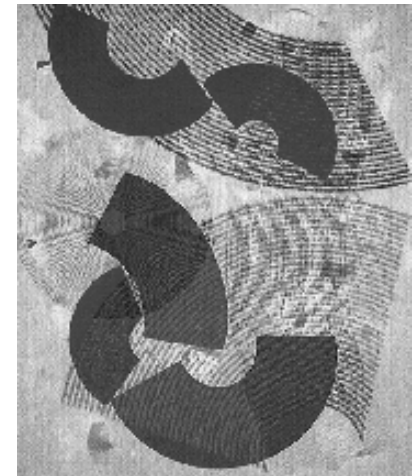


# Software-Management

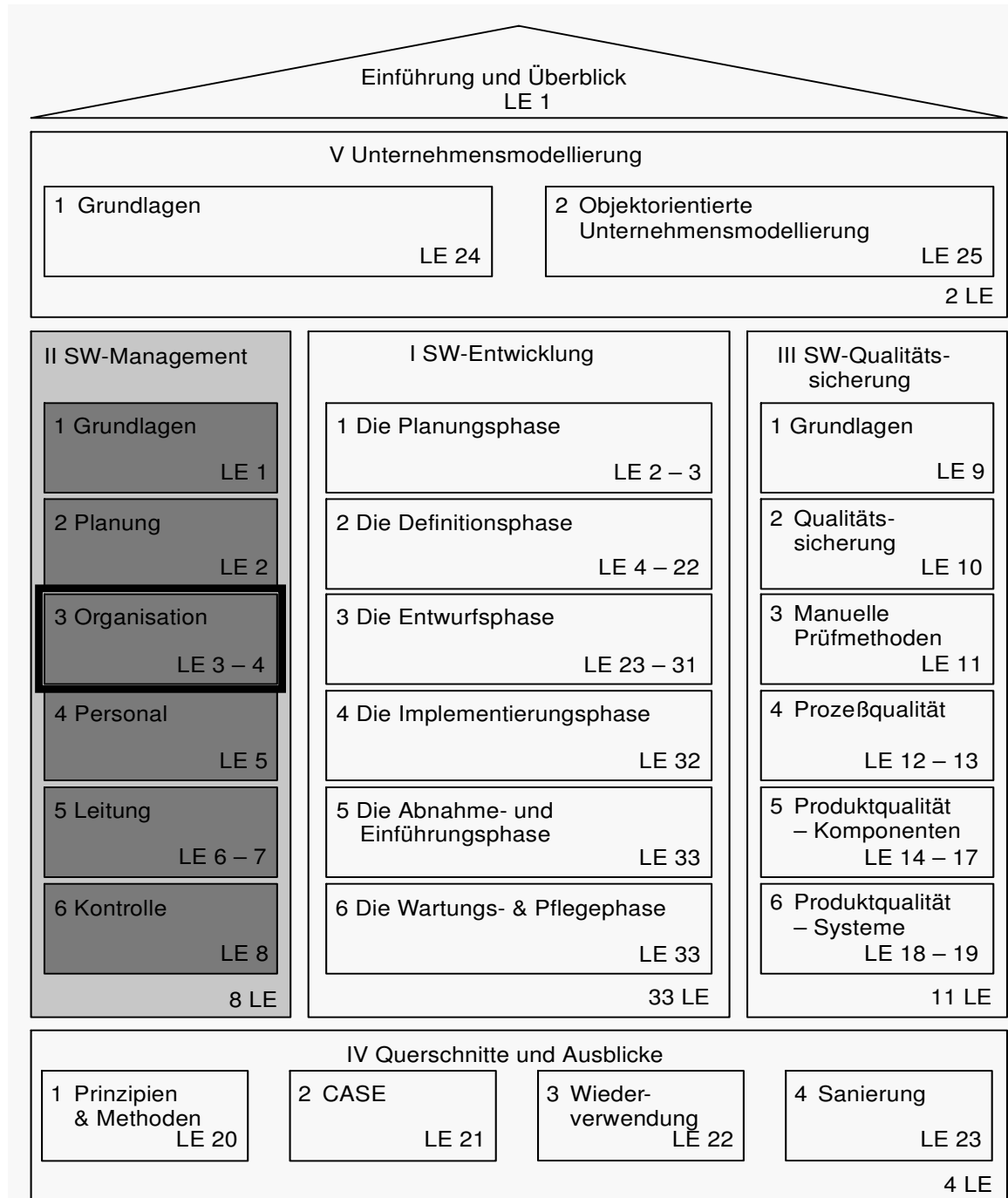
## 3 Organisation

**Prof. Dr. Joachim Hertel**  
**Fachrichtung Informatik**  
**Universität des Saarlandes**



# II Software-Management - Organisation

LE 3  
2



Legende: LE = Lehreinheit (für jeweils 1 Unterrichtsdoppelstunde)

### Lernziele

- ▲ **Die Methodik zur Organisationsgestaltung auf Fallbeispiele anwenden können**
- ▲ **Die Auswirkung des Kommunikationsaufwands auf die Produktivität und Zeitdauer berechnen können**
- ▲ **Ausgehend von vorgegebenen Charakteristika Vorschläge für Organisationsstrukturen erstellen können.**

# Inhalt

## 3.1 Einführung

## 3.2 Grundlagen der Organisationsgestaltung

### 3.2.1 Koordinationsmechanismen

### 3.2.2 Die fünf Teile einer Organisation

### 3.2.3 Gestaltung von Positionen

### 3.2.4 Gestaltung der Aufbauorganisation

### 3.2.5 Projektleiter und Matrixstrukturen

### 3.2.6 Situative Faktoren

### 3.2.7 Die Projektstruktur

### 3.2.8 Die Profibürokratie

### 3.2.9 Mischstrukturen

### 3.2.10 Kooperation Fachabteilung - Systemanalyse.

### 3.1 Einführung

#### ▲ Ziel des Software-Management

- ◆ Software-Produkte entwickeln zu lassen

#### ▲ 2 grundlegende Aufgaben:

##### 1 Arbeitsablauf festlegen

- Ablauforganisation, Prozeßmodell
- Daraus ergeben sich Einzelaufgaben und ihre Interdependenzen

##### 2 Koordinierung der Einzelaufgaben

- Aufbauorganisation.

### 3.1 Einführung

#### ▲ 2 organisatorische Aufgaben:

##### ◆ Mittelfristig

- Geeignete Aufbauorganisation identifizieren und etablieren
- Verbunden damit sind
  - Organisatorische Positionen
  - Verantwortungsbereiche
  - disziplinarische Vollmachten
  - Qualifikationsprofile für die Positionen

##### ◆ Kurzfristig

- Projektspezifische Festlegungen beim Start eines Projekts.

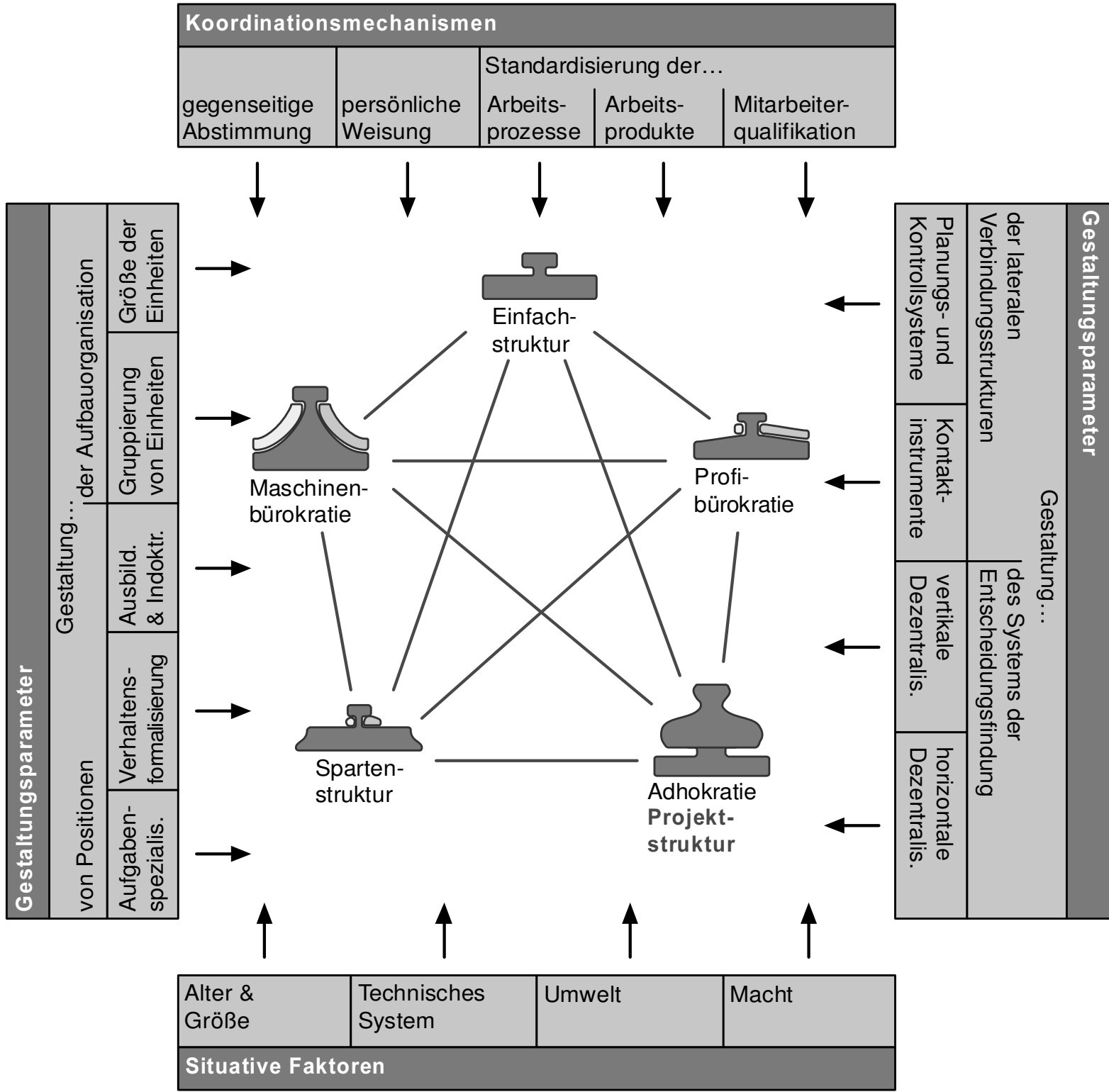
## 3.2 Grundlagen der Organisationsgestaltung

### ▲ Gestaltung einer Organisation durch

- ◆ 5 Koordinationsmechanismen
- ◆ 9 Gestaltungsparameter
- ◆ 4 situative Faktoren

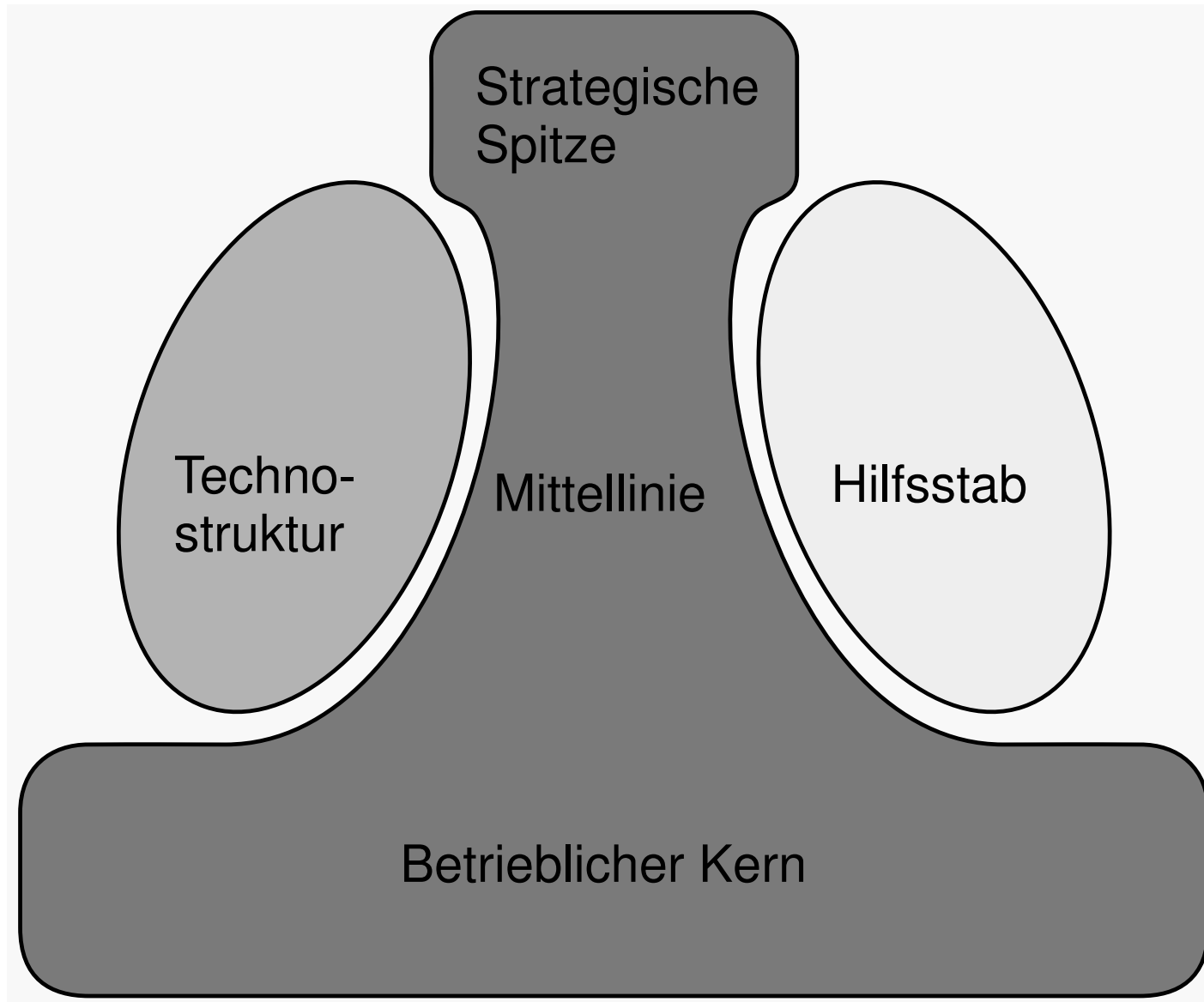
### ▲ Daraus ergeben sich

- ◆ 5 Organisationskonfigurationen
- ◆ + Mischformen.





