

# WZB

Wissenschaftszentrum Berlin  
für Sozialforschung



HELMUT SCHMIDT  
UNIVERSITÄT  
Universität der Bundeswehr Hamburg

---

# Formen und Folgen der Softwaregestaltung

Soziotechnische Netzwerkarbeit für die digitale Transformation  
am Beispiel der deutschen Energiewirtschaft

---

**Johannes Sonnenholzner**

Doktorand an der Helmut-Schmidt-Universität/Universität der Bundeswehr Hamburg  
Gast der Forschungsgruppe "Globalisierung, Arbeit und Produktion" am WZB

## Agenda

1. Warum ist Softwaregestaltung relevant?
2. Forschungsfragen/-design
3. Grundlagen: Arbeitsprozess der Softwaregestaltung und sein Verhältnis zur Softwareanwendung
4. Einzelne Ergebnisse
  - Analyserahmen und Idealtypen
  - Darstellung der Fallstudien (Skizzen)
5. Zusammenfassung und weitere Ergebnisse

## Teil der Digitalisierung: Softwaregestaltung

- **Fokus:** industriespezifische Softwareentwicklung
  - zwischen Individual- und Standardsoftware
  - Primat der Softwareentwicklung bei digitalen Start-ups
  - Wettbewerbsfaktor v.a. in softwareintensiven Industrien
  - Abgleich softwaretechnischer Möglichkeiten und industriespezifischer Bedarfe
- **These:** Softwaregestaltung entscheidend
  - was zwischen Anwendung und Programmierung passiert
  - damit Programmierende wissen, was sie programmieren sollen

## 3 Bereiche soziologischer Forschung

**A**

**Arbeitsprozess Softwareanwendung**

**anwendende  
Organisation**

**B**

**Arbeitsprozess Softwaregestaltung**  
(zwischen Anwendung und Programmierung)

?

**C**

**Arbeitsprozess Softwareentwicklung**  
(Programmierung)

**Softwarefirma,  
Entwicklerteam etc.**

## Softwaregestaltung ist kein Nischenthema

- 96.486 ERP-Berater\*innen (2012) und 22.734 IT-Projektleiter\*innen (2012)
- weitere Stellenbezeichnungen:  
Anforderungsmanager\*in, Product Owner\*in, Scrum Master\*in, Applikationsbetreuer\*in, Digitalisierungsmanager\*in, Key User\*in, Anwendungsbetreuer\*innen, IT-Prozessmanager\*in etc.
- 2019 Investitionen von 239 Milliarden Dollar von Firmen in proprietäre Software in den USA (Bessen 2022)

## Forschungsfrage-/design

- Forschungsfrage:
  - Formen: Wie kontrollieren Organisationen den Arbeitsprozess der Softwaregestaltung?
  - Folgen: Welche Auswirkungen hat die jeweilige Form der Softwaregestaltung für die Arbeit der Softwaregestaltenden und für die Arbeitsgestaltung der Softwareanwendung?
- analytische Perspektive: Transformationsproblem von Arbeit bzw. Kontrolle von Arbeit (vgl. Überblick von Marrs 2010, ein klassisches Beispiel Friedman 1977)
  - eigenes Kontrollkonzept entwickeln
- 62 Experten\*inneninterviews zwischen 2020 und 2023 für sieben Fallstudien in der Energiewirtschaft (jeweils zwischen 3 und 17 Interviews)

## Formen des Arbeitsprozess der Softwaregestaltung

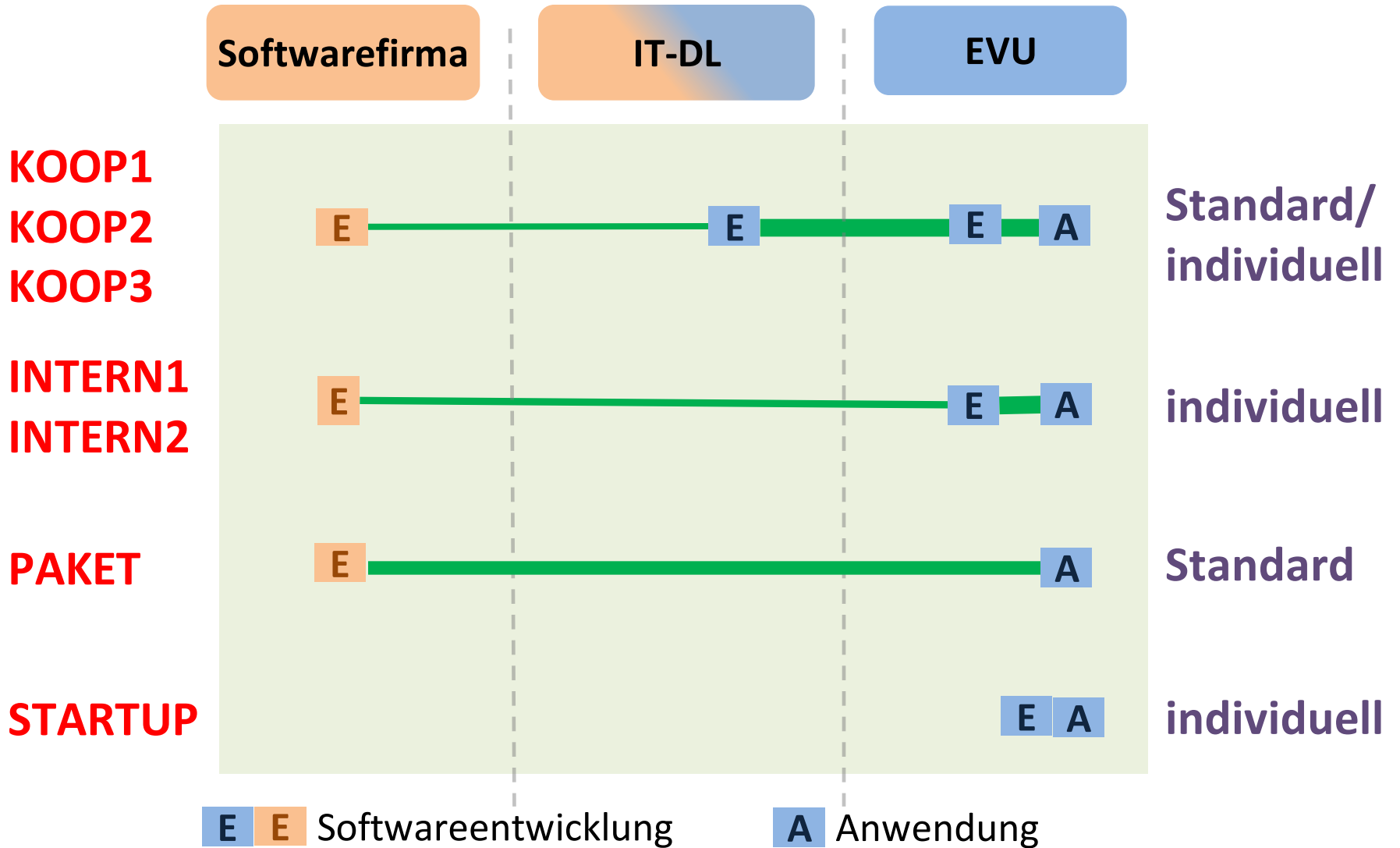
- Kernprobleme der Softwaregestaltung
  - softwaretechnische Interdisziplinarität
  - softwaretechnische Gestaltungsmöglichkeiten
    - a) Zuschnitt: individuell – Standard
    - b) Ausrichtung anwendende Organisation: auf Softwaregestaltung oder auf Softwareanwendung
- zentrale Arbeitsmittel: Wissen und Kommunikation

