

Vorlesung Wintersemester 2010/11

Konzepte und Anwendung von Workflowsystemen

Kapitel 10: Anpassung und Änderungen von Geschäftsprozessen

Lehrstuhl für Systeme der Informationsverwaltung, Prof. Böhm
Institut für Programmstrukturen und Datenorganisation (IPD)

Überblick Kapitel 10

Anpassung und Änderungen von Geschäftsprozessen

- ◆ Methoden des Geschäftsprozess-Reengineering im Unternehmen
 - CPI
 - BPR
 - Schritte
 - Richtlinien
 - Bewertungskriterien
 - Varianten
- ◆ Flexibilität in Workflow-Management-Systemen
 - Motivation
 - Kriterien
 - Methoden zur Unterstützung von Flexibilität

Zwei wichtige Methoden: CPI und BPR

Methoden
CPI
BPR
Flexible WfMS

Continuous Process Improvement (CPI)

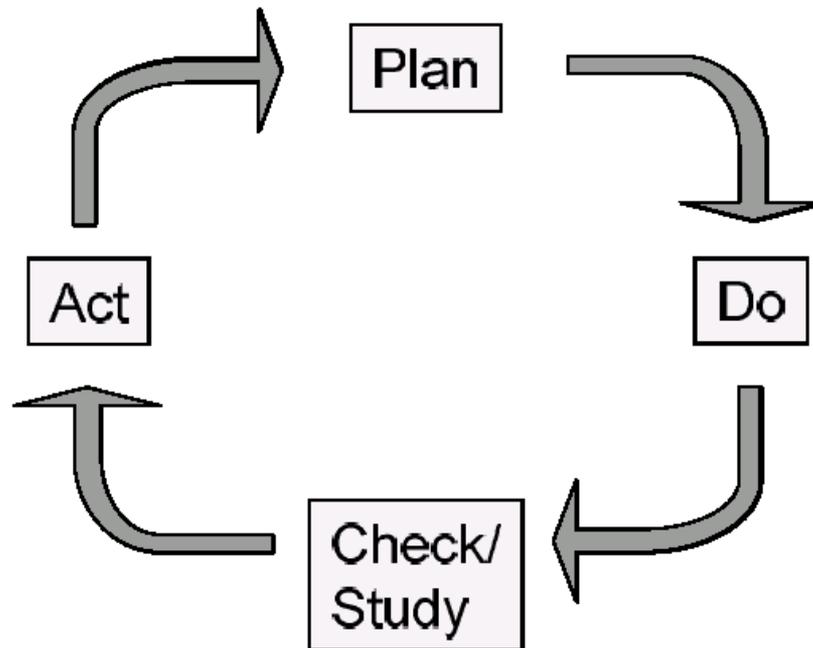
- ◆ Anstelle eines „radikalen Durchbruchs“
 - Optimierung der Prozesse durch kontinuierliche, inkrementelle Verbesserung
- ◆ CPI ist wichtiger Bestandteil des Total Quality Management-Ansatzes (TQM)
 - Kundenbezogen
 - geht vom Qualitätsgedanken an die Problematik heran
 - Ziel: Erhöhung des Qualitätsbewusstseins im Unternehmen

Continuous Process Improvement als Bestandteil des TQM

- ◆ stetige Verbesserung der Qualität des Prozesses
 - -> hohe Prozessqualität
 - stetige Verbesserung der Qualität des Produktes
 - -> hohe Produktqualität
 - -> hohe Kundenzufriedenheit
- ◆ jede Aktivität beinhaltet auch Messungen/Überprüfungen als Basis zur Aufdeckung von Verbesserungsmöglichkeiten
- ◆ Verbesserungszyklen z.B. für Produktentwicklung, CRM, etc.
- ◆ regelmäßige Eliminierung von Überflüssigem/Inaktuellem (z.B. unnötig gewordene Aktivitäten, ...)

Continuous Process Improvement als Bestandteil des TQM

- ◆ stetige, systematische Verbesserung der Qualität durch Verbesserungszyklen:
Plan-Do-Check-Act (PDCA) – Zyklus von Deming:



Weitere TQM-Aspekte (Einschub)

- ◆ Handeln nach Fakten
 - Fakten / statistische Analysen sind Basis für Planung, Überprüfung, Leistungsüberwachung und -verbesserung
 - objektive Daten ermöglichen rationale Entscheidungen
- ◆ Einbezug aller Angestellten
 - ständige Aus- und Weiterbildung der Angestellten,
 - Motivation zu höherer Verantwortung, Kreativität, Innovation
- ◆ TQM Kultur
 - offene, kooperative Unternehmenskultur
 - Verantwortung der einzelnen Angestellten für Kundenzufriedenheit

Business Process Reengineering (1)

- ◆ Faktoren:
 - die Kunden übernehmen das Kommando
 - der Wettbewerb wird intensiver
 - der permanente Wandel wird zur Konstante
- ◆ Hammer und Champy: „Reengineering the Corporation“ (1993):

*“The fundamental rethinking and radical redesign of business processes to bring about dramatic improvements in ...
cost, quality, service, and speed.”*
- ◆ Existierende Prozesse und Organisationsstrukturen werden nicht beachtet!
- ◆ Statt „organize before automate“: „process thinking“!

