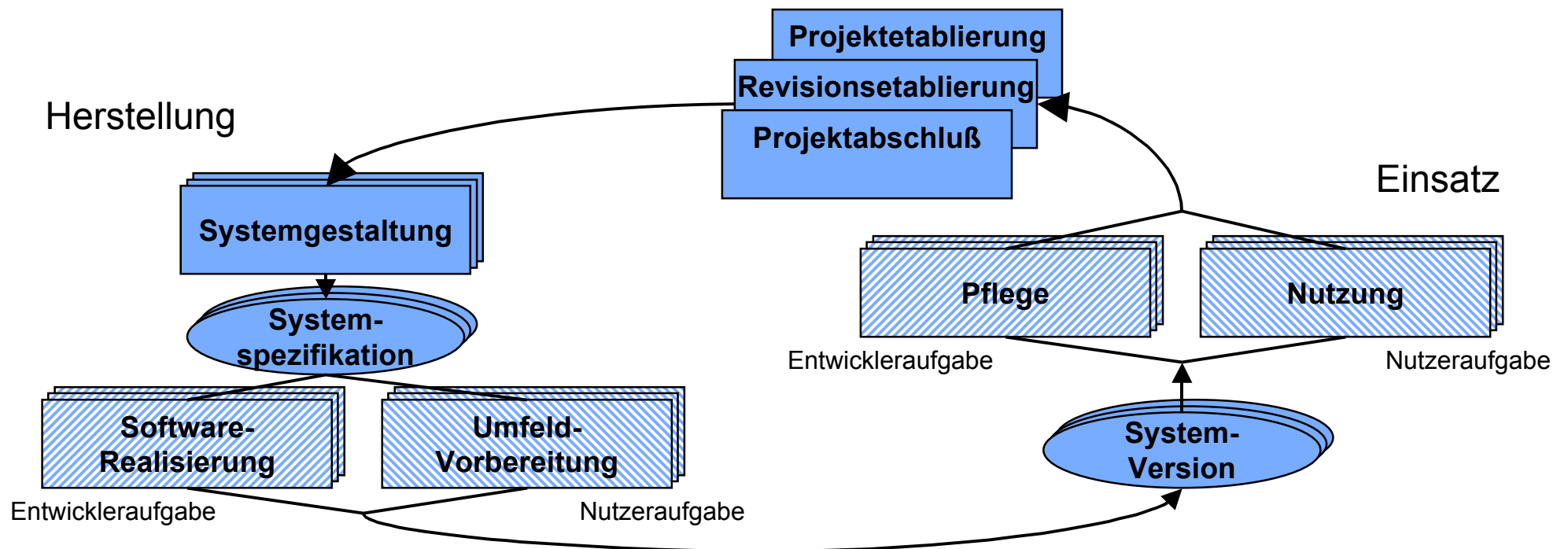
Das Prototypen-Modell (Wdhlg.)

- Ein fertiges Software-Produkt besteht aus vielen Komponenten und Ebenen.
- Unterscheidung zwischen **horizontalen** und **vertikalen Prototypen**:



Das evolutionäre/inkrementelle Modell

- **Beobachtung:** Software-(Weiter) Entwicklung unterliegt Änderungen
 - Lernen zwischen Entwicklern und Anwendern nötig, da
 - Veränderungen im technischen und Einsatzkontext stattfinden
 - sich durch den Einsatz des Systems neue Anforderungen ergeben
- Systementwicklung in Ausbaustufen, inkrementelle Entwicklung, Prototyping