## Folgen als Verhältnis von Softwaregestaltung zur - anwendung

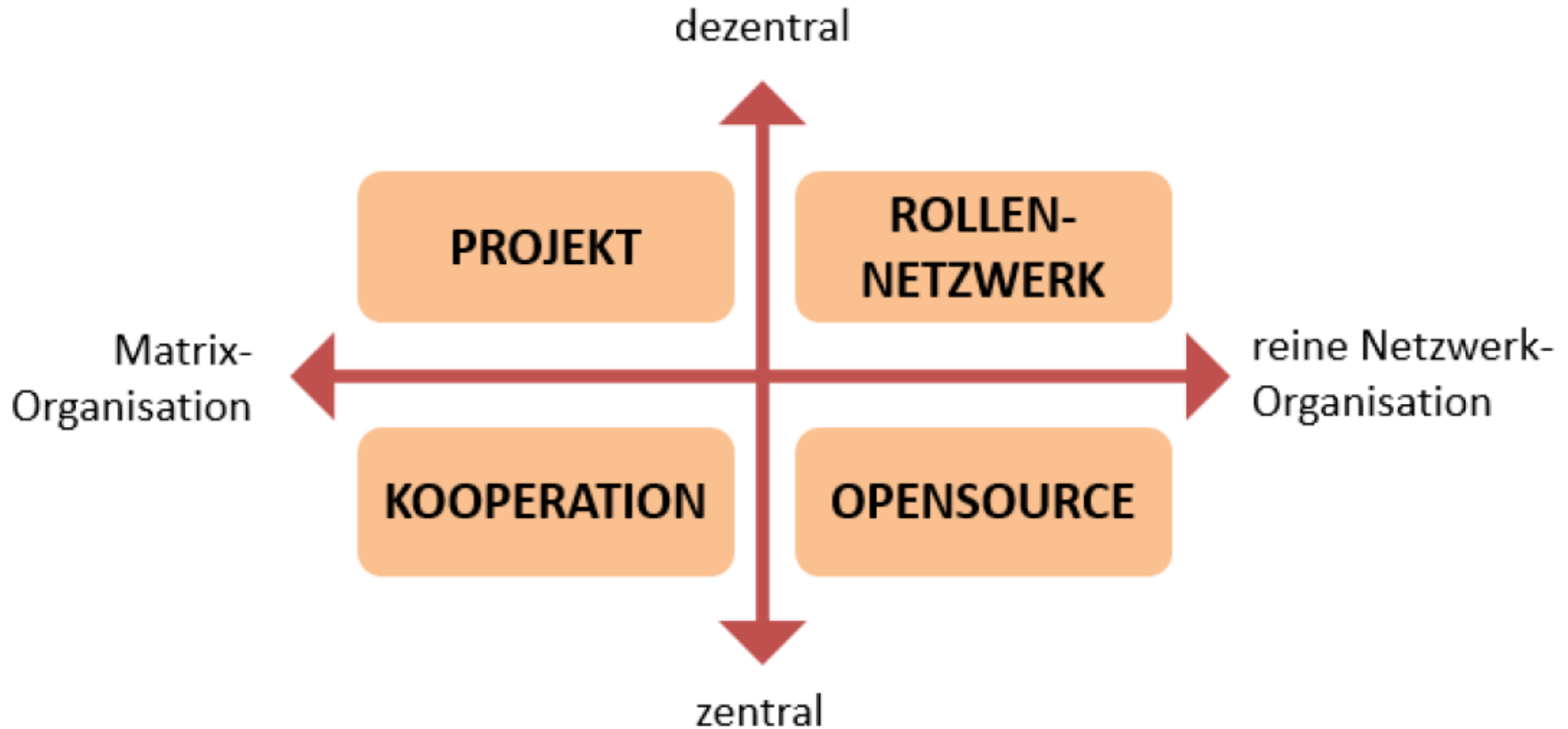
- Softwaregestaltung ist
  - a) kontrollierte Arbeit
  - b) kontrolliert Arbeit der Softwareanwendung
- andere Arbeitsteilung: statt z.B. Kopf- vs. Handarbeit
- Konflikte: innerhalb Management, unterschiedliche Organisationseinheiten, mit Betriebsrat
- Unterschiede: Möglichkeiten überhaupt nützen können vs. Möglichkeiten ausschöpfen



