



Vorlesung Softwaretechnik I (SS 2025)

2. Projekte, Personen, Prozesse, Produkte

Prof. Dr. Jens Grabowski

Tel. 39 172022

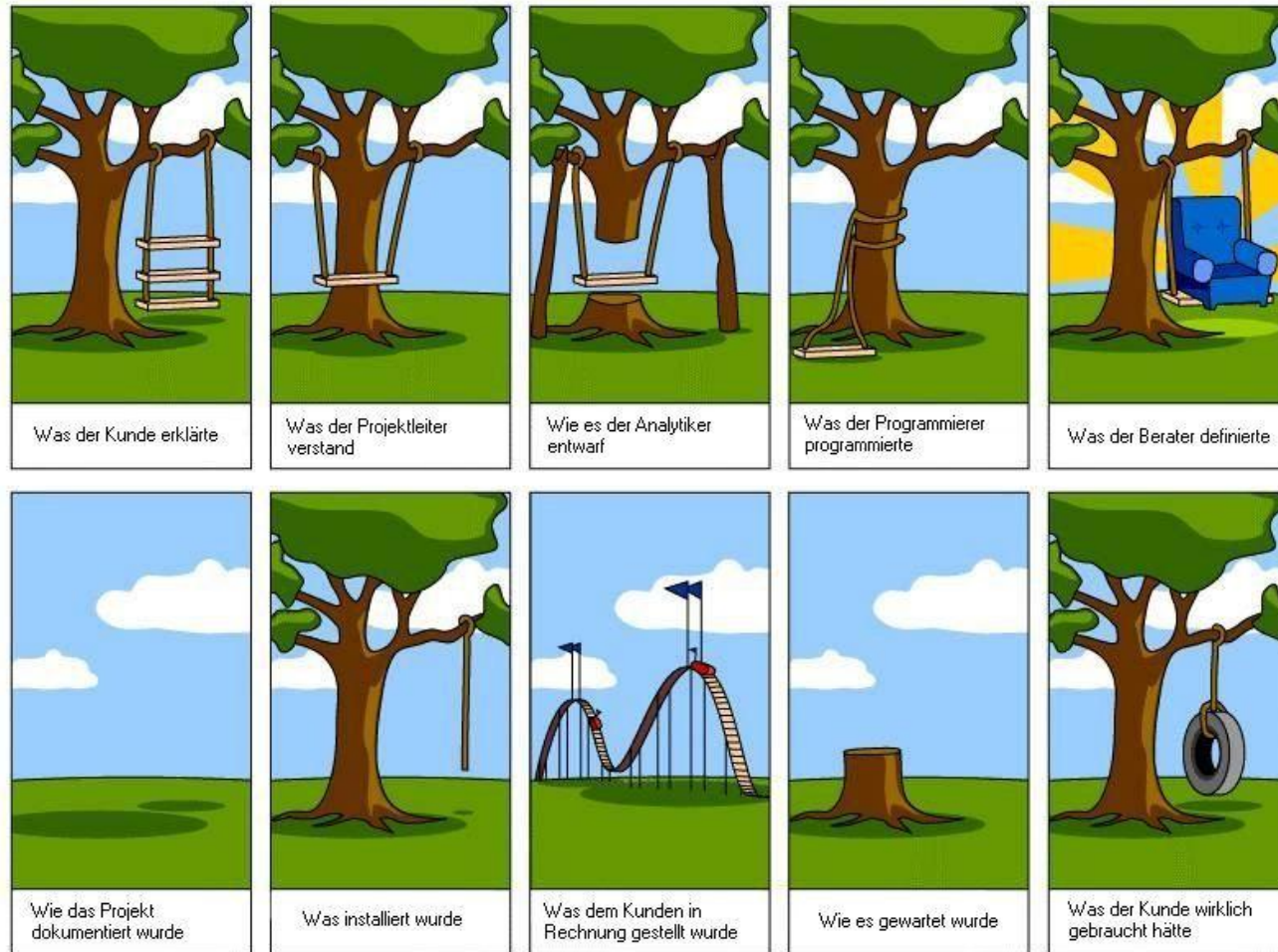
grabowski@informatik.uni-goettingen.de



Inhalt

- Einführung
- Das Projekt
- Personen
- Prozesse
- Produkte
- Lernziele

Einführung – Typisches Softwareprojekt?



[Grechenig et. al.: Softwaretechnik. Pearson Studium, 2010]

Einführung – Top 10 Liste der Risikoelemente des Software Engineering

- Personalmängel
- Unrealistische Zeit- und Kostenpläne
- Entwicklung der falschen Funktionalität
- Unpassende Benutzerschnittstellen
- „Vergolden“ eines Systems
- Häufige Änderungen der Anforderungen
- Qualitätsmängel bei extern vergebenen Komponenten
- Qualitätsmängel bei extern vergebenen Aufgaben
- Unzureichende Produktionsleistungen
- Projektziele an der Grenze des „State-of-the-Art“

Nach: [Dumke, R.: Software Engineering. Vieweg Verlag, Wiesbaden, 2003]

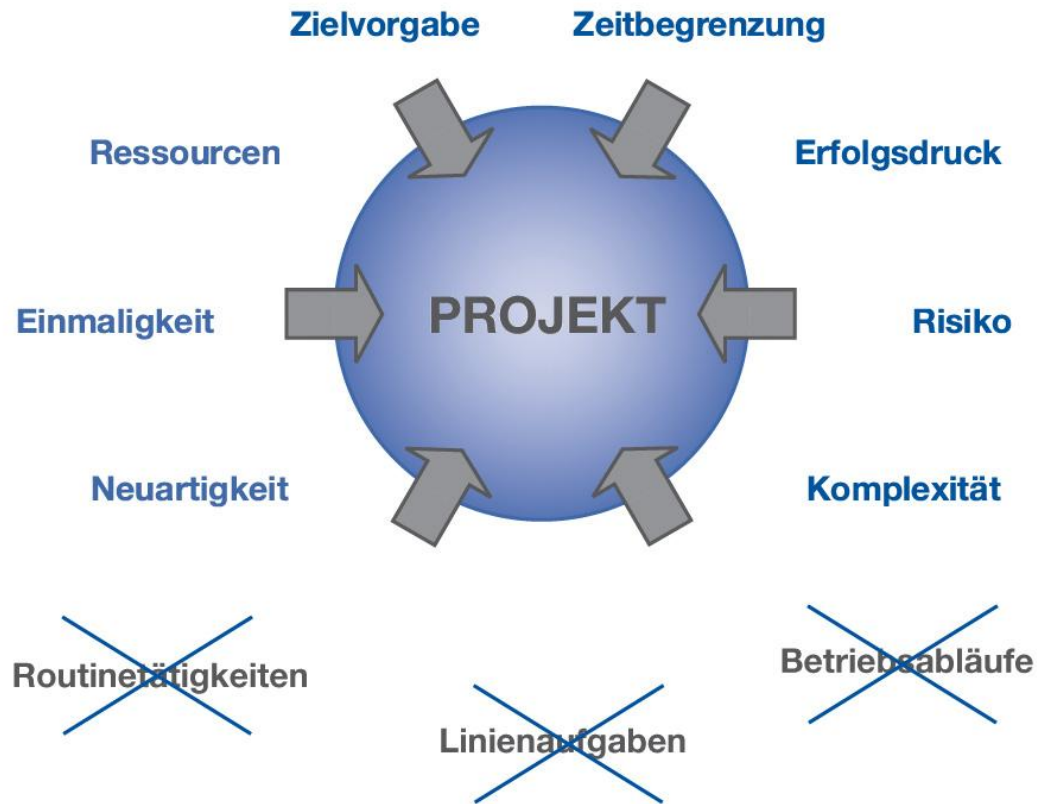
Inhalt

- Einführung
- **Das Projekt**
- Personen
- Prozesse
- Produkte
- Lernziele

Das Projekt - Definition

- In sich inhaltliches abgeschlossenes Vorhaben, in dem neuartige und unbekannte Probleme gelöst werden.
- Ein Projektteam.
- Klare Ziele.
- Beinhaltet eine oder mehrere Phasen.
- Unterschiedliche Methoden.
- Besonderes Risiko.
- Zeitlich begrenzt.
- Eigenes Budget.

Das Projekt



[Grechenig et. al.: Softwaretechnik. Pearson Studium, 2010]



Das Projekt – Merkmale eines Projekts

- Größe
- Dauer
- Anwendungsbereiche
- Quellen der Komplexität
- Zielsetzungen
- Ausgangs- & Zielprodukte
- Projekttypen

Merkmale eines Software-Projekts – Größe & Dauer & Anwendungsbereiche

Projektgrößen aus Softwaresicht (Fallbeispiele)

	<u>Jahre</u>	<u>#Team</u>	<u>PM</u>	<u>€ (2010)</u>
■ Sehr kleines Projekt	0,25	1	3	15.000
■ Kleineres Projekt	0,5	2	12	80.000
■ Mittleres Projekt	1,5	5	60	600.000
■ Große Hochschule	3,0	15	500	6 Mio.
■ Tickets Staatl. Bahn	3,5	40	1400	25 Mio.
■ ID Paraguay	2,5	120	3500	55 Mio.
■ eHealth-Germany	5,0	500	30000	600 Mio.