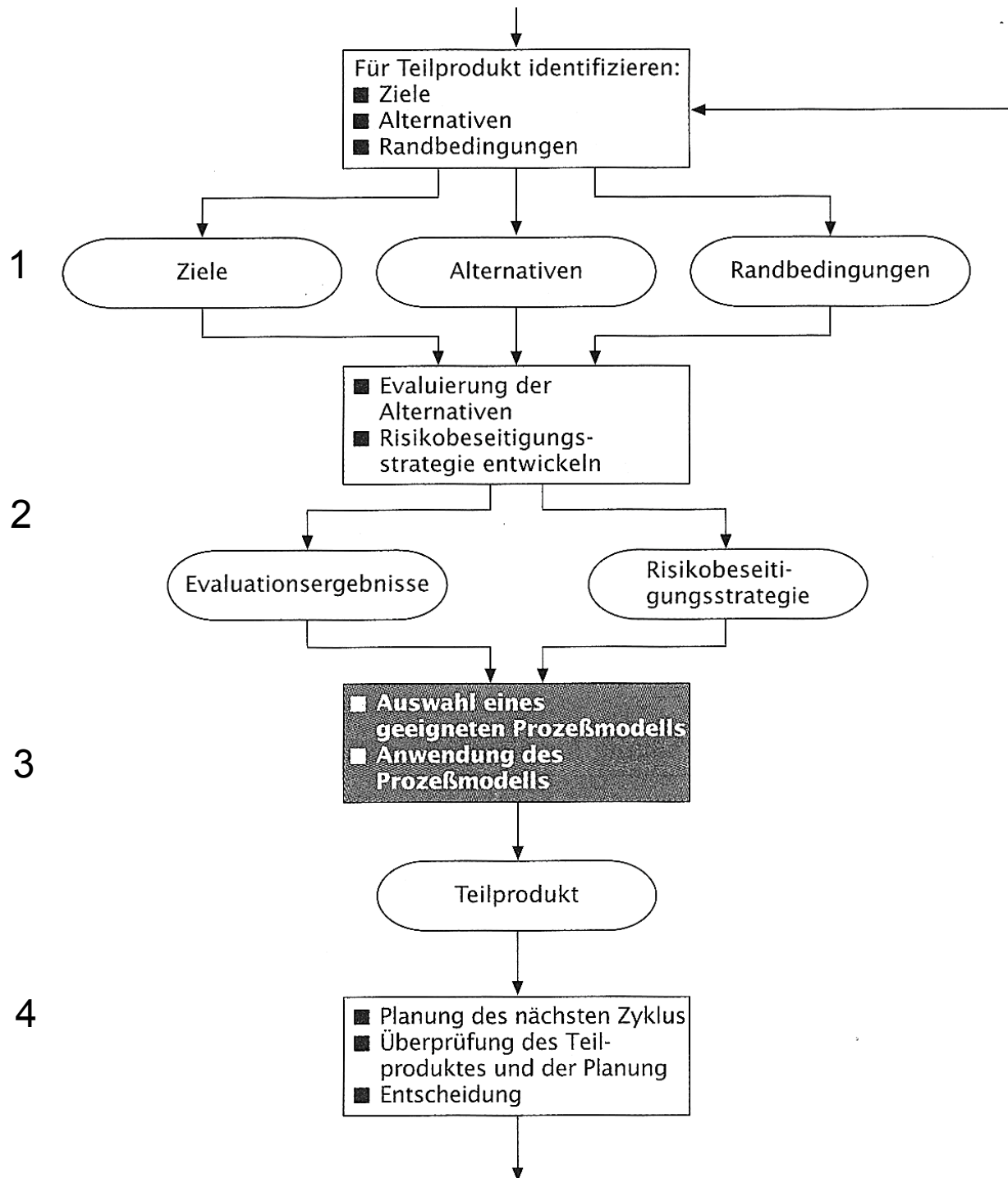
Risiken bei der Software-Entwicklung

Risk	Risk type	Description
Staff turnover	Project	Experienced staff will leave the project before it is finished.
Management change	Project	There will be a change of organisational management with different priorities.
Hardware unavailability	Project	Hardware which is essential for the project will not be delivered on schedule.
Requirements change	Project and product	There will be a larger number of changes to the requirements than anticipated.
Specification delays	Project and product	Specifications of essential interfaces are not available on schedule
Size underestimate	Project and product	The size of the system has been underestimated.
CASE tool under-performance	Product	CASE tools which support the project do not perform as anticipated
Technology change	Business	The underlying technology on which the system is built is superseded by new technology.
Product competition	Business	A competitive product is marketed before the system is completed.

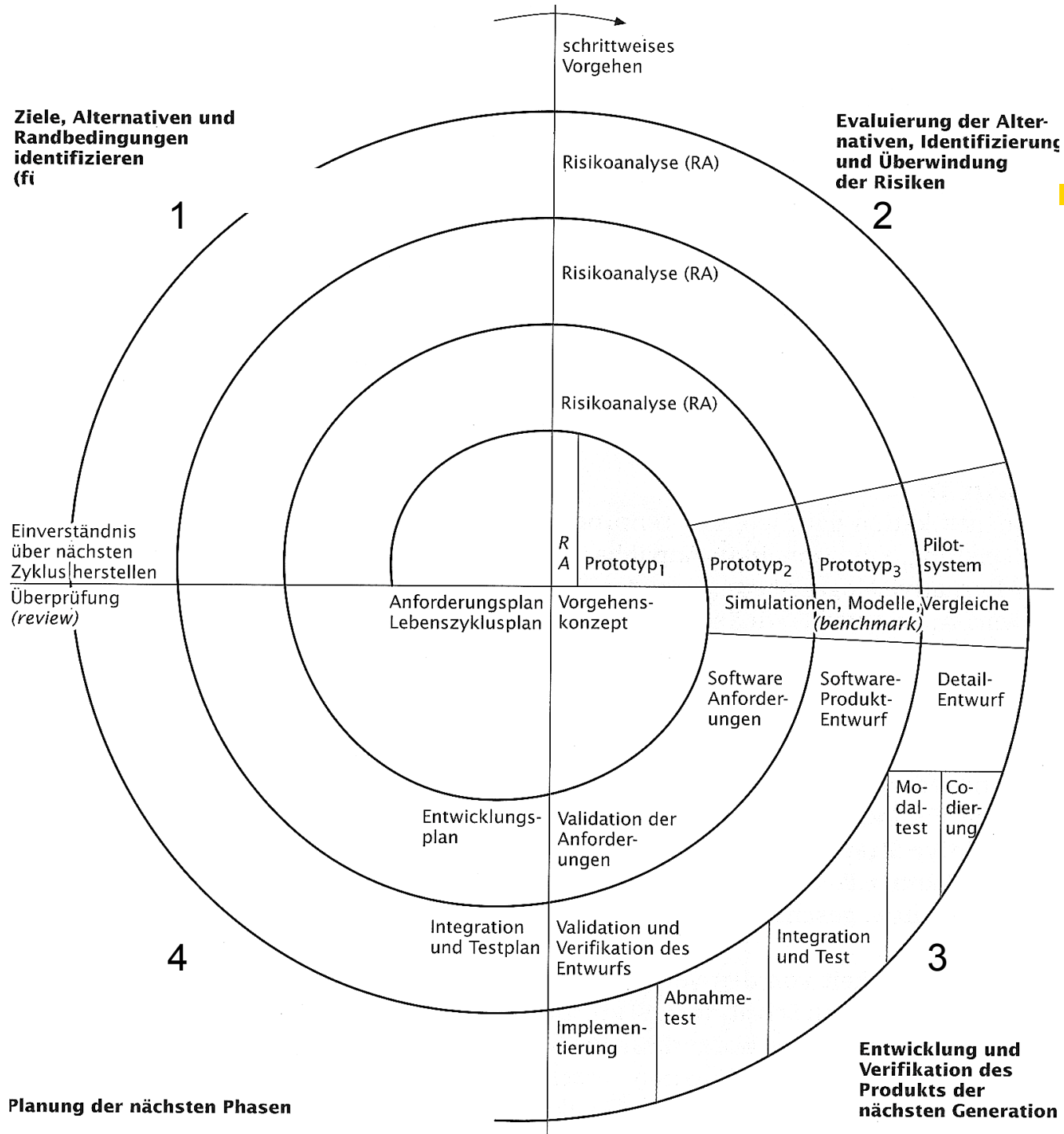
Das Spiralmodell (1)