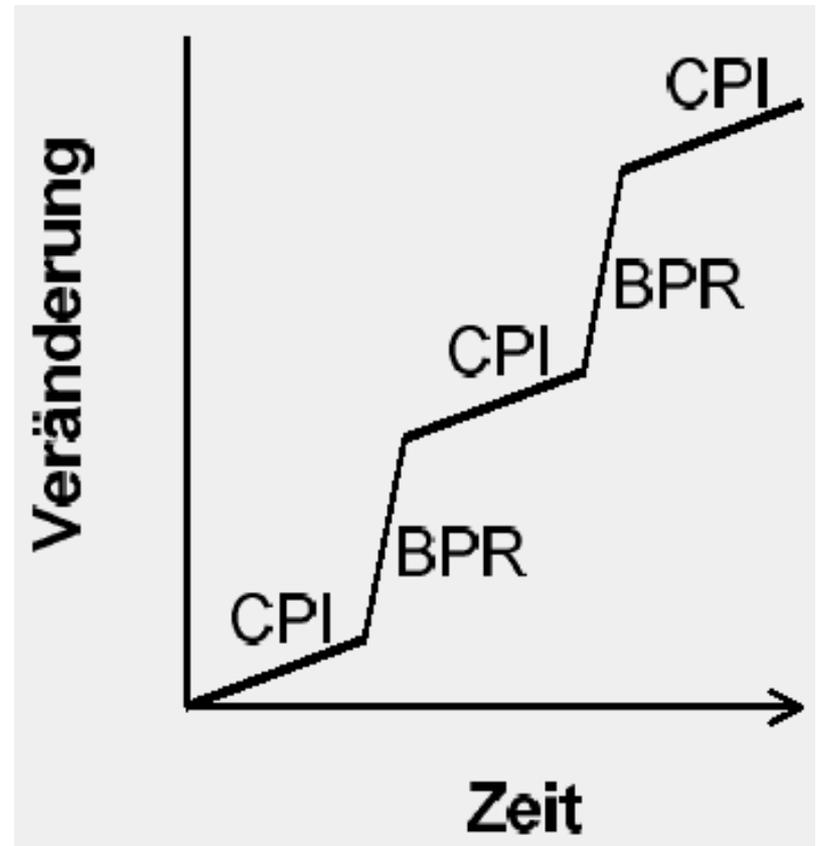
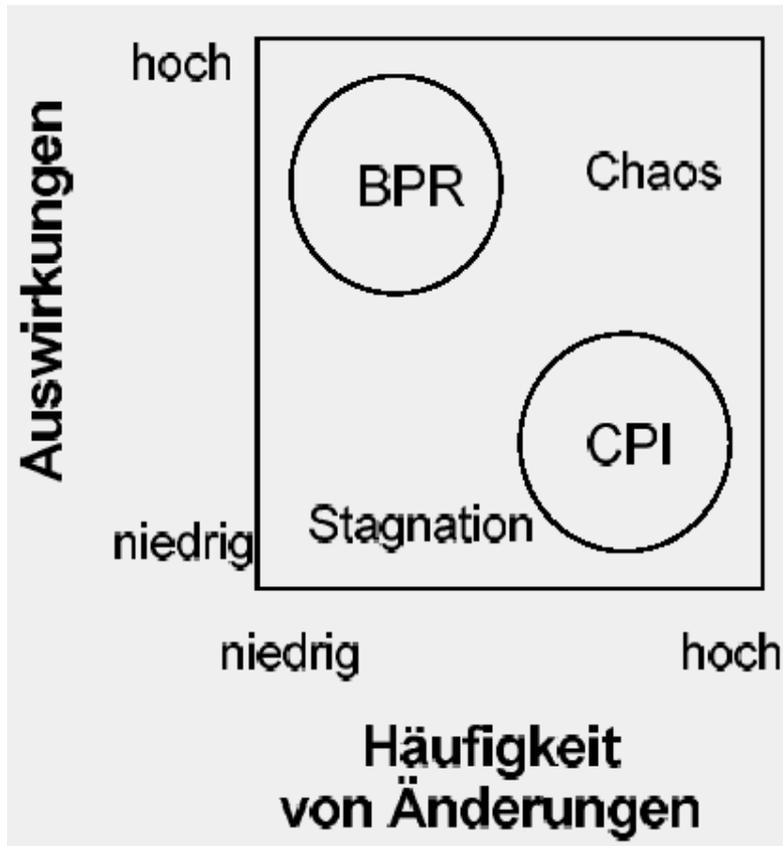
Business Process Reengineering (2)

- ◆ BPR ist fundamentales Überdenken und radikale, dramatische Neugestaltung der Unternehmensprozesse
 - fundamental:
 - Warum machen wir die Dinge, die wir tun?
 - Weshalb machen wir sie auf diese Art und Weise?
 - radikal:
 - völlige Neugestaltung
 - Missachtung aller bestehenden Strukturen u. Verfahrensweisen
 - dramatisch:
 - als Resultat sind Verbesserungen um Größenordnungen angestrebt

Unternehmensprozesse:

*Augenmerk nicht auf Aufgaben, Positionen, Menschen,
Strukturen SONDERN auf Prozessen*

Vergleich von CPI und BPR



Vergleich von CPI und BPR nach Davenport

	Verbesserung (CPI)	Innovation (BPR)
Veränderung	schrittweise	radikal
Ausgangspunkt	existierender Prozess	völlig neuer Zustand
Änderungshäufigkeit	selten/ständig	selten/einmalig
Erforderliche Zeit	kurz	lang
Vorgehensweise	bottom-up	top-down
Reichweite	begrenzt	unternehmensweit, funktionsübergreifend
häufiger Auslöser	statistische Kontrollen	IT
Verbesserung	meist Aufgaben	Prozess, bzgl. der gewünschten Ziele

Symptome für Verbesserungsbedarf

- ◆ zu viele Fälle im System
- ◆ Durchlaufzeit im Verhältnis zur Servicezeit zu hoch (niedriger Servicelevel)
- ◆ zu hohe Ressourcenauslastung
- ◆ hohe Varianz in der Durchlaufzeit (instabiles System)
- ◆ Zahl der Organisationsbrüche
- ◆ Zahl der Medienbrüche

Schritte des Reengineering (1)

- ◆ Beim BPR wird von Neuem begonnen
- ◆ ein existierender Prozess als Ausgangspunkt, wie beim CPI, wird nicht betrachtet

Unklar:

Welche Punkte müssen beim neuen Entwurf eines „guten“ Workflows berücksichtigt werden?

Schritte des Reengineering (2)