## 3.2.2 Die fünf Teile einer Organisation



### 3.2.2 Die fünf Teile einer Organisation

#### ▲ Betrieblicher Kern

- ◆ **Ausführende Mitarbeiter**
- ◆ **Arbeit dient direkt der Erstellung von Produkten bzw. der Bereitstellung von Dienstleistungen.**

## 3.2.2 Die fünf Teile einer Organisation

### ▲ Strategische Spitze

#### ◆ 3 Aufgabenbereiche

##### ● Operativer Bereich

- Verantwortlich für effektive Organisation
- Berücksichtigt Wünsche von Gewerkschaften, Eigentümern usw.
- Verteilt Ressourcen
- Vergibt Aufträge
- Genehmigt Entscheidungen
- Löst Konflikte
- Erstellt Strukturpläne
- Stellt Personal ein
- Kontrolliert Arbeitsleistung
- Motiviert und belohnt Mitarbeiter.

## 3.2.2 Die fünf Teile einer Organisation

### ▲ Strategische Spitze

- **Repräsentativer Bereich**
  - Vertretung der Organisation nach außen
- **Strategischer Bereich**
  - Planung der Fortentwicklung des Unternehmens

### ◆ Charakterisierung der Arbeitsweise:

- **Geringer Wiederholungsanteil**
- **Hohe Entscheidungsfreiheit**
- **Lange Entscheidungszyklen**
- **Koordinierung durch gegenseitige Abstimmung.**

## 3.2.2 Die fünf Teile einer Organisation

### ▲ Mittellinie

- ◆ Verbindet die strategische Spitze mit dem betrieblichen Kern
  - Führungskräfte der Mittellinie erhalten über persönliche Weisung Anweisungen aus der strategischen Spitze und geben sie an den betrieblichen Kern weiter
  - Entscheidungen über Ressourcen, Vorschriften und Pläne werden weitergegeben
- ◆ Ereignisse und Vorschläge aus dem betrieblichen Kern werden in komprimierter Form nach oben geleitet.

## 3.2.2 Die fünf Teile einer Organisation

### ▲ Technostruktur

- ◆ **Gruppe von Analytikern, die...**
  - **Arbeitsabläufe**
    - gestalten
    - planen
    - verändern
    - standardisieren
  - **Innovationen auf das Unternehmen übertragen**
  - **Mitarbeiter ausbilden**
  - **die Organisation an Veränderungen der Umwelt anpassen**
- ◆ **Außerhalb der Hierarchie der Linienführungskräfte**
- ◆ **In der Software-Technik: Methodenberater.**

### 3.2.2 Die fünf Teile einer Organisation

#### ▲ Technostruktur

##### ◆ 3 Arten von Analytikern

- **Arbeitsstudienanalytiker**
  - Standardisieren Arbeitsprozesse
  - In der Software-Technik: Software-Prozeßgruppe
- **Planungs- und Kontrollanalytiker**
  - Standardisieren Arbeitsprodukte
  - Beispiel: Ingenieure in der Qualitätskontrolle
- **Personalanalytiker**
  - Standardisieren die Qualifikationen der Mitarbeiter.

## 3.2.2 Die fünf Teile einer Organisation

### ▲ Hilfsstab

- ◆ Unterstützt die Organisation außerhalb des betrieblichen Arbeitsablaufs
  - Poststelle
  - Telefonzentrale
  - Kantine
  - Rechtsabteilung
  - Öffentlichkeitsarbeit.



## 3.2.2 Die fünf Teile einer Organisation

### ▲ Weitere Begriffe

#### ◆ Mittleres Management

- Führungskräfte, die weder zur strategischen Spitze noch zum betrieblichen Kern gehören

#### ◆ Linie

- Hierarchie der Führungskräfte von der strategischen Spitze bis zum betrieblichen Kern
- Linienpositionen sind i.d.R. mit Entscheidungsbefugnissen ausgestattet

#### ◆ Stab

- Oberbegriff für Technostruktur und Hilfsstab
- Meist ohne formale Entscheidungsbefugnisse
- Leitungshilfsstellen mit Unterstützungscharakter.

## 3.2.1 Koordinationsmechanismen

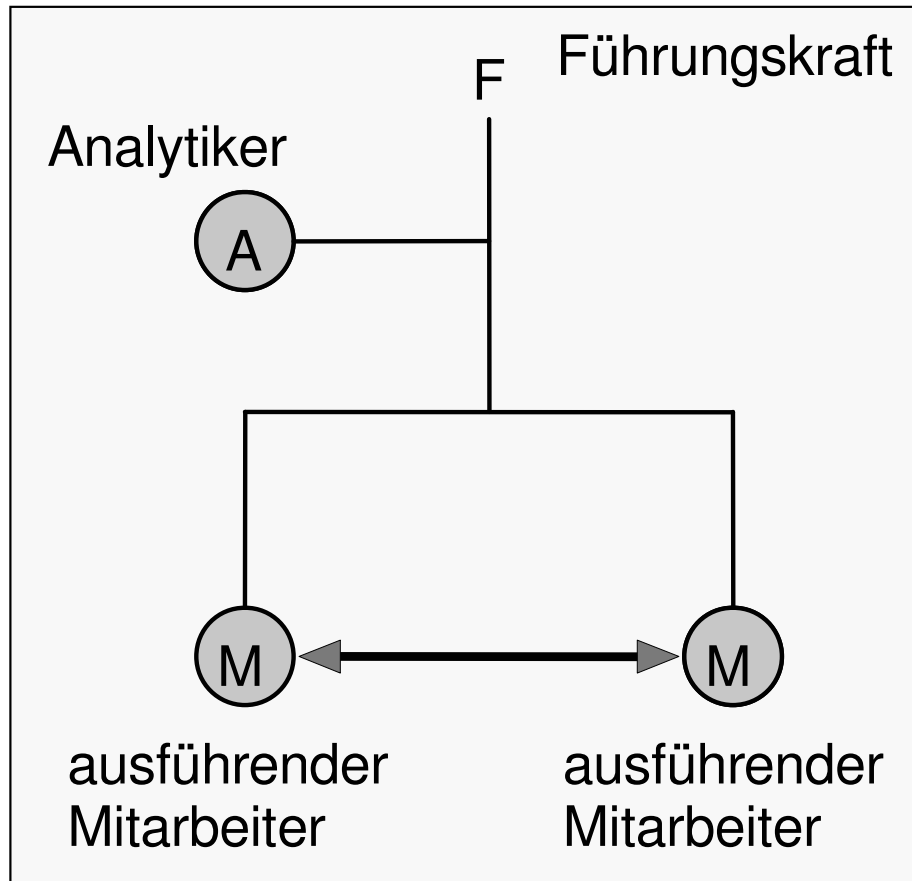
### ▲ Koordination von Arbeitsabläufen durch...

- ◆ gegenseitige Abstimmung
- ◆ persönliche Weisung
- ◆ Standardisierung der Arbeitsprozesse
- ◆ Standardisierung der Arbeitsprodukte
- ◆ Standardisierung der Qualifikationen

▲ Kontroll- und Kommunikationsaspekte werden mit abgedeckt.

## 3.2.1 Koordinationsmechanismen

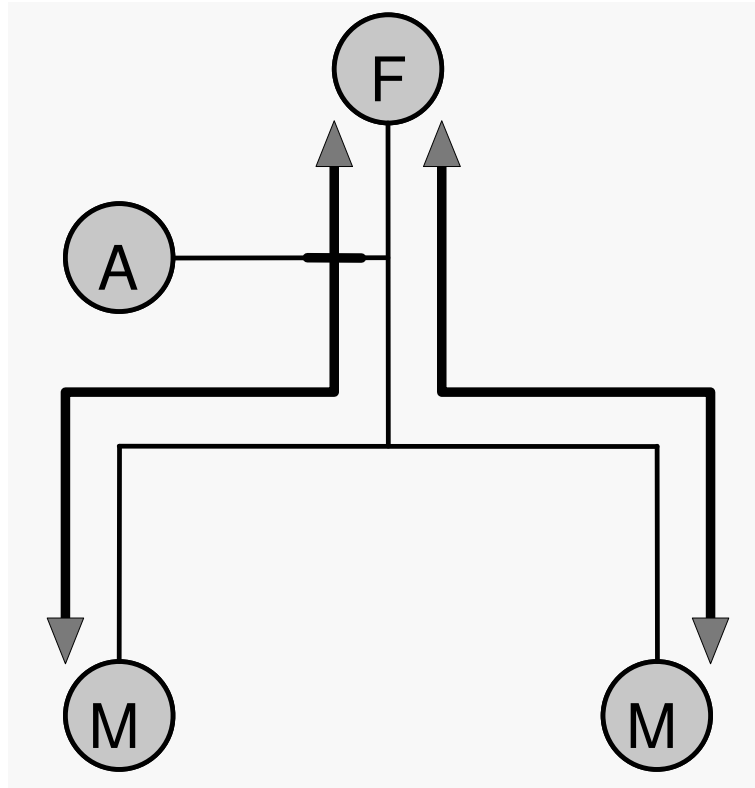
### ▲ Gegenseitige Abstimmung



- ◆ **Koordinierung der Arbeitsabläufe durch informelle Kommunikation**
- ◆ **Arbeitskontrolle liegt bei den Mitarbeitern.**

## 3.2.1 Koordinationsmechanismen

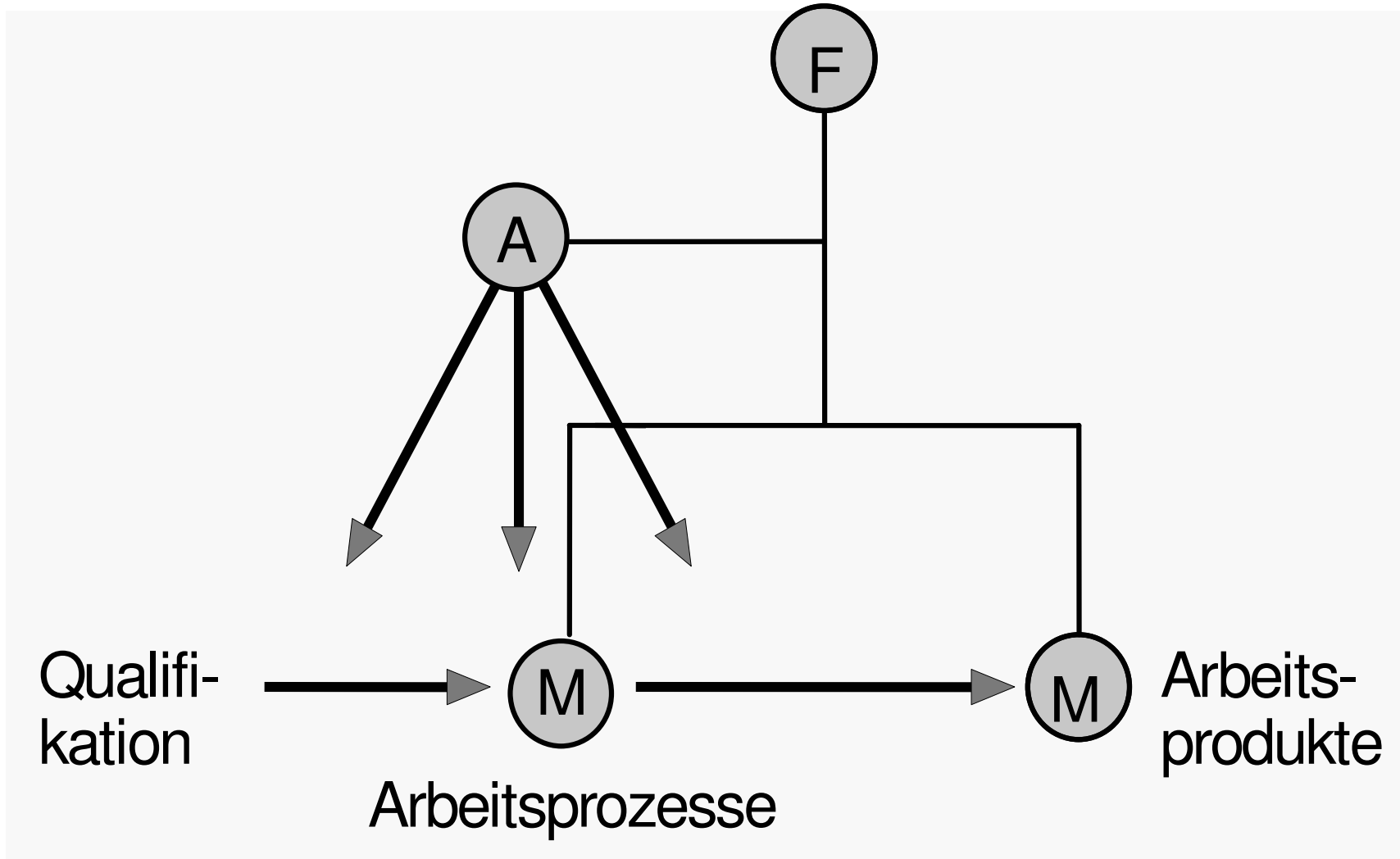
### ▲ Persönliche Weisung



- ◆ Ein Mitarbeiter erteilt Anweisungen an andere Mitarbeiter und kontrolliert die Durchführung.

## 3.2.1 Koordinationsmechanismen

### ▲ Standardisierung



## 3.2.1 Koordinationsmechanismen

### ▲ Standardisierung

#### ◆ Arbeitsprozeß

- Festlegung der einzelnen Arbeitsgänge, z.B. durch Planungsanalytiker
- Beispiel:
  - Für jedes Modul ist ein Funktionstest durchzuführen

#### ◆ Arbeitsprodukt

- Festlegung von Funktionsumfang und Qualität des Ergebnisses
- Beispiel:
  - Die im Pflichtenheft festgelegten Qualitätsziele sollen eingehalten werden.

## 3.2.1 Koordinationsmechanismen

### ▲ Standardisierung

#### ◆ Qualifikation

- **Art und Umfang der Ausbildung ist in Abhängigkeit von den durchzuführenden Tätigkeiten festgelegt**
- **Beispiel:**
  - **Mitarbeiter, die objektorientiertes Programmieren in ihrer Ausbildung gelernt haben, schreiben objektorientierte Programme**
  - **Ein Mitarbeiter der Qualitätssicherung weiß daher, welche Art Programme er zu erwarten hat**

### ▲ Vorteil

- ◆ **Auf indirektem Weg wird eine Kontrolle und Koordination des Arbeitsablaufs erreicht.**

### 3.2.1 Koordinationsmechanismen

#### ▲ Koordinationsmechanismen

- ◆ Die 5 Mechanismen werden meist kombiniert eingesetzt
- ◆ Persönliche Weisung und gegenseitige Abstimmung sind immer vorhanden.