- Für jedes (Teil-)Produkt sind zyklisch vier Schritte zu durchlaufen:
- **Schritt 1:**
 - Identifizierung der Ziele des Teilprodukts (Leistung, Funktionalität, Anpaßbarkeit, ...)
 - Alternative Möglichkeiten zur Realisierung des Teilprodukts finden.
 - Randbedingungen bei verschiedenen Alternativen finden
- **Schritt 2:**
 - Evaluierung der Alternativen unter Berücksichtigung aller Alternativen
 - Identifizieren und ggf. Überwinden von Risiken (durch Prototypen, Simulation, ...)
- **Schritt 3:**
 - Abhängig vom Risiko wird ein Prozeßmodell festgelegt (oder eine Kombination).
 - Anwendung des Modells
- **Schritt 4:**
 - Planung des nächsten Zyklus, Überprüfung der nächsten 3 Schritte im nächsten Zyklus, Einverständnis mit Beteiligten sichern.
- Das Spiralmodell ist eigentlich ein **Modell höherer Ordnung**

Das Spiralmodell (2)



Das Spiralmodell (3)



Das Spiralmodell (4)

■ Eigenschaften

- **Risikogetriebenes** Modell, da Hauptziel die Minimierung des Risikos ist.
- Ziel: Beginne im Kleinen, halte die Spirale so eng wie möglich und erreiche das Ziel mit minimalen Kosten.

■ Vorteile:

- Periodische Überprüfung und ggf. Neufestlegung des Prozeßmodells
- Prozeßmodell ist nicht für die gesamte Dauer des Projekts festgelegt.
- Flexibel, leichtere Umsteuerung
- Erleichtert Wiederverwendung von Software durch Betrachtung von Alternativen.

■ Nachteile:

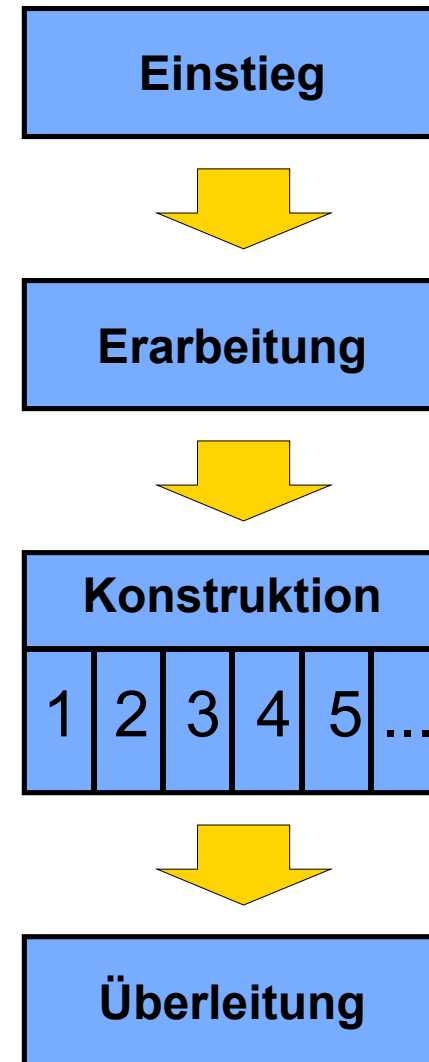
- Hoher Managementaufwand
- Für kleine und mittlere Projekte weniger gut geeignet.
- Wissen über Identifizierung und Management von Risiken ist noch nicht sehr verbreitet.

OO-Modell: *Unified Process* (1)

