

Engineering IT-basierter Dienstleistungen

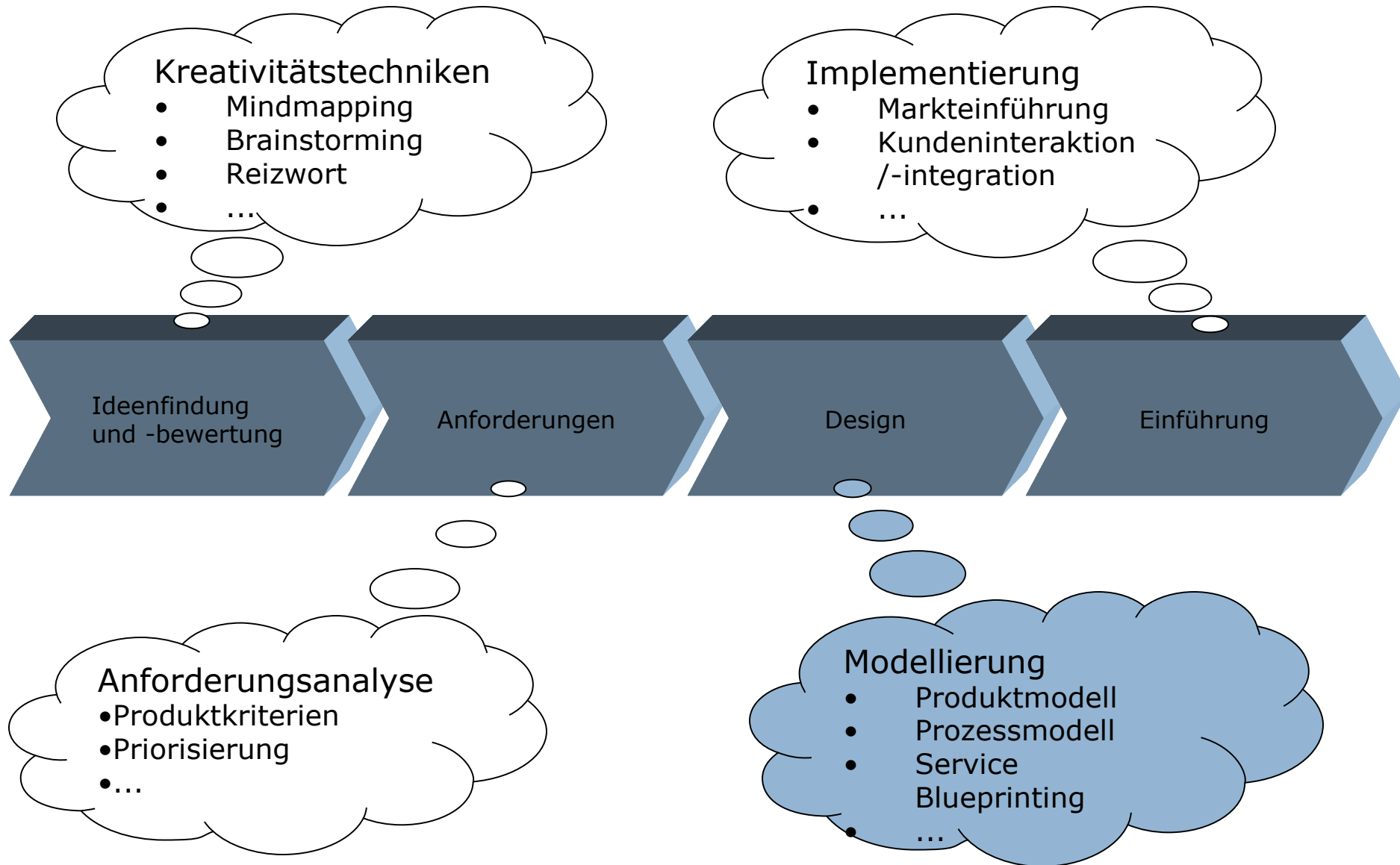
Prof. Dr. Klaus-Peter Fährnich

Teil 7: Methoden und Werkzeuge

Engineering IT-basierter Dienstleistungen

1. Einführung
2. Typologisierung von Dienstleistungen
3. Grundlagen des Service Engineering
4. Vorgehensmodelle
5. Plattformstrategie: Produktmodelle und Modularisierung
6. Methoden und Werkzeuge I
- 7. Methoden und Werkzeuge II**
8. Methoden und Werkzeuge III
9. Methoden und Werkzeuge IV
10. Werkzeuganwendung I
11. Werkzeuganwendung II
12. Zusammenfassung Werkzeuge
13. Service-Technologien
14. Kundenintegration und Kundenmanagement
15. Management der Dienstleistungsentwicklung
16. Fallstudie IT-Services
17. Standardisierung im Dienstleistungsbereich
18. Dienstleistungen im internationalen Wettbewerb
19. Praxisteil I
20. Praxisteil II

Methoden und Werkzeuge: Ein Überblick



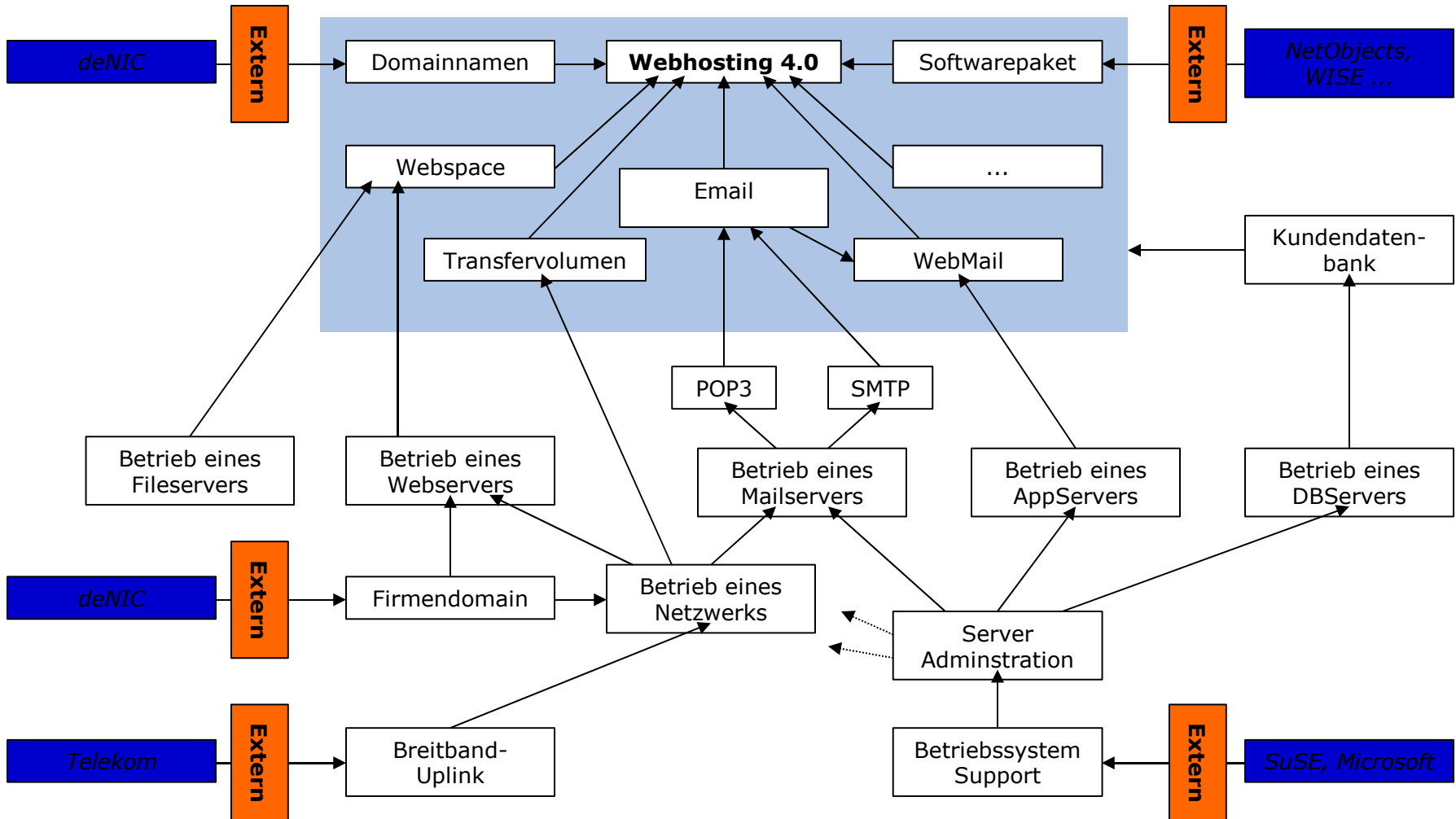
Warum Modellierung?



- Ein Modell ist eine formale Beschreibung eines Ausschnitts der realen Welt, also z.B. eines Produktes oder Geschäftsprozesses.
- Modelle helfen, die Welt besser zu verstehen, da komplexe Dinge abstrahiert werden
- nur durch die möglichst umfassende Modellierung des geplanten Service-Prozesses können die wichtigen Fragen der Implementierung geklärt werden
- Die Modellierung von Dienstleistungen umfasst
 - Produkte
 - Prozesse
 - Ressourcen (menschliche, materielle und immaterielle)
 - Organisation

Produktmodelle

- kurze Wiederholung: Produktmodell IT-Service



Produktmodelle/Produktkonfiguration



- ein (Service-)Produkt besteht aus Komponenten
- die Zusammenstellung der Komponenten heißt auch Konfiguration
- die Komponenten haben Abhängigkeiten von anderen Komponenten, um diese aufzulösen, gibt es verschiedene Ansätze



- regelbasiert (rule-based)
 - feste Regeln, nicht flexibel
 - wenn eine neue Komponente hinzukommen, müssen evtl. viele bestehende Regeln angepasst werden
- fallbasiert (case-based)
 - basiert auf Wissensdatenbank mit bekannten Lösungen
 - neue Konfigurationen werden evtl. falsch auf bekannte Lösungen abgebildet
 - erstellen der Datenbank ist sehr aufwändig
- ressourcenbasiert (resource-based)
 - z.B. Webserver braucht Ressource „Speicherplatz“, diese bietet ein Fileserver an
 - vielversprechendes Modell

Produktkonfiguration (2)



- constraintbasiert
 - es wird von einem Lösungsraum ausgegangen, der durch Constraints (Bedingungen) eingeschränkt wird
 - die Constraints beziehen sich auf Beziehungen zwischen Objekten oder deren Eigenschaften
 - Nachteil: das CSP (Constraint Satisfaction Problem) ist NP-vollständig, also kann das Finden einer Lösung sehr aufwändig oder auch unmöglich sein
- strukturbasiert (structure-based)
 - Aufbau eines Dekompositionsbaums, der alle möglichen Konfigurationen repräsentiert
 - jede Konfiguration ist ein Teilbaum des Dekompositionsbaums
 - aufwändige Taxonomie-Erstellung, nicht für dynamische Konfigurationen geeignet, da hoher Änderungsaufwand



- Produktkonfiguratoren werden als Teilbereich der KI entwickelt
- wichtigste Forschungsprojekte sind:
 - EngCon für regelbasierte Systeme, aber auch hybrid für strukturbasiert
 - ILOG für constraint-basierte Systeme
 - Cosmos für ressourcen-basierte System
 - KONWERK für strukturbasierte System

Produktkonfigurator: EngCon Screenshot

The screenshot shows the EngCon product configurator interface. On the left is a tree view of the configuration, and on the right is a 'Configuration - Specialize' dialog for a VGA card.