### **3.2.3 Gestaltung von Positionen**

- ▲ 4 Gestaltungsparameter beeinflussen Arbeitsteilung und Aufgabenkoordinierung**
  - ◆ Gestaltung von Positionen**
  - ◆ Gestaltung der Rahmenstruktur**
  - ◆ Gestaltung der lateralen Verbindungsstrukturen**
  - ◆ Gestaltung des Systems der Entscheidungsfindung**
- ▲ Alle Parameter werden mit Schwerpunkt Software-Management untersucht.**

## 3.2.3 Gestaltung von Positionen

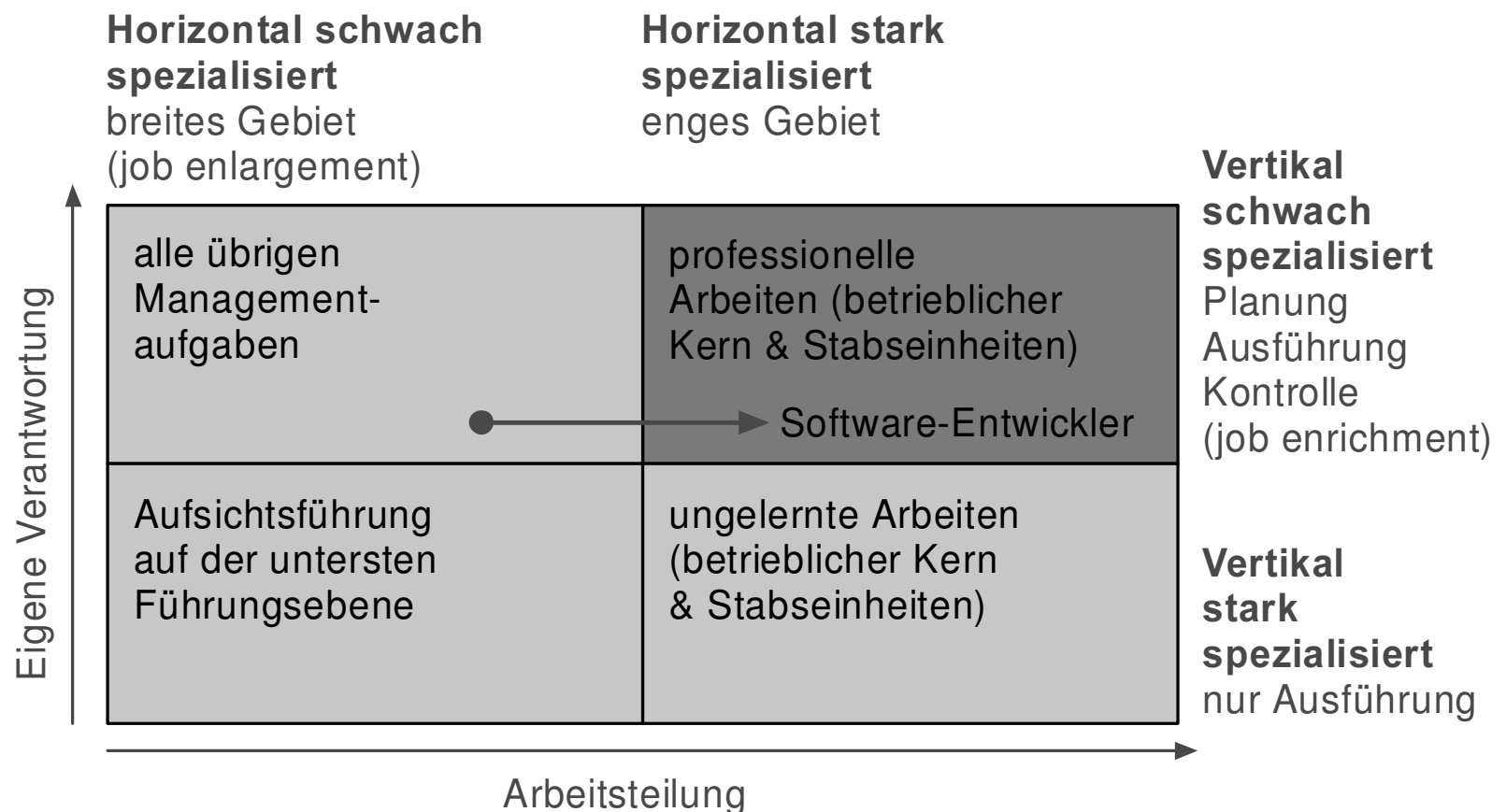
### ▲ Positionsgestaltung ist abhängig von...

- ◆ Aufgabenspezialisierung
- ◆ Verhaltensformalisierung
- ◆ Ausbildung und Indoktrination.

### 3.2.3 Gestaltung von Positionen

#### ▲ Aufgabenspezialisierung

- ◆ **Aufgabenbereiche lassen sich horizontal und vertikal spezialisieren**



## 3.2.3 Gestaltung von Positionen

### ▲ Aufgabenspezialisierung

#### ◆ Horizontale Spezialisierung

- Enges Aufgabengebiet
- Vorrangige Form der Arbeitsteilung
- Repetitive Arbeitsgänge
- Standardisierung wird erleichtert
- Wird durch hohen Arbeitsanfall begünstigt
- Führt zu Kommunikations- und Koordinationsproblemen
- Kann zu Auslastungsproblemen führen.

## 3.2.3 Gestaltung von Positionen

### ▲ Aufgabenspezialisierung

#### ◆ Vertikale Spezialisierung

- Arbeit wird ausgeführt, aber nicht vom selben Mitarbeiter geplant und kontrolliert

#### ◆ Software-Technik

- Arbeit oft horizontal und vertikal erweitert
- Verstärkte horizontale Spezialisierung ist notwendig.

## 3.2.3 Gestaltung von Positionen

### ▲ Formalisierung von Verhaltensweisen

- ◆ Entscheidungsfreiheit der Mitarbeiter kann durch Vorschriften eingeschränkt werden
- ◆ Je stärker das Verhalten standardisiert ist, desto bürokratischer wird eine Struktur
- ◆ Eine Struktur ohne Standardisierung ist organisch
- ◆ In bürokratischen Organisationen werden Teile, die innovativ sein müssen, bewußt von der Organisation getrennt.

## 3.2.3 Gestaltung von Positionen

### ▲ Ausbildung und Indoktrination

#### ◆ Ausbildung

- Vermittelt Kenntnisse und Qualifikationen für eine Position

#### ◆ Indoktrination

- Dient zum Erwerb organisatorischer Normen

#### ◆ Ergebnis:

- Mitarbeiter verinnerlichen standardisierte Verhaltensmuster

#### ◆ Professionelle Arbeit

- Eine komplexe, nicht rationalisierte aber zum Teil formal erfaßte und spezifizierte Arbeit
- Im betrieblichen Kern
  - Oft komplexe Qualifikationen & detaillierte Kenntnisse.

## 3.2.3 Gestaltung von Positionen

### ▲ Ausbildung und Indoktrination

- ◆ **Ausbildung ist Gestaltungsparameter für komplexe, nicht rationalisierte Arbeiten**
- ◆ **Professionelle Mitarbeiter brauchen lange Ausbildungszeiten**
- ◆ **Ausbildung erfolgt meist außerhalb des Unternehmens, z.B. in Universitäten**
- ◆ **Verhaltensformalisierung und Ausbildung sind austauschbar**
- ◆ **Die Mitarbeiter sind Experten auf wohldefinierten Gebieten und damit horizontal spezialisiert.**



## 3.2.3 Gestaltung von Positionen

### ▲ Software-Organisation

- ◆ Im betrieblichen Kern und in der Technostruktur sind professionelle Arbeiten (horizontal spezialisiert, vertikal erweitert) durchzuführen
- ◆ Für jede Position sind die notwendigen Qualifikationen und Kenntnisse zu definieren
- ◆ Bei unzureichender horizontaler Spezialisierung ist diese vorzunehmen.

### 3.2.3 Gestaltung von Positionen

#### ▲ Software-Organisation

##### ◆ Beispiel:

- **Positionen für ein Software-Haus in der technischen Software-Entwicklung**
  - **Betrieblicher Kern**
    - ★ **Systemanalytiker**
    - ★ **Konstrukteur (Entwurf und Implementierung)**
  - **Technostruktur**
    - ★ **Methodenberater**
    - ★ **Prozeßplaner**
  - **Hilfsstab**
    - ★ **Software-Ergonom.**

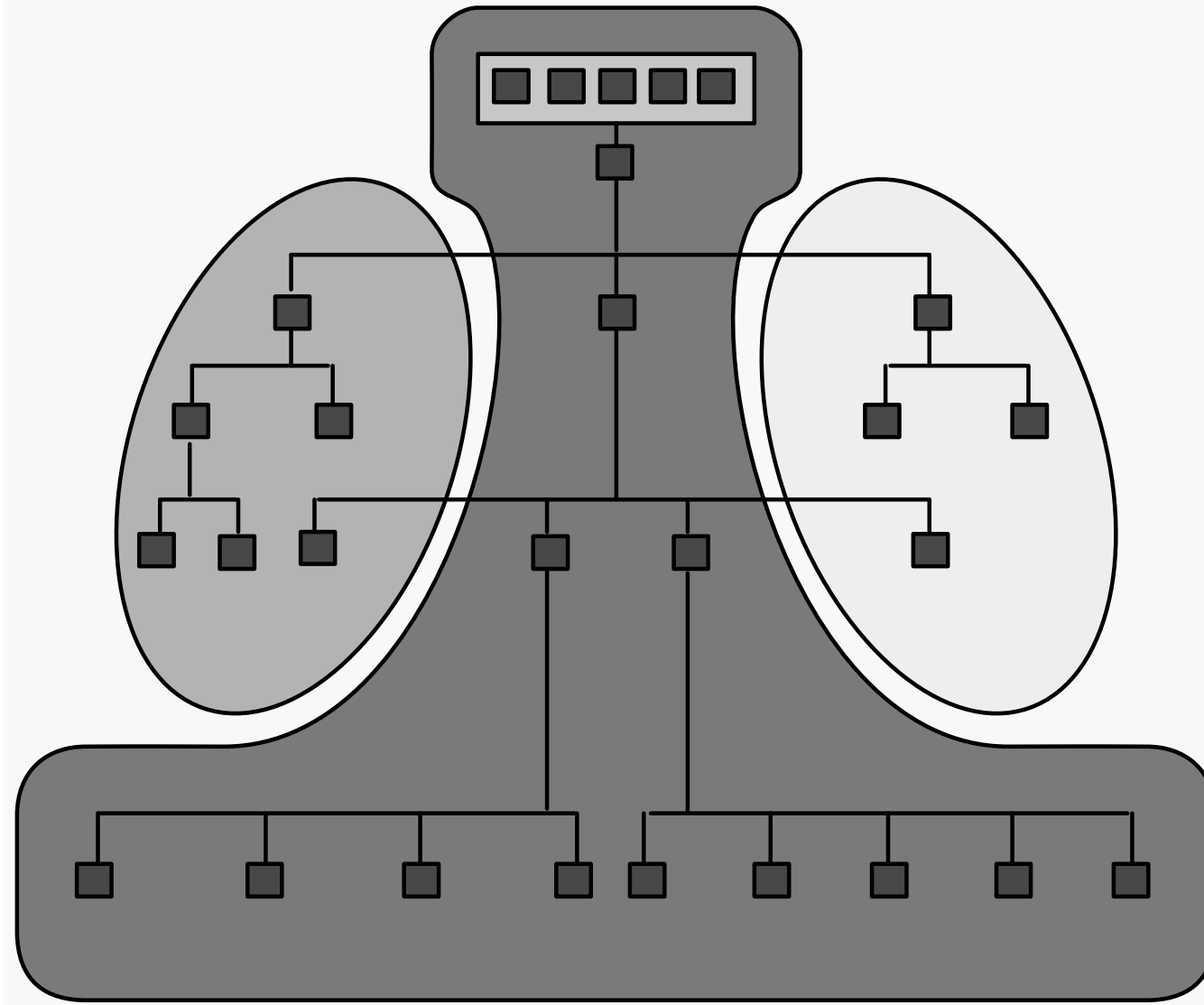
## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Gestaltung der Aufbauorganisation

- ◆ Positionen müssen zu Einheiten zusammengefaßt werden
- ◆ Die Größe der Einheiten muß bestimmt werden
- ◆ Ergebnis des Gruppierungsvorgangs als Organigramm darstellen
  - Bild der Arbeitsteilung
  - Veranschaulicht den Koordinationsmechanismus der persönlichen Weisung.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Aufbau eines Organigramms



## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Methodik

#### ◆ Kombiniertes top-down- und bottom-up-Verfahren

##### 1 Sämtliche Aufgaben zusammenstellen

- Dabei z.B. Ziele und Strategien der Organisation beachten

##### 2 Einzelaufgaben nach Grad der gewünschten Spezialisierung unterschiedlichen Positionen zuordnen

- Formalisierung und Ausbildung festlegen.

## 3.2.4 Gestaltung der Aufbauorganisation

- 3 Festlegen, wie viele Positionen in den Einheiten der untersten Ebene zusammengefaßt werden**
  - Dann entscheiden, welche und wie viele Einheiten zu den jeweils nächsthöheren Einheiten zusammengefaßt werden
  - Fortsetzen, bis die vollständige Hierarchie festliegt
- 4 Aufbauorganisation ausbauen**
  - Entscheidungsbefugnisse festlegen.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Methodik: Beispiel

- ◆ Ein Softwarehaus entwickelt Standardsoftware für ausgewählte Branchen
  - 1 Zusammenstellung sämtlicher Aufgaben
    - a Beobachtung und Analyse des Marktes
    - b Entwicklung der Branchensoftware
    - c Vertrieb der Software
    - d Führung des Unternehmens.