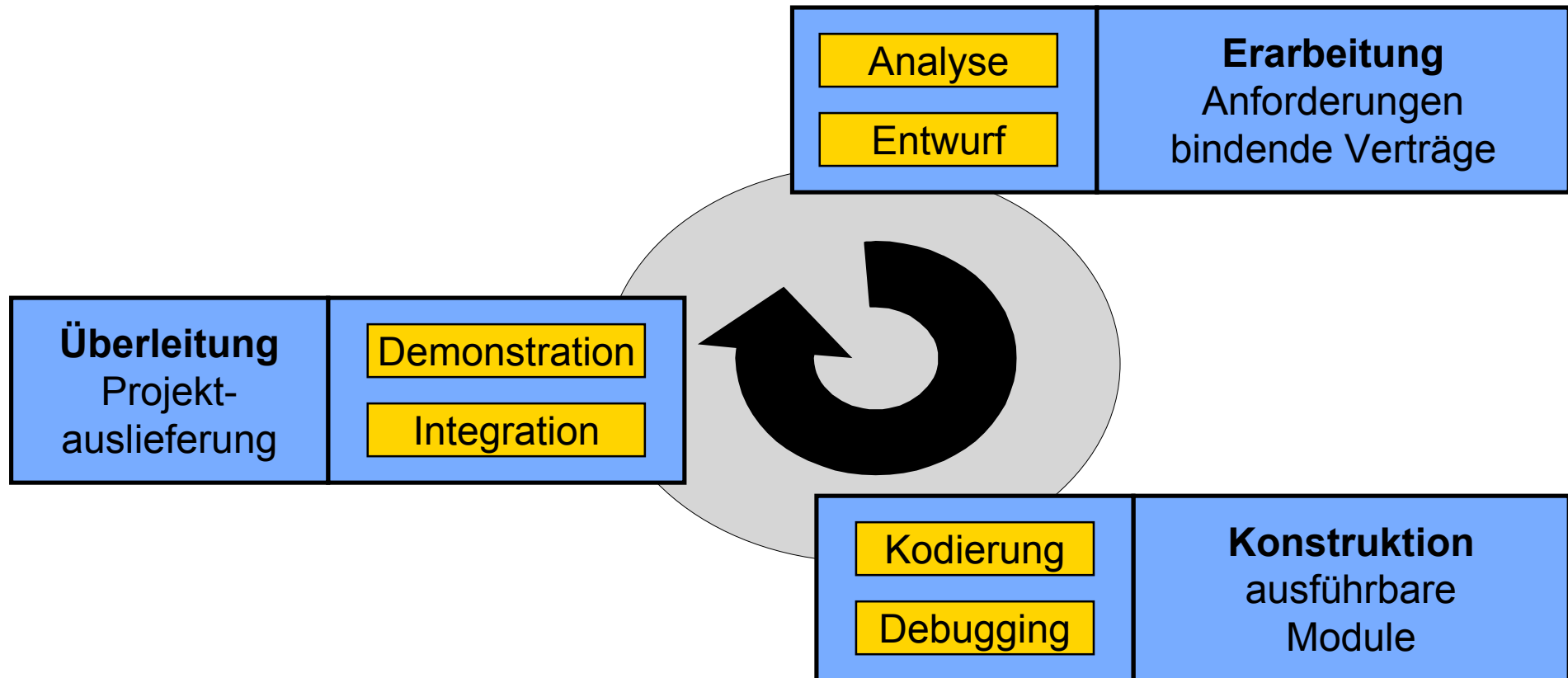
■ **Unified Process:**

Der UML-Software-Entwicklungsprozeß:

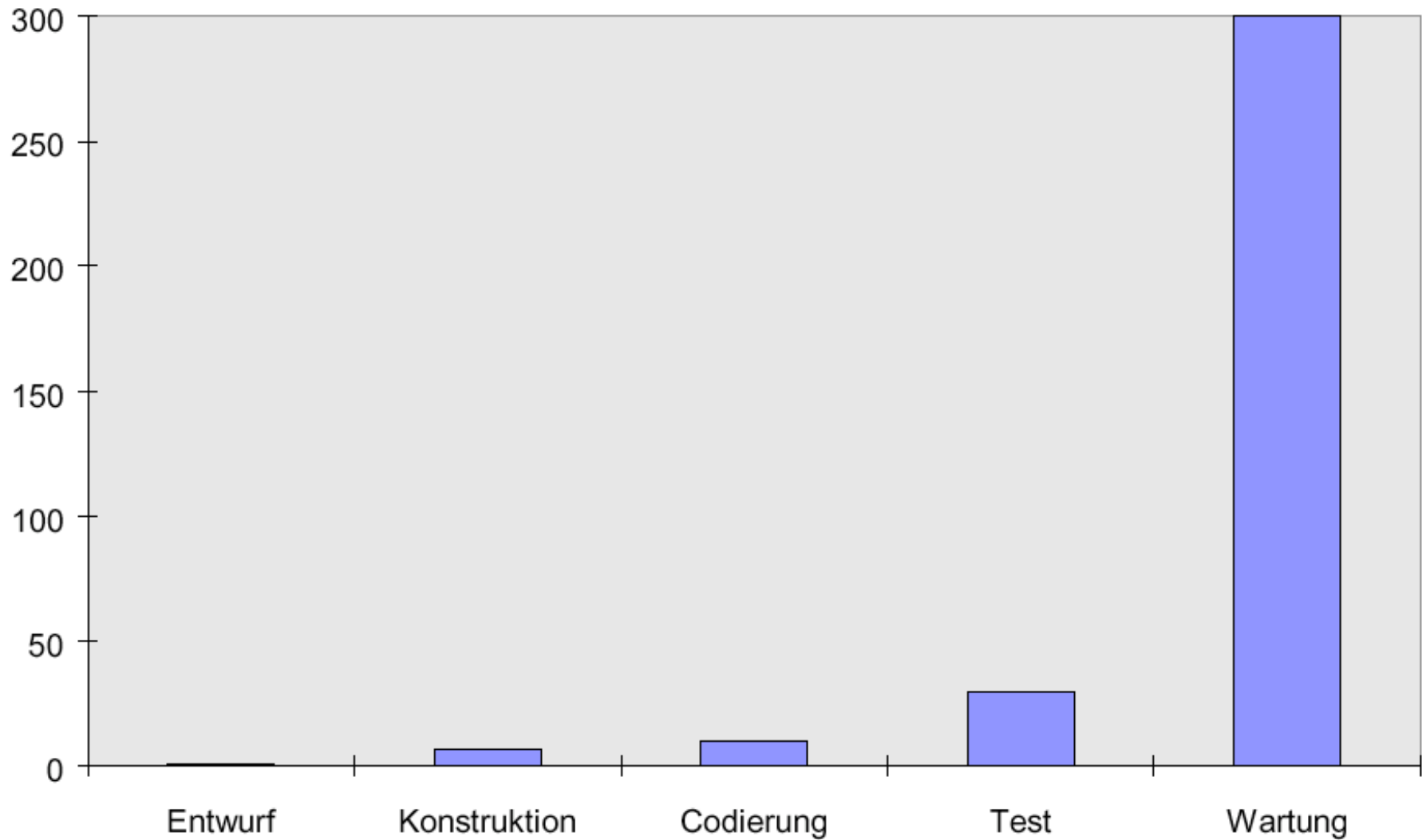
- Der **Einstieg** etabliert das Geschäftsziel und legt den Umfang des Projektes fest.
- In der **Erarbeitungsphase** werden detaillierte **Anforderungen gesammelt**, **Analyse** betrieben und **Entwurf** grundsätzliche Architekturentscheidungen getroffen sowie der **Plan** für die Konstruktion gemacht.
(use case diagrams)
- Die Konstruktion ist ein **iterativer** und **inkrementeller** Prozeß. Jede Iteration dieser Phase baut Software-**Prototypen** mit Produktqualität, die getestet werden und einen Teil der Anforderungen des Projekts umsetzen.
(use-case driven)
- Die Überleitungsphase enthält den **Beta-Test**, **Leistungssteigerung** und Benutzer-**Training**.