[Grechenig et. al.: Softwaretechnik. Pearson Studium, 2010]

Merkmale eines Software-Projekts – Größe & Dauer & Anwendungsbereiche

„Size creates Complexity“: Erläuterung und Grobschätzung
der relativen Aufwände abhängig von der Projektgröße

	PM	Aufw.	Dauer	Größensymptom (Bsp.)
■ Sehr kleines Projekt	3	0.5	0.8	User = AG = Develop.
■ Kleineres Projekt	12	1	1	Benutzerhandbuch
■ Mittleres Projekt	60	3	1.5	Reviews, autom. Tests
■ Große Hochschule	500	6	4	perm. User-Workshops
■ Tickets Staatl. Bahn	1400	15	4	Call-Center, 99,995%
■ ID Paraguay	3500	10	8	präventive Medienarbeit
■ eHealth–Germany	30000	50	20	Projekt wird zur Behörde

Auswirkungen von
Projektgröße und
-typ auf Dauer und
Aufwand für die
Umsetzung von
Anforderungen

[Grechenig et. al.: Softwaretechnik. Pearson Studium, 2010]

Merkmale eines Software-Projekts – Quellen der Komplexität 1(2)

- Jedes der bisher genannten Projektmerkmale kann als Quelle von Komplexität auftreten.
 - Projektgröße
 - falls die Größe für das Unternehmen ungewohnt ist.
 - Zielsetzungen
 - falls diese unklar sind, oder sich während der Projektlaufzeit ändern.
 - Domäne
 - falls die Begriffe und Sprache der Domäne nicht ausreichend geklärt sind.
 - Neue Technologien
 - erfordern ggf. neue Vorgehensweisen, zusätzliche Mitarbeiterschulungen usw.

Merkmale eines Software-Projekts – Quellen der Komplexität 2(2)

Projektkomplexität aus User- und Interessens-Dynamik Beispiele und Größenordnungen zur Veranschaulichung					
	Promotion	Core	User	Visitors	User-Kompl.
■ Sehr kleines Projekt	1	2	10	200	0.01
■ Kleineres Projekt	1	10	30	3000	0.25
■ Mittleres Projekt	4	100	2000	20.000	1
■ Große Hochschule	100	4000	40.000	1 Mio.	100
■ Tickets Staatl. Bahn	15	8000	4 Mio.	20 Mio.	200
■ ID Paraguay	50	15K	8 Mio.	> 8 Mio.	400
■ eHealth-Germany	1000	200K	80 Mio.	> 80 Mio.	4000

Abbildung 2.5: Projektkomplexität. Promotoren sind die Führungsspitzen, die hinter dem System stehen, Core-User die professionellen Anwender und Informationsbereitsteller, User bezeichnen die normalen Nutzer, Visitor besuchen das System gelegentlich, surfend oder zufällig. Der Wert für User-Komplexität ist ein aggregierter Expertenschätzwert aus allen vier Teilkomplexitäten. Als Faktorbasis fungiert das mittlere Projekt.

[Grechenig et. al.: Softwaretechnik. Pearson Studium, 2010]

Merkmale eines Software-Projekts – Zielsetzungen

- Beweggrund, warum ein Projekt durchgeführt werden soll:
 - Kurzfristige ökonomische Interessen,
 - Strategisches Investitionsprojekt,
 - Projekt in der Krise,
 - Forschungsprojekt,
 - ...

Merkmale eines Software-Projekts – Ausgangs- und Zielprodukte

■ Ausgangsprodukte

- ☐ Keine Vorgaben
- ☐ Produkte aus einem aktuellen Vorgängerprojekt
- ☐ Bestehendes System als Vorlage

■ Projekttypen

- ☐ Analyseprojekt
- ☐ Realisierungsprojekt
- ☐ Wartungsprojekt
- ☐ Gesamtprojekt
(beinhaltet Aspekte aller Projektarten)

Merkmale eines Software-Projekts – Projekttypen (nach Auftraggeber) 1(2)

Projekttyp	Ergebnis	Auftraggeber			
		internes Marketing	externer Kunde	internes Management	Verkauf / externer Kunde
Entwicklungsprojekt	Produkte, Systeme für den Markt	X	(X)	(X)	
Auftragsprojekt	Kundenspezifisches Softwaresystem		X		
EDV-Projekt	Datenverwaltung, Informationssysteme		(X)	X	
Systemprojekt	Industrieanlagen, technische Systeme			(X)	X

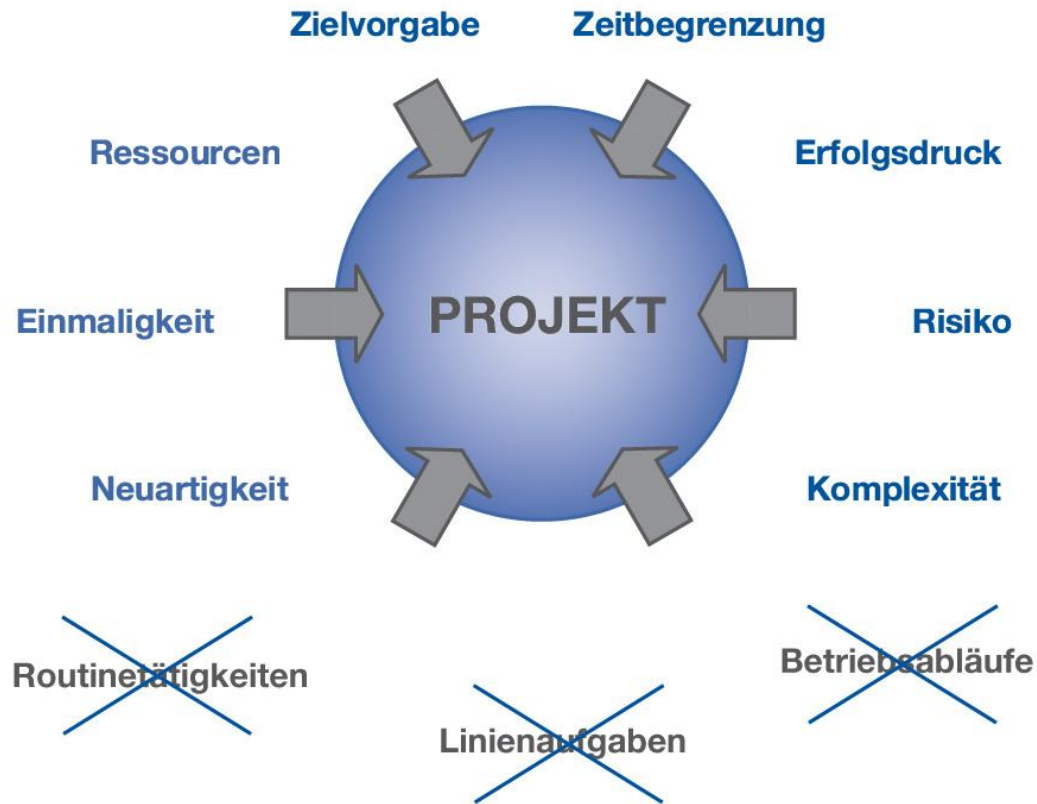
[Ludewig, Lichter: Software Engineering. dpunkt.verlag, 2010]



Das Projekt – Merkmale eines Projekts

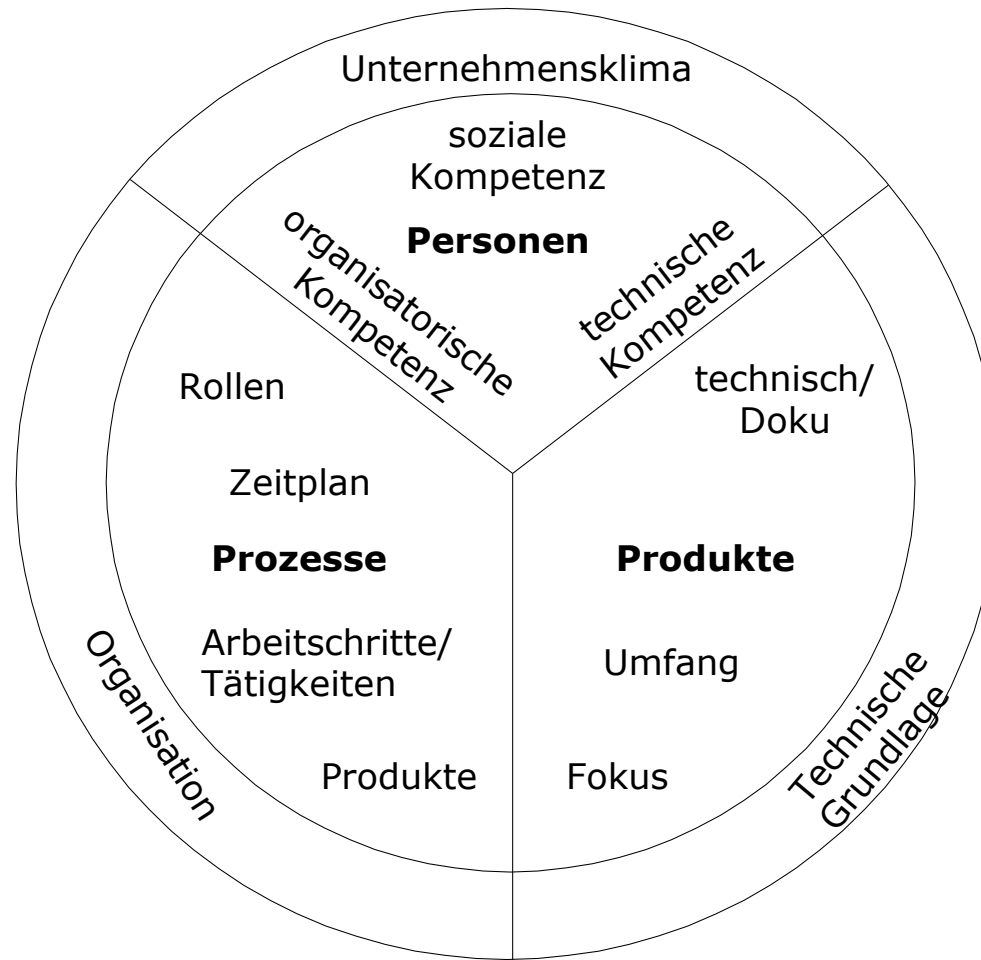
- Größe
- Dauer
- Anwendungsbereiche
- Quellen der Komplexität
- Zielsetzungen
- Ausgangs- & Zielprodukte
- Projekttypen