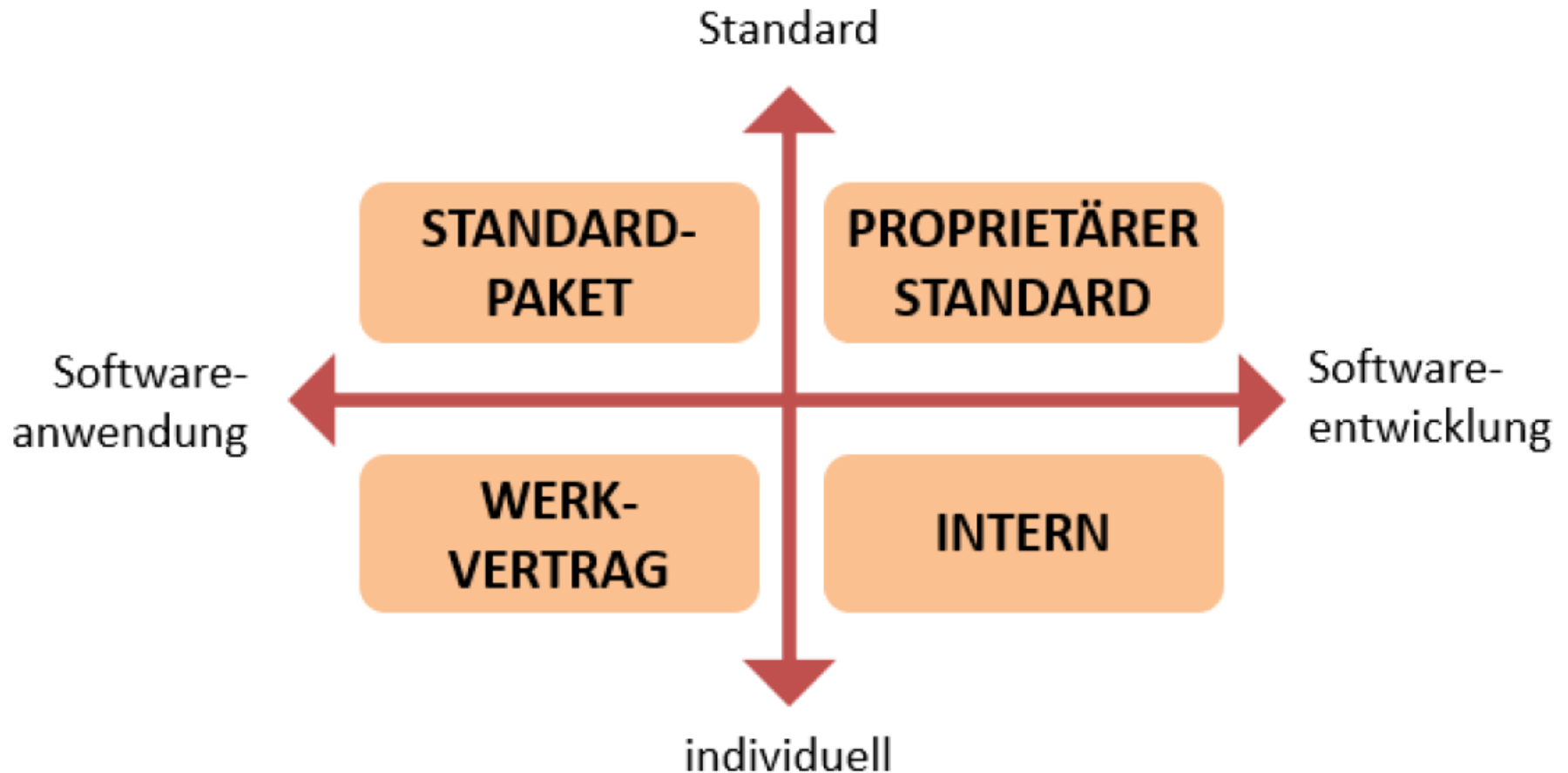
# Fallübersicht



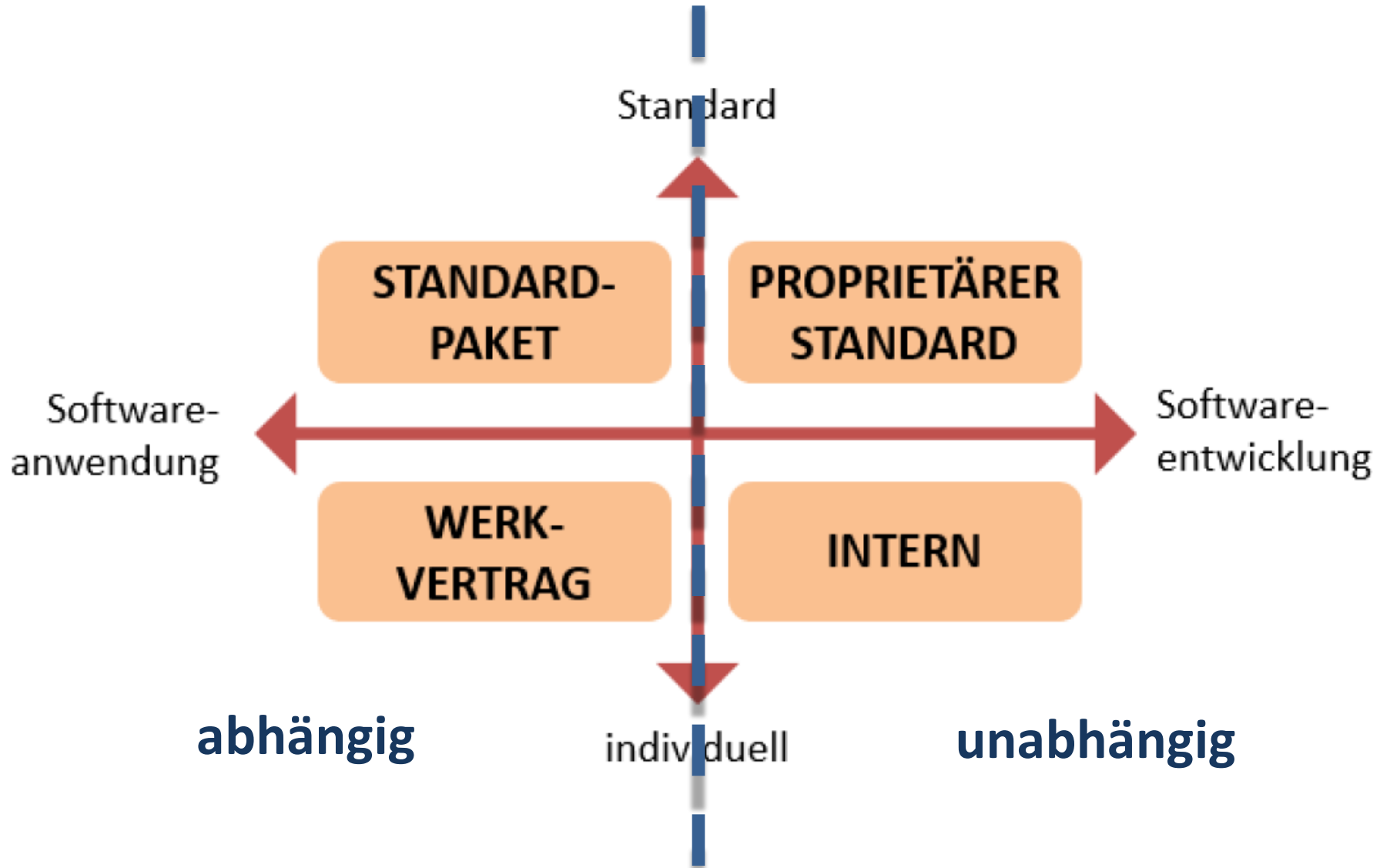
# Idealtypische Formen