OO-Modell: *Unified Process* (2)



Wdhlg.: Zeitaufwand je nach Entwicklungsphase



Konsequenz

- Erfahrungsgemäß hohe Kosten bei Änderungen in späten Phasen rechtfertigen hohen Aufwand in frühen Phasen zur Vermeidung von späteren Änderungen
- Einfluß auf vorgeschlagene „klassische“ Vorgehensmodelle
- Aber: Kostenreduktion durch aufwendiges Vorgehen in frühen Phasen umstritten
- These: Änderungen sind kaum vermeidbar
- Durch neue Vorgehensweisen soll Änderungsflexibilität erhalten bleiben

Neuere Entwicklungen

- Extreme Programming
- Agile Modeling
- Software Product Lines
- Component-Oriented Software Engineering
- Model-Driven Architecture
- ...

Wir greifen diese Themen etwas später wieder auf.

Produktplanung (1)

■ **Produktauswahl**

Trendstudien, Marktanalysen, Forschungsergebnisse, Kundenanfragen, Vorentwicklungen, ...

■ **Voruntersuchung des Produkts**

- u.U. gezielte Ist-Aufnahme, wenn bereits Vorgängerprodukt vorhanden;
anschließend Ist-Analyse
- Festlegen der Hauptanforderungen
 - | Festlegen der Hauptfunktionen
 - | Festlegen der Hauptdaten
 - | Festlegen der Hauptleistungen
 - | Festlegen der wichtigsten Aspekte der Benutzungsschnittstelle
 - | Festlegen der wichtigsten Qualitätsmerkmale.

■ **Durchführbarkeitsuntersuchung**

- Prüfen der fachlichen Durchführbarkeit (softwaretechnische Realisierbarkeit, Verfügbarkeit von Entwicklungs- und Zielmaschinen, ...)
- Prüfen alternativer Lösungsvorschläge (Beispiel: Kauf und Anpassung von Standardsoftware vs. Individualentwicklung)
- Prüfen der personellen Durchführbarkeit: Verfügbarkeit qualifizierter Fachkräfte für die Entwicklung
- Prüfen der Risiken

■ **Prüfen der ökonomischen Durchführbarkeit**

- Aufwands- und Terminschätzung
- Wirtschaftlichkeitsrechnung

Problemanalyse und Planung

- Analyse des Ist-Zustandes (Aufgabenbereiche)
- Systemabgrenzung
 - Festlegung, welche Teile zum System gehören und damit Gegenstand der weiteren Untersuchung sind
 - Ermittlung der Umgebungsbedingungen des Systems (Schnittstellen)
- Systemerhebung
 - Sammeln und Strukturieren von Informationen über das System und seine Eigenschaften (insbes. Anforderungen u. Änderungswünsche)

Erhebungstechniken

- Interview-Technik
 - Direkte Befragung der Benutzer/Auftraggeber durch den Analytiker
- Schriftliche Befragung
 - Verteilen, Einsammeln und Auswerten von Fragebögen
- Beobachtung
 - Erfassung von Fakten durch den Analytiker ohne direkten Kontakt mit dem beobachteten Aufgabenträger/Arbeitsprozeß
- Berichte
 - Schriftliche Darstellung von Tätigkeitsbereichen

Gliederung der Systemerhebung

- **Strukturanalyse**
- Aufgaben- / Ablaufanalyse
- Kommunikationsanalyse
- Dokumentenanalyse
- Datenanalyse

- Schwachstellenanalyse

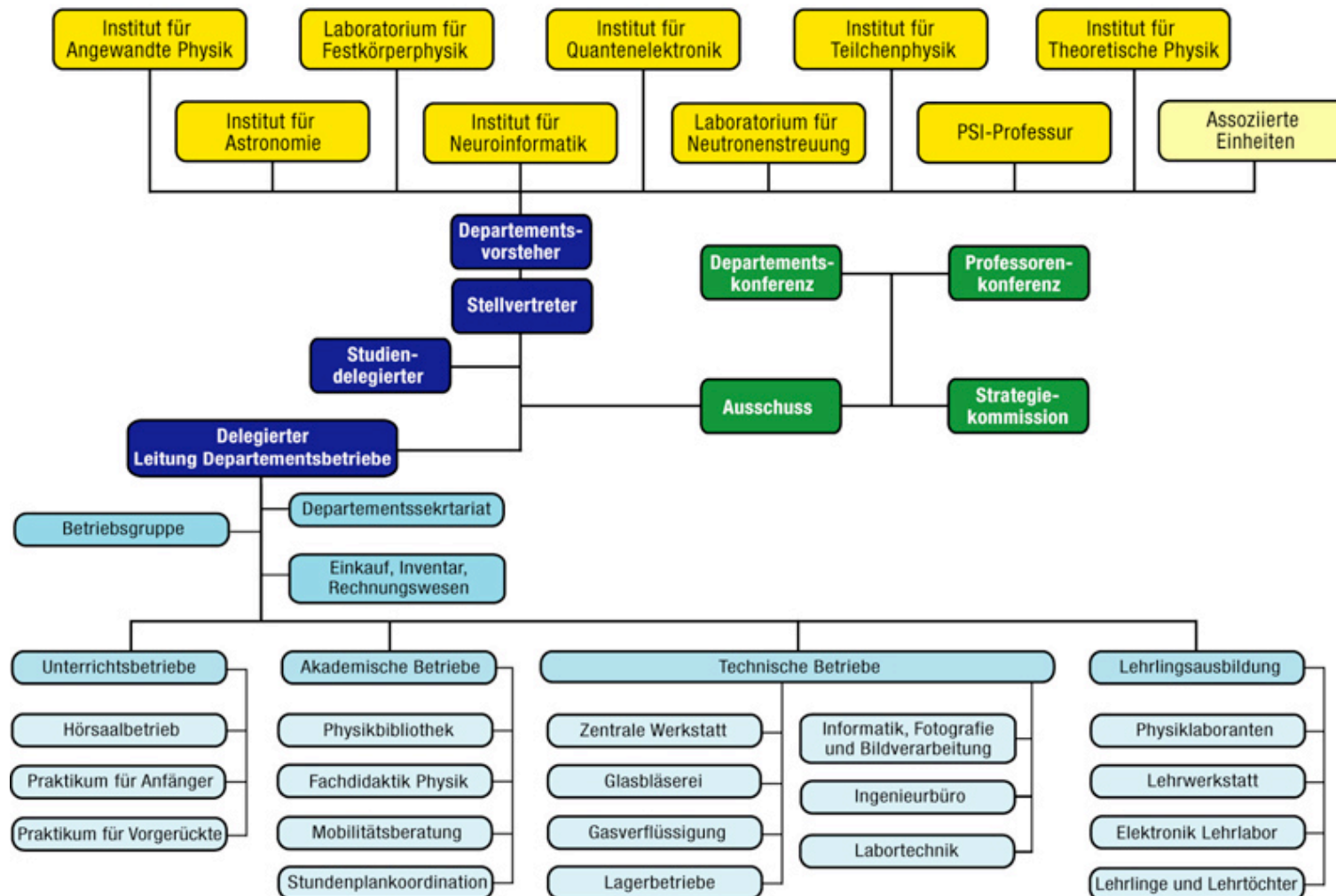
(nach Pomberger/Blaschek)