Das Projekt



[Grechenig et. al.: Softwaretechnik. Pearson Studium, 2010]

Faktoren eines Softwareprojekts



Inhalt

- Einführung
- Das Projekt
- **Personen**
- Prozesse
- Produkte
- Lernziele

Die Personen – Industrial Needs ...

... for software engineers coming freshly from university.

- Knowledge of state-of-the-art technology.
- Knowledge of theoretical background.
- Ability to understand new methods and tools.
- Familiarity with current major paradigms of SE.

- Understanding of the whole process of SE.
- Ability and modesty to do any job within that process.
- Ability and modesty to do any job within the team.
- Understanding and respecting the value of actual project experience.

Die Personen – Rollen je nach Projektgröße

Projekt- Größe	Rollen
<i>Sehr klein</i>	<i>Keine besondere Rolleneinteilung notwendig.</i>
Klein	Kunde; Management, Programmierer und Tester vereinigt in einer Person.
Mittel	Kunde, Projektleiter, Analytiker, Integrator, Programmierer und Tester.
Groß	Kunden, oberes Management, wirtschaftlicher Projektleiter, technischer Projektleiter, Gruppenleiter, Analytiker, Integrator, Programmierer, Tester, Dokumentierer, Qualitätssicherer.
<i>Sehr groß</i>	<i>Alle Rollen aus Projektgröße "groß" und zusätzliche Rollen je nach Projekt (z.B. Berater, Domänenexperten, Technologieexperten usw.).</i>

Die Personen – Rollen

■ Umfeld

- ☐ Kunde
- ☐ Management

■ Team

- ☐ Projektleiter
- ☐ Gruppenleiter
- ☐ Analytiker
- ☐ Integrator
- ☐ Programmierer
- ☐ Tester
- ☐ Qualitätssicherer

Die Personen – Anzahl der Mitarbeiter nach Rollen

Unternehmensgröße	Projekt-leiter	Arbeits-gruppen-leiter	Entwick-ler	Spezia-listen	Qualitäts-sicherer
Groß (ab 120 Mitarbeiter)	8-10	15-20	50-70	5-12	4-10
Mittel (bis 40 Mitarbeiter)	3-5	5-6	15-20	2-5	2-5
Klein (bis 15 Mitarbeiter)	1-2	2-3	5-7	0-2	0-1
Sehr klein (bis 5 Mitarbeiter)	1	1	5	0	0
Ungefähre Relation im Vergleich zu Projektleitern	1	2-3	5-10	0,5-1	0,5-1

Die Personen – Arbeitsgruppe 1(8)

- Zur Bedeutung von Arbeitsgruppen für die einzelne Person:
 - *Stellen Sie sich vor, Ihr Chef hat Ihnen gerade eine Spezifikation auf den Tisch gelegt und gefragt: „Wie lange brauchen Sie mit einer weiteren Person, um das zu erledigen?“ Welche Frage stellen Sie als Erstes?*
 - *Würden Sie fragen: „Können wir objektorientierte Methoden verwenden?“ oder „Welches CASE-System können wir kaufen?“ Natürlich nicht. Ihre erste Frage lautet: „Wer ist die andere Person?“*

[T. DeMacro. Warum ist Software so teuer? Und andere Rätsel des Informationszeitalters. Hanser Verlag, München, 1997.]

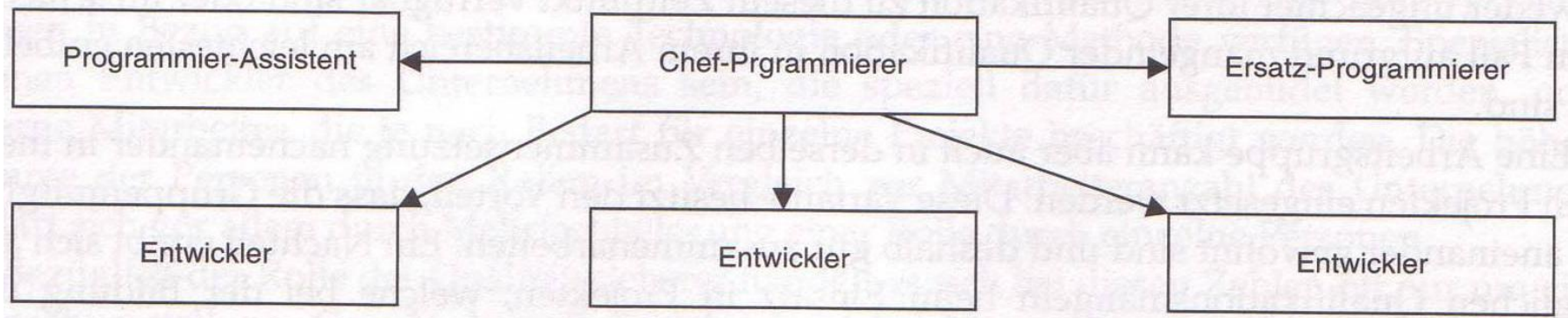
Die Personen – Arbeitsgruppe 2(8)

- Arbeitsgruppen werden nach Bedarf zusammengestellt. Sie können in einer Zusammensetzung nur ein Projekt oder nacheinander für mehrere Projekte durchführen.
- (Theoretische) Modelle für die Zusammensetzung von Arbeitsgruppen:
 - Anarchische Arbeitsgruppe
 - Demokratische Arbeitsgruppe
 - Chefprogrammierer-Team
 - Typische Gruppenstruktur

Die Personen – Arbeitsgruppe 5(8)

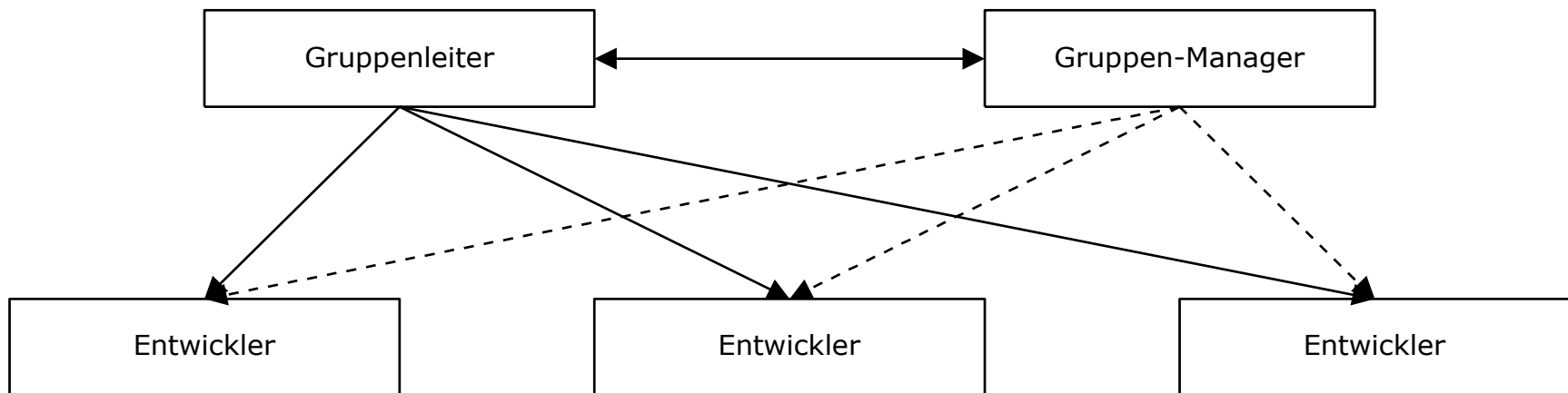
■ Chefprogrammierer-Team

- Vergleichbar mit Operationsteam, in dem ein Chefarzt von einem Team von Spezialisten unterstützt wird.