der Softwaregestaltung



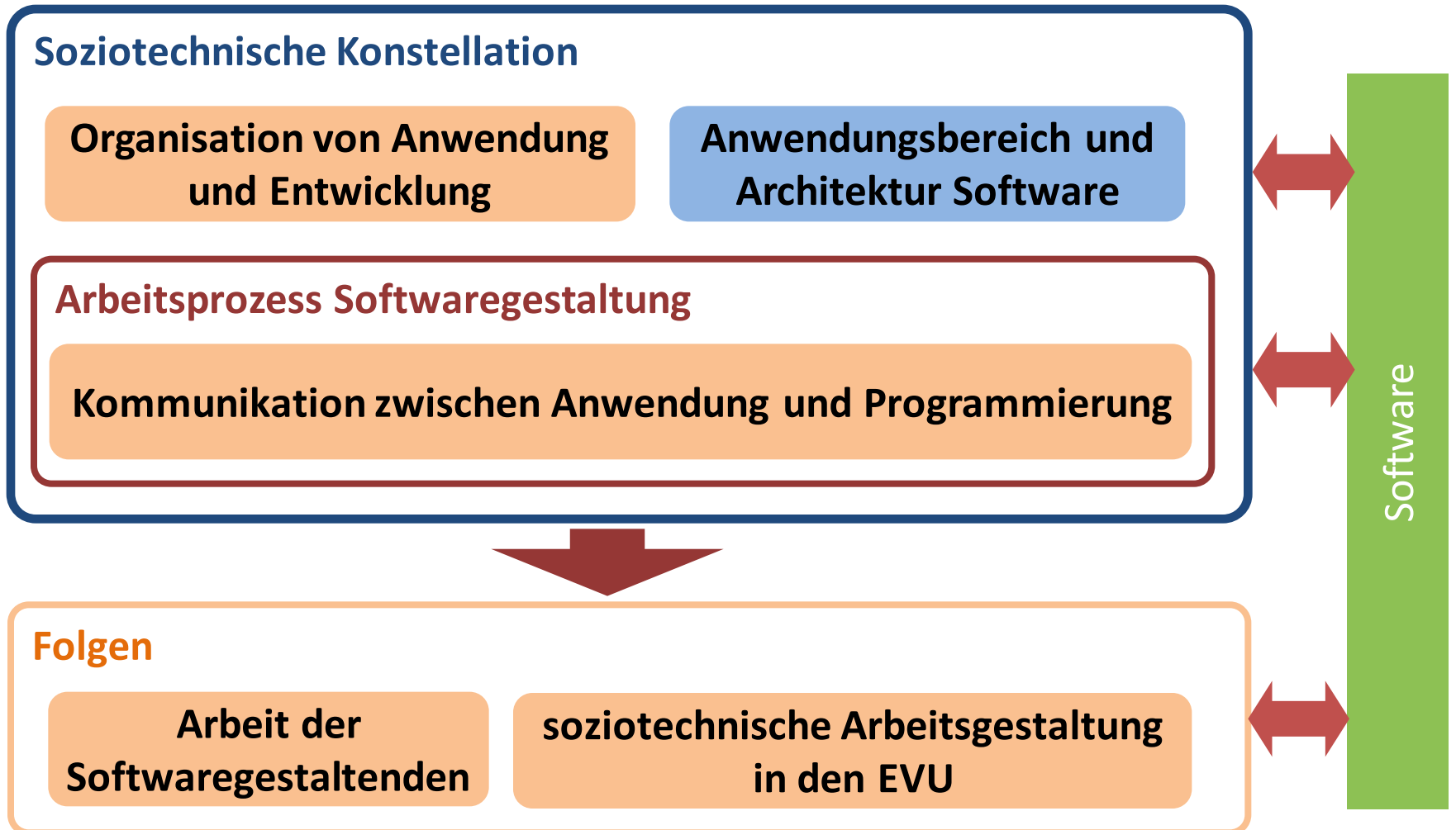
# Idealtypen soziotechnischer Arbeitsgestaltung