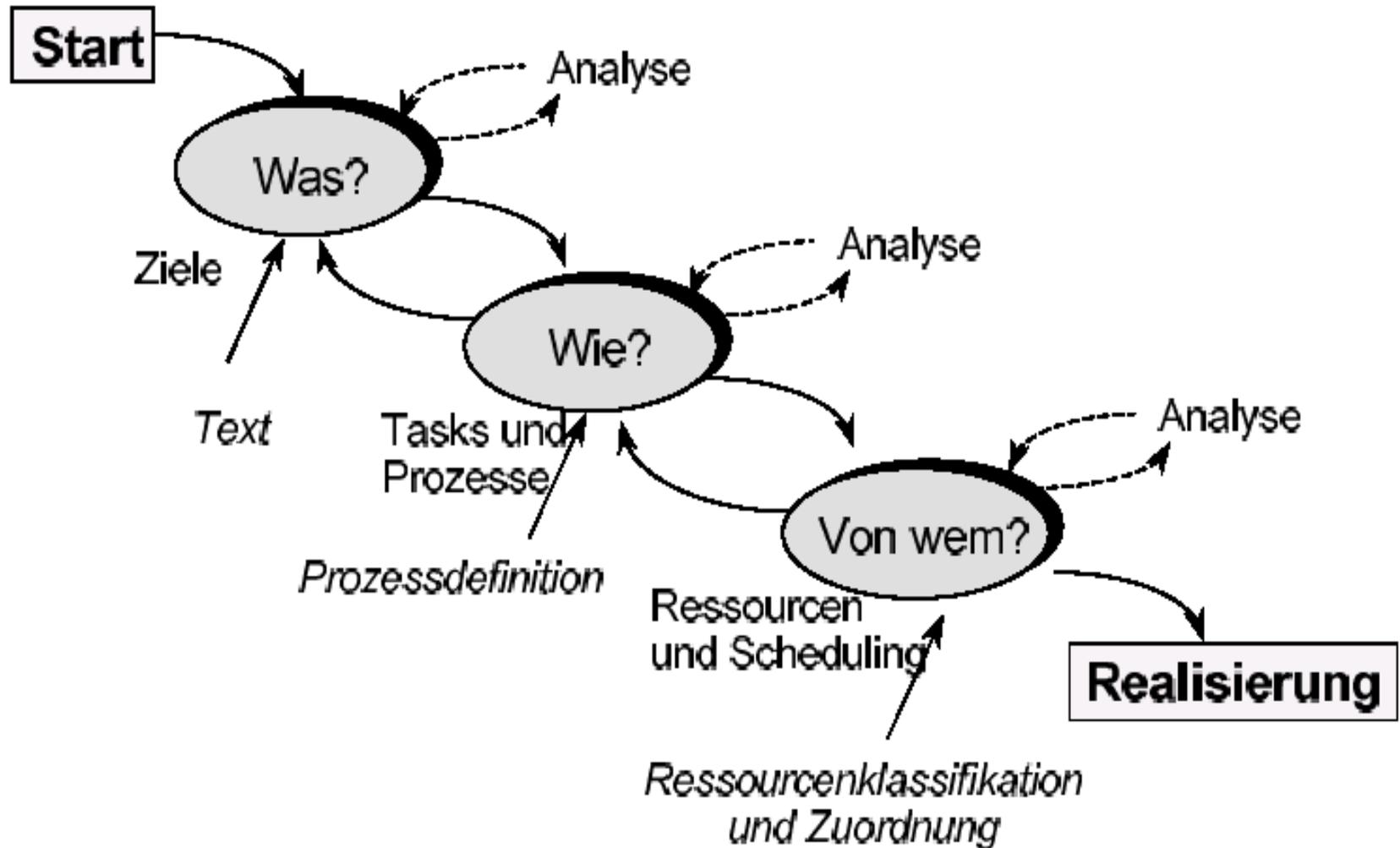
- ◆ Start mit Zieldefinition:
 - messbare Zielvorgaben, Bewertungskriterien definieren
 - Beispiele: Kennziffern, wie Produktivität, Anzahl der Reklamationen

“Start from the future and work backwards!”

Schritte des Reengineerings (3)

- ◆ Orientierung an den kritischen Geschäftsprozessen, d.h. an allen Prozessen, die direkt mit der Leistungserstellung zu tun haben
- ◆ Ausrichtung dieser Geschäftsprozesse am Kunden
- ◆ Konzentration auf Kernkompetenzen, d.h. auf spezifische Fähigkeiten des Unternehmens, durch die es sich von den anderen Unternehmen abhebt
- ◆ Nutzung modernster Informationstechnologie

Ablaufschema



Richtlinien

- ◆ Beginn mit der **Identifizierung des Falles** (Case)
- ◆ Ein Fall wird meist durch einen (internen oder externen) Kunden initiiert
- ◆ der Prozess sollte zu einer Wertschöpfung bezüglich des Falles führen
- ◆ ein Fall besitzt einen Lebenszyklus mit Beginn und Ende
- ◆ ein Fall kann nicht unterteilt werden, die Arbeit jedoch schon

Prozessfestlegung

Aus der Fallidentifizierung kann die **Definition des Prozesses** abgeleitet werden:

- ◆ möglichst genaue Festlegung des Prozess-Ziels
- ◆ möglichst frühe Festlegung des Prozess-Umfangs
- ◆ Festlegung der logischen Abhängigkeiten
- ◆ Abstraktion von Ressourcen beim Entwurf der Prozessdefinition
- ◆ Prozess-Entwurf ist iterativer Entwicklungsvorgang (keine Angst vor Fehlern)
- ◆ Granularität der Tasks verändert sich
- ◆ Hierarchie-Konzepte sollten genutzt werden (top-down)

Logische Arbeitseinheiten - Tasks

Tasks als LUWs (Logical Units of Work) während des Reengineering-Prozesses:

- ◆ atomar: Commit oder Rollback
- ◆ eine Aufgabe wird von einer Ressource zu einer Zeit an einem Ort ausgeführt
- ◆ Rüstzeiten sollten minimiert werden
- ◆ Umfang der LUW`s sollte so klein wie möglich und nur so groß wie notwendig werden (Commit zum frühest möglichen Zeitpunkt)

ACID-Eigenschaften

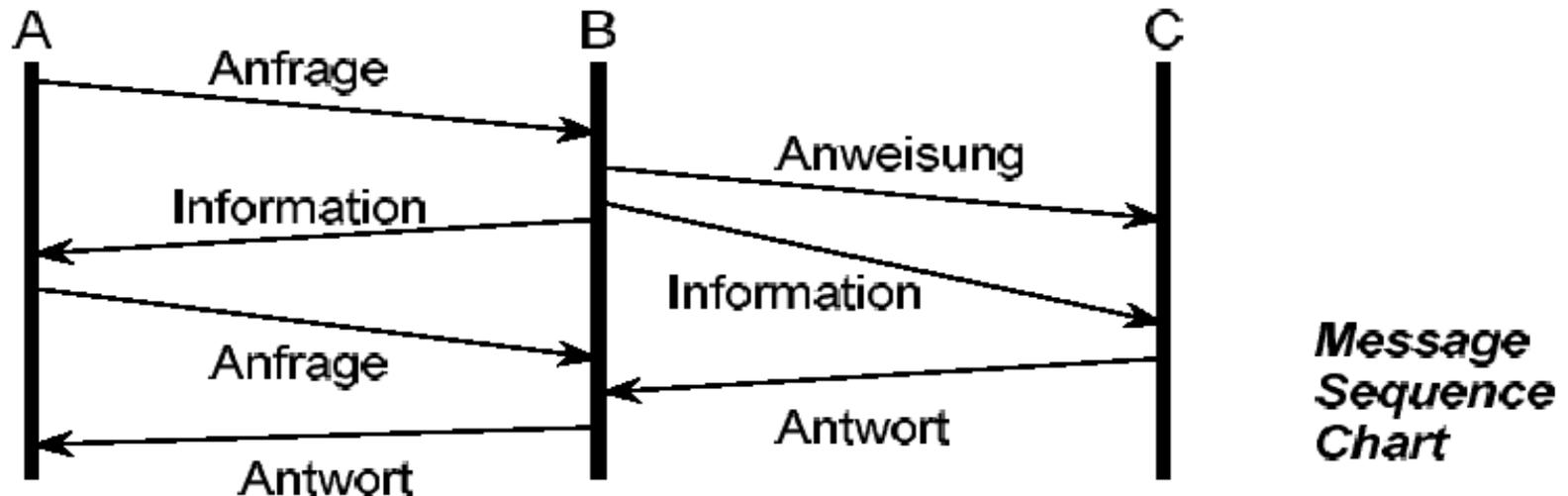
ACID-Eigenschaften aus der Datenbankwelt /
Transaktionsverwaltung übertragen:

- ◆ **Atomizität** ("alles oder nichts", Commit oder Rollback)
- ◆ **Consistency** (Konsistenz) (eine beendete Task überführt das System in einen gültigen Zustand)
- ◆ **Isolation** (Tasks beeinflussen sich nicht gegenseitig, auch wenn sie parallel ausgeführt werden)
- ◆ **Dauerhaftigkeit** (Ergebnisse einer komplettierten Task gehen nicht verloren)

Kommunikationswege

Informationen über einen Prozess sammeln:

- ◆ Weg der Dokumente
- ◆ Kommunikationswege zwischen Einzelpersonen, Teams, Abteilungen
- ◆ Kommunikationsmuster feststellen



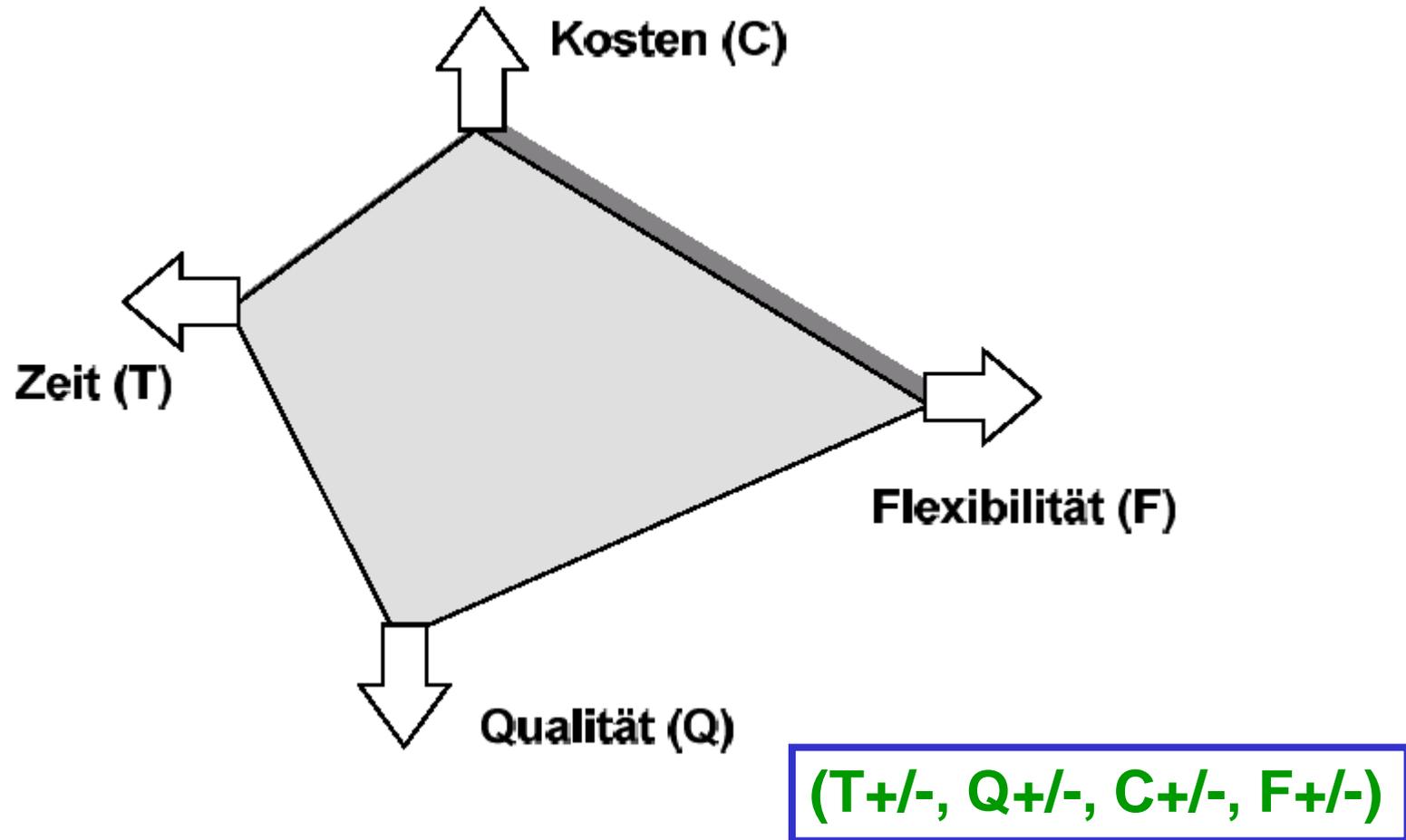
Bewertungskriterien

Vier Kriterien zur Bewertung eines Prozessentwurfs:
(= kritische Leistungsgrößen)

Zeit (T)
Qualität (Q)
Kosten (C)
Flexibilität (F)

Problem: in der Regel kommt es zu einem Trade-Off
(**Abwägen der Kriterien**)

Reengineering-Kriterien: Trade-Off



Reengineering-Kriterium I: Zeit

Durchlaufzeit setzt sich zusammen aus

- ◆ Servicezeit (inklusive Rüstzeit)
- ◆ Transportzeit (kann oft verhindert werden)
- ◆ Wartezeit
 - limitierte Ressourcen
 - externe Einflüsse (Events)

Kennzahlen im Zusammenhang mit Durchlaufzeit

- ◆ Durchschnitt
- ◆ Varianz
- ◆ Service-Level
- ◆ Termintreue

Reengineering-Kriterium II: Qualität

Externe Qualität: Zufriedenheit des Kunden in Bezug auf

- ◆ Produkt: entspricht den Anforderungen / Erwartungen
- ◆ Prozess: Service-Level

Interne Qualität: Arbeitsbedingungen, z. B.