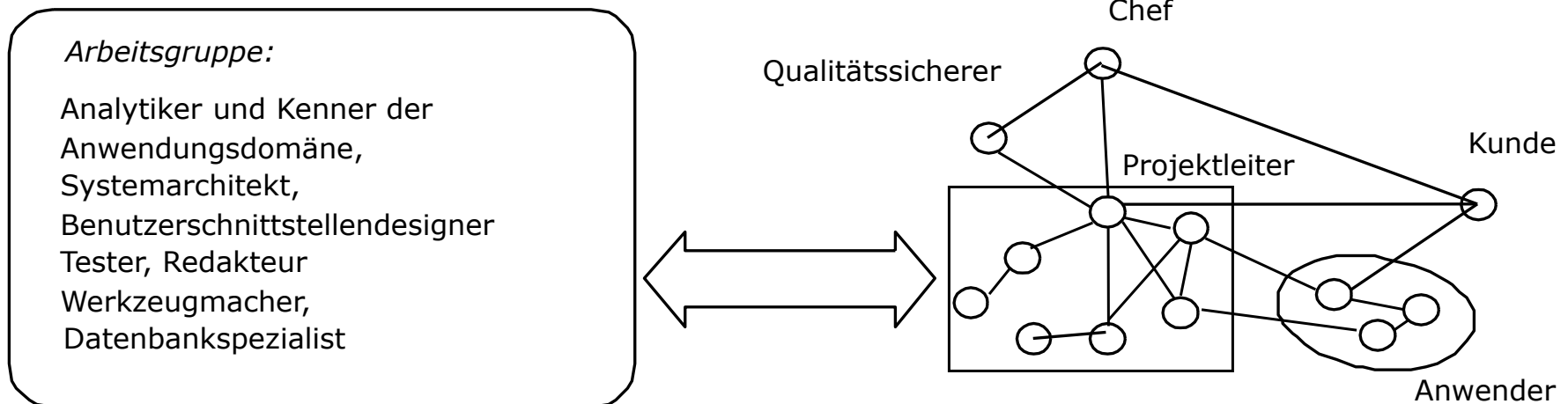
Die Personen – Arbeitsgruppe 7(8)

■ Typische Gruppenstruktur

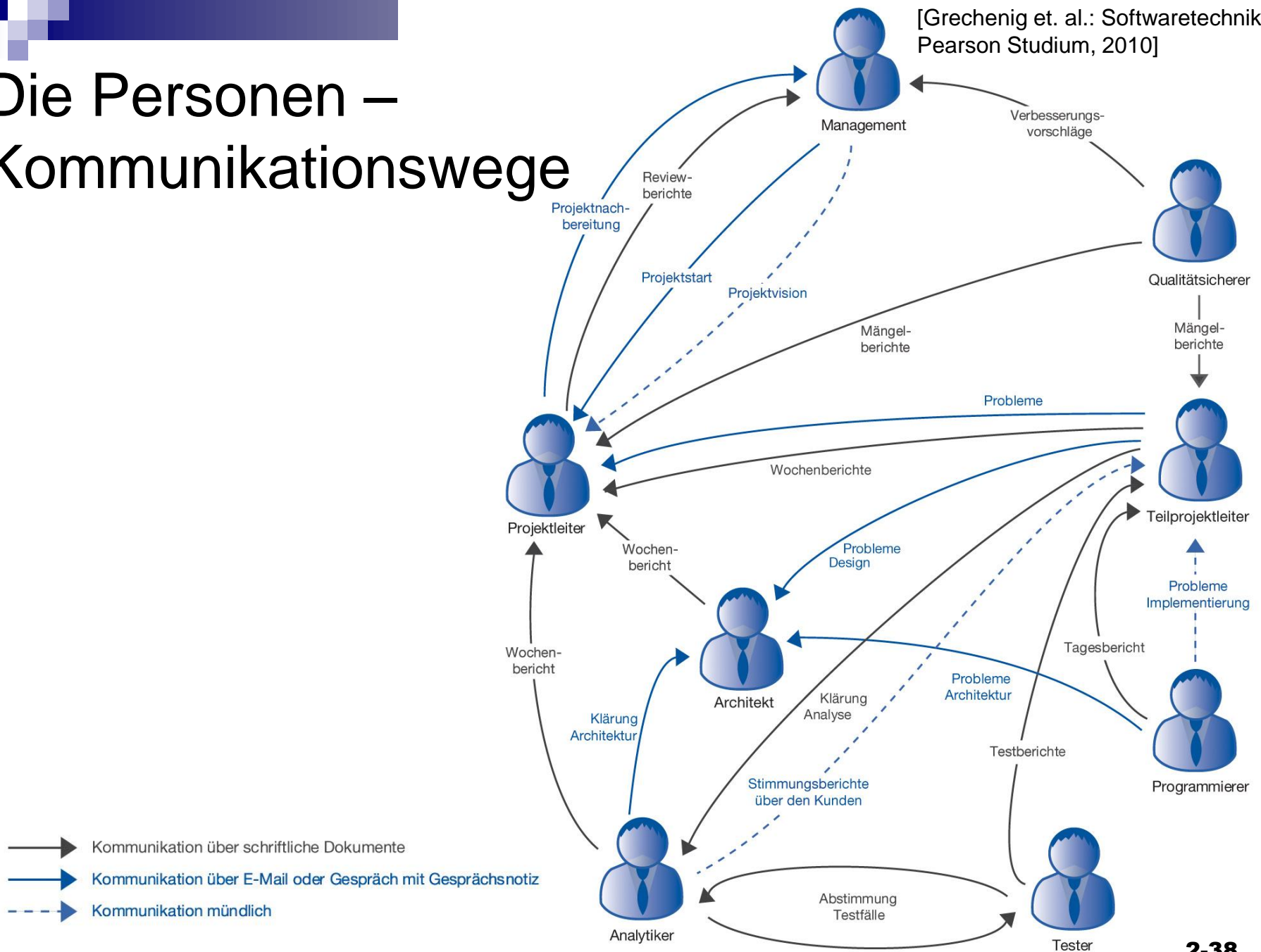


Die Personen – Arbeitsgruppe, Unternehmen & Kunde

- Arbeitsgruppe und ihre Einbindung im Unternehmen und ihre Verbindung zum Kunden.



Die Personen – Kommunikationswege



Die Personen – 1(12)