Configuration Tree (Left):

- Teil-Konfiguration
 - PC_0
 - hat-komponente [9...9]
 - Gehäuse [1...1]
 - Gehäuse_1
 - Mainboard [1...1]
 - Mainboard_2
 - Prozessor [1...1]
 - Prozessor_3
 - Speicher [2...2]
 - Speicher_4
 - Speicher_11
 - Netzwerkkarte [1...1]
 - Netzwerkkarte_12
 - VGA_Karte [1...1]
 - VGA_Karte_5
 - TV_Karte [0...0]
 - Soundkarte [0...0]
 - Festplatte [1...1]
 - Festplatte_6
 - CD_Rom [1...1]
 - CD_Rom_7
 - hat-peripherie [6...6]
 - Monitor [1...1]
 - Monitor_8
 - Maus [1...1]
 - Maus_9
 - Tastatur [1...1]
 - Tastatur_10
 - Drucker [1...1]
 - Drucker_13
 - Scanner [1...1]
 - Scanner_14
 - Joystick [0...0]
 - Boxen_Sef [1...1]

Configuration - Specialize Dialog (Right):

Name: VGA_Karte_5
 Concept: AGP_VGA_Karte
 Specialization: < select >
 Parameters: < select >
 ELSA_Gladiac511_GeForce2_MX400

Name	Value
Bildwiederholrate	Gainward_GeForce_2_Pro
RAMDAC	[350 .. 350]
DVI	false
Preis	[0.0 .. 3.40282346...]
VGA_out	2
Bus	agp
Chip	NVidia_GeForce2...
Kapazität	64

Relations:

Name	Concept	Min	Max	Value	Detailed
komponente-von	"PC"				

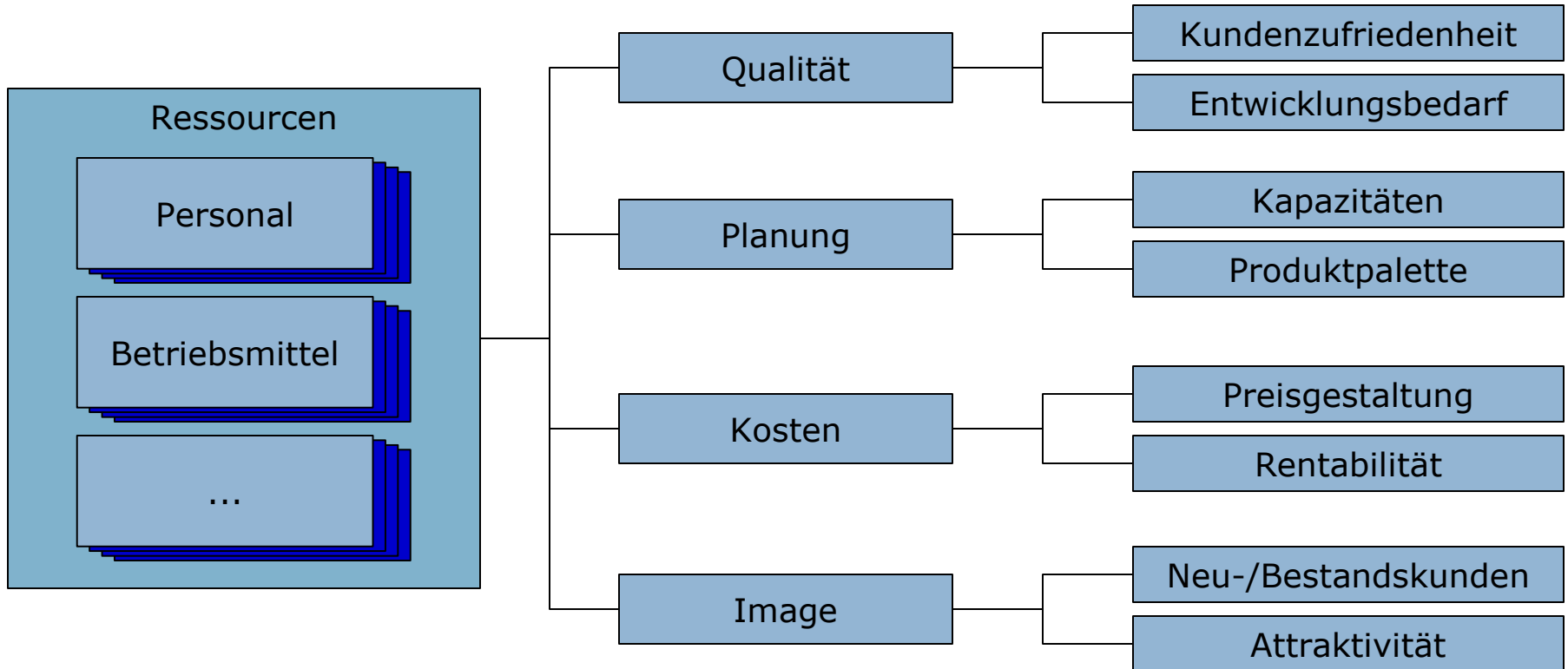
Buttons: OK, New Entry

Ressourcenmodelle



- Ressourcenmodelle beschreiben die Gesamtheit aller für den Engineering-Prozess benötigten Ressourcen.
- Ressourcen sind
 - menschlich, also interne und externe Mitarbeiter (human resources)
 - materiell, also Betriebsmittel wie Hardware
 - immateriell, also z.B. Software oder auch Zeit
- Ressourcen können modelliert werden durch
 - Organigramme (Organisationssicht)
 - Entity-Relationship-Modelle (Datensicht)
 - Funktionszuordnungsdiagramme (Funktionssicht)
 - Leistungsbäume (Leistungssicht)
 - ...

Warum brauchen wir Ressourcenmodelle?

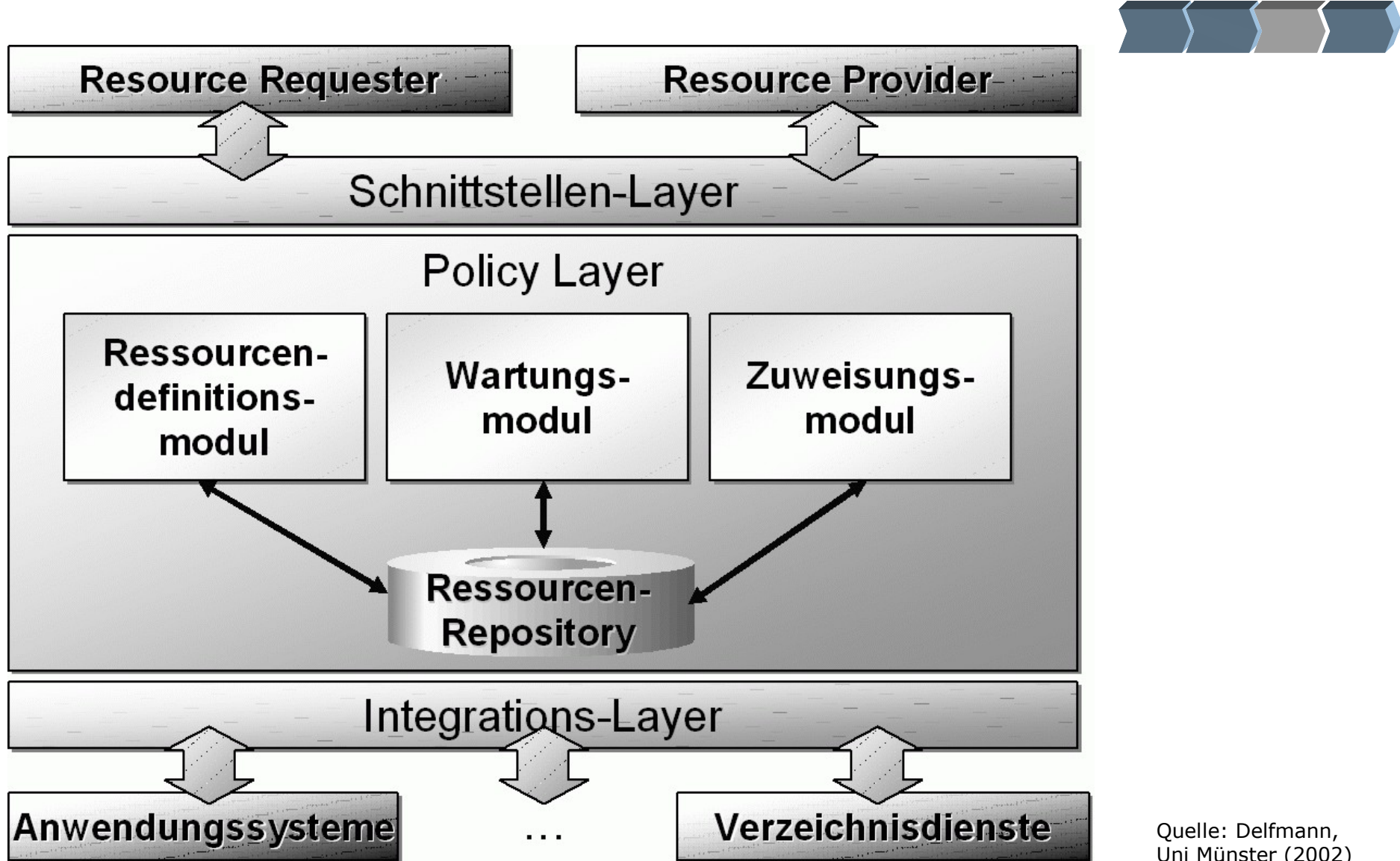


Merkmale von Ressourcen

Merkmal	Ausprägung		
Typ	human	technisch	
Beständigkeit	wiederverwendbar	konsumierbar	
Nutzung	exklusiv	gemeinsam	
Zuweisung	Push	Pull	Mischform
Beitrag	aktiv	passiv	
Ort	stationär	mobil	
Unabhängigkeit	autonom	abhängig	
Verfügbarkeit	nicht verfügbar	unverzöglich	verzögert
Kapazität	begrenzt	unbegrenzt	
Qualification			