- ◆ Anspruch
- ◆ Abwechslung
- ◆ Kontrolle

Anmerkung:

Es existiert oft eine positive Korrelation zwischen interner und externer Qualität

Reengineering-Kriterium III: Kosten

Kostenarten

- ◆ fixe und variable Kosten
- ◆ Personal-, Material- und Systemkosten
- ◆ externe Kosten
- ◆ Bearbeitungs-, Verwaltungs- und Support-Kosten
- ◆ ...

Anmerkung:

Es kommt oft zu einem Trade-Off zwischen Personal- und Systemkosten

Reengineering-Kriterium IV: Flexibilität

- ◆ Flexibilität = Fähigkeit, auf Veränderungen zu reagieren.
- ◆ Flexibilität in Bezug auf
 - Ressourcen:
Fähigkeit, verschiedene und neue Tasks auszuführen
 - Prozess:
Fähigkeit, verschiedene Fälle zu handhaben, verschiedene Auslastungen zu verkraften
 - Management:
Fähigkeit, Regeln und Ressourcen-Allokation zu ändern
 - Organisation:
Fähigkeit, die Struktur den Anforderungen des Marktes und der Partner anzupassen

Varianten des Reengineering (1)

mögliche Varianten bzgl. der BPR-Richtlinien:

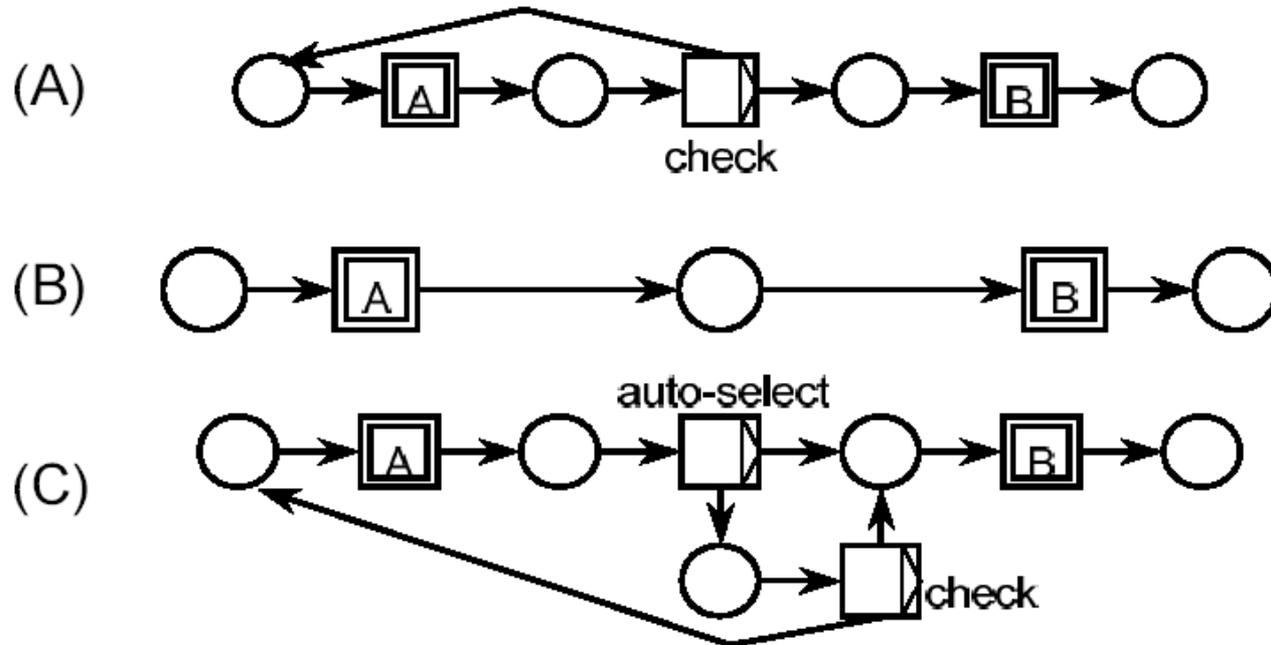
- ◆ Notwendigkeit von Tasks überprüfen
- ◆ Prozess-Verantwortliche bestimmen
- ◆ Case-Verantwortliche bestimmen
- ◆ Größe der Tasks überdenken
- ◆ Trade-Off zwischen einem generischen Prozess und verschiedenen Versionen für einen Prozess überdenken
- ◆ Trade-Off zwischen generischen Tasks und verschiedenen spezialisierten Tasks überdenken
- ◆ Grad der Parallelität erhöhen

Varianten des Reengineering (2)

mögliche Varianten bzgl. der BPR-Richtlinien:

- ◆ Neue Möglichkeiten durch Einsatz von Technologie untersuchen
- ◆ Kommunikationswege überprüfen und optimieren
- ◆ Bedenke: ein elektronisches Dokument ist überall und nirgends
- ◆ Ressourcen dazu nutzen, wozu sie gut sind
- ◆ Möglichst hohe Flexibilität bewahren
- ◆ Rüstzeiten (Setup Time) verringern
- ◆ Setups verringern
- ◆ Nicht den Weg des Papiers automatisieren!

Notwendigkeit von Tasks



Beispiel: jede Check-Task kann ausgelassen werden:
 Trade-Off zwischen den Kosten der Überprüfung und den
 Kosten durch Nicht-Überprüfung

(T+,Q-,C+/-)

Verantwortliche

- ◆ Ein Prozessverantwortlicher überwacht den Prozess und achtet auf Engpässe, Kapazitätsprobleme und Fristen.
- ◆ Case-Verantwortliche werden einem Fall (Case) zugeordnet. Sie sind für die korrekte Bearbeitung des Falls zuständig.