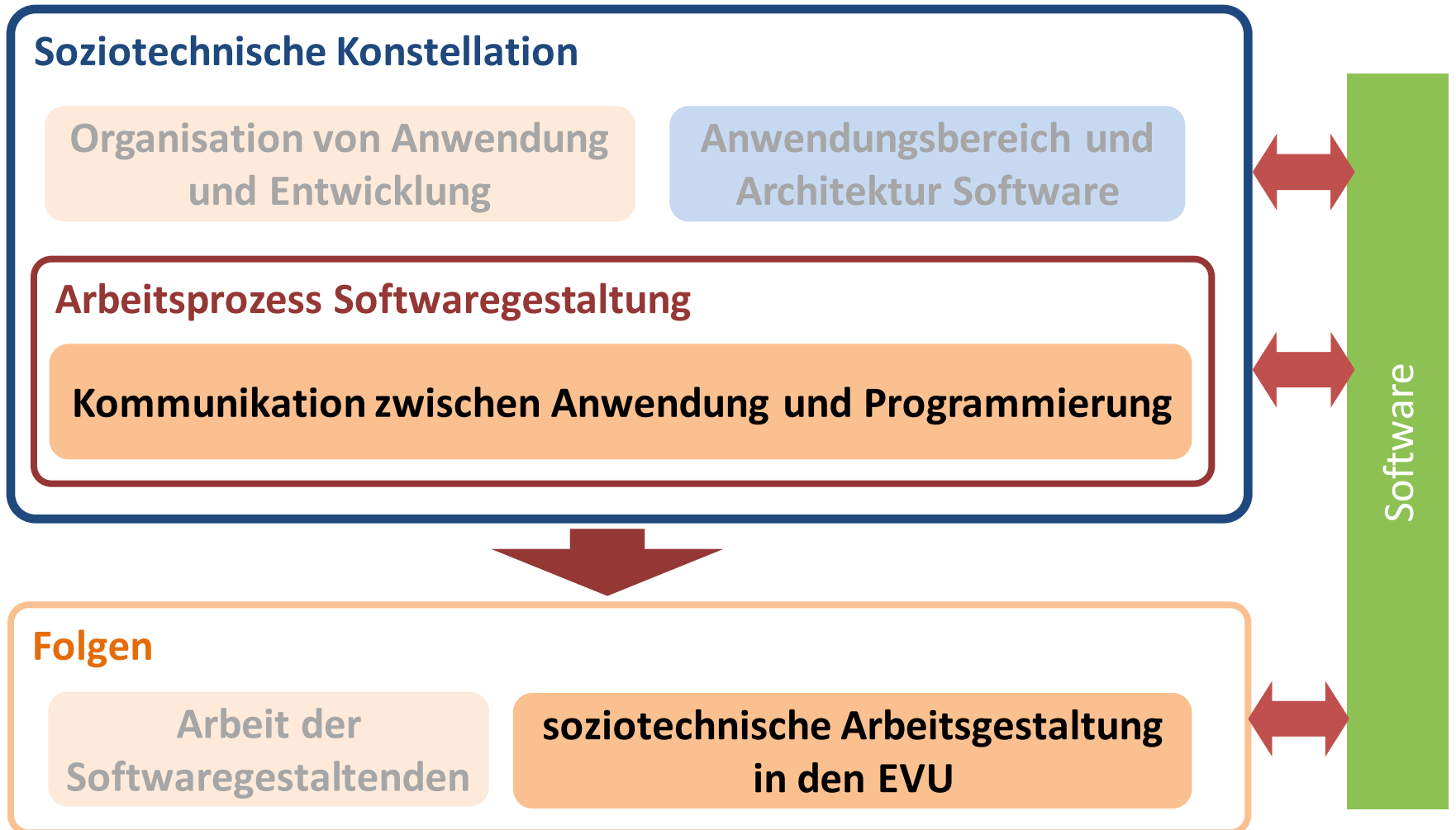
# Idealtypen soziotechnischer Arbeitsgestaltung