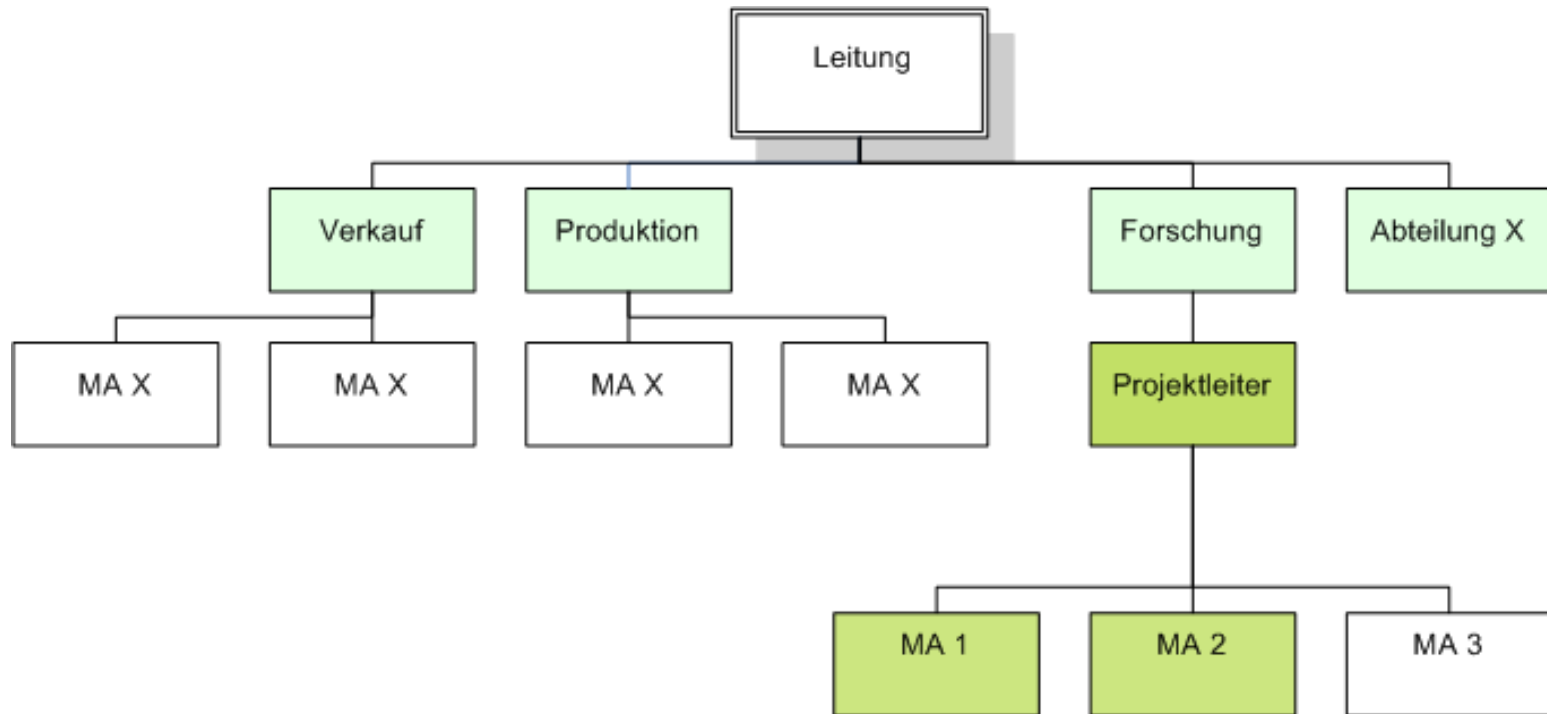
Projekt- & Primärorganisation

- Primärorganisation
 - Organisationsstruktur des Herstellers
 - Aufgabenverteilung/Beziehungen zwischen den Mitarbeitern
- Sekundärorganisation
 - Projektgruppe
- Mögliche Organisationsformen
 - Linienprojektorganisation
 - Stabslinienprojektorganisation (Einflussprojektorganisation)
 - Matrixprojektorganisation
 - Reine Projektorganisation
 - Auftragsprojektorganisation
 - Projektgesellschaft

Die Personen – 2(12)

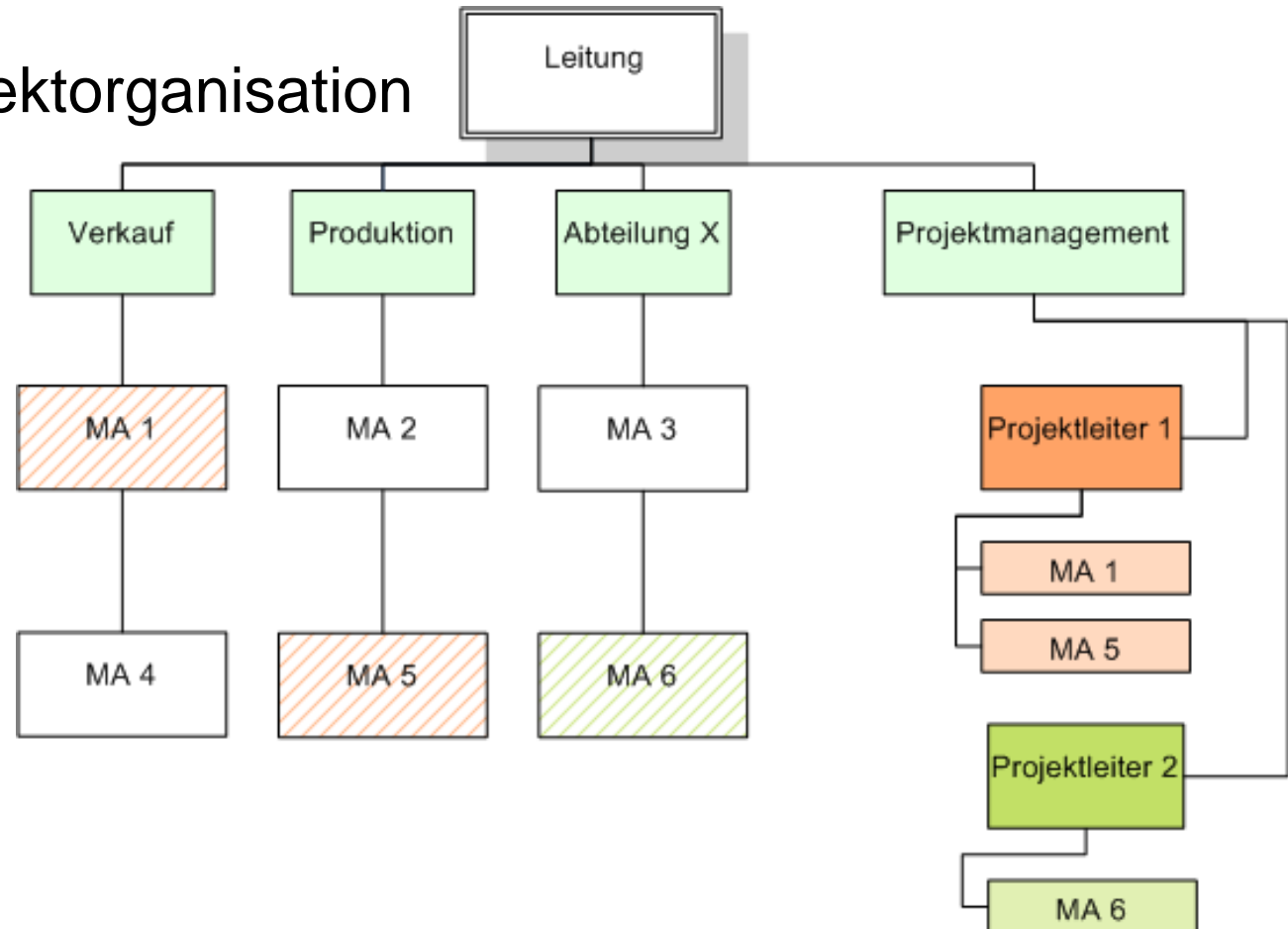
Projekt- & Primärorganisation

■ Linienprojektorganisation

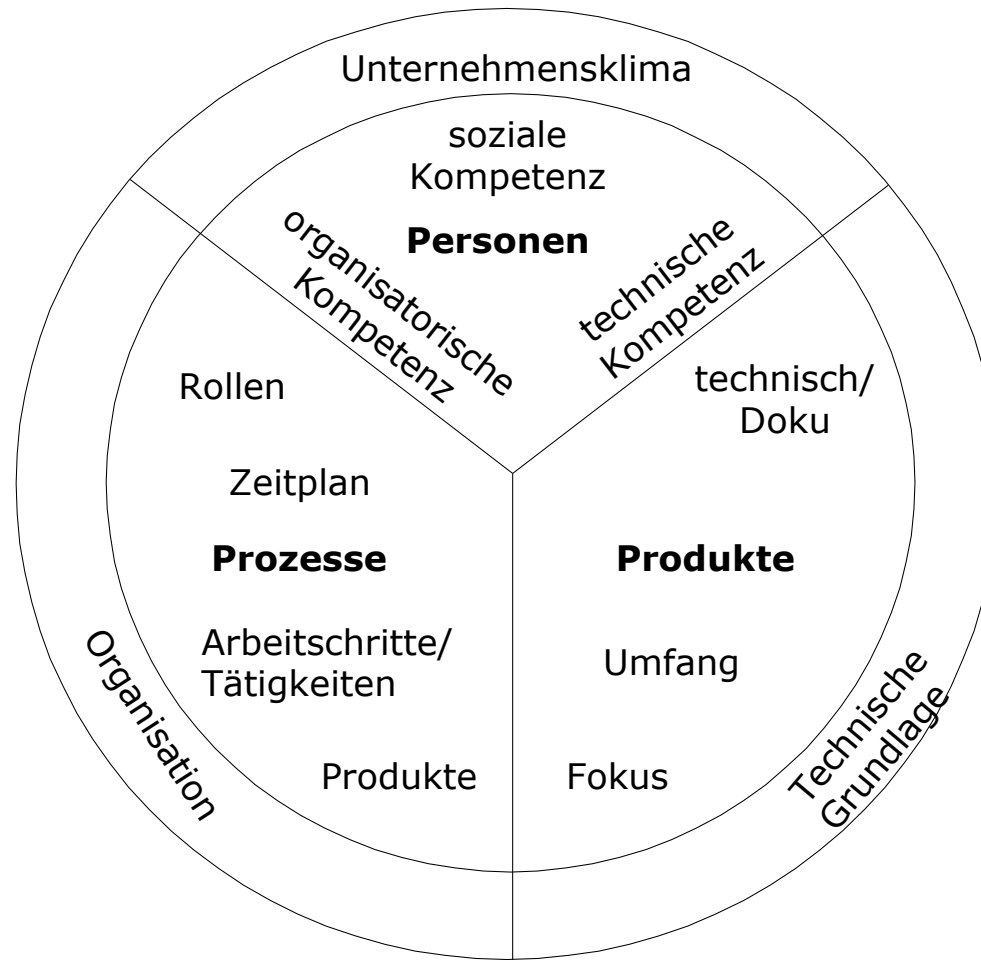


Die Personen – 10(12) Projekt- & Primärorganisation

■ Auftragsprojektorganisation



Faktoren eines Softwareprojekts





Inhalt

- Einführung
- Das Projekt
- Personen
- **Prozesse**
- Produkte

Vorgehens- & Prozessmodelle – Software-Life-Cycle 1(2)

- Wiederkehrende Schritte bei der Softwareentwicklung:

Analyse

Spezifikation der Anforderungen

Grobentwurf, Spezifikation der Module

Feinentwurf

Codierung und Modultest

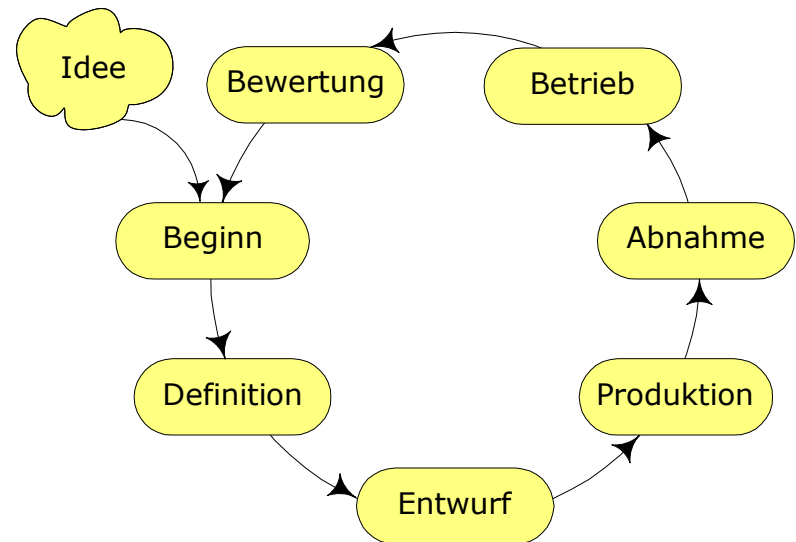
Integration, Test, Abnahme

Betrieb, Wartung

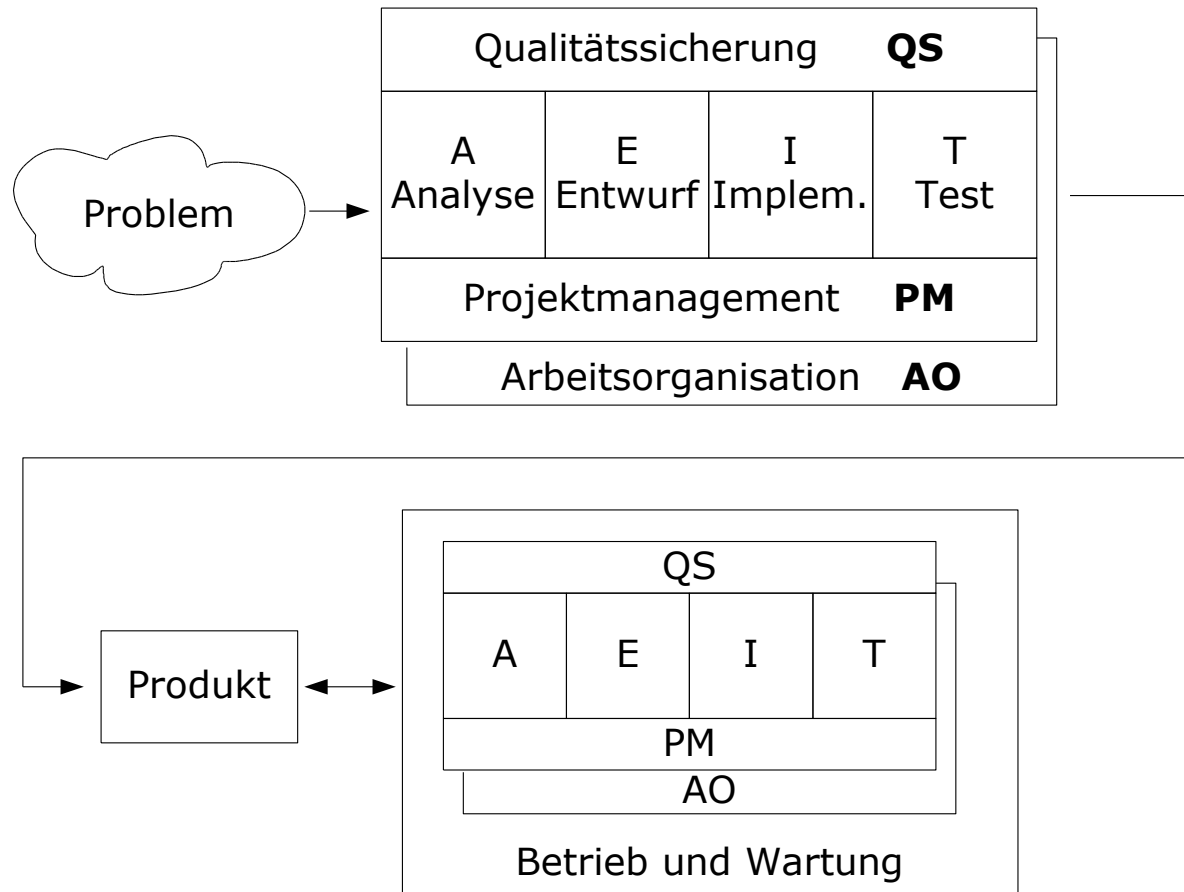
© Ludewig, Lichter, 2006

Vorgehens- & Prozessmodelle – Software-Life-Cycle 2(2)

- Def. IEEE Glossar
(IEEE Std. 610.12, 1990):
 - The period of time that begins when a software product is conceived and ends when the software is no longer available for use. The software life cycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and, sometimes, retirement phase.
 - Note: These phases may overlap or be performed iteratively.



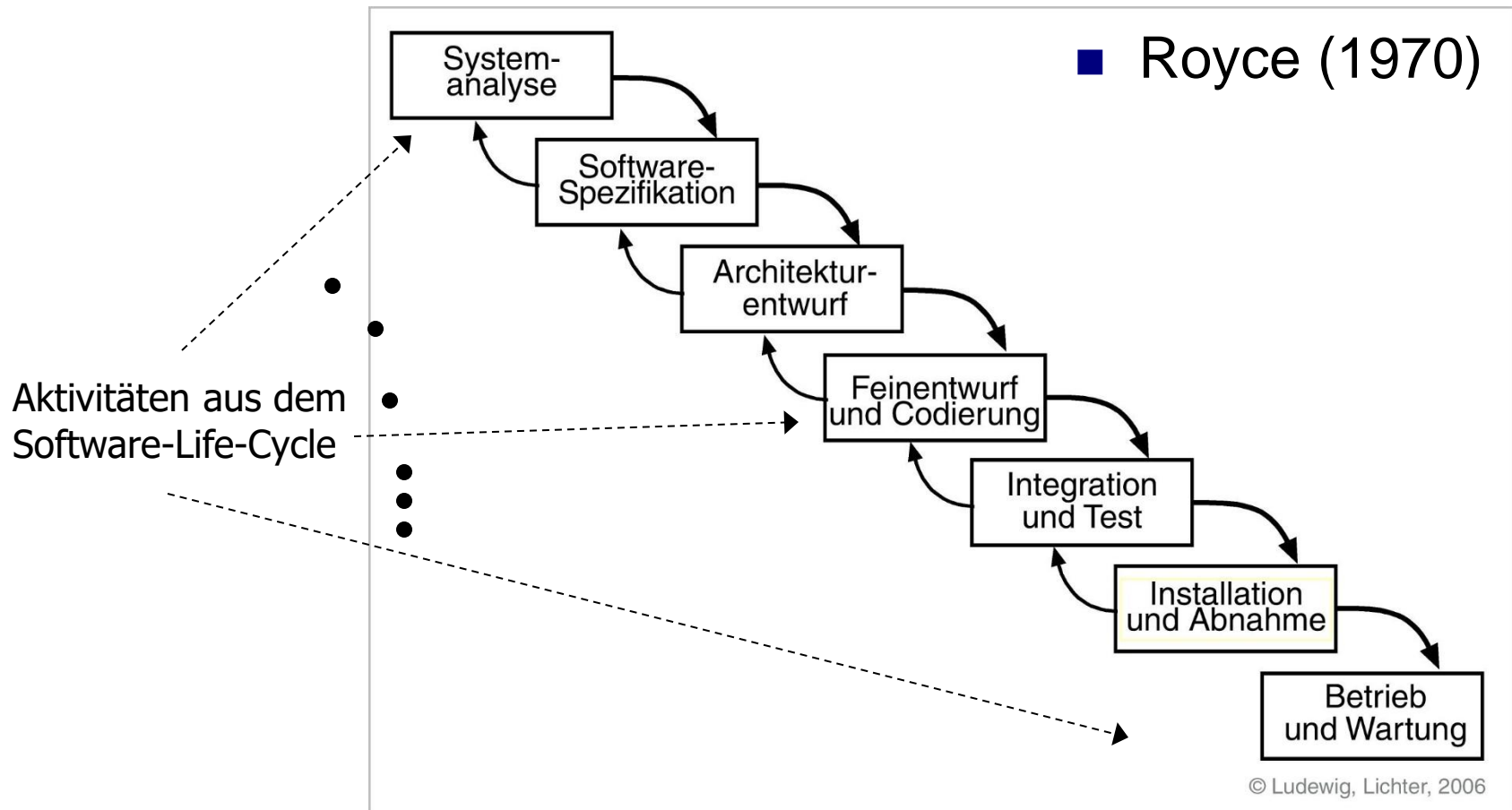
Vorgehens- & Prozessmodelle – Basismodell