Vorteile:

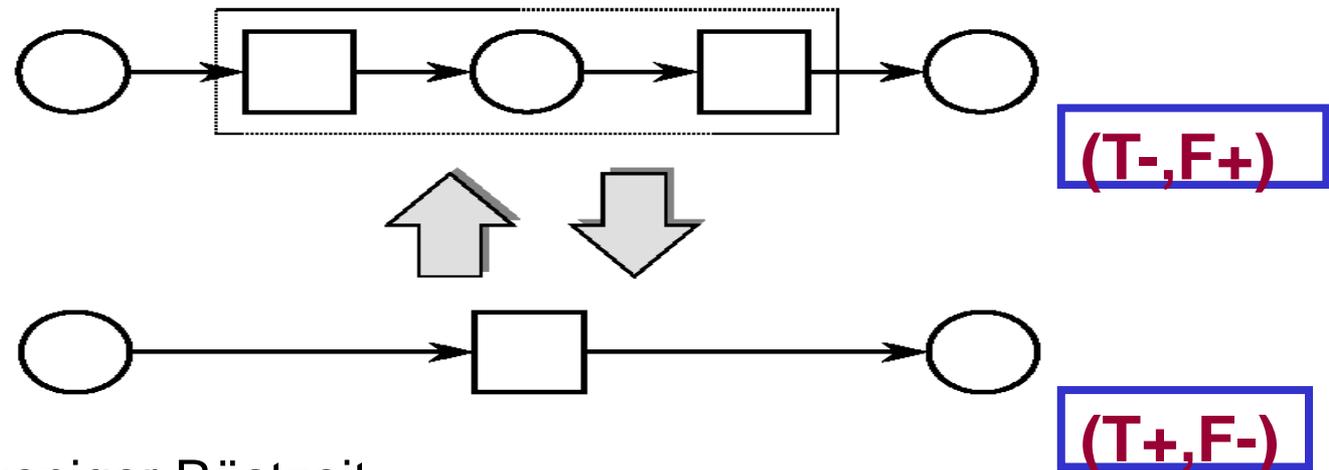
- ◆ Verantwortung (Commitment)
- ◆ **ein** Ansprechpartner
- ◆ Verringerung der Rüstzeit (Setup Time) bei Fällen

(Q+)

Größe von Tasks

Pro: weniger Arbeit bei Commit, ermöglicht Spezialisierung

Contra: Rüstzeit, Fragmentierung, weniger Verantwortung

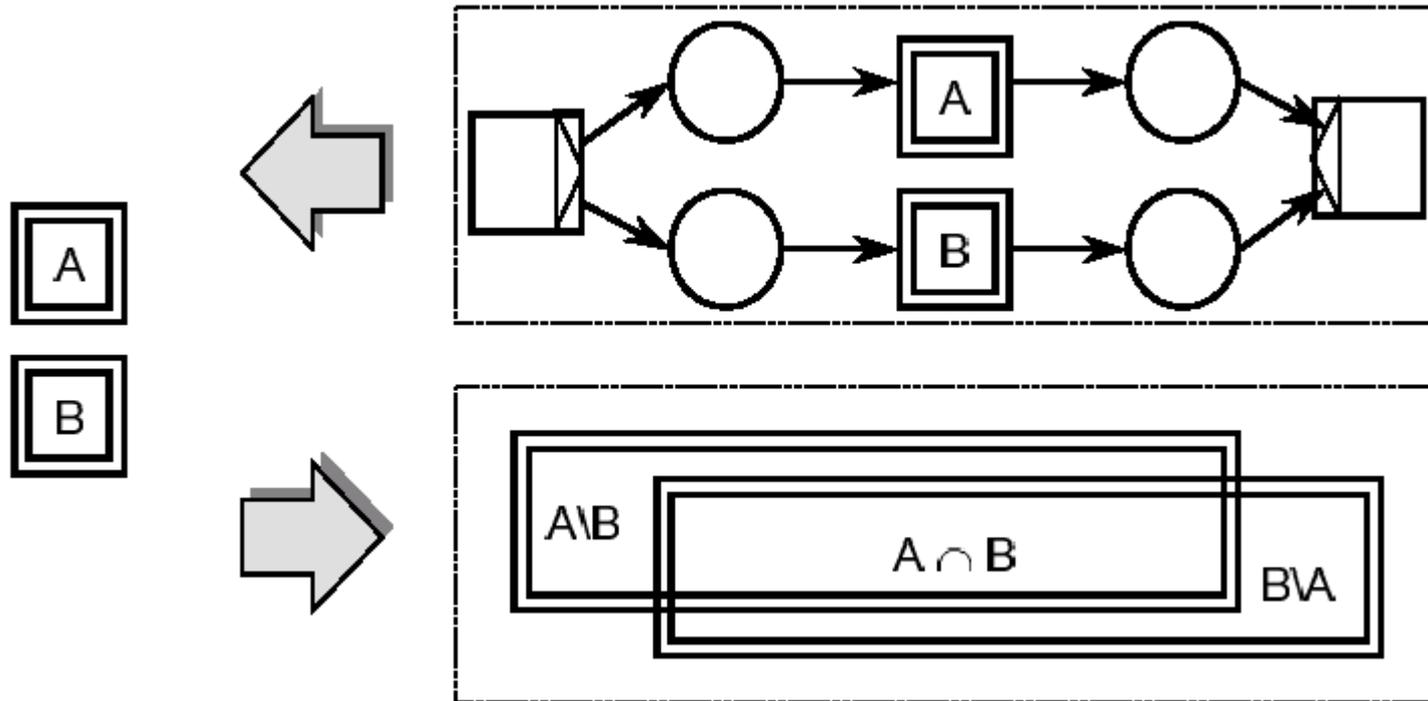


Pro: weniger Rüstzeit, ...

Contra: Commit, erfordert Qualifizierung

Anmerkung: Trade-Off zwischen Komplexität des Prozesses und Komplexität der Tasks

Generische Prozesse vs. Versionen



Aspekte: Einfachheit, Effizienz, Kontrollierbarkeit, Wartbarkeit,...

(F+/-)

Generische Tasks vs. Versionen

vgl. generische Prozesse

Spezialisierung kann die folgenden Auswirkungen haben:

- ◆ verbesserte Allokation der Ressourcen
- ◆ bessere Ausführung der Tasks
- ◆ weniger Flexibilität
- ◆ komplexere Prozesse
- ◆ Monotonie