# Vereinfachter Analyserahmen Softwaregestaltung



# Vereinfachter Analyserahmen Softwaregestaltung



## INTERN2 – dezentraler Arbeitsprozess der SG

- **Rollen** für Anforderungsaufnahme: Anforderungsmanagende, Product Owner\*innen und Key User\*innen in den Fachbereichen
- **Ablauf** um Anforderungen direkt zu sammeln, auszuarbeiten und zu verhandeln: gemeinsame Anforderungsrunde mehrerer Fachbereiche, Interviews und Workshops in den Fachbereichen
- **Digitale Werkzeuge** für direkten Input: Ticketsysteme und MS Excel
- **Kommunikative Beziehungen** sind direkt: unabhängig von Führungskräften und auch informell

*„Aber wir haben halt versucht, gerade halt möglichst mehr auf eine direkte Kommunikation umzusteigen, weil gerade in der Vergangenheit gab es dann in den Tickets häufig das Phänomen, dass dann so ein Frage-Antwort-Pingpongspiel hin und her ging. Und der Durchschnitt war da tatsächlich, dass wir zu einem Ticket so 30 Kommentare hatten.“ (Scrum Master)*

- individueller **softwaretechnische Zuschnitt**: Anforderungsrunde vs. Anwendende, Begriffe der Anwendenden fließen ein; doch: Anpassung SAP-Standard

## INTERN2 – unabhängige soziotechnische Arbeitsgestaltung

- **Konflikte** sind intern: interpersonelle zwischen Fachbereichen beseitigt; inhaltliche in Anforderungsrunde ausdiskutieren; Betriebsrat interveniert bei Leistungskontrolle
- **Einfluss** besteht sowohl auf Software, wie auch auf Organisation – EVU schöpft die Möglichkeiten aber nicht aus
  - **Kontrollverhältnis:** SA könnte SG untergeordnet werden, aber: Softwaregestaltende können nichts gegen Führungskräfte der Fachbereiche durchsetzen
  - **Reorganisation** wechselseitig von Softwaregestaltung und -anwendung: etabliert AP SG (neue Rollen etc.), aber Status Quo in Anwendung bleibt erhalten
  - Softwaregestaltung und -anwendung auf die **Ziele** des EVUs ausgerichtet: abhängig bei Softwarestandard von SAP
  - **Partizipation:** Anwendende können direkt einbezogen werden: einerseits fragt PO Anwendende über Interviews ab, andererseits entscheidet Anforderungsrunde



## INTERN2 – Bsp. für **unabhängige** soziotechnische Arbeitsgestaltung

*„Ja, den Eindruck hatte ich auch, dass praktisch durch diese Software, diese alten Strukturen auf einmal feststellen: Wir haben ja was gemeinsam. Was vorher halt einfach nicht war; weil man fachlich getrennt war; weil es eben diese Fachexperten gab für Montage oder Hausanschluss oder was weiß ich, was es da alles gibt. Und jetzt haben die aber doch irgendwie so bestimmte Sachen gemeinsam oder müssen sich jetzt versuchen abzustimmen und das passt nicht mehr zu den alten Strukturen.“ (Anforderungsmanager)*

→ in Zukunft: Prozessteam für end-to-end  
Prozess ohne Fachabteilungs- und  
Teamgrenzen?

## KOOP1 – Beispiel zentralisierter AP SG

- **Rollen** für die Koordination beim IT-DL: Mediator, Key Account Managende, Anforderungsmanagende
- **Ablauf** für Verhandlungen, Erwartungsabgleich und Konfliktlösung: zweistufiges Anforderungsmanagement

*„Ich glaube, eines der größten Learnings ist, dass die größten Herausforderungen, einer der großen Kostenpunkte, die Abstimmung überhaupt und den Austausch über den Kunden (unv.). Also dieses ganze Anforderungsmanagement und alles was nichts mit Technik zu tun hat, [...] sondern wirklich miteinander reden und abstimmen und irgendwie sich auf einen gemeinsamen Nenner zu einigen. Dass das eigentlich die größten Hürden sind und das über Jahre hinweg irgendwie so am Laufen zu halten. Dafür haben wir eine Lösung gefunden, auch sogar irgendwann extern moderiert und auch immer noch.“ (Digitalisierungsmanager IT-DL)*

## KOOP1 – Beispiel **zentralisierter AP SG**

- **digitale Werkzeuge** für Transparenz und Abstimmungen: Ticketsystem und MS Sharepoint
- **kommunikative Beziehungen**: langfristige Beziehungen „auf Augenhöhe“, direkte Kommunikation trotz formal-bürokratischer Abläufe, Konflikte persönlich klären

*„Es muss ein Geben und Nehmen sein. Selbst bei den SLAs: Es gibt natürlich SLA-Verstöße. [...] Diese Dinge werden immer erst besprochen, die Gesamtgemengelage, Ursachen, vielleicht schwierige Rahmenbedingungen mit besprochen, Herstellerqualität, Corona[...]“ (Key Account Manager)*
- Standard als **softwaretechnischer Zuschnitt**: Verhandlung (3-Schichten-Architektur), Vertrieb vs. Netz

## STARTUP – Primat der Softwareentwicklung

STARTUP organisierten den Quoten- bzw. Emissionshandel für E-Autos für einzelne Autobesitzer und für Ladesäulen

**Netzwerk-Organisation:** rollenbasiert, weder Markt noch Hierarchien stören (keine Organisations-, Team- oder Abteilungsgrenzen)

- **Rollenträger** treffen sich in Kreisen, um Anforderungen zu besprechen
- **Ablauf** der Konzeptionierung zeichnet sich durch direkte Feedbacks und den direkten Austausch zwischen Anwendung, Gestaltung und Programmierung aus; angepasstes Scrum
- offen, direkte, langfristige **kommunikative Beziehungen:** z.B. spontan ohne Absprache mit einem Vorgesetzten via direkter Kommunikation Themen bearbeiten