## 3.2.4 Gestaltung der Aufbauorganisation

### 2 Zuordnung zu Positionen

#### a Marketingspezialist:

- Beobachtung und Analyse des Marktes (betrieblicher Kern)

#### b Anwendungsspezialist, Systemanalytiker, Konstrukteur:

- Entwicklung der Branchensoftware (betrieblicher Kern)

#### c Vertriebsbeauftragter:

- Vertrieb der Software (betrieblicher Kern)

#### d Manager:

- Führung der Firma (Mittellinie und strategische Spitze)

#### e Methodenberater, Prozeßplaner, Qualitätssicherer (Technostruktur)

#### f Software-Ergonom, Buchhaltung (Hilfsstab).



## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Gruppierung in Einheiten

- ◆ Durch Gruppierung werden Arbeitsbereiche einer Organisation koordiniert
- ◆ Auswirkungen:
  - 1 Für alle Gruppenmitglieder wird ein gemeinsames Weisungssystem geschaffen
  - 2 In einer Gruppe werden Ressourcen gemeinsam genutzt
  - 3 In einer Gruppe entwickeln sich gemeinsame Leistungsmaßstäbe
  - 4 Gruppierungen fördern die gegenseitige Abstimmung.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Gruppierungsalternativen

#### ◆ Gruppierung nach Funktionen

##### 1 Gruppierung nach Kenntnissen und Qualifikationen

- Beispiel: Alle Diplom-Informatiker

##### 2 Gruppierung nach Arbeitsprozeß und -funktion

- Beispiel: Nach Geschäftsfunktionen: Marketing, ...

##### 3 Gruppierung nach Zeit

- Beispiel: Einteilung nach Schichten

#### ◆ Gruppierung nach Märkten

##### 4 Gruppierung nach Arbeitsprodukten

- Beispiel: Nach den Produkten für die Branchen

##### 5 Gruppierung nach Kunden

##### 6 Gruppierung nach Orten.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Gruppierungskriterien

#### 1 Interdependenzen der Arbeitsabläufe

- Aufgaben so gruppieren, daß sie den Interdependenzen des natürlichen Arbeitsablaufs entsprechen
- Entstehende Arbeitsbereiche sind eine arbeitspsychologisch vollständige Aufgabe
- **Beispiel Software-Technik**
  - Gruppierungen Definition und Konstruktion (mit Entwurf und Implementierung) sinnvoll
  - Gruppierungen Definition und Implementierung sowie die Gruppe Entwurf nicht sinnvoll.

### 3.2.4 Gestaltung der Aufbauorganisation

#### ▲ Gruppierungskriterien

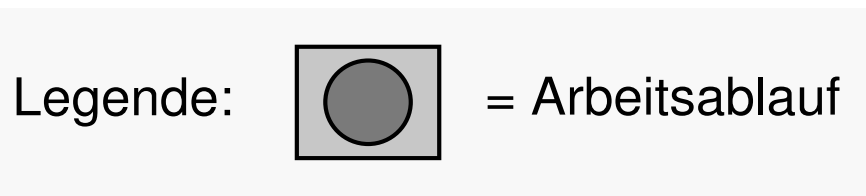
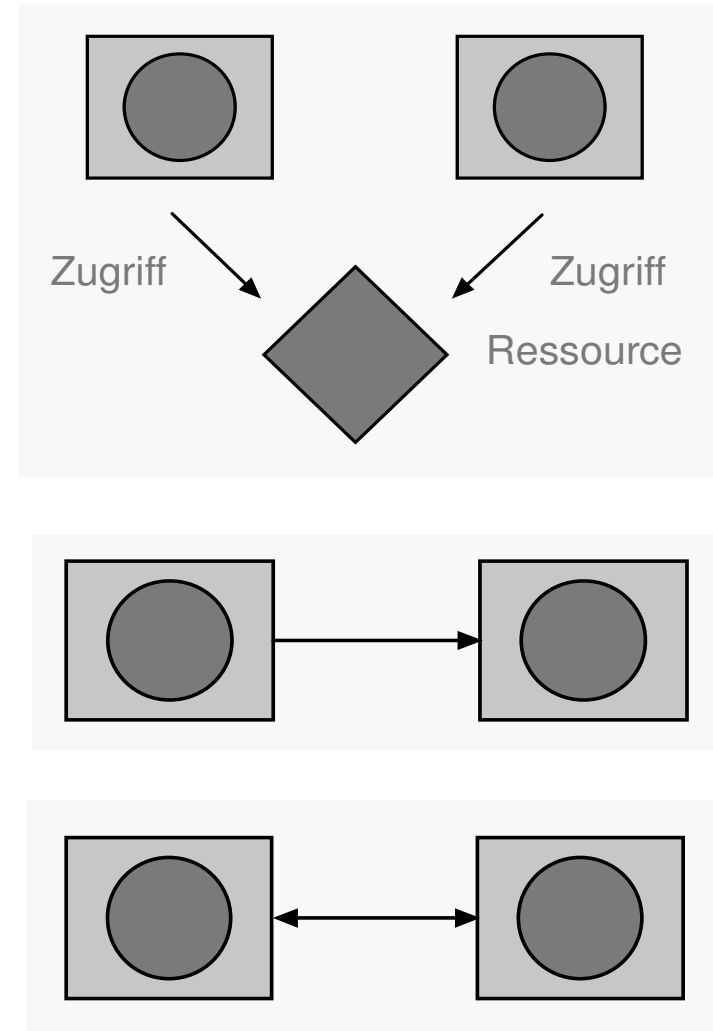
#### ◆ 3 Arten von Interdependenzen zwischen Arbeitsabläufen

##### a Gepoolter Arbeitsablauf

- Gemeinsame Ressourcennutzung

##### b Sequentieller Arbeitsablauf

##### c Reziproker Arbeitsablauf



## 3.2.4 Gestaltung der Aufbauorganisation

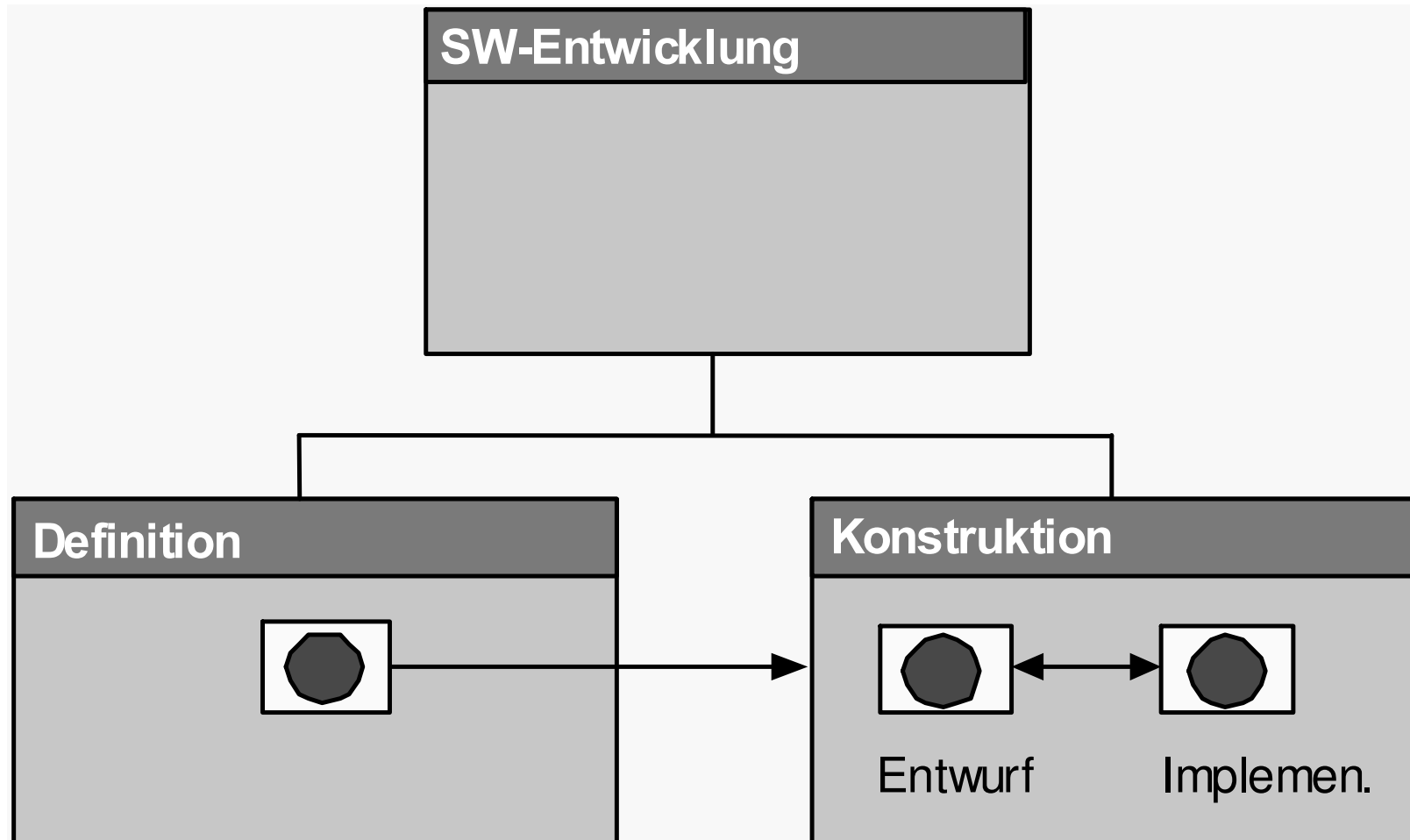
### ▲ Gruppierungskriterien

- **Gruppen auf der untersten Ebene:**
  - Reziproke Interdependenzen berücksichtigen
- **Gruppen auf höheren Ebenen:**
  - Sequentielle Interdependenzen berücksichtigen
- **Gruppen auf der höchsten Ebene:**
  - Gepoolte Interdependenzen berücksichtigen.

### 3.2.4 Gestaltung der Aufbauorganisation

#### ▲ Gruppierungskriterien: Beispiel

- ◆ Aufbauorganisation unter Berücksichtigung von Arbeitsablauf-Interdependenzen



## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Gruppierungskriterien

#### 2 Interdependenzen im Hinblick auf Arbeitsprozesse

- Gruppierung der Positionen, so daß Interaktionen bzgl. der Arbeitsprozesse gefördert werden
- Experten können ihr Wissen besser austauschen
- Kontakt zu Personen außerhalb der Fachrichtung wird eingeschränkt
- Beispiel
  - Vom Arbeitsablauf her sollten Software-Ergonomen der Gruppe Systemanalyse zugeordnet werden
  - Vom Arbeitsprozeß her kann es sinnvoll sein, alle Ergonomen in einer Gruppe zusammenzufassen
- Interdependenzen im Zusammenhang mit der Spezialisierung können eine Gruppierung nach Funktionen nahelegen.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Gruppierungskriterien

- 3 Interdependenzen im Hinblick auf wirtschaftlich optimale Arbeitsbereiche
  - Gruppen werden so eingeteilt, daß ihre Größe effizientes Arbeiten zuläßt
- 4 Interdependenzen im Hinblick auf soziale Arbeitsbeziehungen
  - Mitarbeiter sollen gut miteinander auskommen
- ◆ Diese 4 Kriterien bei der Gruppierung nach Funktionen bzw. Märkten beachten.



## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Gruppierung nach Funktionen

- ◆ Berücksichtigt Interdependenzen im Hinblick auf Arbeitsprozesse und optimale Arbeitsbereiche
- ◆ Vernachlässigt Interdependenzen beim Arbeitsablauf
- ◆ Spezialisierung wird gefördert
  - Aufstiegsmöglichkeiten für Spezialisten
- ◆ Mitarbeiter sind nicht an den Zielsetzungen der gesamten Organisation ausgerichtet
- ◆ Leistung ist schwieriger zu messen.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Gruppierung nach Funktionen

- ◆ Kein eingebauter Mechanismus zur Koordinierung des Arbeitsablaufs
- ◆ Gegenseitige Abstimmung unter verschiedenen Spezialisten wird erschwert
- ◆ Persönliche Weisung durch die Führungskräfte wird behindert
- ◆ Zusätzliche Koordinationsinstrumente werden benötigt
- ◆ Es entsteht zusätzlicher Bürokratismus
- ◆ Es werden mehr Führungskräfte benötigt.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Gruppierung nach Märkten

- ◆ Einheiten umfassen alle sequentiellen und reziproken Interdependenzen
- ◆ Ressourcen und bestimmte Hilfsdienste werden von einer gemeinsamen Struktur bezogen
- ◆ Führt alle Funktionen für eine Kategorie von Produkten, Dienstleistungen, Kunden oder Orten aus
- ◆ Leistung der Einheit ist leicht zu beurteilen
- ◆ Organisation in der Regel weniger bürokratisch.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Gruppierung nach Märkten

- ◆ Spezialisierung von Arbeitsprozessen nur in geringem Umfang möglich
- ◆ Für spezialisierte oder repetitive Aufgaben weniger geeignet
- ◆ Gut für viele verschiedene Aufgaben
- ◆ Kann flexibel auf neue Aufgaben reagieren, da Einheiten relativ unabhängig voneinander sind
- ◆ Selbstbewußtsein der Mitarbeiter kann abnehmen
- ◆ Höherer Ressourcenverbrauch.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Empfehlungen

#### ◆ Marktorientierte Struktur

- Wenn die mit dem Arbeitsablauf verbundenen Interdependenzen entscheidend sind und sich durch Standardisierung nicht gut erfassen lassen

#### ◆ Funktionsorientierte Struktur

- Wenn die Interdependenzen beim Arbeitsablauf sich durch Standardisierung leicht auffangen lassen und wirtschaftlich optimale Arbeitsbereiche im Vordergrund stehen.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Beispiel: Software-Haus