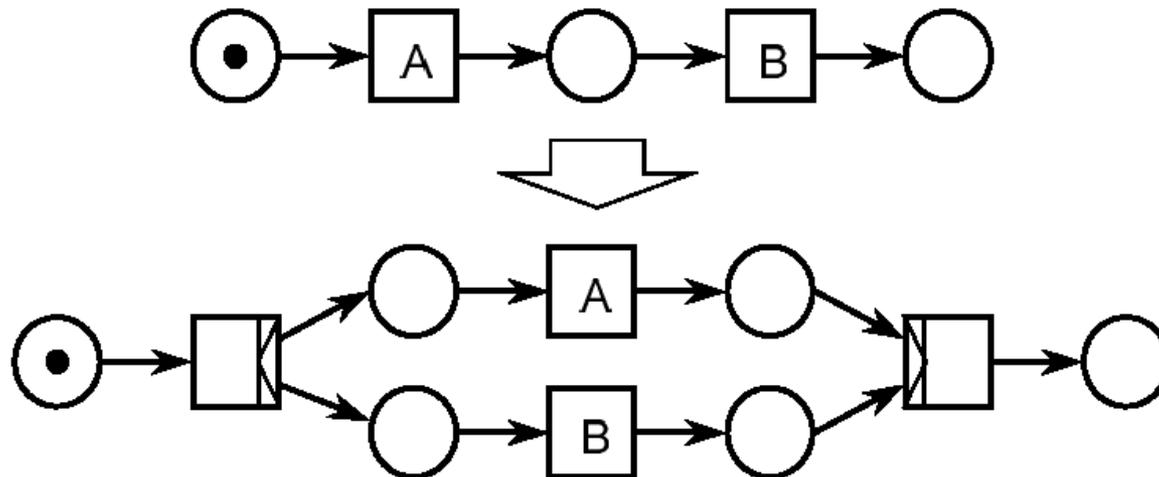
(T+,F-)

Grad an Parallelität

Hoher Grad an Parallelität führt zu einer besseren Performanz:

- ◆ Verringerung der Wartezeiten
- ◆ bessere Nutzung von Kapazitäten

Benötigt wird: IT-Infrastruktur, die gleichzeitige Nutzung von Daten ermöglicht, entsprechende Prozesslogik



Einsatzmöglichkeiten für IT (1)

- ◆ DBMS: Sharing von Daten
„ein elektronisches Dokument ist überall und nirgends“
- ◆ Netzwerke
- ◆ Kommunikation: E-Mail, Intranet, WWW, ...
- ◆ Verteilung von Information: Transport von Daten ist schnell und kostengünstig
- ◆ Automatisierung von Tasks oder automatisierte Unterstützung bei der Ausführung

Einsatzmöglichkeiten für IT (2)

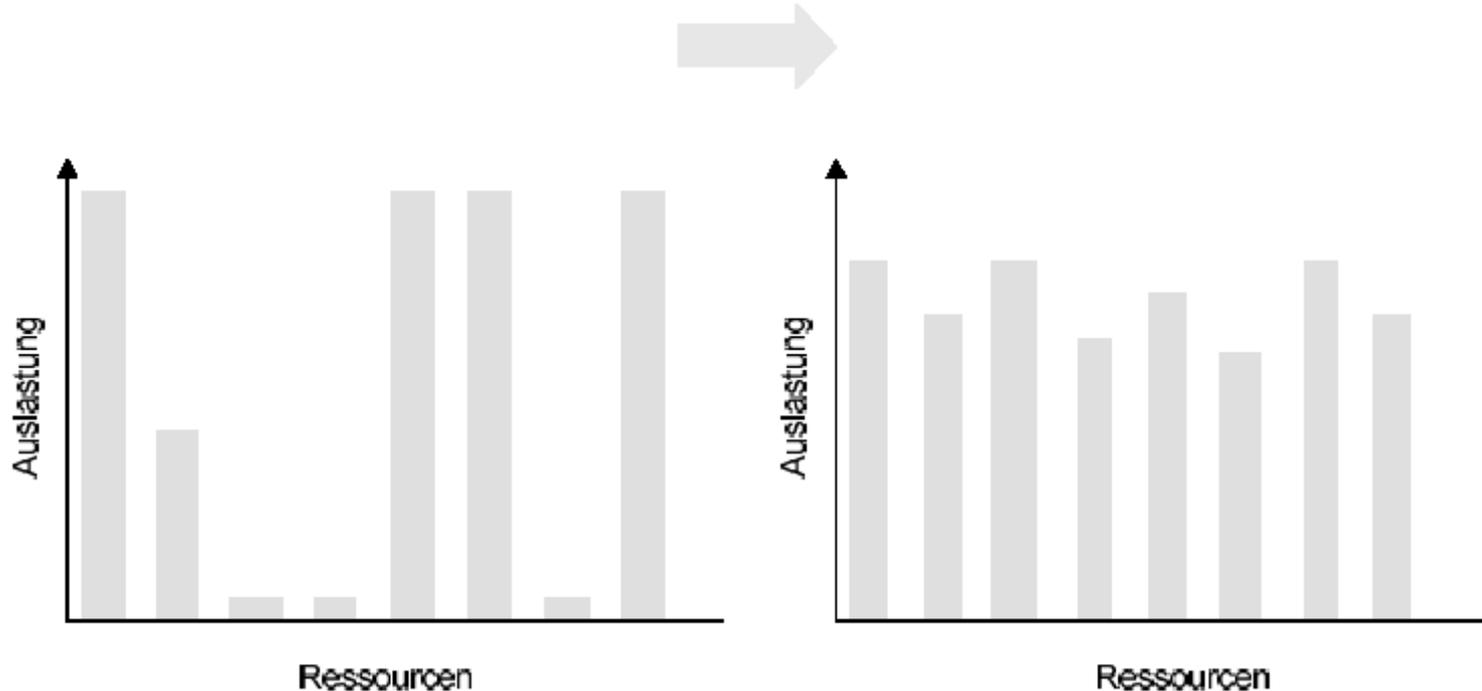
Beispiele für Einflüsse

- ◆ Parallelität (Sharing of data)
- ◆ Kundenbezug (Web-Dokumente)
- ◆ Asynchrone statt synchrone Kommunikation
- ◆ Analyse historischer Daten (z.B. Risikoanalyse)

(T+,Q+,C+/-,F+)

Ressourcen-Allokation (1)

Ressourcen gleichmäßig auslasten
(Vermeiden von Überlasten, während andere warten)



Ressourcen-Allokation (2)

- ◆ Ressourcen ihren Fähigkeiten entsprechend nutzen
- ◆ jedoch auf Inflexibilität durch Spezialisierung achten
- ◆ Routine entstehen lassen
- ◆ Rüstzeit so gut wie möglich verringern:
 - für Cases
 - für Tasks

(T+,Q+/-)

Kommunikationsstruktur

- ◆ Anzahl der auszutauschenden Nachrichten verringern (insbesondere externe Kommunikation)
- ◆ Handhabung der Nachrichten automatisieren
- ◆ Kommunikationsfehler vermeiden (EDI, ebXML, WWW)
- ◆ möglichst asynchrone statt synchrone Kommunikation verwenden (z.B. Email).