*„[D]a ziehen halt alle am gleichen Strang, ohne viele persönliche Befindlichkeiten, die dahinterstecken, ohne irgendwelche Ego-Geschichten.“ (Programmierer1)*

## STARTUP – Primat der Softwareentwicklung

- **digitale Werkzeuge** für direkten Input: Ticketsystem, MS-Teams-Chat
- **softwaretechnischer Zuschnitt**: individuell und Standard-White-Label-Software: Webseite und App zur E-Auto-Anmeldung für andere Firmen z.B. der Automobil-Branche

### Soziotechnische Arbeitsgestaltung

- **Einfluss und Konflikte**: Softwareanwendung klar der Softwaregestaltung untergeordnet und nachgelagert

# Übersicht

Fall	Grund- koordination	Interdisziplinäre Netzwerke	Arbeitsprozess Softwaregest.	Arbeitsgestaltung Softwareanw.
<b>INTERN2</b>	Hierarchie	IT-Abteilung: organisiert Anforderungsrunde mit Fachbereichen	PROJEKT	INTERN
<b>KOOP1</b>	Markt	IT-Dienstleister: organisiert Anforderungsrunde mit EVU	KOOPERATION	STANDARD- PAKET/INTERN/ WERKVERTRAG
<b>STARTUP</b>	Netzwerk	Kreistreffen und Chatgruppen	ROLLEN- NETZWERK	INTERN/ PROPRIETÄRER STANDARD

## Weitere Ergebnisse

- Konzepte
  - I. **soziotechnische Netzwerkarbeit** (Kontrolle Softwaregestaltung)
  - II. **soziotechnische Arbeitsgestaltung** (der Softwareanwendung durch Softwaregestaltung)
  - III. Typ der **technikentwicklungsbezogenen Rationalisierung**
    - Wissens- statt Verhaltenskontrolle
    - soziotechnische Begriffsbildung um moderne Organisationen zu beschreiben
- Rolle Betriebsrat
- Rolle Management/Führungskräfte

Vielen Danke für Ihre  
Aufmerksamkeit!



## Analyserahmen inkl. Kategorien

### Soziotechnische Konstellation (Ausgangsbedingungen)

#### Organisation von Anwendung und Entwicklung

**Arbeitsteilung**

**Grundkoordination**  
(Markt, Hierarchie, Netzwerk)

#### Software

**Anwendungsbereich**

**Architektur**

### Soziotechnischer Arbeitsprozess der Softwaregestaltung

#### Rollen

(Key User, Anwendungsbetreuer, IT Berater, Product Owner etc.)

#### Ablauf

(Scrum, IT Projekt, Anforderungsmanagement etc.)

#### kommunikative Beziehung

(gem. Sprache, langfristig, kooperativ etc.)

#### digitale Werkzeuge

(Ticketsystem, Email, Sharepoint etc.)

#### softwaretechn. Zuschnitt

(Synergien, Priorisierung, Standardisierung etc.)

### Arbeit der Softwaregestaltenden

**Kontrolle**

**Beschäftigungssystem**

**Verteilung von Wissen**

### Arbeitsgestaltung der Anwendung

**Konflikte**

**Einfluss**

# Softwaregestaltung als soziotechnische Netzwerkarbeit



## Softwaregestaltung als soziotechnische Netzwerkarbeit

- **Ablauf**
  - feedbackorientiert, iterativ, durch Lernprozesse entstanden
  - kombiniert mit Netzwerk aus Wissensarbeiter\*innen verteilt auf Organisationen, Abteilungen und Teams
- **Beziehungen**
  - partnerschaftlich und auf Augenhöhe, für eine interpersonelle Abstimmung auf operativer und strategischer Ebene
  - Erwartungen regelmäßig abgeglichen und Konflikte professionell lösen (u. U. mit Mediator\*innen).
- **Software**
  - Arbeitsgegenstand und -mittel: Bezugspunkt der Beteiligten, koordiniert, dokumentiert und kontrolliert Kommunikation
- **Softwaregestaltende**
  - handeln rollenbasiert und situativ
  - typische Erwartungen sind: kooperativ und selbstorganisiert zu sein, sich auf Software einzulassen, mit Nicht-Wissen umgehen und sich auf wechselnde zwischenmenschliche Konstellationen einzustellen.

## Warum Netzwerk als analytische Grundkategorie?

- Zusammenhang von Organisation und Software:
  - neue Gestaltungsnetzwerke innerhalb der Firmen entstehen aus Projektteam der SAP-Einführung (Hohlmann 2007)
  - Folgen von Softwarearchitektur für die Zusammenarbeit zwischen Fachabteilungen (Remer 2008)
- für Softwaregestaltung relevante Organisationsformen integrierbar: IT Projekte wie ERP-Einführungen, Scrum, organisations-/abteilungsübergreifende Zusammenarbeit
- kombinierbar mit Markt und Hierarchien
- Dynamik zwischen dezentral und zentralisiert
- Fokus auf Wissensaustausch, Kommunikation und Kooperation
- Netzwerke als aus mehreren Ebenen bestehend (Sydow/Windeler 2000, Apitzsch 2006, Kalkowski/Mickler 2015)

## Methode und Wirklichkeit

### **Manifesto for Agile Software Development**

- “The best architectures, requirements, and designs emerge from self-organizing teams.”
- “The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.” (Beck et al. 2001)