Vorgehens- & Prozessmodelle – Vorgehensmodelle

- **Strategien** für die Durchführung eines Projektes, z.B.
 - Build-and-Fix (Code-and Fix)
 - Wasserfallmodell
 - Prototyping
 - Nichtlineare Vorgehensmodelle
 - Rapid Prototyping
 - Evolutionäre Software-Entwicklung
 - Iterative Software-Entwicklung
 - Inkrementelle Software-Entwicklung
 - Treppenmodell
 - Phasenmodell

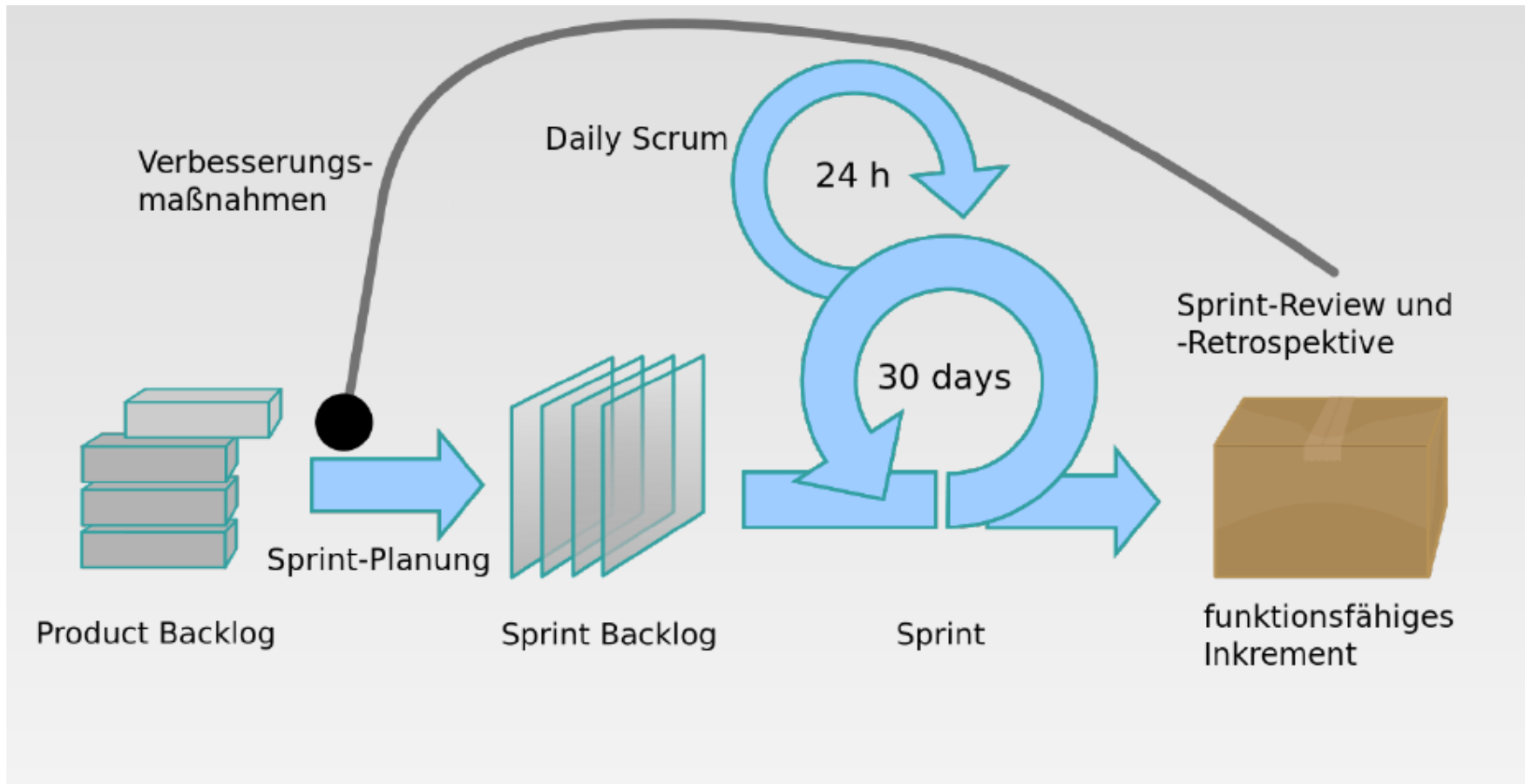
Beispiel: Vorgehensmodell – Wasserfallmodell



Vorgehens- & Prozessmodelle – Prozessmodelle

- **detaillierte Anweisungen** zur Durchführung von Softwareentwicklungsprojekten (Aussagen zu Organisation, Verantwortlichkeiten, zu erstellenden Dokumenten, zu verwendenden Methoden, usw.), z.B.
 - V-Modell
 - V-Modell XT
 - Unified Process
 - Agile Modelle
 - eXtreme Programming (XP)
 - Scrum

Beispiel Prozessmodell – Scrum

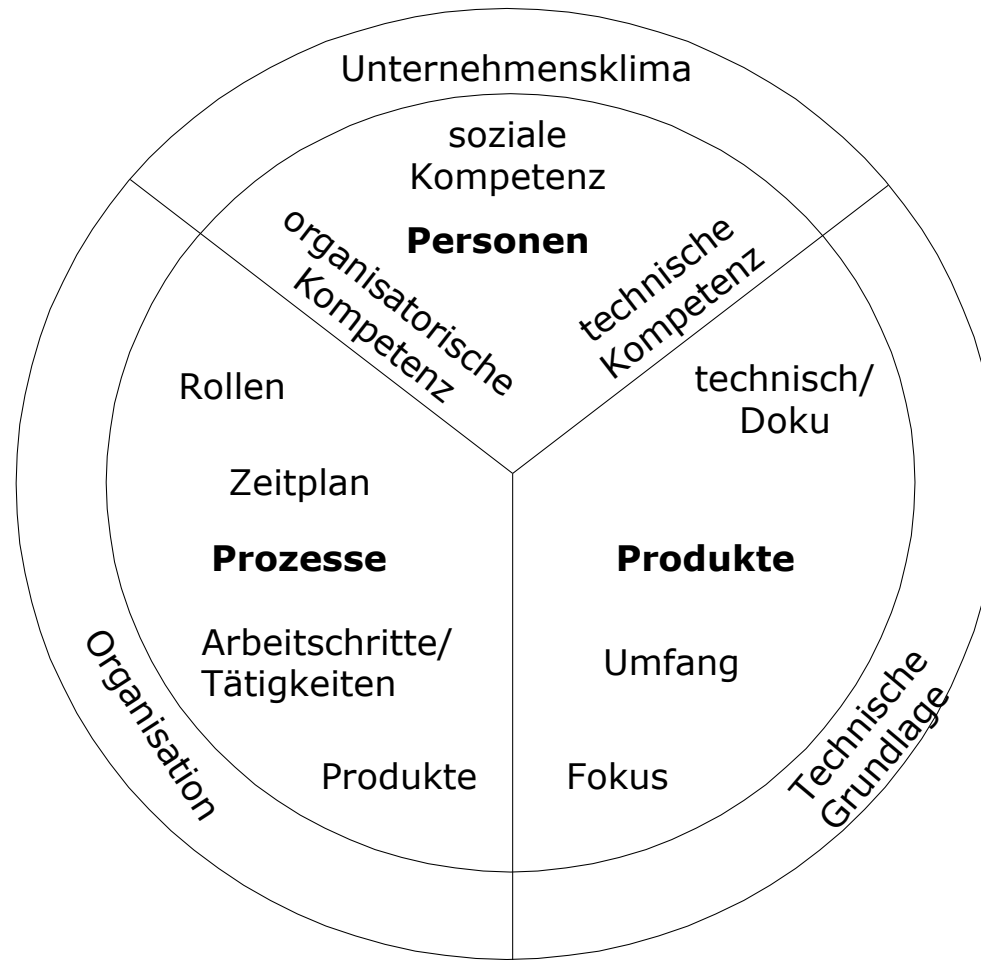




Erwartungen an Vorgehens- und Prozessmodelle

- Gemeinsame Sprache für Projekte.
- Weniger Nacharbeit.
- Vereinfachung von Routineabläufen.
- Basis kontinuierlicher Verbesserung.
- Wettbewerbsvorteil Zertifizierung.
- Hilfe für wenig erfahrene Projektmanager.
- Rascher Nutzen (z.B. Schulung neuer Mitarbeiter).
- Rücksicht auf typische Sonderfälle der Entwicklerrealität.