Strukturanalyse (auch Organisationsanalyse)

- Erfassung und Darstellung des organisatorischen Aufbaus und der damit verbundenen Regelungen (Verantwortlichkeiten, Kompetenzen)
- Nachweis der einzelnen bearbeitenden Stellen
- Erfassung des logistischen Zusammenhangs von Aufgaben in der Organisationsstruktur
- Arbeits- und Darstellungsmittel:
 - Organigramme
 - Stellenbeschreibungen
 - Stellenbezogene Prozeßablaufdiagramme

Organigramm – Beispiel

ORGANISATION des Departements Physik der ETH Zürich



Bereiche der Systemerhebung

- Strukturanalyse
- Aufgaben- / Ablaufanalyse
- Kommunikationsanalyse
- Dokumentenanalyse
- Datenanalyse
- Ablaufanalyse
- Schwachstellenanalyse

Aufgabenanalyse (auch Prozeßanalyse)

- Erfassung und Darstellung der anfallenden Operationen/Prozesse zur Erledigung von Aufgaben und ihrer internen Charakteristika
- Für jede Operation Angaben zu
 - benutzten Daten
 - produzierten Daten
 - Ablauf der Operation (Verarbeitungsalgorithmus)
- Beispiele für Arbeitstechniken und Hilfsmittel:
 - Einfache Ablaufpläne/Struktogramme
 - Black-Box-Analysen
 - Entscheidungstabellen
 - ...

Bereiche der Systemerhebung

- Strukturanalyse
- Aufgaben- / Ablaufanalyse
- **Kommunikationsanalyse**
- Dokumentenanalyse
- Datenanalyse

- Schwachstellenanalyse

Kommunikationsanalysen (1)

- Gegenstand u. Ziele
 - Darstellung der Austauschbeziehungen von Informationen und Daten zwischen Elementen der organisatorischen Struktur (Aufgaben, Aufgabenträgern)
 - Quantifizierung der Kommunikation (Volumen, Zeiten, Aufwand/Kosten)
 - Ermittlung qualitativer Merkmale der Kommunikation (Sicherheit, Rechtzeitigkeit etc.)
- Unterscheidbare Elemente der Kommunikation:
 - Kommunikationspartner
 - Kommunikationsträger
 - Kommunikationskanäle

Kommunikationsanalysen (2)

- Verschiedene Hilfsmittel zur Erfassung und Darstellung der Kommunikationsbeziehungen, z.B.:
 - Graphische Kommunikationsnetze
 - Kommunikationsdiagramme (Kommunikationsmatrix)

	S1	S2	Sm
E1	K11	K12	K1m
E2	K21	K22	K2m
E3	K31	K32	K3m
...
En	Kn1	Kn2	Knm

E: Empfänger

S: Sender

K: Kenndaten der Kommunikationsbeziehung

Bereiche der Systemerhebung

- Strukturanalyse
- Aufgaben- / Ablaufanalyse
- Kommunikationsanalyse
- Dokumentenanalyse
- Datenanalyse

- Schwachstellenanalyse

Dokumentenanalyse

- Erhebung aller im untersuchten System verwendeten und produzierten Dokumente
- Grundlage für die Gestaltung von Ein-/Ausgabemasken oder -formularen
- Inhalt der Dokumentenbeschreibung:
 - Bezeichnung
 - Inhalt
 - Zweck
 - Grad der Formalisierung
 - Verteiler
 - Archivierung

Datenanalyse (1)

- Ziel: Klarheit über Art und Umfang der zu verarbeitenden Daten gewinnen
- Teilbereiche:
 - Formale, einheitliche Darstellung aller Datenbestände (Struktur, Wertebereiche)
 - Erfassung der Verarbeitungscharakteristika der Daten (Speicherungsformen, Zugriffsarten, Sicherheitsbedingungen, Datenträger)

Datenanalyse (2)

- Erfassung des Datenvolumens (Einzeldaten und Dateien, Wachstum) und Prognose
- Häufigkeit und Art der Verarbeitung (Nutzung) und Änderung (Transaktionsanalyse)
- Abhängigkeiten zwischen den Daten (Relationen, Konsistenzbedingungen, Reihenfolge der Erstellung etc.)