- ◆ Es wird Standardsoftware für ausgewählte Branchen entwickelt

1\_ Zusammenstellung sämtlicher Aufgaben

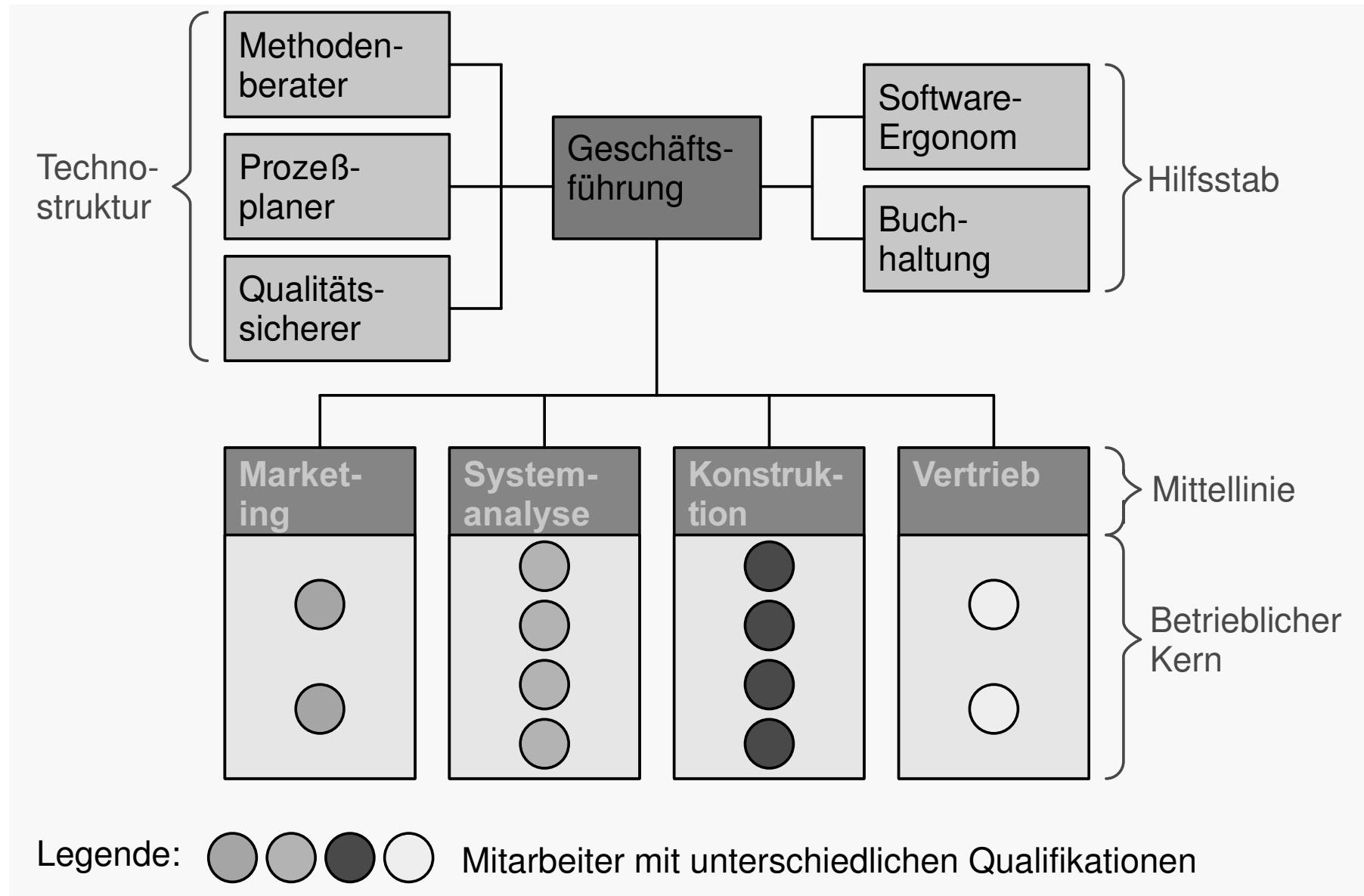
2 Zuordnung zu Positionen

3 Gruppierung von Positionen

- Gruppierung des Software-Hauses nach Funktionen
- Gruppierung des Software-Hauses nach Märkten.

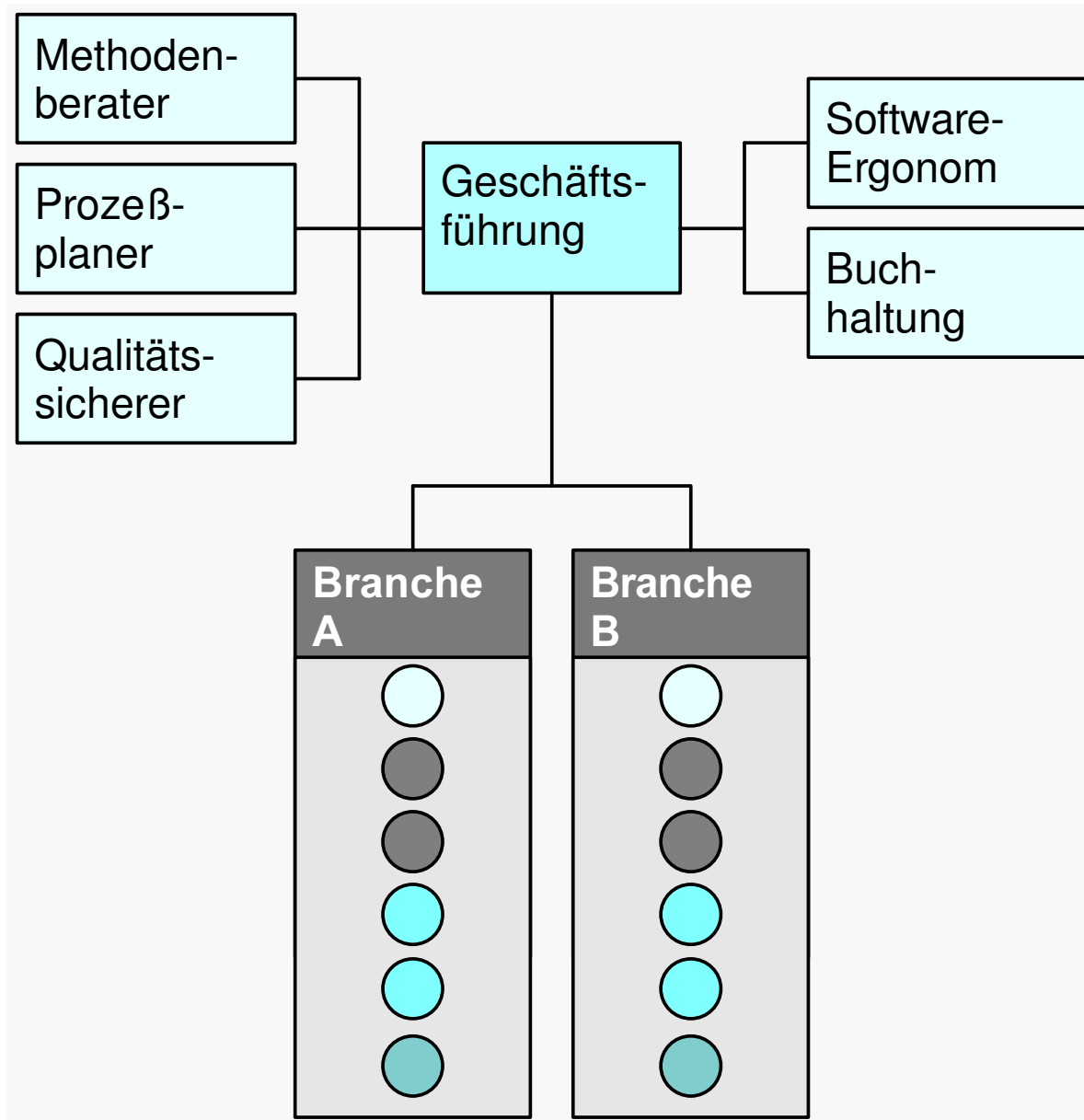
### 3.2.4 Gestaltung der Aufbauorganisation

#### ◆ Gruppierung des Software-Hauses nach Funktionen



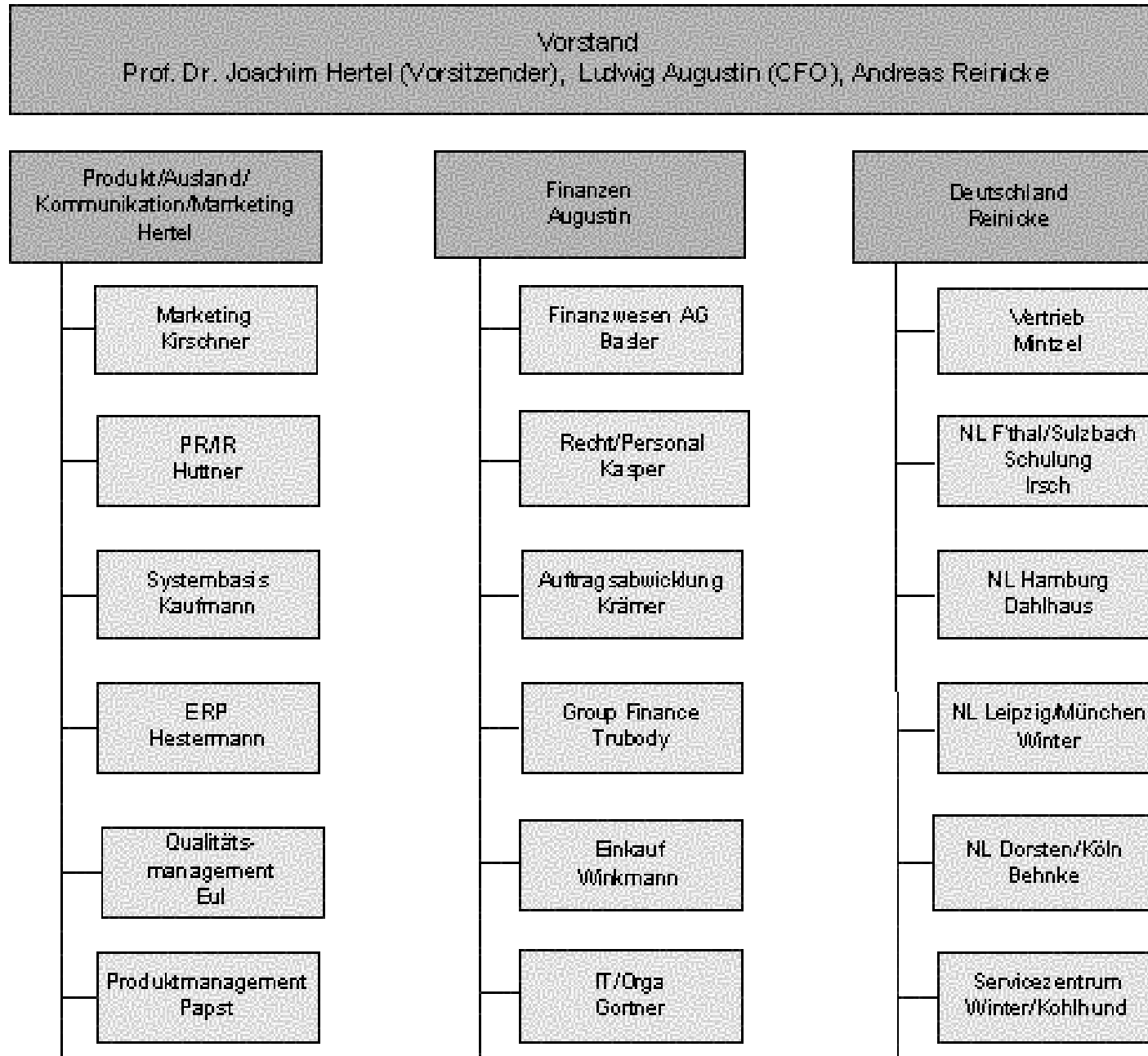
### 3.2.4 Gestaltung der Aufbauorganisation

#### ◆ Gruppierung des Software-Hauses nach Märkten





## Organisationsbeispiel infor AG



## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Größe der Einheiten

#### ◆ Größe wird gefördert durch...

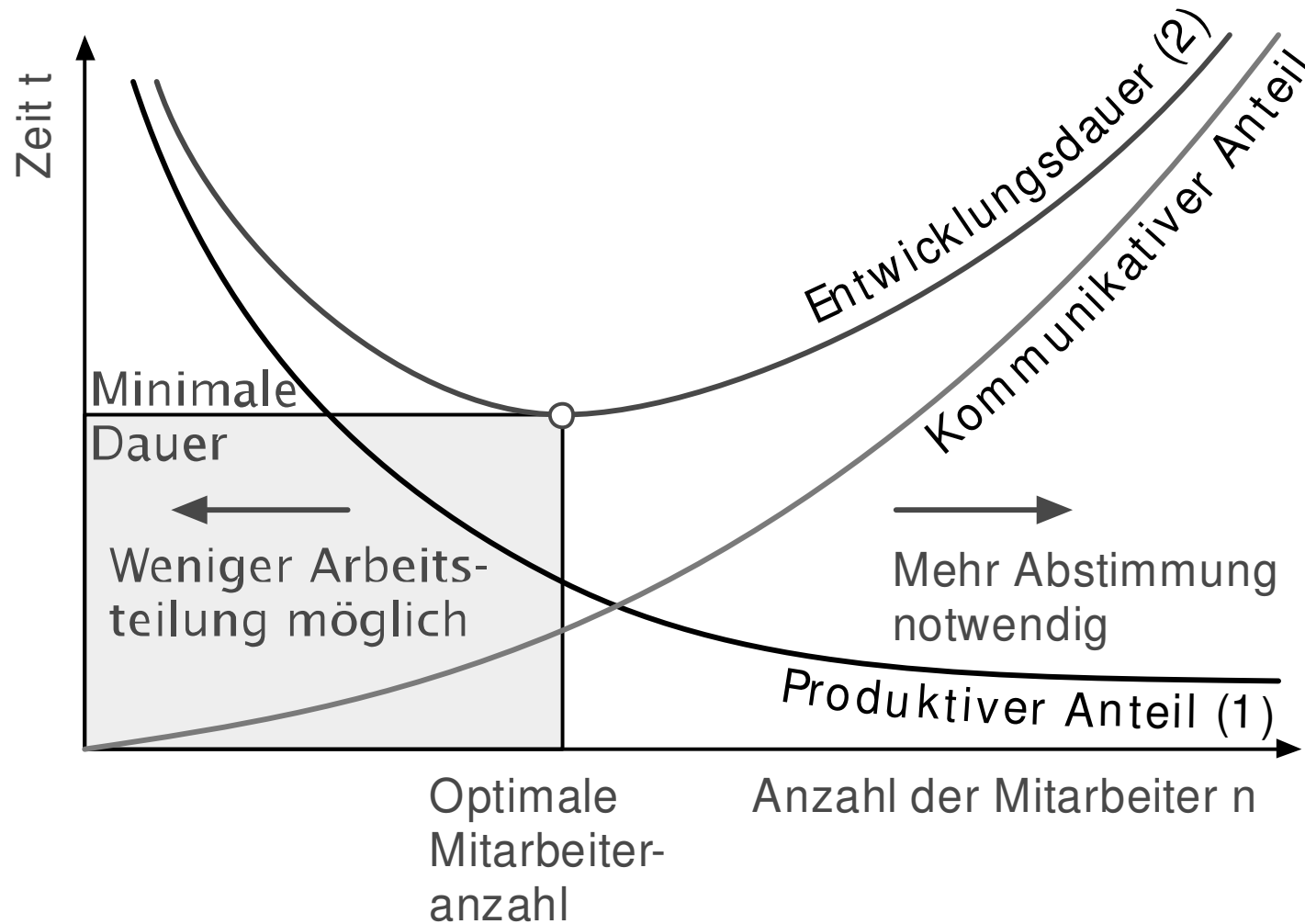
- Standardisierung
- gute Ausbildung der Mitarbeiter

#### ◆ Größe wird erschwert durch...

- interdependente Aufgaben, die durch gegenseitige Abstimmung koordiniert werden müssen
- notwendige persönliche Weisung.