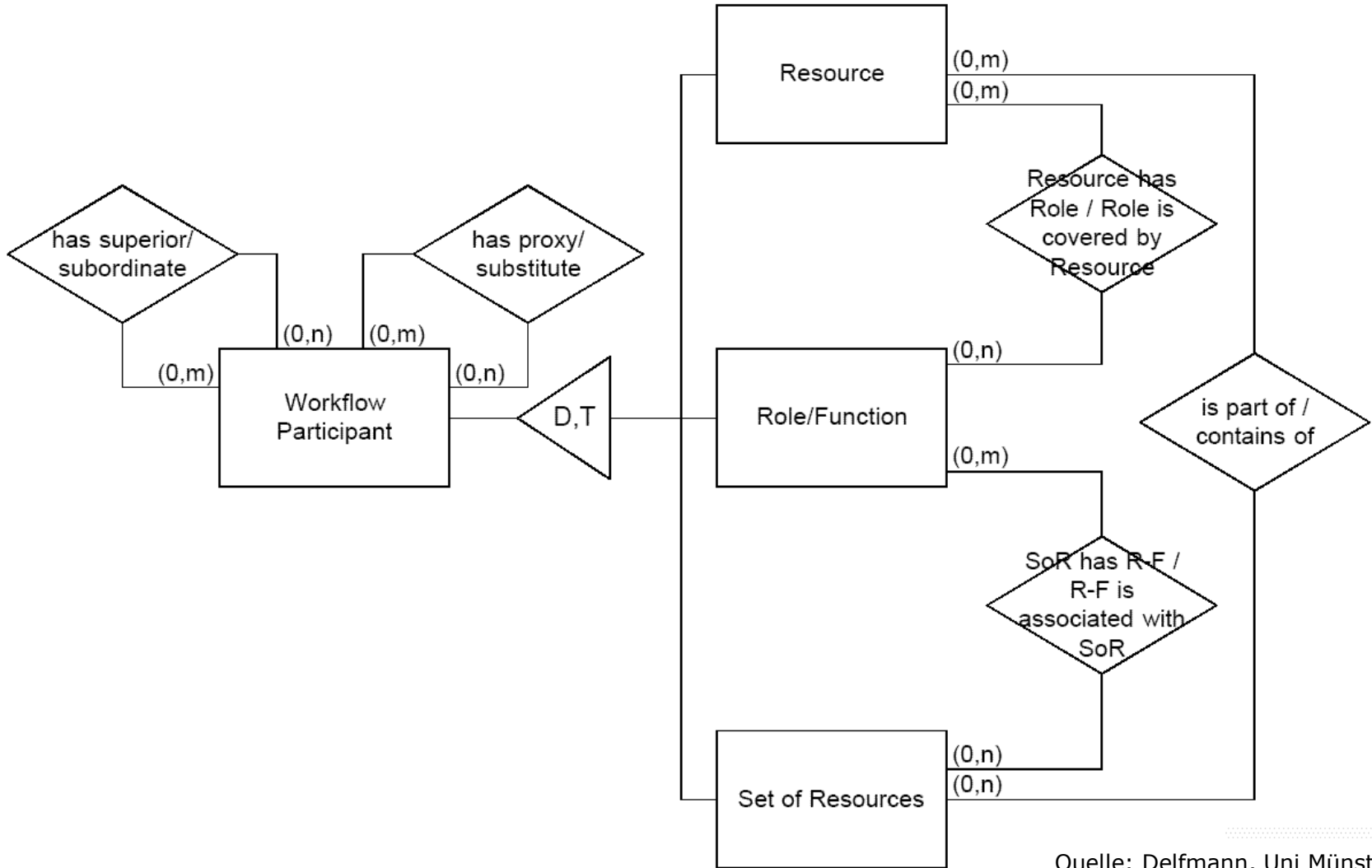
Quelle: Delfmann, Uni Münster (2002)

Ressourcenmanagement-System

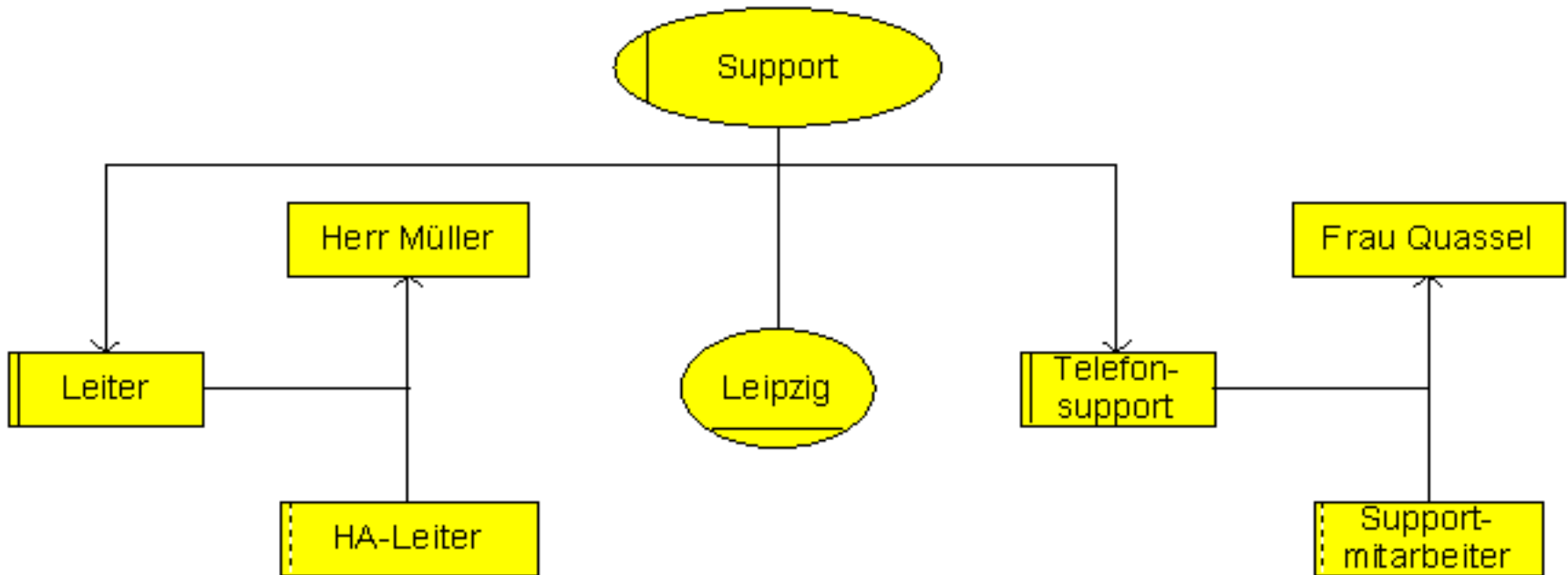


Quelle: Delfmann, Uni Münster (2002)

Beispiel: Generisches WfMC-Ressourcenmodell

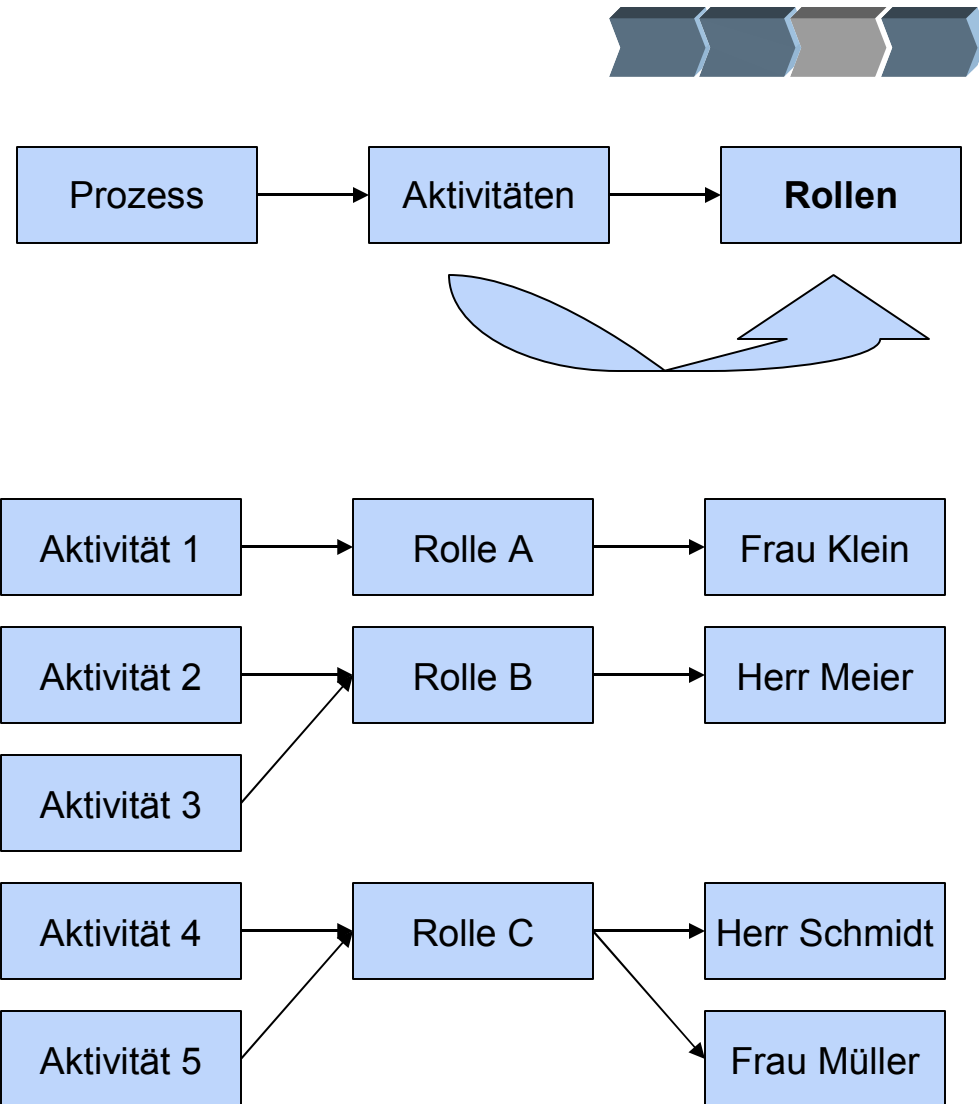


Quelle: Delfmann, Uni Münster (2002)



Einsatz von Rollenkonzepten

- Um die Zuordnung von Personen zu den einzelnen Aktivitäten im Entwicklungsprozess flexibel zu halten, lassen sich sogenannte Rollenkonzepte einsetzen.
- Ein Mitarbeiter kann dabei mehrere Rollen übernehmen, gleichzeitig kann auch eine Rolle von mehreren Mitarbeitern übernommen werden.