Ablaufanalyse

- Erfassung der Reihenfolge der Operationen und des
- Informationsflusses zwischen diesen ("Informationsflußanalyse")
- Keine Beschreibung der Operation und der ausgetauschten Daten
- Zahlreiche graphische Notationen zur Darstellung

Schwachstellenanalyse

- Ermittlung von systematischen Mängeln im System
- Basis: andere Ergebnisse der Ist-Analyse
- Systematische Mängel: Lücken, Redundanzen, Abweichungen von Plan- und Vergleichswerten
- Indikatoren für Schwachstellen:
 - Lange Durchlaufzeiten bei Prozessen
 - Hohe Lagerbestände
 - Abteilungsweise Datenhaltung,
 - Medienbrüche bei der Datenübermittlung
 - Bezugsgrößen: Planwerte oder Vergleichswerte aus Literatur, früheren Erhebungen, von anderen Unternehmungen

Lastenheft: Produktanforderungen

Pflichtenheft: Systemanforderungen

- **Aufgabe:** Zusammenfassung aller fachlichen Basisanforderungen aus Sicht des Auftraggebers
- **Adressat:** Auftraggeber sowie Auftragnehmer (Projektleiter, Marketing, ...)
- **Inhalt:** Basisanforderungen („*Was?*“, nicht „*Wie?*“)
- **Form:** standardisiertes, numeriertes Gliederungsschema (s. Beispiel)
- **Sprache:** verbale Beschreibung
- **Umfang:** wenige Seiten

Beispiel für ein Lastenheft: Seminarorganisation (1)

Version	Autor	QS	Datum	Status	Kommentar
2.1	Schmidt	Hupe	2/03	akzeptiert	
2.2	Schmidt	Hupe	3/03	akzeptiert	/LF40/ gelöscht

Versions-
historie

■ 1 Zielbestimmung

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

informell

■ 2 Produkteinsatz

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

informell

Beispiel für ein Lastenheft: Seminarorganisation (2)

■ Produktfunktionen

■ /LF10/

Ersterfassung, Änderung und Löschung von Kunden (Teilnehmer, Interessenten)

■ /LF20/

Benachrichtigung der Kunden (Anmeldebestätigung, Abmeldebestätigung, Änderungsmitteilungen, Rechnung, Werbung)

■ /LF30/

Ersterfassung, Änderung und Löschung von Seminarveranstaltungen und Seminartypen

...

■ /LF70/

Erstellung verschiedener Listen
(Teilnehmerliste, Umsatzliste, Teilnehmerbescheinigungen)

■ /LF80/

Anfragen der folgenden Art sollen möglich sein: Wann findet das nächste Seminar X statt? Welche Mitarbeiter der Firma Y haben das Seminar X besucht?

*Label /LF.../ zur Referenzierung von **Funktionen***

Beispiel für ein Lastenheft: Seminarorganisation (3)

*Label /LD.../ zur
Referenzierung
von **Daten***

■ 4 Produktdaten

■ /LD10/

Es sind relevante Daten über die Kunden zu speichern.

■ /LD20/

Falls ein Kunde zu einer Firma gehört, dann sind relevante Daten über die Firma zu speichern.

■ /LD30/

Es sind relevante Daten über Seminarveranstaltungen, Seminartypen und Dozenten zu speichern.

■ /LD40/

Bucht ein Kunde eine Seminarveranstaltung, dann sind entsprechende Buchungsdaten zu speichern.

Beispiel für ein Lastenheft: Seminarorganisation (4)

■ 5 Produktleistungen

■ /LL10/

Die Funktion /LF80/ darf nicht länger als 15 Sekunden Interaktionszeit benötigen, alle anderen Reaktionszeiten müssen unter 2 Sekunden liegen.

■ /LL20/

Es müssen maximal 50.000 Teilnehmer und maximal 10.000 Seminare verwaltet werden können.

*Label /LL.../ zur Referenzierung von **Leistungen***

Beispiel für ein Lastenheft: Seminarorganisation (5)

■ 6 Qualitätsanforderungen

Produktqualität	sehr gut	gut	normal	irrelevant
Funktionalität		x		
Zuverlässigkeit			x	
Benutzbarkeit		x		
Effizienz		x		
Änderbarkeit			x	
Übertragbarkeit			x	

■ 7 Ergänzungen

[keine]

Aufwandsschätzung

- Sicht des Software-Herstellers bzw. des Auftragnehmers:
- Kosten eines Software-Systems: **Entwicklungskosten**
 - Hauptanteil der Entwicklungskosten: **Personalkosten**

Faustregel:

Personalkosten: 50 Tsd EUR / Jahr pro Mitarbeiter

Verrechnungspreise: 100 – 150 Tsd EUR / Jahr pro Mitarbeiter

- Anteilige Umlegung der CASE-Umgebungskosten (einschließlich Hardware und Systemsoftware) für die Produktentwicklung
- Kosten für andere Dienstleistungen, Büromaterial, Druckkosten, Dokumentation, Reisekosten usw. sind im Verhältnis zu den Personalkosten eher gering.