# 3.2.4 Gestaltung der Aufbauorganisation

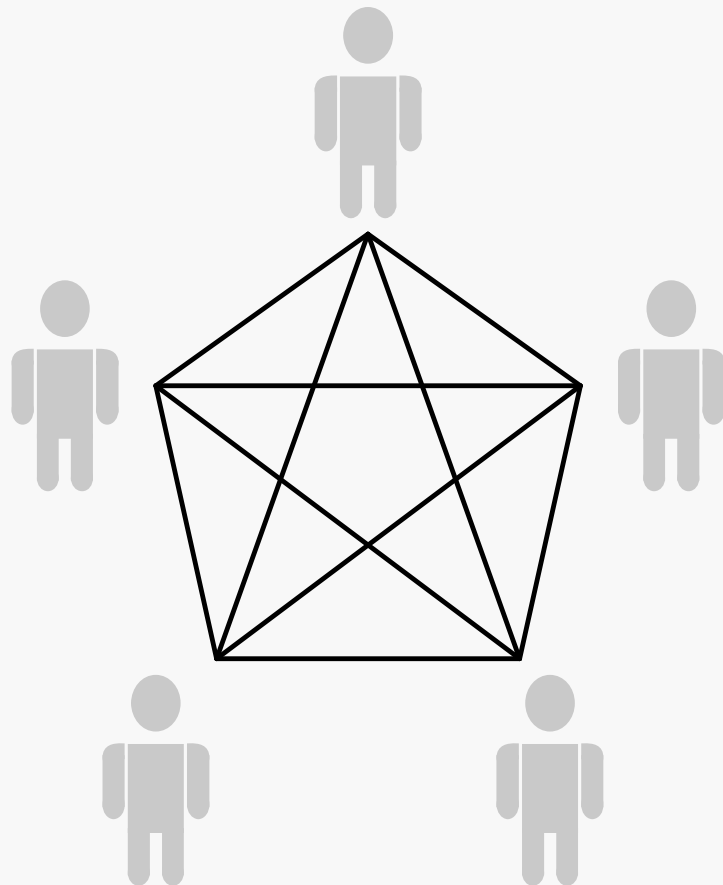
## ▲ Kommunikationsaufwand



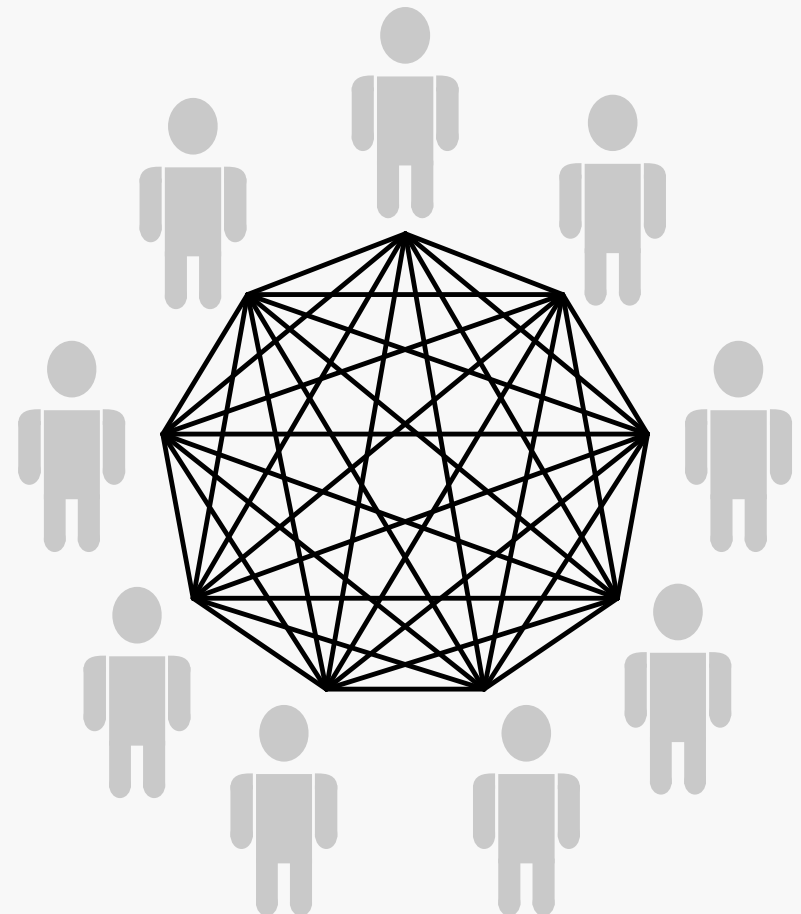
### 3.2.4 Gestaltung der Aufbauorganisation

#### ▲ Grafische Darstellung der Kommunikationspaare bei 5 und 9 MA

(10 Kommunikationspaare)



(36 Kommunikationspaare)



## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Kommunikation

- ◆ Kann ein Produkt ohne Kommunikation erstellt werden, gilt:
  
- ◆ Unter Berücksichtigung des Kommunikationsaufwandes  $k$  pro Kommunikationspaar ergibt sich:

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Brooks'sches Gesetz

- ◆ Adding manpower to a late software project makes it later
  - Zusätzliche Mitarbeiter müssen von den vorhandenen geschult und eingearbeitet werden
  - Es muß eine weitere Aufgabenteilung und erneute Abstimmung vorgenommen werden
- Die Produktivität der alten Mitarbeiter sinkt
- Die optimale Mitarbeiterzahl muß rechtzeitig zur Verfügung stehen.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Zur Historie

- ◆ Prof. Dr. Frederick P. Brooks, Jr.
- ◆ \*1931 in Durham
- ◆ USA, Wegbereiter des Software-Managements
- ◆ Buch »*The Mythical Man-Month*« 1975
- ◆ Promotion an der Harvard-Universität 1956
- ◆ langjähriger Mitarbeiter der Firma IBM
- ◆ heute: Professor für Informatik an der *University North Carolina*.



## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Größe der Einheiten

#### ◆ Größere Einheiten bilden, wenn...

- 1 alle drei Formen der Standardisierung vorliegen
- 2 die ausführenden Aufgaben in einer Einheit ähnlich sind (in Software- Entwicklung : ja)
- 3 die Mitarbeiter Wert auf Selbständigkeit und Selbstverwirklichung legen (in Software- Entwicklung : ja)
- 4 die Informationsvermittlung von unten nach oben optimal funktionieren soll (in Software- Entwicklung : ja).



## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Größe der Einheiten

#### ◆ Kleinere Einheiten bilden, wenn...

- 1 eine straffe persönliche Weisung notwendig ist
- 2 komplexe interdependente Aufgaben eine gegenseitige Abstimmung erfordern  
(in Software-Entwicklung: ja)
- 3 die Führungskraft einer Einheit neben der Kontrollfunktion viele andere Aufgaben hat
- 4 die Mitglieder der Einheit häufige Beratung beim Vorgesetzten suchen.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Größere Einheiten in der Software-Entwicklung

- ◆ Durch geeignete Software-Methoden muß die Interdependenz zwischen Aufgaben drastisch reduziert werden, so daß eigenständige Aufgaben entstehen.

## 3.2.4 Gestaltung der Aufbauorganisation

### ▲ Aussagen zur Größe der Einheiten

- ◆ **Größten Einheiten sind im betrieblichen Kern, da dort die Koordination vorwiegend durch Standardisierung der Arbeitsprozesse erfolgt**
- ◆ **Nur wenige funktionsorientierte Einheiten lassen sich in einer übergeordneten Einheit zusammenfassen**
- ◆ **Gegenteil bei marktorientierten Einheiten**
- ◆ **Führungshierarchie wird zur Spitze hin immer steiler, da mehr gegenseitige Abstimmung**
- ◆ **Bei umfangreicher Technostruktur und vielen Hilfsstabseinheiten sind in der Mittellinie kleine Einheiten zu bilden**
- ◆ **Bei professionellen Stabseinheiten kleine Gruppen.**

## 3.2.5 Projektleiter und Matrixstrukturen

### ▲ Aufbauorganisation

- ◆ Um laterale Verbindungsstrukturen ergänzen

### ▲ Kontaktinstrumente

- ◆ Um eine reibungslose gegenseitige Abstimmung sicherzustellen
- ◆ Möglichkeiten:
  - Kontaktpositionen
  - Arbeitskreise
  - permanente Ausschüsse
  - Matrixstrukturen
- ◆ Beispiel: In der Fachabteilung, die die Software-Abteilung beauftragt, wird die Position Software-Beauftragter eingerichtet.

## 3.2.5 Projektleiter und Matrixstrukturen

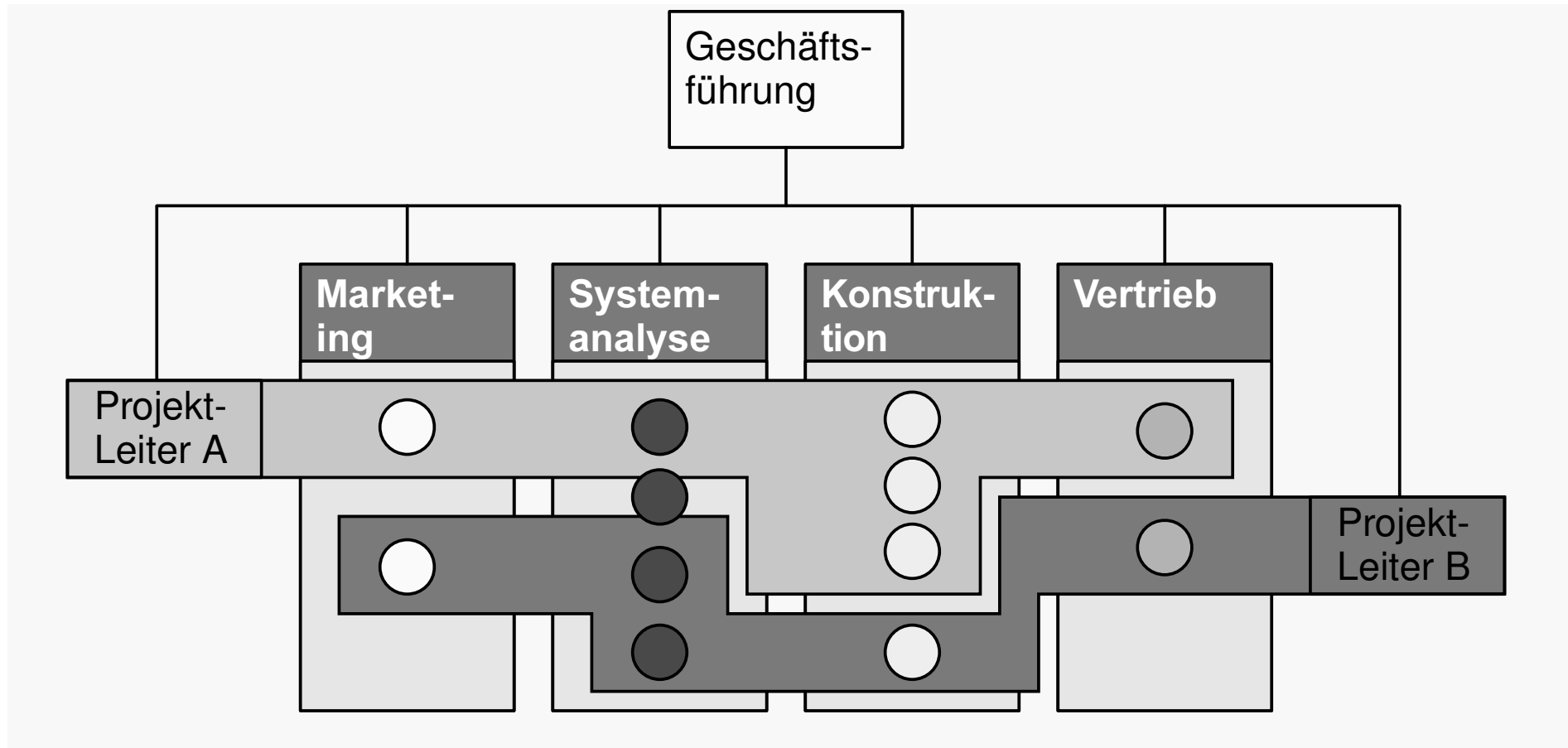
### ▲ Projektleiter

- ◆ Wird als Kontaktposition eingerichtet, wenn viel Koordination durch gegenseitige Abstimmung erforderlich ist
- ◆ Erhält formale Autorität für einige Aspekte des Entscheidungsprozesses
- ◆ Hat keine formale Autorität über die Mitarbeiter
- ◆ Muß das Verhalten der Mitarbeiter koordinieren durch
  - Entscheidungsautorität
  - Überzeugungskraft
  - Verhandlungsgeschick.