Faktoren eines Softwareprojekts

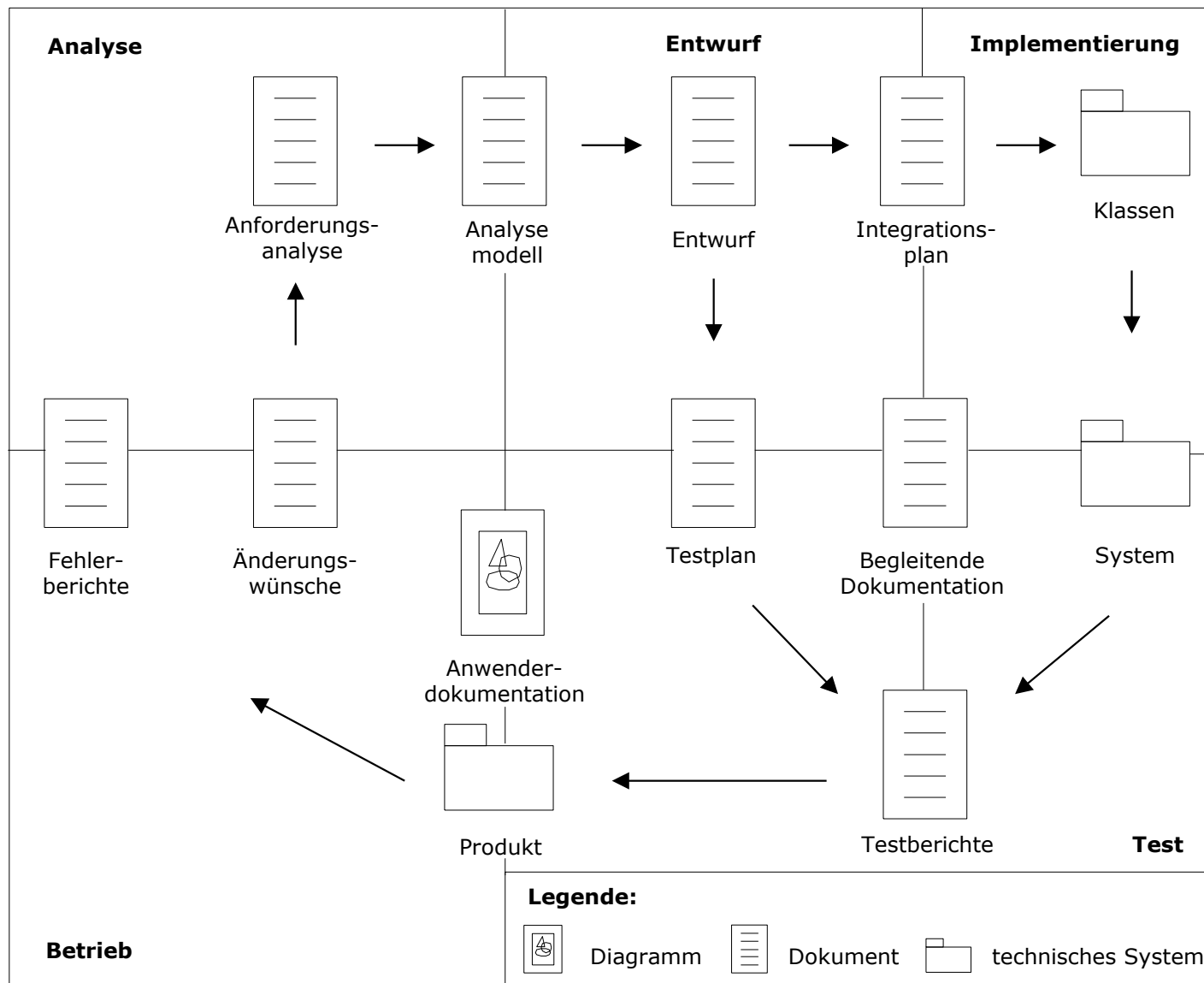




Inhalt

- Einführung
- Das Projekt
- Personen
- Prozesse
- **Produkte**
- Lernziele

Produkte – Zwischen- und Endprodukte



Produkte

- Produkte stellen die überprüfbaren Resultate eines Projektes dar.
- Produktmerkmale
 - ☐ Zweck
 - ☐ Zielpublikum
 - ☐ Art (Doku, Techn. ...)
 - ☐ Detailgrad
 - ☐ Reifestufe
 - ☐ Abnahmekriterien
 - ☐ Referenzen
 - ☐ Erstellungszeitpunkt



Produkte – Formaler Rahmen

- Begriffe
- Dokumentationsrichtlinien
- Dokumentationsstruktur
- Produktvorlagen
- Checklisten
- Versionsmanagement
- Änderungsmanagement
- Archivierungsmanagement

Produkte – Qualitätsmerkmale

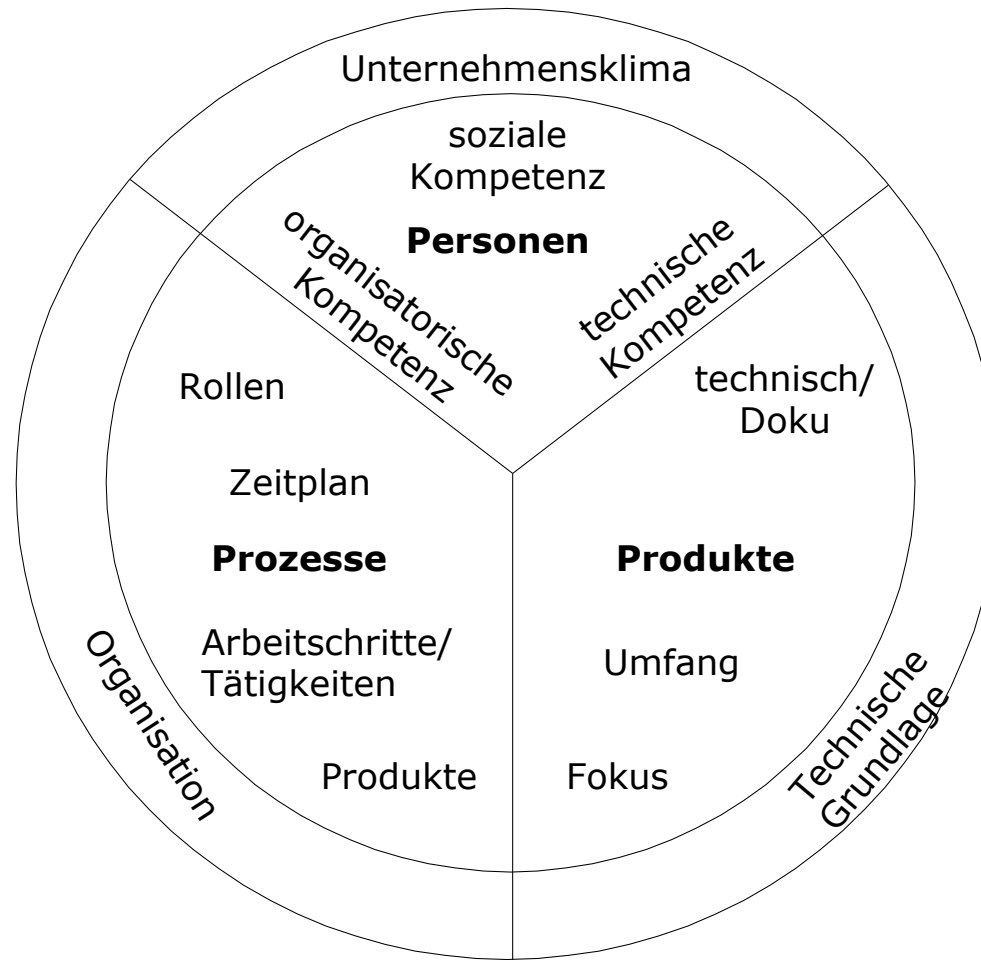
■ Dokumente

- ☐ Vollständigkeit
- ☐ Korrektheit
- ☐ Konsistenz
- ☐ Verständlichkeit
- ☐ Änderbarkeit

■ Technische Produkte

- ☐ Benutzerfreundlichkeit
- ☐ Korrektheit
- ☐ Zuverlässigkeit
- ☐ Effizienz
- ☐ Wartungsfreundlichkeit

Faktoren eines Softwareprojekts





Inhalt

- Einführung
- Das Projekt
- Personen
- Prozesse
- Produkte
- **Lernziele**



Lernziele

■ Softwareprojekte

- ☐ Was ist ein Projekt (Definition)?
- ☐ Was ist das Besondere an Softwareprojekten?
- ☐ Was sind die Merkmale von Softwareprojekten?

■ Personen

- ☐ Welche Rollen gibt es in Softwareprojekten?
- ☐ Wie setzen sich Arbeitsgruppen zusammen?
- ☐ Wie lassen sich Arbeitsgruppen in Unternehmen einbinden?

Lernziele (2)

■ Vorgehens- und Prozessmodelle

- ☐ Was ist der Software-Life-Cycle?
- ☐ Wie sieht das Basismodell für die Softwareentwicklung aus?
- ☐ Welche Erwartungen hat man an Vorgehens- und Prozessmodelle?

■ Produkte

- ☐ Welche Merkmale haben Produkte in der Softwareentwicklung?
- ☐ Welchen formalen Rahmen benötigt man für die Produkterstellung?
- ☐ Welche Qualitätsmerkmale sollten Produkte erfüllen?