Quelle: Fraunhofer IAO

Rollenkonzepte: Matrixform



<p>A ausführend</p> <p>M mitwirkend</p> <p>B beratend</p>	Disponent	EDV-Koordinator	EDV-Assistenz	Erfasskraft	Fachabteilungsleiter	Foto-Laborant	Fuhrparkmanager	Hotline-Agent	Kaufm. Sachbearbeiter	Prüfingenieur	Prüfmittelverwalter	Kunde
Terminvereinbarung mit Kunden	A											M
Routenplanung	A											
Reservierung Dienstwagen	A						M					
Auswahl Prüfingenieur	A				B							
Zusammenstellung Prüfgeräte										A	M	
Zusammenstellung Prüfformulare				M						A	M	
Fahrt zum Kunden										A		

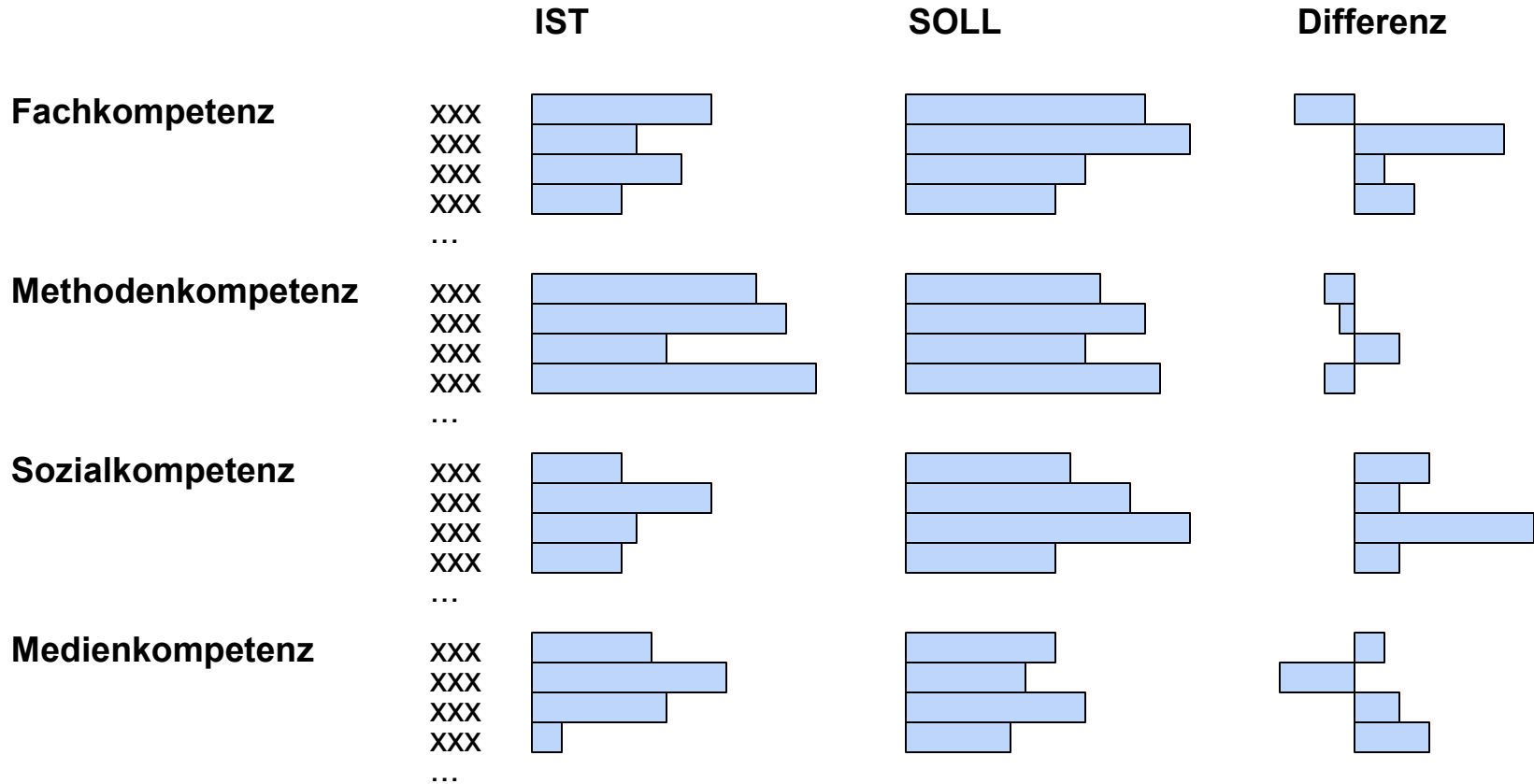
Quelle: Fraunhofer IAO

Aufbau einer Rollenbeschreibung



- verständlicher Name
- Beschreibung der Aktivitäten
- Dokumentation der notwendigen Kompetenzen:
 - Fachkompetenz
 - Methodenkompetenz
 - Sozialkompetenz
 - Medienkompetenz
- Beziehungen zu anderen Rollen

Auswahl und Qualifizierung von Mitarbeitern



Quelle: Fraunhofer IAO

Nutzen von Rollenbeschreibungen



- Unternehmenssicht
 - Erleichterung der Personalplanung
 - Festlegen von Verantwortlichkeiten, Aufgaben und Ergebnissen
 - Ressourcen-Engpässe können früher erkannt werden
 - Unterstützung bei der Rekrutierung und Qualifizierung
 - Abgrenzung der Kompetenzen ist klar definiert
 - Teamstrukturen und -abläufe werden entwickelbar
- Mitarbeitersicht
 - Transparenz der Tätigkeiten und Ergebnisse
 - Transparenz des Rekrutierungs- und Qualifizierungsbedarfs
 - Transparenz der Rollen und der Vernetzung der Rollen
 - Grundlage für Personalentwicklungsmaßnahmen

Quelle: Fraunhofer IAO

Prozessmodelle: Definition und Motivation

- Ein Prozess besteht aus miteinander verbundenen **Aktivitäten** zur Bearbeitung einer **Aufgabe**.



- Das Prozessmodell definiert alle zur Erbringung der Dienstleistung notwendigen Prozesse
- die Optimierungsmöglichkeiten im Service Engineering liegen auch bei der organisatorischen Gestaltung der Unternehmensabläufe

Vorgehensmodelle

- Vorgehensmodelle bzw. Prozessmodelle definieren den Ablauf des Softwareentwicklungsprozesses
- im Software-Engineering sind bereits viele Vorgehensmodelle entwickelt, getestet und standardisiert worden
 - Übertragung auf Service Engineering ist sinnvoll


Prozess-Modell legt fest ...

- Reihenfolge des Arbeitsablaufs,
- jeweils durchzuführende Aktivitäten,
- Definition der Teilprodukte einschließlich Layout und Inhalt,
- Fertigstellungskriterien,
- notwendige Mitarbeiterqualifikation,
- Verantwortlichkeiten und Kompetenzen,
- anzuwendende Standards, Richtlinien, Methoden und Werkzeuge.

Vorgehensmodelle im Software Engineering

Prozessmodell	Primäres Ziel	Antreibendes Moment	Benutzerbeteiligung	Charakteristika
V-Modell	max. Qualität	Dokumente	gering	sequentiell, volle Breite, Validation, Verifikation
Prototypen-Modell	Risikominimierung	Code	hoch	nur Teilsysteme
Inkrementelles Modell	min. Entwicklungszeit und Risiko	Code	mittel	volle Definition, dann zunächst nur Kernsystem
OO-Modell	Zeit und Kostenminimierung	wiederverwendbare Komponenten	?	volle Breite in Abh. von WV-Komponenten
Spiralmodell	Risikominimierung		mittel	Entscheidung pro Zyklus über weiteres Vorgehen

Service Blueprinting

- 
- Ein Service Blueprint (englisch »Blaupause«) ist eine Prozessdarstellung (in Form eines Ablaufdiagramms) einer Dienstleistung. Das Erarbeiten und Aufzeichnen eines Blueprint nennt man Blueprinting.
 - wichtige Grundlage ist die Betrachtung der Dienstleistung aus Kundensicht
 - es wird klar zwischen Kundenaktionen und Unternehmensaktionen unterschieden
 - Vorteile:
 - detaillierte und transparente Aufzeichnung der Arbeitsabläufe zur Erbringung der Dienstleistung
 - mögliche Fehler und die wichtigsten Entscheidungssituationen können modelliert werden
 - objektive und qualifizierbare Aussagen werden ermöglicht
 - Einbeziehung der Kundenaktivitäten und Kundenperspektive
 - Herausstellen der „Momente der Wahrheit“

Service Blueprinting (2)



- weitere Anwendungen:
 - Suche nach Fehlerquellen, Problemstellen, Schwachpunkte
 - ergänzende Analysen, wie Zeitrahmen, Bewegungsstudien, etc.
- meist mangelnde Übersicht durch hohe Komplexität, daher werden Sichtbarkeitslinien verwendet
- wesentliches Instrument, um
 - Prozess-Evidenz für den Kunden erhöhen
 - Rollenklarheit gewährleisten
- Ziel:
 - zukünftige Dienstleistungen (Produkt, Prozess) detailliert planen – in Zusammenarbeit mit dem Kunden

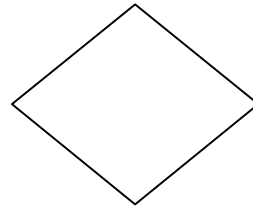
Service Blueprinting – Konventionen



Symbole des Service Blueprinting:



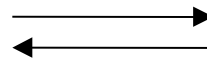
Aktion des
Anbieters



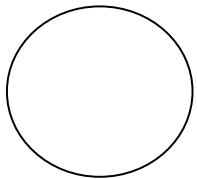
Entscheidung



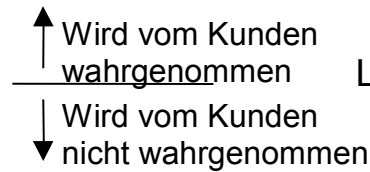
Aktion des
Kunden



Eingabe
Ausgabe



bestimmtes
Ereignis

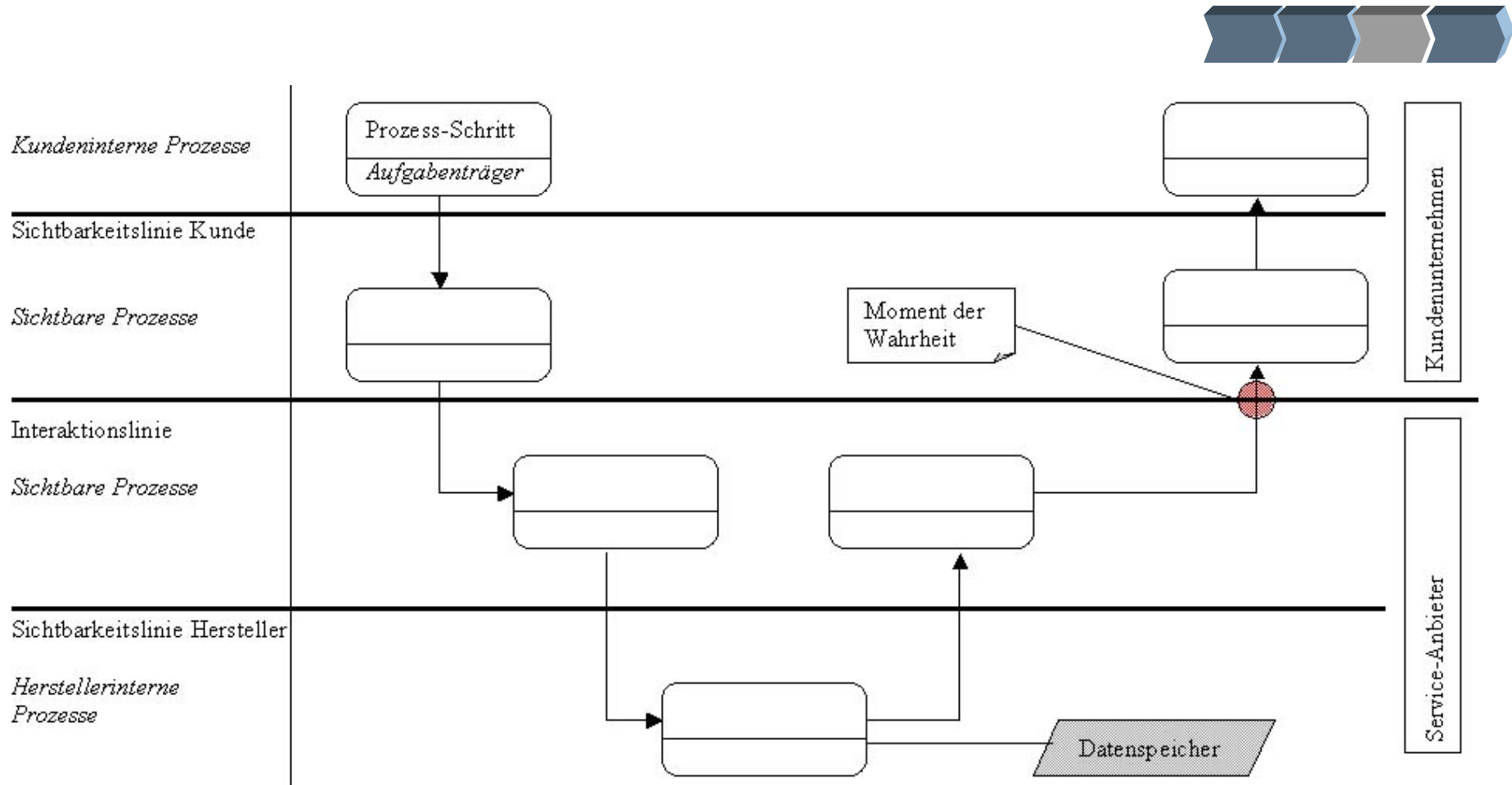


Linie der Sichtbarkeit

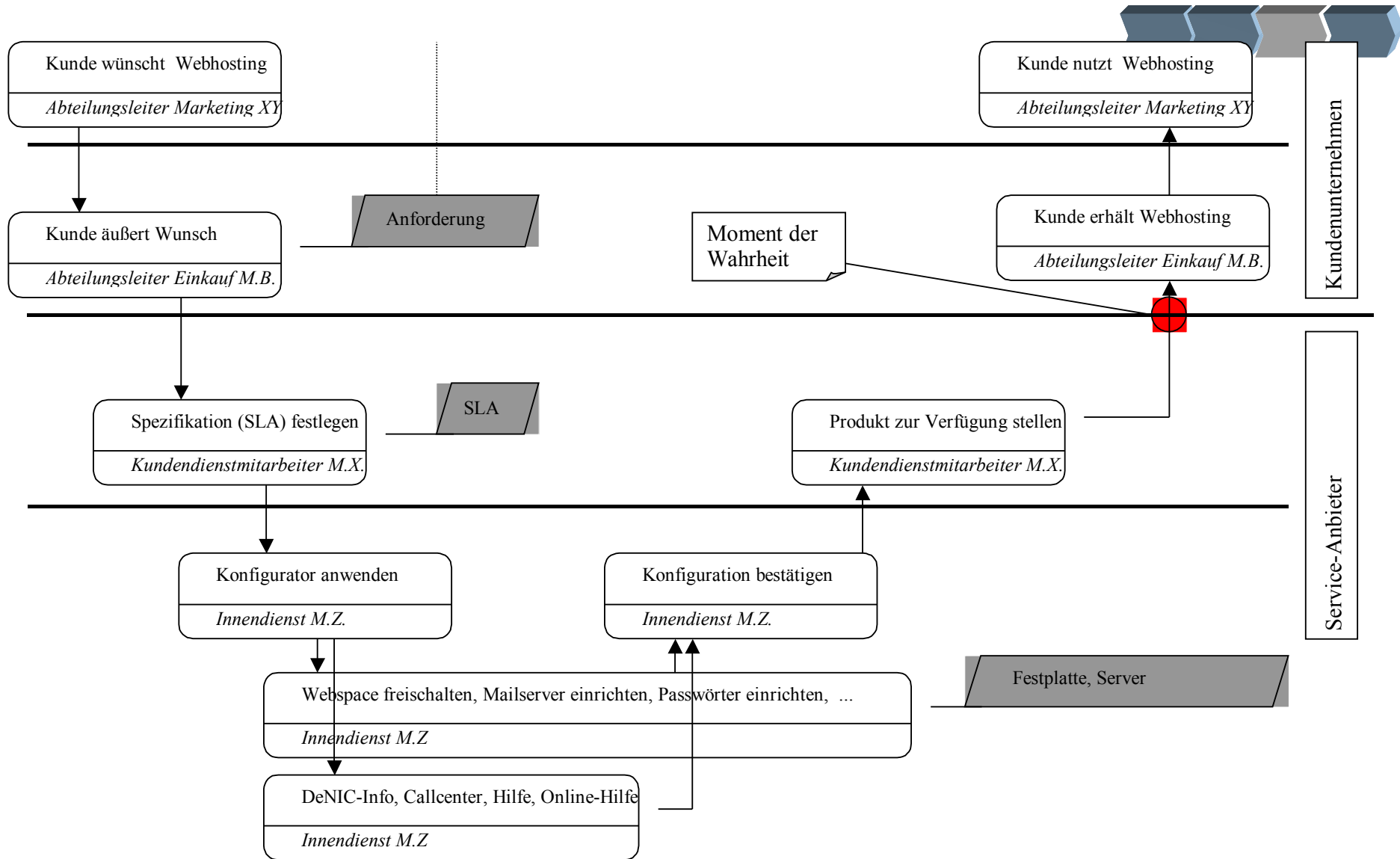


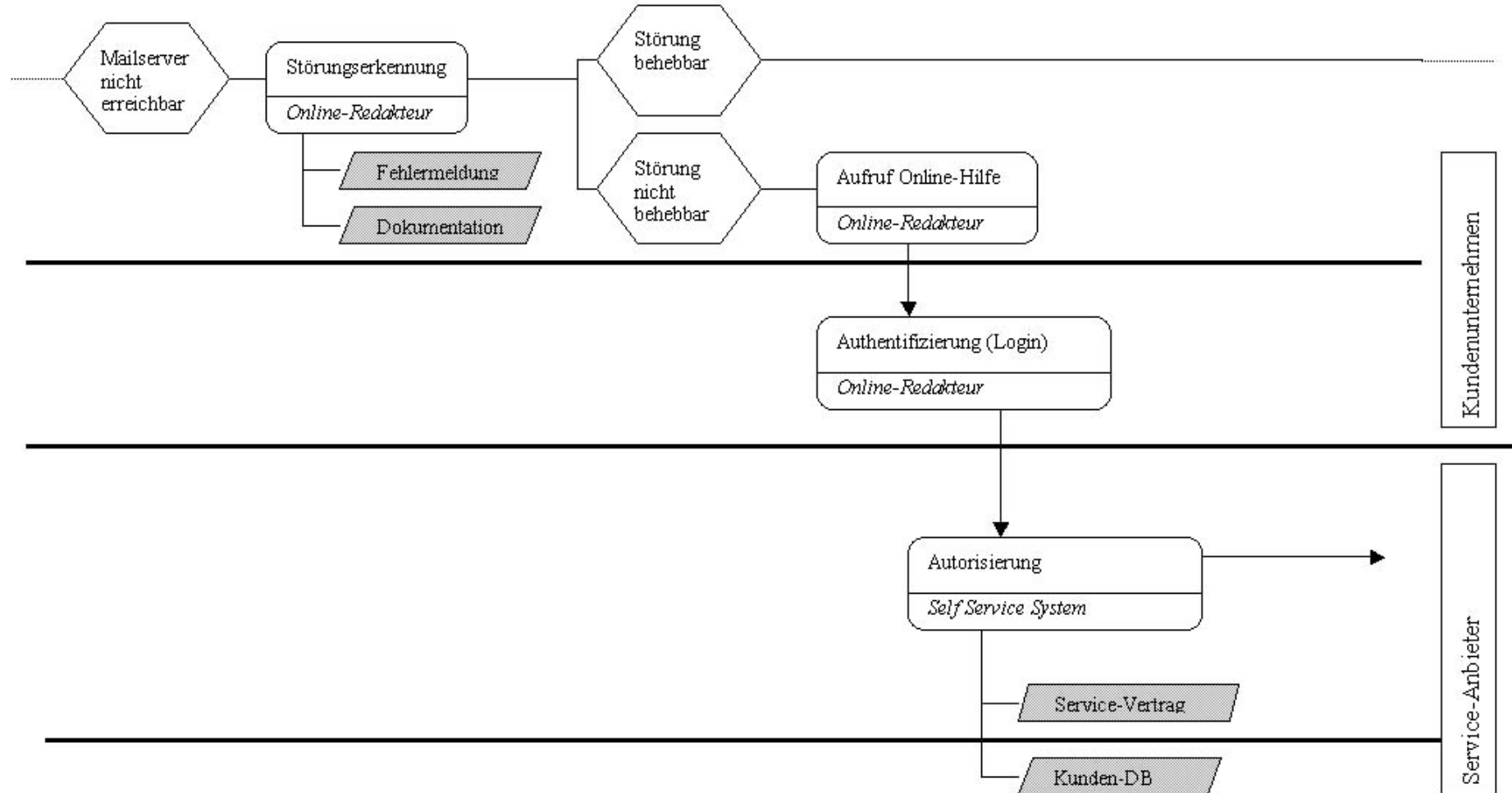
Fehlerquelle

Quelle: Fraunhofer IAO



Blueprinting – Beispiel





Kundeninteraktion (Ausblick)



- durch das Service Blueprinting wird die Interaktion mit dem Kunden und innerhalb des Unternehmens zu einem der zentralen Punkte
- das zentrale Vorgehen zur Modellierung der Interaktionen ist
 - Identifikation von Interaktionen (z.B. durch entsprechendes Blueprinting, Kontextdiagramme, ...)
 - falls nötig, Verfeinern des Prozessmodells
 - Kooperationspartner zu identifizieren und klassifizieren
 - Auswahl einer geeigneten Kommunikationsmethode und Erstellen bzw. Auswählen eines Kommunikationsprotokolls
- in der Regel sind unternehmensinterne Interaktionen einfacher aufzulösen, wenn entsprechende Vorkehrungen (Planung!) getroffen wurden
- mehr in Vorlesung „Kundenintegration und Kundenmanagement“

Prozessmodellierung mit eEPKs



- Ereignisgesteuerte Prozessketten (EPKs)
 - 1992 von Prof. Scheer entwickelt
 - basieren im Wesentlichen auf der Petri-Netz-Theorie
- Wegen des Fehlens von Bezügen zur Daten- und Organisationssicht Weiterentwicklung der EPKs zu erweiterten **ereignisgesteuerten Prozessketten (eEPKs)**.
- Mit Hilfe von (e)EPKs wird die Ablauforganisation von Unternehmen dargestellt, d. h., die Darstellung der Verbindungen zwischen den Objekten der Daten-, Funktions- und Organisationssicht und demzufolge die Darstellung von Prozessen.

Softwarewerkzeug: ARIS

- ARIS bietet eine einheitliche Methode, die sämtliche Aspekte der Leistungserstellung berücksichtigt und integriert. Das ARIS-Haus dient dabei als Bezugsrahmen und ermöglicht die verschiedenen Perspektiven von Dienstleistungen (Potenzial, Prozess, Ergebnis) modellbasiert zu vereinen.