### 3.2.5 Projektleiter und Matrixstrukturen

#### ▲ Projektleiter: Beispiel

- ◆ Funktionsorientierte Struktur ergänzt um marktorientierte Projektleiter



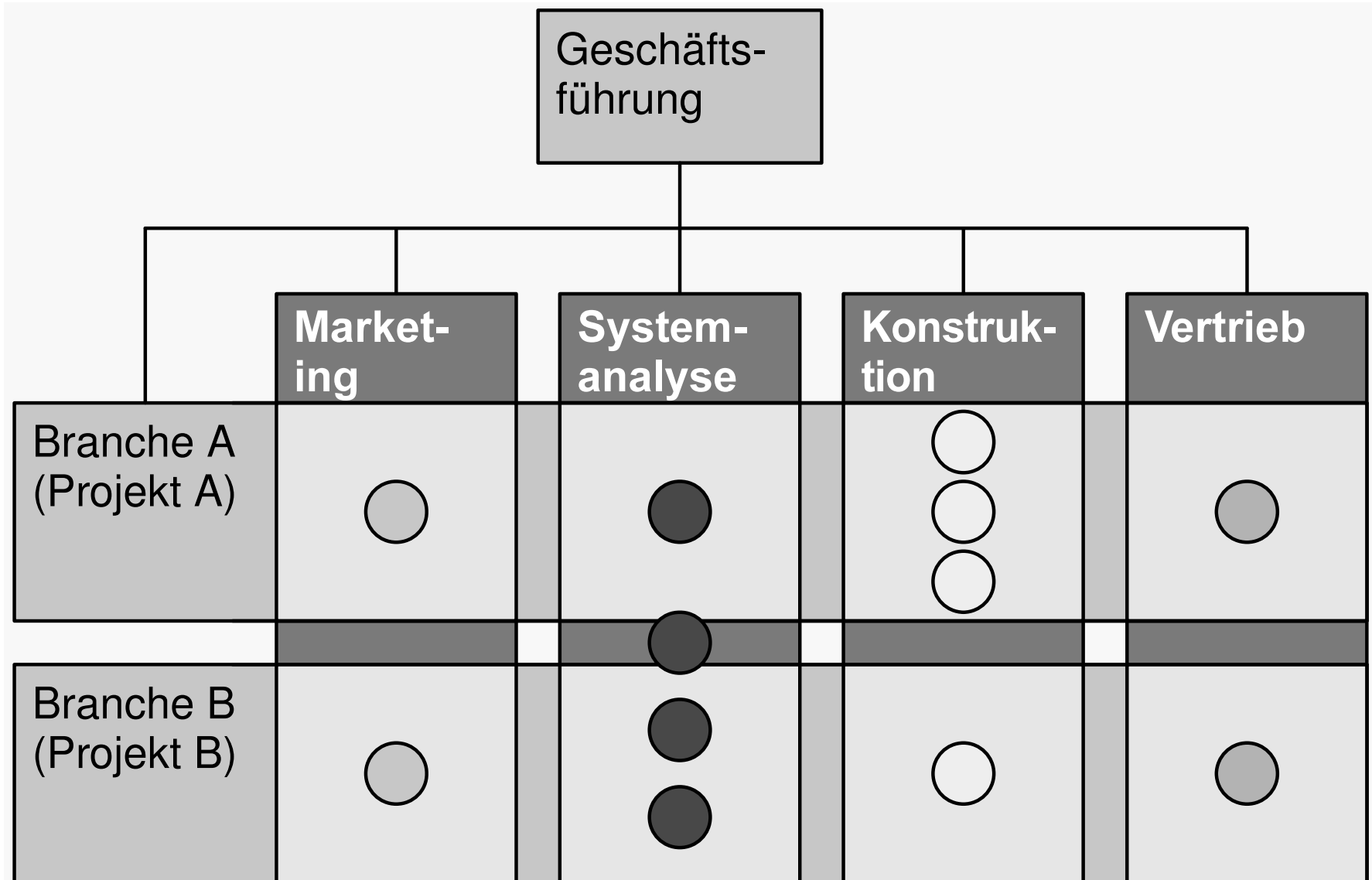
## 3.2.5 Projektleiter und Matrixstrukturen

### ▲ Matrixstrukturen

- ◆ Beide Gruppierungsalternativen beibehalten
  - funktionsorientiert
  - marktorientiert
- ◆ Permanente Matrixstruktur
  - Interdependenzen bleiben stabil
  - Zusammensetzung der Einheiten und Mitarbeiter bleiben konstant
- ◆ Alternierende, projektorientierte Matrixstruktur
  - Interdependenzen wechseln häufig
  - Projektteams werden eingerichtet
  - Für Unternehmen, die ständig wechselnde Produkte herstellen.

### 3.2.5 Projektleiter und Matrixstrukturen

#### ▲ Beispiel für eine Matrixstruktur





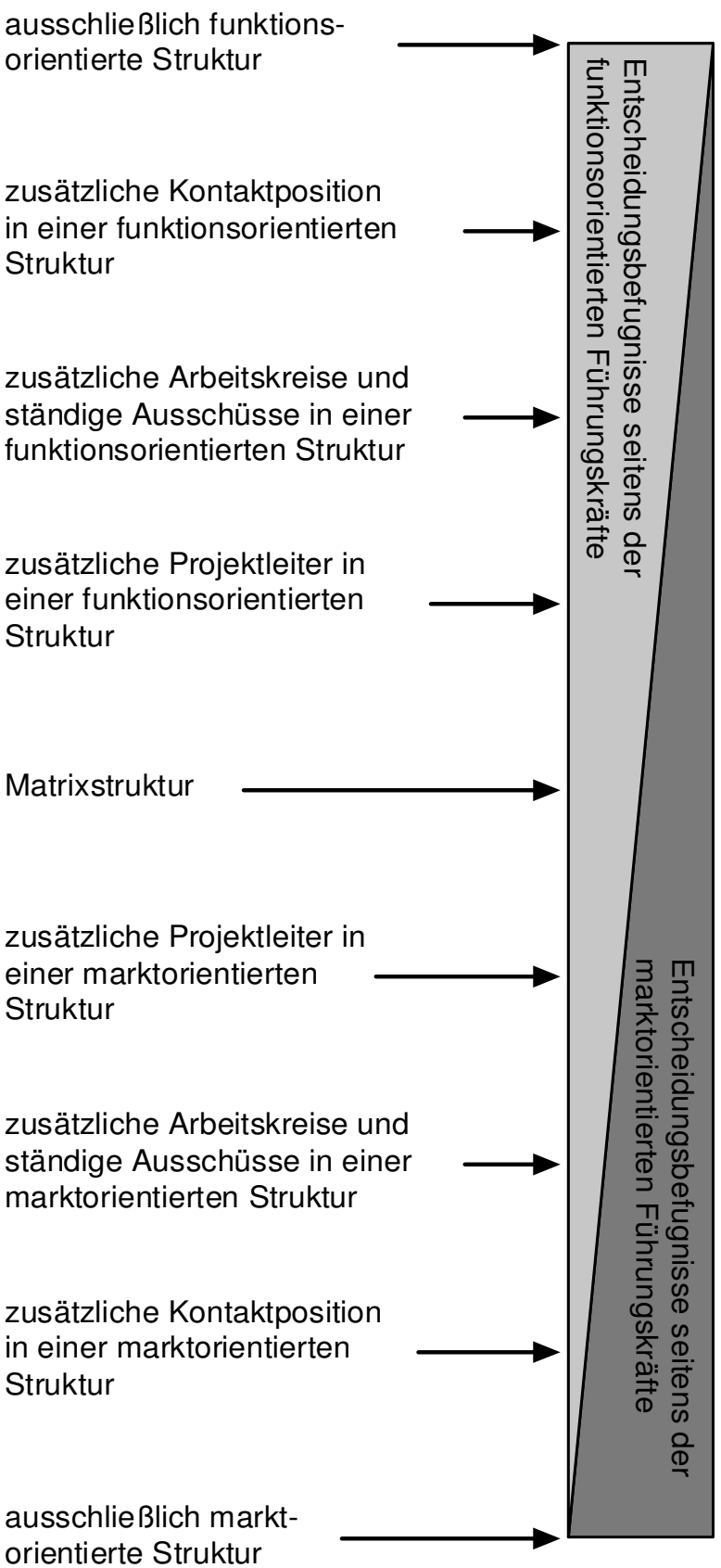
## 3.2.5 Projektleiter und Matrixstrukturen

### ▲ Matrixstrukturen

- ◆ **Sind effektiv für...**
  - **Entwicklung neuer Aktivitäten**
  - **Koordinierung komplexer und multipler Interdependenzen**
- ◆ **Sind nicht effektiv für...**
  - **Organisationen mit hohem Anspruch an Sicherheit und Stabilität**
  - **Firmen mit empfindlichen Gleichgewichten zwischen den Machtbefugnissen der Führungskräfte**
- ◆ **Benötigen mehr Zeit für Sitzungen und mehr Führungskräfte.**

### 3.2.5 Projektleiter und Matrixstrukturen

## ▲ Kontinuum von der rein funktionalen Struktur zur rein marktorientierten Struktur



### 3.2.6 Situative Faktoren

- ▲ **Die Struktur einer erfolgreichen Organisation ist konsistent mit den äußeren Bedingungen**
- ▲ **Zunehmendes Alter und/oder Größe führen zu einer stärkeren Formalisierung der Struktur.**

## 3.2.6 Situative Faktoren

### ▲ Technisches System beeinflusst die Struktur

#### ◆ Einzelfertigung

- Keine Standardisierung der Produkte und Verfahren
- Organische Struktur der Betriebe
- Vorgesetzte führen wenige Mitarbeiter

#### ◆ Massenfertigung

- Technische Systeme sind standardisiert
- Verhaltensformalisierung und Bürokratie
- Im betrieblichen Kern hauptsächlich Routinearbeiten
- Ausgebaute Technostruktur.

### 3.2.6 Situative Faktoren

#### ◆ Prozeßfertigung

- **Weitgehend automatisches technisches System**
- **Regeln, Vorschriften und Standards gelten für Maschinen, nicht Mitarbeiter**
- **Viele Spezialisten werden zur Gestaltung des technischen Systems gebraucht**
- **Betrieblicher Kern besteht aus hochqualifizierten indirekten Arbeitskräften, z.B. Wartungsexperten.**

### 3.2.6 Situative Faktoren

#### ▲ **Bezug zur Software-Technik**

- ◆ **Software-Entwicklung beinhaltet Elemente der verschiedenen Produktionssysteme**
- ◆ **Software-Entwicklung ist eine standardisierte Einzelfertigung mit anspruchsvollen, komplexen, teilweise formalisierten Routinetätigkeiten**
- ◆ **Teilbereiche sind automatisiert.**

### 3.2.6 Situative Faktoren

#### ▲ Umwelteinflüsse

##### ◆ Stabilität

- Ist die Umwelt dynamisch, muß die Struktur organisch sein
- Die Umkehrung gilt nicht!

##### ◆ Komplexität

- Je komplexer die Techniken, desto dezentralisierter ist die Struktur
- Software-Entwicklung:
  - Um Software zu erstellen, ist es erforderlich, komplexe Software-Techniken zu beherrschen.

### 3.2.6 Situative Faktoren

- ◆ Die Faktoren **Stabilität und Komplexität** führen zu **4 Organisationstypen**

	Stabile Umwelt	Dynamische Umwelt
Komplexe Umwelt (komplexe Technologie)	dezentralisiert bürokratisch (Standardisierung von Qualifikationen) Beispiele: Universitäten, Krankenhäuser	dezentralisiert organisch (gegenseitige Abstimmung)
Einfache Umwelt (einfache Technologie)	zentralisiert bürokratisch (Standardisierung von Arbeitsprozessen)	zentralisiert organisch (persönliche Weisung)



## 3.2.6 Situative Faktoren

### ▲ Die Umwelt der Software-Technik

- ◆ Die Umwelt der Software-Technik ist immer komplex
- ◆ Komplexe, stabile Umwelt
  - Bürokratische, dezentralisierte Struktur
  - Koordination durch Qualifikationen
  - Beispiele: Universität, Krankenhaus
- ◆ Komplexe, dynamische Umwelt
  - Schnelle Reaktion auf nicht vorhersehbare Ereignisse erforderlich
  - Entscheidungsbefugnisse an Spezialisten
  - Koordination durch gegenseitige Abstimmung.

## 3.2.6 Situative Faktoren

### ▲ Umwelteinflüsse

#### ◆ Marktdiversität

- **Marktorientierte Struktur, wenn...**

- **Produkte vielfältig**
- **Dienstleistungen vielfältig**
- **Kunden vielfältig**

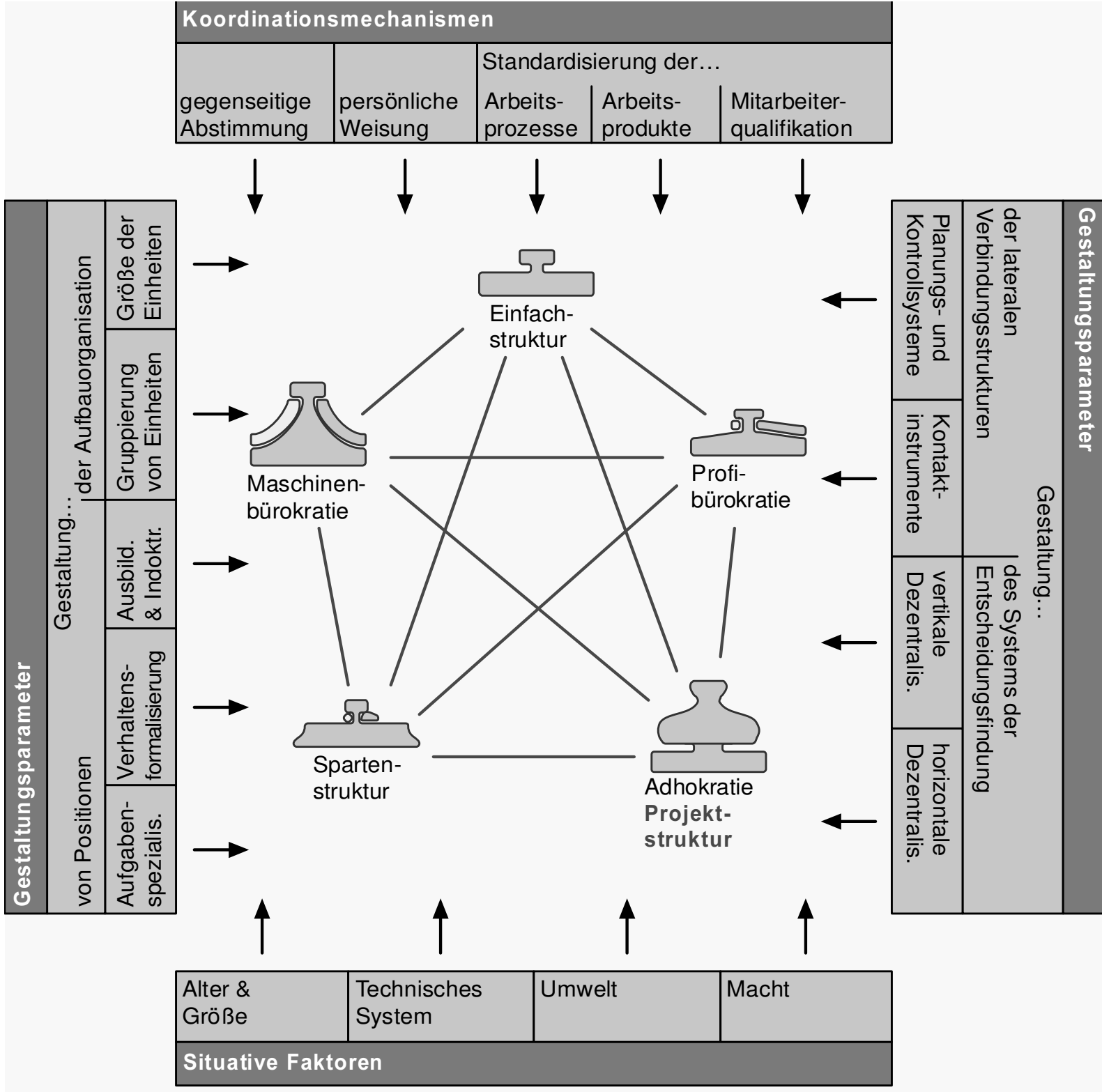
#### ◆ Macht

- **Je größer die von außen ausgeübte Kontrolle ist, desto zentralisierter und formalisierter wird die Struktur einer Organisation.**

### 3.2.7 Die Projektstruktur

#### ▲ 5 Organisationsstrukturen:

- ◆ Einfachstruktur
- ◆ Maschinenbürokratie
- ◆ Spartenstruktur
- ◆ Projektstruktur
  - relevant für das Software-Management
- ◆ Profibürokratie
  - relevant für das Software-Management.