Methoden zur Kosten- und Terminschätzung

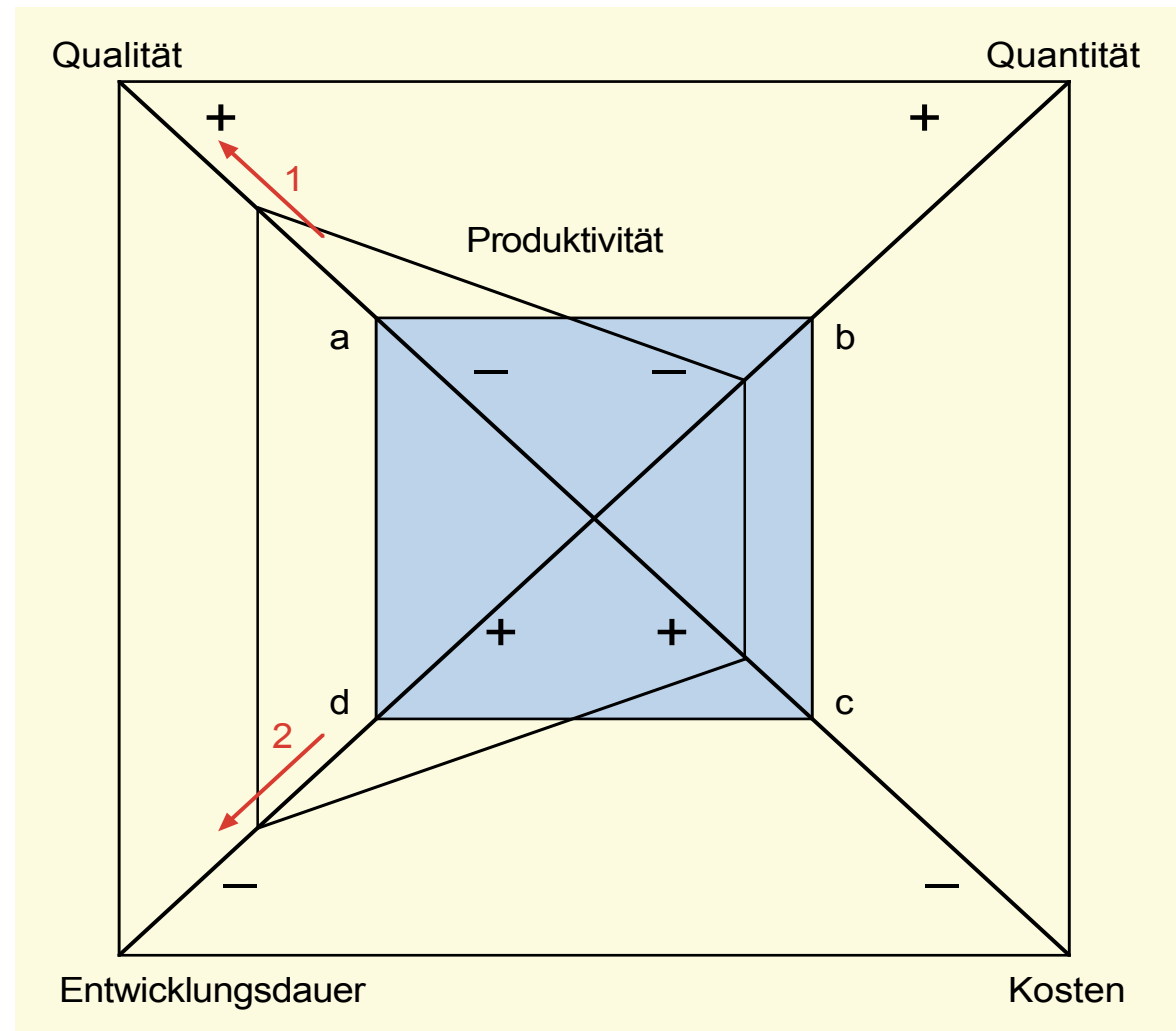
- Die meisten Modelle basieren auf dem geschätzten Umfang des zu erstellenden Software-Produktes in „**Anzahl der Programmzeilen**“ bzw. in ***Lines of Code (LOC)***.
 - Bei höheren Sprachen werden z.B. alle Vereinbarungs- und Anweisungszeilen geschätzt.
 - Der geschätzte Umfang wird durch einen Erfahrungswert für die Programmierproduktivität (in LOC) eines Mitarbeiters pro Jahr oder Monat geteilt.
 - Ergebnis: geschätzter Aufwand in **Personenjahren (PJ, auch MJ)** oder **Personenmonaten (PM, auch MM)**
 - $1 \text{ PJ} = 9 \text{ PM}$ oder 10 PM (Urlaub, Krankheit, Schulung, ...)
 - Der so ermittelte Aufwand wird durch die nach der Terminvorgabe zur Verfügung stehende Entwicklungszeit geteilt.
 - Ergebnis: Anzahl der einzusetzenden, parallel arbeitenden Mitarbeiter.

Einflussfaktoren der Aufwandsschätzung (1)

- Quantität
- Qualität
- Entwicklungsdauer
- Kosten

bedingen einander

➔ **Teufelsquadrat**



Einflussfaktoren der Aufwandsschätzung (2)

■ Quantität

■ Größe des Programmtextes

in Planungsphase unbekannt

- | Maß „Anzahl Programmzeilen“ (LOC)
- | lineare oder überproportionale Beziehung zwischen LOC und dem Aufwand

■ Funktions- und Datenumfang

- | Maß unabhängig von einer Programmiersprache

früh bekannt

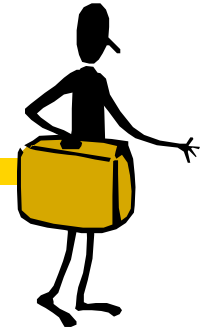
■ evtl. zusätzliche Gewichtung mit **Komplexität**

- | qualitative Maße, z.B. „leicht“, „mittel“ und „schwer“
- | Abbildung auf Zahlenreihe. Beispiel: Noten zwischen 1 und 6.

■ Qualität

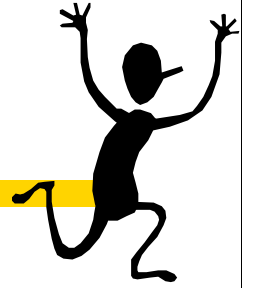
- | Je höher die Qualitätsanforderungen, desto größer ist der Aufwand.
- | Es gibt nicht **die** Qualität, sondern es gibt verschiedene Qualitätsmerkmale.
- | Jedem Qualitätsmerkmal lassen sich Kennzahlen zuordnen.

Zusammenfassung, Kernpunkte



- Vorgehensmodelle
- Planungs- und Analysephase
- Ist-Analyse
- Lastenheft
- Durchführbarkeitsuntersuchung
 - Einfache Techniken der Aufwandsschätzung

Was kommt beim nächsten Mal?



- Erweiterte Techniken der Aufwandsschätzung
 - COCOMO-Methode
 - Function-Point-Methode