eEPK-Werkzeug: ARIS Web Designer

ARIS

ARIS

File Edit View Insert Format Help

Kundenanfrage bearbeiten Kundenangebot bearbeiten Vertriebsprozesse (eEPK)

Historie

- Unternehmensprozesse
 - Marketing
 - Produktions- und Beschaffungsplanung
 - Vertriebsprozesse
 - Kundenanfrage bearbeiten [eEPK]
 - Kundenangebot bearbeiten [eEPK]
 - Vertriebsprozesse (eEPK) [eEPK]
 - Vertriebsprozesse (Office Process) [Office Process]
 - General Cars Corp. [eBusiness Scenario Diagram]

Attributname	Wert
Name	Kundenkontakt bearbeiten
Identifizierer	IDS.5063
Beschreibung/Definition	Durch der Kundenkontaktabwicklung werden die Kontakte der Vertriebsmitarbeiter zu den Kunden aufgebaut und gepflegt. Ziel ist die Anbahnung von Aufträgen. Kundenkontakte können persönlich oder über sonstige Medien erfolgen (e-Mail, Fax, Telefon etc.)
Verfasser	IDS Scheer AG
Typ	Funktion
Erstellzeitpunkt	20.02.03 08:40:15
Ersteller	system
letzte Änderung	20.02.03 08:40:15
letzter Bearbeiter	XMLIMPORT
Synonyme	Kontaktabwicklung
Kann-Funktion	<input type="checkbox"/> Kann-Funktion
Klassifikation	Modifizierte Standardsystem-Funkt...

Weitere Attribute...

Designer Explorer

Zoom: 80%

```

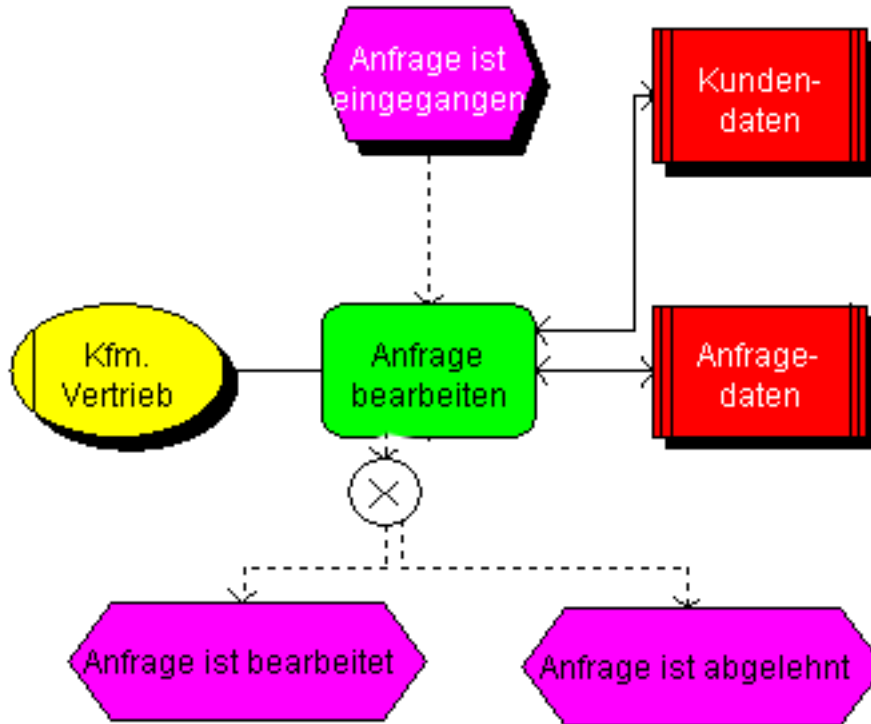
graph TD
    A{{Kundenkontakt vorhanden}} --> B[Kundenkontakt bearbeiten]
    B --> C((X))
    C --> D{{Anfrage eingetroffen -ohne Ref.-}}
    C --> E{{Anfrage aus Kontakt anzulegen}}
    C --> F{{Kundenkontakt erfolglos}}
    D --> G((X))
    G --> H[Kundenanfrage bearbeiten]
    H --> I((X))
    F --> I
  
```

Modellierungselemente einer eEPK



- Funktionen
 - aktive, zeitverbrauchende Komponente
 - generieren Ereignisse
- Ereignisse
 - passive, zeitpunktbezogene Komponente
 - lösen Funktionen aus
- Verknüpfungsoperatoren
 - stellen eine logische Verknüpfung zwischen Ereignissen und Funktionen dar
- Kontrollfluss
 - stellt zeitlich logische Abhängigkeiten zwischen den Komponenten Ereignis und Funktion her
- Zusätzliche Modellierungselemente
 - Informations-, Material-, Ressourcenobjekte
 - Dokumente, Dateien
 - Organisationseinheiten
 - Anwendungssysteme
 - ...