## 3.2.7 Die Projektstruktur

### ▲ Projektstruktur

- ◆ Entwickelt mit Projektteams anspruchsvolle Innovationen
- ◆ Eigenschaften
  - Ausgeprägte organische Struktur mit geringer Verhaltensformalisierung
  - Hohe horizontale Aufgabenspezialisierung
  - Tendenz zur verwaltungsinternen Gruppierung der Mitarbeiter in funktionale Einheiten, aber auch marktorientierten Projektteams (projektorientierte Matrixstruktur)
  - Verwendung von Kontaktinstrumenten zur Förderung der gegenseitigen Abstimmung.

### 3.2.7 Die Projektstruktur

#### ▲ Projektmanager

- ◆ **Leitet die Projektteams**
- ◆ **Keine klassischen Führungskräfte**
- ◆ **Widmen sich der Kontaktpflege und Verhandlungsführung**
- ◆ **Oft selbst Experten, die im Team mitarbeiten**
- ◆ **Erledigen administrative und betriebliche Arbeiten.**

### 3.2.7 Die Projektstruktur

#### ▲ Projektstruktur

- ◆ **Keine Technostruktur**
  - **Da keine Koordination durch Standardisierung**
- ◆ **Existiert nur solange das Projekt existiert**
- ◆ **Führungskräfte benötigen viel Zeit, da innovative Projekte schwer zu kontrollieren sind**
- ◆ **Kann mit einer dynamischen und komplexen Umwelt umgehen**
- ◆ **Tendiert mit zunehmendem Alter zur Bürokratisierung.**

### 3.2.7 Die Projektstruktur

- ◆ **Beispiele**
  - **Forschungsorientierte Organisationen**
  - **Forschungsabhängige High-Tech-Industrien**
  - **Unternehmen mit häufigem Produktwechsel**
  - **Innovative Beratungsfirmen**
- ◆ **Die Arbeit ist komplex, nicht vorhersehbar und oft wettbewerbsintensiv**
- ◆ **Häufig ist jeder Kundenauftrag ein neues Projekt**
- ◆ **Ein Produkt wird in Einzelfertigung hergestellt.**



### 3.2.7 Die Projektstruktur

#### ▲ Probleme

- ◆ Individuelle Ziele müssen dem Team untergeordnet werden
- ◆ Struktur ist nicht effizient

#### ▲ Übertragbarkeit auf Software-Entwicklung

- ◆ In der Regel keine Innovationen
- ◆ Produkte lassen sich bestimmten Klassen zuordnen
- ◆ Oft Entwicklung für anonymen Markt
- ◆ Umwelt oft stabil
- ◆ Arbeit ist komplex, aber vorhersehbar
- ◆ Arbeitsprozeß teilweise standardisiert.

## 3.2.8 Die Profibürokratie

### ▲ Profibürokratie

- ◆ **Koordination durch Standardisierung der Qualifikationen**
- ◆ **Betrieblicher Kern am wichtigsten**
- ◆ **Mitarbeiter arbeiten alleinverantwortlich**
- ◆ **Arbeitsaufgaben sind horizontal spezialisiert, aber vertikal erweitert**
- ◆ **Komplexe, aber stabile Umwelt**
- ◆ **Technische Systeme sind nicht regulativ und nicht kompliziert**
- ◆ **Die Struktur ist funktions- und marktorientiert (Matrixstruktur).**

### 3.2.8 Die Profibürokratie

#### ▲ Profibürokratie sinnvoll, wenn...

- ◆ betrieblicher Kern einer Organisation überwiegend aus hochqualifizierten, professionellen Mitarbeitern besteht, die schwer zu erlernende, aber gut zu definierende Verfahren anwenden
- ◆ Umwelt so komplex ist, daß schwierige Verfahren erforderlich sind
- ◆ Umwelt so stabil ist, daß die von den Mitarbeitern zu erwartenden Qualifikationen gut zu definieren und folglich auch zu standardisieren sind.

## 3.2.8 Die Profibürokratie

### ▲ **Eigenschaften der Software-Entwicklung, die für eine Profibürokratie sprechen**

- ◆ **Erhebliche Ermessensfreiheit bei der Ausführung der Arbeiten**
- ◆ **Noch keine Standardisierung der Arbeitsprozesse und Arbeitsprodukte**
- ◆ **Anwendung von Standardverfahren zur Lösung wohldefinierter Probleme**
- ◆ **Kategorisierung der Probleme und Auswahl von Standardverfahren**
  - **kommerzielle Anwendungen**
  - **technische Anwendungen**
  - **Echtzeit-Anwendungen.**

### 3.2.8 Die Profibürokratie

- ▲ **Eigenschaften der Software-Entwicklung, die gegen eine Profibürokratie sprechen**
  - ◆ **Die Ausbildung ist noch nicht standardisiert**
    - **Daher reicht die Standardisierung der Qualifikationen zur Koordinierung nicht aus**
  - ◆ **Mitarbeiter können nicht allein und autonom arbeiten, um das Produkt oder die Dienstleistung zu erbringen**
    - **Gegenseitige Abstimmung ist nötig**
  - ◆ **Es wird ein kompliziertes, z.T. automatisiertes technisches System eingesetzt**
    - **CASE-Umgebungen.**

### 3.2.9 Mischstrukturen

#### ▲ **Software-Entwicklung**

- ◆ **Weder Projektstruktur noch Profibürokratie berücksichtigen vollständig die Charakteristika einer Software-Entwicklung**

#### ▲ **Zwischen einer Projektstruktur und einer Profibürokratie gibt es Übergänge und Mischformen.**

### 3.2.9 Mischstrukturen

#### ▲ Charakteristika verschiedener Strukturen (1/5)

Charakteristika	<b>Software-Entwicklung</b>	Projektstruktur	Profibürokratie
Vorrangiger Koordinationsmechanismus	gegenseitige Abstimmung, Standardisierung der Qualifikationen und Arbeitsprozesse	gegenseitige Abstimmung	Standardisierung der Qualifikationen
Wichtigster Organisations- teil	betrieblicher Kern, Hilfsstab, Technostruktur	betrieblicher Kern und Hilfsstab	betrieblicher Kern

## 3.2.9 Mischstrukturen

### ▲ Charakteristika verschiedener Strukturen (2/5)

Charakteristika	Software- Entwicklung	Projektstruktur	Profibürokratie
<b>Gestaltungsparameter:</b>			
Aufgaben-Spezialisierung	horizontale	Spezialisierung, vertikale Erweiterung	
Ausbildung		viel Ausbildung	
Verhaltensformalisierung	beschränkt/ organisch & bürokratisch	Kaum/ organisch	Kaum/ bürokratisch
Gruppierung		funktional und marktorientiert	
Größe der Einheiten	eher klein	überall klein	groß unten, sonst klein
Kontaktinstrumente		überall & viel	in der Administration



### 3.2.9 Mischstrukturen

#### ▲ Charakteristika verschiedener Strukturen (3/5)

Charakteristika	Software-Entwicklung	Projektstruktur	Profibürokratie
<b>Funktionen:</b>			
Strategische Spitze		externe Kontakte, Konfliktlösung, Arbeitsauslastung, Projektüberprüfung	externe Kontakte, Konfliktlösung
Betrieblicher Kern	qualifizierte, grob standardisierte Arbeit im Team mit innovativen, repetitiven und routinehaften Aspekten	qualifizierte, innovative Arbeit, multidisziplinäre Arbeit im Team	qualifizierte, standardisierte Arbeit, viel individuelle Autonomie, weitgehend Einzelarbeit

### 3.2.9 Mischstrukturen

#### ▲ Charakteristika verschiedener Strukturen (4/5)

Charakteristika	Software-Entwicklung	Projektstruktur	Profibürokratie
<b>Funktionen:</b>			
Mittellinie		umfassend, Trennung vom Stab; Beteiligung an Projektarbeit	kontrolliert durch professionelle Mitarbeiter, gegenseitige Abstimmung
Technostruktur	Methodenberater, Prozeßplaner, Qualitätsplaner	klein und mit Projektarbeit vermischt	kaum
Hilfsstab	z.B. Software-Ergonom	ausgebaut, aber mit Projektarbeit vermischt	ausgebaut zur Unterstützung der professionellen Mitarbeiter

### 3.2.9 Mischstrukturen

#### ▲ Charakteristika verschiedener Strukturen (5/5)

Charakteristika	Software-Entwicklung	Projektstruktur	Profibürokratie
<b>Situative Faktoren</b>			
Technisches System	regulativ, kompliziert, teilweise automatisiert	nicht regulativ, unkompliziert	
Umwelt	komplex und semi-stabil bzw. semi-dynamisch	individuelle Einzelfertigung	standardisierte Einzelfertigung
Macht	Kontrolle durch Techno- struktur, Experten und externe Kontrolle	Kontrolle durch Experten	Kontrolle durch professionellen betrieblichen Kern

### 3.2.9 Mischstrukturen

▲ **Zusätzlich ist bei einer Software-Entwicklung noch zu berücksichtigen:**

- ◆ **Die Arbeitsprozesse sind zumindest teilweise standardisiert**
- ◆ **Es gibt eine Technostruktur, die zuständig ist für...**
  - **die Innovation**
  - **den Arbeitsprozeß**
  - **die Qualität**
- ◆ **Das technische System ist...**
  - **teilweise regulativ**
  - **kompliziert**
  - **teilweise automatisiert**
    - **CASE-Umgebungen.**

## 3.2.9 Mischstrukturen

### ▲ Eine »optimale« Organisationsstruktur für eine Software-Entwicklung:

- ◆ Horizontal spezialisierte, vertikal erweiterte Aufgaben
- ◆ Definierte Prozeßmodelle mit Zuordnung der durchzuführenden Aufgaben
- ◆ Rechnerunterstützung der Prozeß-Modelle
- ◆ Projektorientierte Matrixstruktur mit relativ kleinen Einheiten
- ◆ Ausgebaute Technostruktur sowie Hilfsstab als Kompetenzzentrum für Spezialaufgaben.

## 3.2.10 Kooperation Fachabteil. – Systemanalyse

### ▲ Probleme

- ◆ Oft »Verwischung« der Verantwortlichkeiten
- ◆ Systemanalyse übernimmt Aufgaben der Fachabteilung
  - Beispiele:
    - Festlegung von Nummernkreisen
    - Festlegung von Voreinstellungen
- ◆ Fachabteilung erstellt in manchen Fällen sogar ein formales Analysemodell
- ◆ Teilweise werden Mitarbeiter aus der Software-Abteilung in die Fachabteilung »abgeworben«.

## **3.2.10 Kooperation Fachabteil. – Systemanalyse**

### **▲ Aufgaben der Fachabteilung (1/2)**

- ◆ **Aufstellung der fachlichen Anforderungen an ein zu entwickelndes Software-Produkt**
- ◆ **Festlegung des Funktions-, Daten-, Leistungs- und Qualitätsumfangs**
- ◆ **Angabe, welche Arbeitsabläufe mit dem Produkt ausgeführt werden**
- ◆ **Festlegung der Standardvoreinstellungen**
- ◆ **Aufstellung der Wünsche und Anforderungen an die Benutzungsoberfläche.**

## **3.2.10 Kooperation Fachabteil. – Systemanalyse**

### **▲ Aufgaben der Fachabteilung (2/2)**

- ◆ Vorgabe von Prioritäten**
- ◆ Erstellung des Benutzerhandbuches in Zusammenarbeit mit der Systemanalyse**
- ◆ Überprüfung der von der Systemanalyse erstellten Modelle und Benutzungsoberflächen des Produkts.**



## 3.2.10 Kooperation Fachabteil. – Systemanalyse

### ▲ Aufgaben der Systemanalyse

- ◆ Erstellung einer Produkt-Definition
  - Pflichtenheft
  - Produktmodell
  - Konzept Benutzungsoberfläche
- ◆ Verständliche, übersichtliche Präsentation des Produkt-Modells
- ◆ Bereitstellung von Prototypen zum Ausprobieren der Konzepte
- ◆ Darstellung der Auswirkungen der Anforderungen auf die Benutzungsoberfläche
- ◆ In kurzfristigen Abständen (ca. halbjährlich) liefern einsatzfähiger Teilsysteme.

## 3.2.10 Kooperation Fachabteil. – Systemanalyse

### ▲ Fachabteilung vs. Systemanalyse

- ◆ Systemanalytiker sollte Projektleiter sein
- ◆ Mitarbeiter der Fachabteilung sollten in den Konzepten der Systemanalyse ausgebildet sein
  - Ausbildung sollte sich auf das Lesen und Verstehen von Modellen konzentrieren.

▲ **Danke!**

▲ **Aufgaben**

▲ **Diese Präsentation bzw. Teile dieser Präsentation enthalten Inhalte und Grafiken des Lehrbuchs der Software-Technik (Band 2) von Helmut Balzert, Spektrum Akademischer Verlag, Heidelberg 1998**