## Anknüpfungspunkte in der Forschung zu Software

- „**requirements engineering**“ als intensiver Verbalisierungs-, Meinungsbildungs- und Kommunikationsprozess (vgl. Andelfinger 1997, Alvarez 2002, Ross/Chiasson 2011, Kaminski 2012, Alsanoosy et al. 2020)
- “**general theory of software engineering**“:  
„Software system development is more of a communication problem than a technical problem“  
(Wohlin/Šmite/Moe 2015)
- **IT-Alignment**: zentrale Bedeutung von Kooperation und Kommunikation (vgl. Chan/Reich 2007, Masak 2008, Schlosser et al. 2015, Valorinta 2011, Ko/Kirsch 2017)
- Implementierung und Entwicklung von **ERP-Systemen** (vgl. u.a. Grant et al. 2006, Hohlmann 2007, Pollock/Williams 2009, Svejvig/Jensen 2013, Lyytinen/Newman 2015, Conrad 2017)
- **Arbeits-/Techniksoziologie**:
  - unterschiedliche Arbeits-/Organisationsformen in der Softwareentwicklung (vgl. u.a. Friedman/Cornford 1989, Schulz-Schaeffer 1996, Andrews et al. 2005, Barrett 2005, Feuerstein 2012, Pfeiffer/Sauer/Ritter 2014, Boes et al. 2018, Schulz-Schaeffer/Bottel 2018)
  - subjektivierendes Arbeitshandeln von **Softwareproduktmanagenden** (vgl. Bolte 2017a, Bolte 2017b, Weishaupt/Hösl 2017)

## Anknüpfungspunkte in der Forschung zu Software

- **Software** verlangt wissensevozierendes Engagement und Interaktion und verändert sich im Arbeitsprozess stetig (Darr 2019, Rennstam 2012).
- Die zu gestaltende Software koordiniert als gemeinsam gestaltbares, zeichenbasiertes **Bezugsobjekt** die Wissensarbeit (vgl. Nicolini/Mengis/Swan 2012, Barrett/Oborn 2010, Carugati et al. 2018, Ponte/Rossi/Zamarian 2009, Bolici/Howison/Crowston 2009 und 2016).
- Software prägt die Kommunikation, wie die Diskussion um **Conways Law** zeigt (1968): kann bei der Softwareentwicklung umgangen werden und ist trotzdem noch prägend ist (vgl. Colfer/Baldwin 2016, Hvatum/Kelly 2005). Das betrifft jedoch die Aufteilung einer Software z.B. in Module. Nutzen mehrere Teams einen Softwarebaustein, müssen sich für dessen Gestaltung abstimmen (vgl. Remer 2008).

## Idealtypische Unterschiede Arbeitsprozess Softwaregestaltung

Typ	typische Erwartungen an die Rollen	Typisch für den Ablauf	typischer Nutzen Werkzeuge	typische Kommunikative Beziehungen	Software-technischer Zuschnitt
<b>zentral</b>	Koordination	Gremien für Verhandlungen, Erwartungsabgleich, Konfliktlösung	Transparenz, Abstimmungen	Spannungen ausgleichen, bürokratisch	Standard
<b>dezentral</b>	Anforderungsaufnahme	Anforderungen direkt sammeln und ausarbeiten	direkter Input	kooperativ, direkt, gemeinsame Sprache	individuell



## Idealtypische Unterschiede Arbeit Softwaregestaltende

Typ	Kontrolle	Beschäftigungssystem	Wissensverteilung
reines NW	Peer- und Objekt-Kontrolle, Fokus auf Koordination	keine Karriereleiter, Aufgaben und Position abhängig von Stellung zur Software	Wissen rein horizontal im Gestaltungsnetzwerk verteilt
Matrix	Mischung mit Markt und Hierarchie	Teil von Märkten oder Hierarchien	Wissen getrennt durch Hierarchien oder Märkte

## Idealtypische Unterschiede soziotechnische Arbeitsgestaltung

Typ	Einfluss der Softwaregestaltung	Konflikte
<b>unabhängig</b>	intern: Software und Organisation	intern
<b>abhängig</b>	extern: nur auf Software	extern

Typ	Kontrollverhältnis Softwareanwendung (SA) - Softwaregestaltung (SG)	Reorganisation SA - SG	Ziele SA - SG	Partizipation SA an SG
<b>unabhängig</b>	SG kontrolliert SA möglich	Software, SG und SA im Wechselspiel möglich	beides auf EVU-Nutzen ausgerichtet	direkter Einbezug Anwendende möglich
<b>abhängig</b>	Kontrolle SA - SG: nur getrennt möglich	nur getrennt möglich	Ziel SG entscheidet Softwarefirma oder IT-DL	(Branchen-)Fachleute, anwendende Organisation entscheidet nicht

## INTERN1 – Beispiel für **dezentralen AP SG**

- **Rollen** für Anforderungsaufnahme: 6 Anforderungsmanagende im Fachbereich, 4 Product Owner\*innen in der IT-Abteilung
- **Ablauf** um Anforderungen direkt zu sammeln und auszuarbeiten: Scrum, Resonanzgruppen, Workshops
- **Digitale Werkzeuge** für direkten Input: Ticketsystem und MS-Teams-Kanal
- **Kommunikative Beziehungen** sind direkt: unabhängig von Führungskräften, Offenheit, interdisziplinäre Sprache

*„Ich musste IT Deutsch lernen. Wenn ich Storys schreibe, dann muss ich dies so schreiben, dass die Softwareentwicklung versteht, obwohl ich aus der Anwender-Ecke gekommen. Ja, ich muss mich da einfach in so eine IT-denke reinschrauben.“ (Anforderungsmanagerin)*

- individueller **softwaretechnische Zuschnitt**: Management vs. Anwendende, Begriffe der Monteur\*innen berücksichtigt

## KOOP1: Firmenübergreifende Gestaltung eines industriespezifischen und verhandelten Standard-Softwarepakets für die Datenverarbeitung