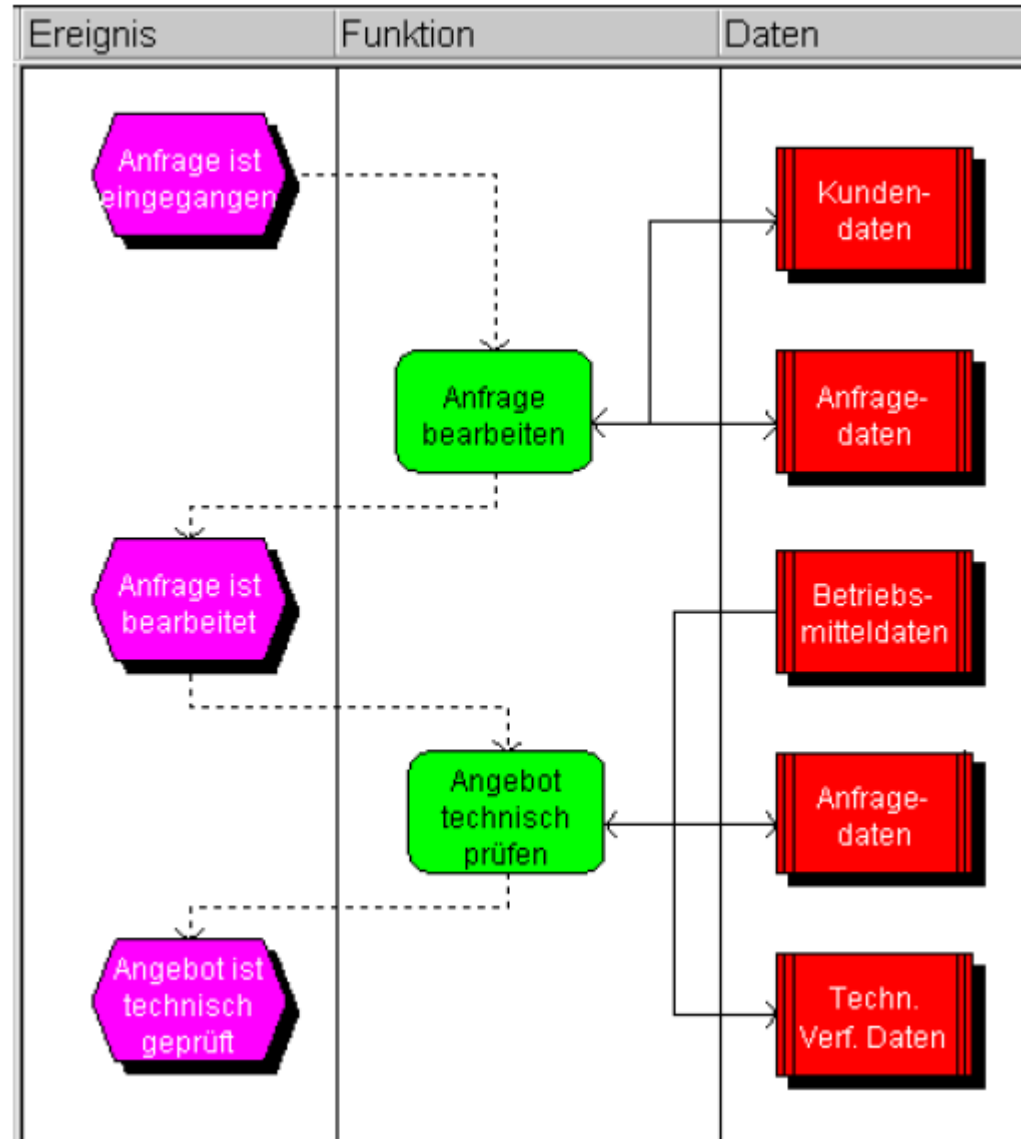
eEPKs: Ein einfaches Beispiel



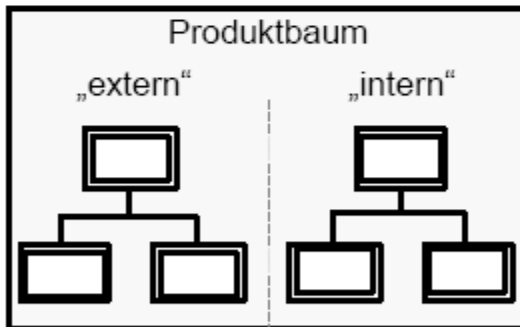
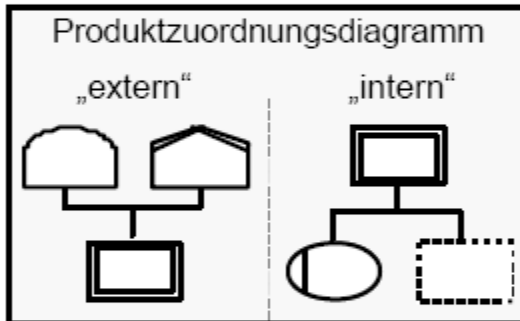
VKD in ARIS

VKD (Vorgangskettendiagramm)

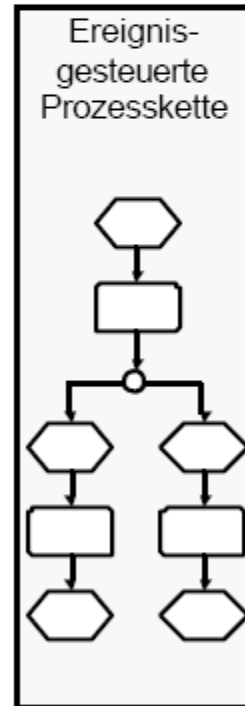
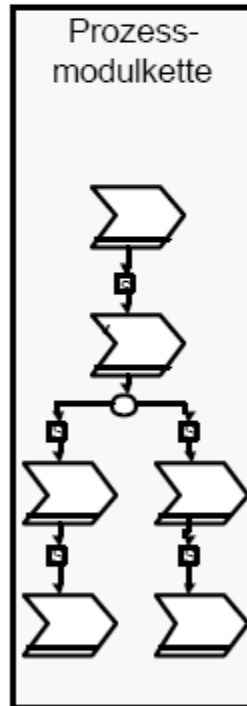
- Erfasst die gleiche Sachverhalte als eine eEPK
- Die Objekte von gleichen Typen sind jedoch spaltenweise angeordnet
- Erhöht die Übersichtigkeit der Modellen



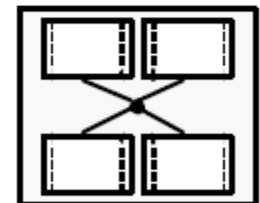
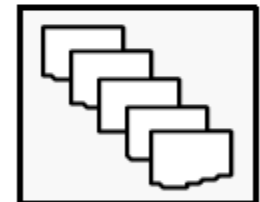
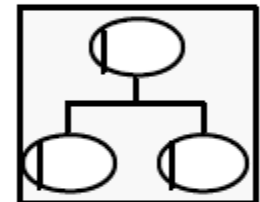
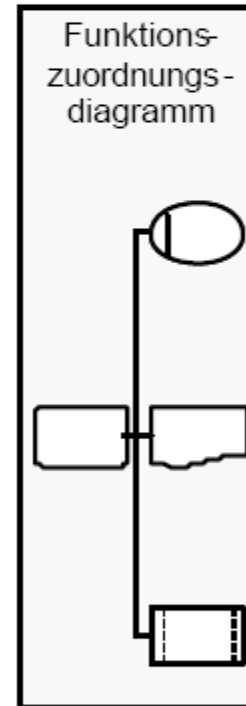
Produktmodell



Prozessmodell



Ressourcenmodell



Rahmenwerk zur Dienstleistungsmodellierung (Quelle: Griebel/Klein/Scheer)

ARIS-Methode, Zusammenfassung



- Merkmale der Modellierung in ARIS:
 - Viele Modell- und Objekttypen
 - Modellübergreifende Objekttypen und Beziehungstypen
 - Keine lose Grafik, Symbole sind immer semantisch hinterlegt
 - Anpassbarer Detaillierungsgrad auf jeder Ebene.
 - Ambition alle wichtigen Unternehmensabläufe erfassbar zu machen (Zeitplanung, Arbeitskräfte (Organigramm, Schichtkalender), Finanzplanung (Kostenartendiagramm), Markt (Wettbewerbskräftemodell), Materialwirtschaft, Lager, ...)
 - Die verschiedenen Modelltypen stellen lediglich verschiedene Ansichten, die in der Firmen-Soll- und Ist-Realität interessant sein könnten.
- Geeignet für Business Reengineering, Prozessmodellierung, z.T. Systementwurf

Prozessmodellierung in UML



- UML: Unified Modelling Language
- graphische Modellierungssprache zur objektorientierten Modellierung eines zu entwickelnden Anwendungssystems beliebiger Komplexität
- beschreibt:
 - Systemzweck
 - **Anwendungsfall-Diagramm (Use Case)**
 - statische Systemstruktur
 - Klassendiagramm
 - Paketdiagramm
 - Verteilungsdiagramm
 - dynamisches Systemverhalten
 - **Interaktionsdiagramme**
 - **Sequenzdiagramm**
 - **Kollaborationsdiagramm**
 - Zustandsdiagramm
 - **Aktivitätsdiagramm**

Rational Rose - Inbetriebnahme - [Activity Diagram: Maschinen aufstellen / Maschinen aufstellen]

File Edit View Format Browse Report Query Tools Add-Ins Window Help

Herr Pretz : Projektleiter Sped 1 : Spedition Herr Müller : Montageleiter Team 1 : Montage-Team

Anlage 1 : Maschinen

Object Specification for Offline

General Incoming Object Flows Outgoing Object Flows

Name: Offline

Class: IB-Ticket

State: (Unspecified)

Stereotype:

Documentation:
Offline Ticket auf dem Laptop des Montageleiters

Persistence
 Persistent Static Transient

Multiple instances

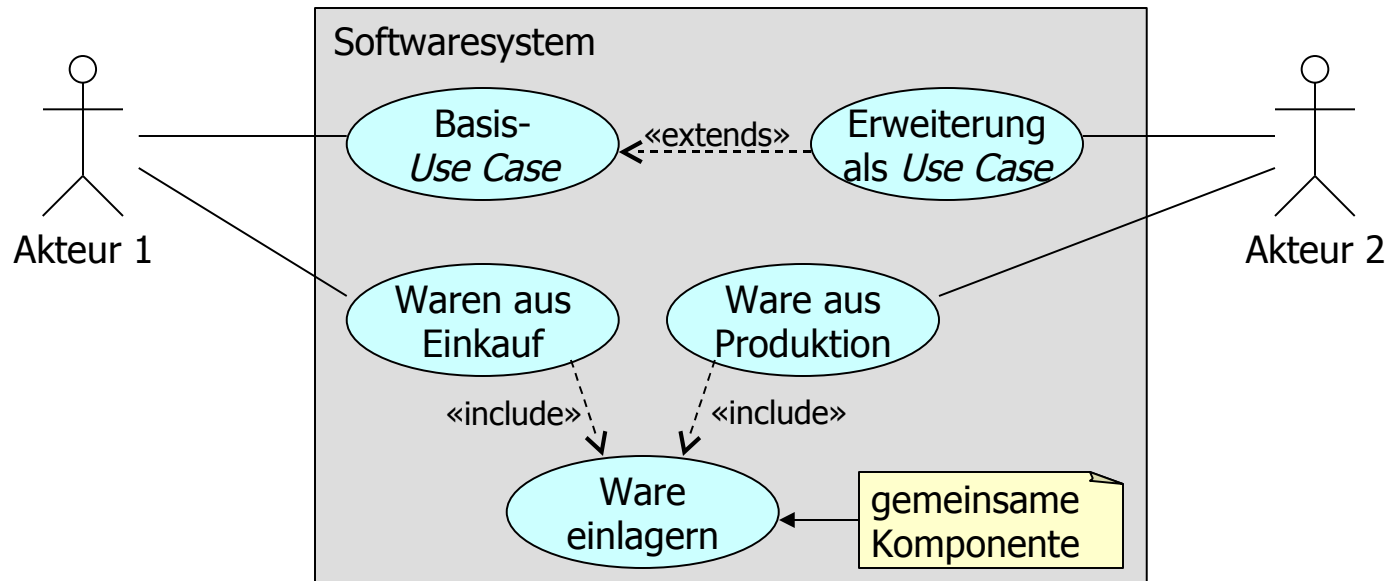
OK Cancel Apply Browse Help

10:32:35 [Customizable Menus]
 10:32:35 [Customizable Menus]
 10:32:35 [Customizable Menus]
 10:32:35 [Customizable Menus]
 Log/

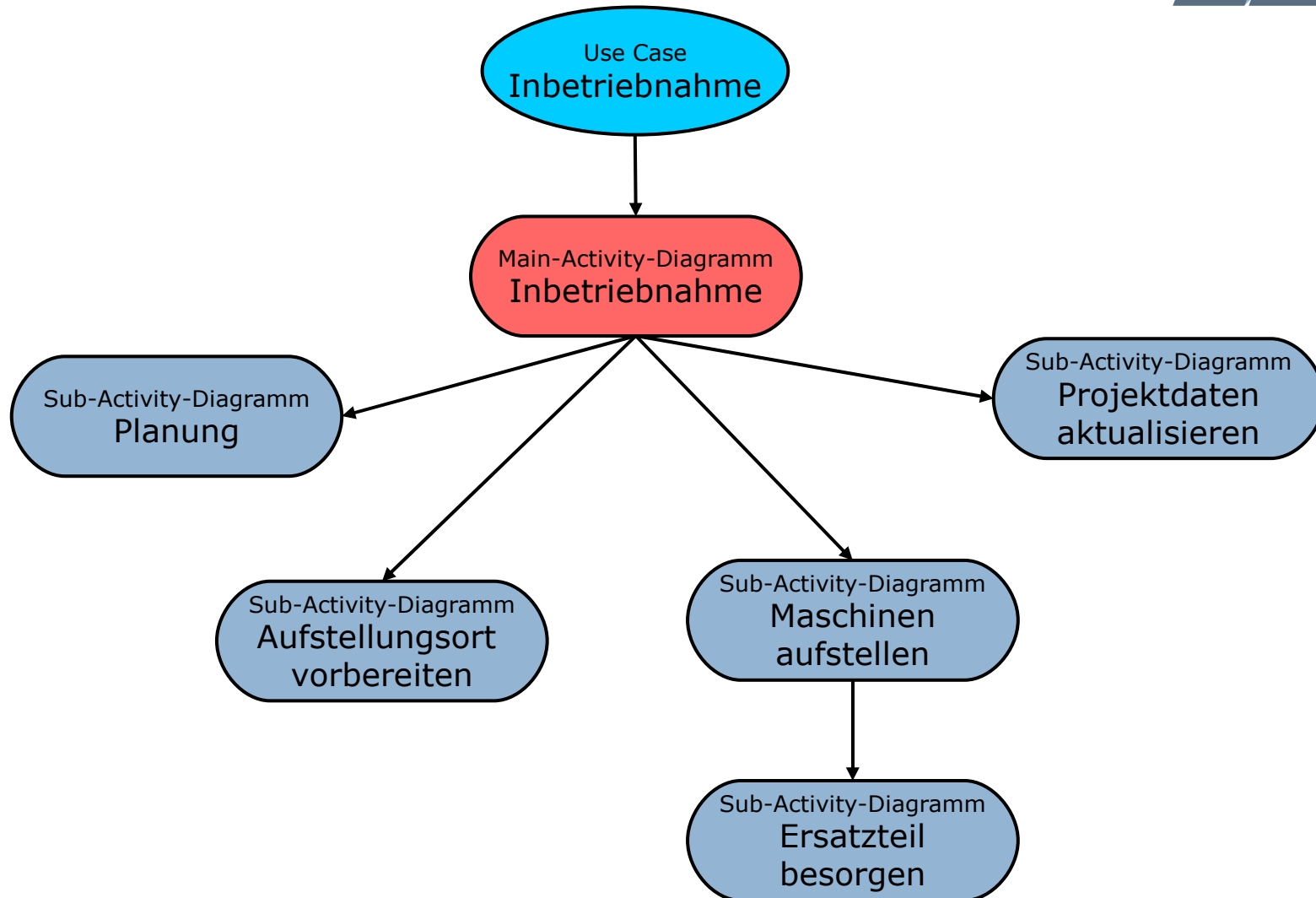
For Help, press F1 Default Language: Analysis

UML: Use Case-Diagramm

- beschreibt die Funktionalität eines (Software-)Systems, die ein Akteur ausführen muss, um ein gewünschtes Ergebnis zu erhalten / Ziel zu erreichen
- geben auf hohem Abstraktionsniveau einen guten Überblick über System und Schnittstellen
- Use Cases ermöglichen es, mit künftigen Benutzern ohne Details über das System zu sprechen
- Akteure sind Benutzer (Personen / automatisierte Systeme), die außerhalb des Systems stehen

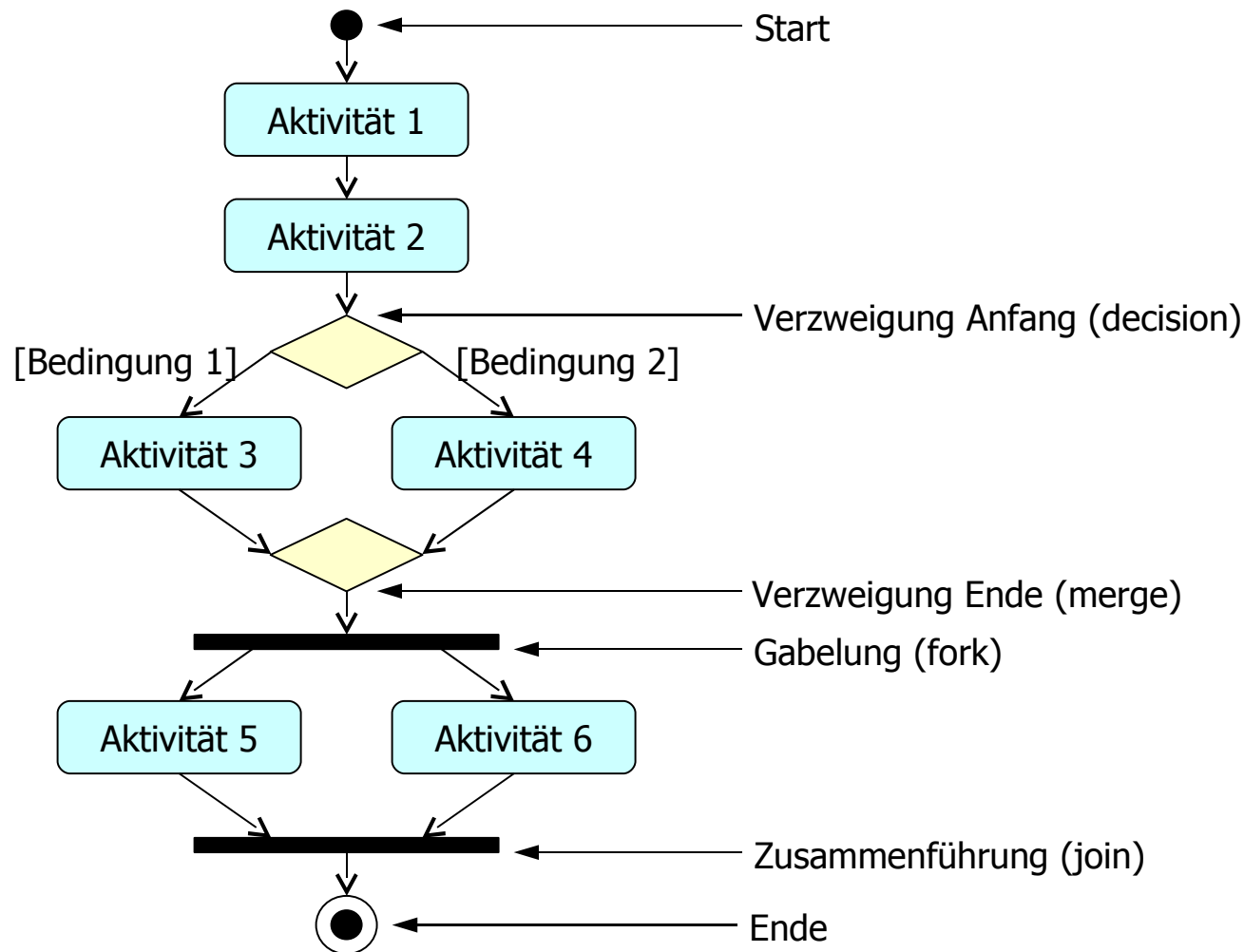


UML: Diagrammhierarchie



Beispiel: Aktivitätsdiagramm

- ist ein Ablaufdiagramm, welches auch komplexe Use Cases graphisch gut modellieren kann



UML-Methode, Zusammenfassung



- Kommunikation und Produktivität von Entwicklern im objektorientierten Umfeld zu ermöglichen und zu verbessern
- eingesetzt in allen Bereiche der Software- und Systementwicklung
-> da Programme wie Rational Rose angefertigte UML-Diagramme analysieren können und daraus z.B. für die Implementation Methodenrumpfe generieren können
- Zur Modellierung mit UML stehen eine Vielzahl von Diagrammtypen zur Verfügung, z.B.:
 - das Anwendungsfalldiagramm (Use Case diagram)
 - das Aktivitätsdiagramm (Activity diagram)
 - das Sequenzdiagramm (Sequence diagram)
- Es fehlt bei UML eine Unterstützung für z.B. folgende Aufgaben:
 - Zeitplanung
 - Ressourcenplanung (ERP – Enterprise Resource Planning)
 - Geschäftsprozessmodellierung (vor allem bei umfangreichen Geschäftsprozessen)

Prozessmodellierungstools



- CASE-Tools
 - ARIS-Familie (IDS Scheer)
 - EasyER (Evergreen) or (Visible)
 - Object Team (Cayenne)
 - ObjecTime Developer (ObjecTime)
 - Rhapsody (i-Logix)
 - Rose (Rational Soft.)
 - SA/Object Architect (Popkin)
 - SELECT Enterprise (Select)
 - StP/UML (Aonix)
 - Visual CASE (Stingray)
 - Visual Thought (Confluent)
- Entwicklungsumgebungen
 - Together/J (OI soft)
 - Together/Professional (OI soft)
- Drawing Tools
 - MagicDraw UML (No Magic)
 - UML v1.1a Template for Visio (Navision)

Quelle: http://sunsite.nus.edu.sg/pub/cetus/oo_uml.html; 04/2004