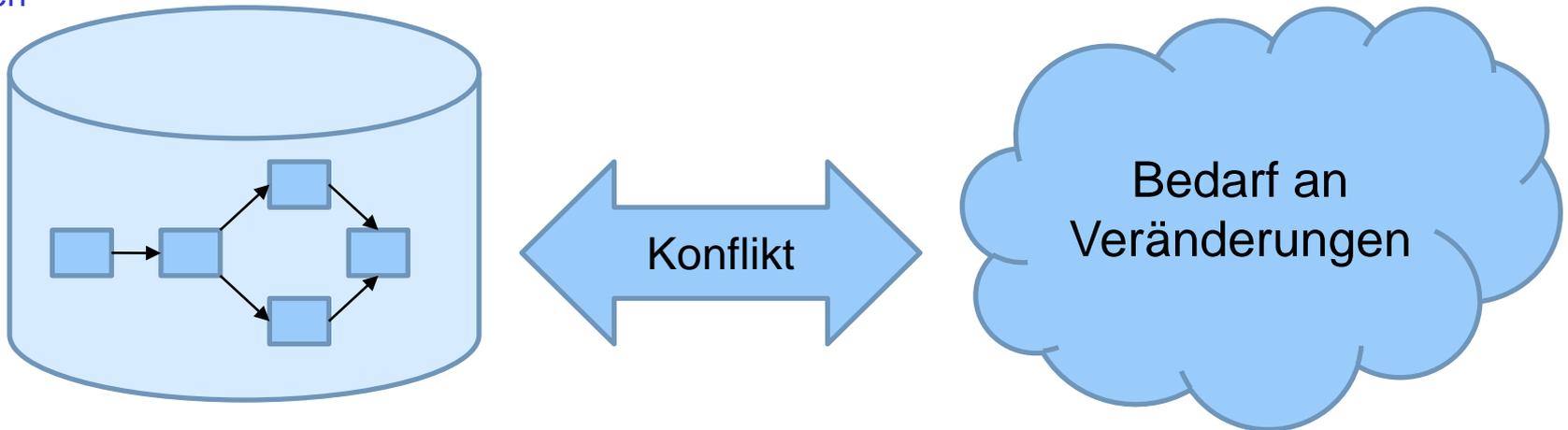
(T+,Q+,C+/-,F-)

Überblick Kapitel 10 – 2. Teil

- ◆ Flexibilität in Workflow-Management-Systemen
 - Motivation
 - Kriterien
 - Methoden zur Unterstützung von Flexibilität

Motivation

Methoden
Flexible WfMS
Motivation
Kriterien
Methoden



Prozess-Schemata: fest vorgegebener, idealisierter Ablauf mit dem Ziel der Wiederverwendung

Flexible Workflow-Management-Systeme

- ◆ Aspekte der Workflow Spezifikation (Kontrollfluss, Ablaufverhalten, Datenfluss, alle Tätigkeiten/ Aktivitäten sind dem Modellierer im Voraus nicht immer bekannt)
- ◆ Ausführungsumgebung und alle Komponenten sind dem WF-Modellierer nicht immer bekannt bzw. ändern sich
- ◆ Workflow Engine sollte neben normaler Ausführung auch Ausnahmebehandlung abwickeln
- ◆ Fehlertypen und mögliche Fehlerbehandlung sind im Voraus nicht absehbar



Notwendigkeit für Flexibilität!

Flexibilität in Workflow-Management-Systemen (1)

- ◆ Ad-hoc Änderungen für spezielle oder nicht vorhersehbare Situationen
 - Umsetzung von (meist spontanen) Abweichungen vom Prozessmodell
 - z.B. spezielle Forderungen von wichtigen Kunden, situationsspezifische Anforderungen
- ◆ Behandlung gelegentlich auftretender (absehbarer) Fehlerfälle
 - Exception-Handling (nicht im Fokus dieses Kapitels)
 - z.B. Abbruch eines Arbeitsschrittes, Service nicht verfügbar
- ◆ Workflow-Spezifizierung zur Laufzeit
 - Deployment eines unvollständigen Workflow-Schemas
 - genaue Spezifizierung zur Laufzeit (instanzspezifisch, z.B. Auswahl von Teilprozessen)
- ◆ Anpassung von Prozessabschnitten
 - Änderungen an zu Grunde liegenden Datenstrukturen oder den Daten selbst
 - Varianten
 - z.B. personalisierte Abläufe für konkrete Kunden

Flexibilität in Workflow-Management-Systemen (2)

- ◆ Änderungen der Workflow-Schemata
 - Veränderung der gesetzlichen/ betrieblichen Rahmenbedingungen, Märkte/ Kundenanforderungen
 - Optimierung der Prozesse (CPI, BPR)
- ◆ Technologieänderungen
 - WfMS / Web Content Management Systeme
 - Umsetzung neuer Standards
- ◆ Berücksichtigung weiterer Kriterien
 - Sicherheitsaspekt, Zugriffsschutz, etc.
 - Organisatorische Aspekte:
 - z.B. Änderung der Verfügbarkeit von Ressourcen, Ressourcen-Allokationsmechanismen
 - Unterstützung von Kunden/Benutzern bei der Änderung von Workflows

Herausforderungen an flexible WfMS

- ◆ Wie kann Flexibilität in WfMS unterstützt werden?
 - Entwurf
 - Ausführungs- (Enactment-) Strategien
- ◆ Umsetzung von Continuous Process Improvement (CPI) durch flexible WfMS

Aspekte von WfMS (Wdh. Kap. 2)

Methoden
Flexible WfMS
Motivation
Kriterien
Methoden

Veränderung von
Arbeitsschritten

Modifikation des
Workflow-Ablaufs

Änderung der Daten
bzw. Datenstruktur

funktionaler Aspekt

verhaltensbezogener
Aspekt

datenbezogener
Aspekt

organisatorischer
Aspekt

operationaler
Aspekt

weitere Aspekte

Änderung der
Organisationsstruktur/
Zugriffsrechte/
Verfügbarkeiten

Modifikationen an
Applikationen/
Infrastruktur

Einflussfaktoren und Komponenten

- ◆ Workflow Modellierungssprache (Meta-Modell)
- ◆ Workflow Spezifikation (Workflow-Schema)
 - Aktivitätenspezifikation
 - Abhängigkeiten zwischen Aktivitäten
 - Rollen, Ressourcen
 - Daten
- ◆ Repository (Workflow-Bibliothek)
- ◆ Ausführungsmaschine (Engine)
 - Umsetzung der Workflow Spezifikation
 - Behandlung von Abweichungen von Instanzen
 - Ressourcenmanagement (z.B. Rollenzuweisung, Zugriffsrechte auf Dokumente)
 - Fehlerbehandlung
 - Lastverteilung (Monitoring)

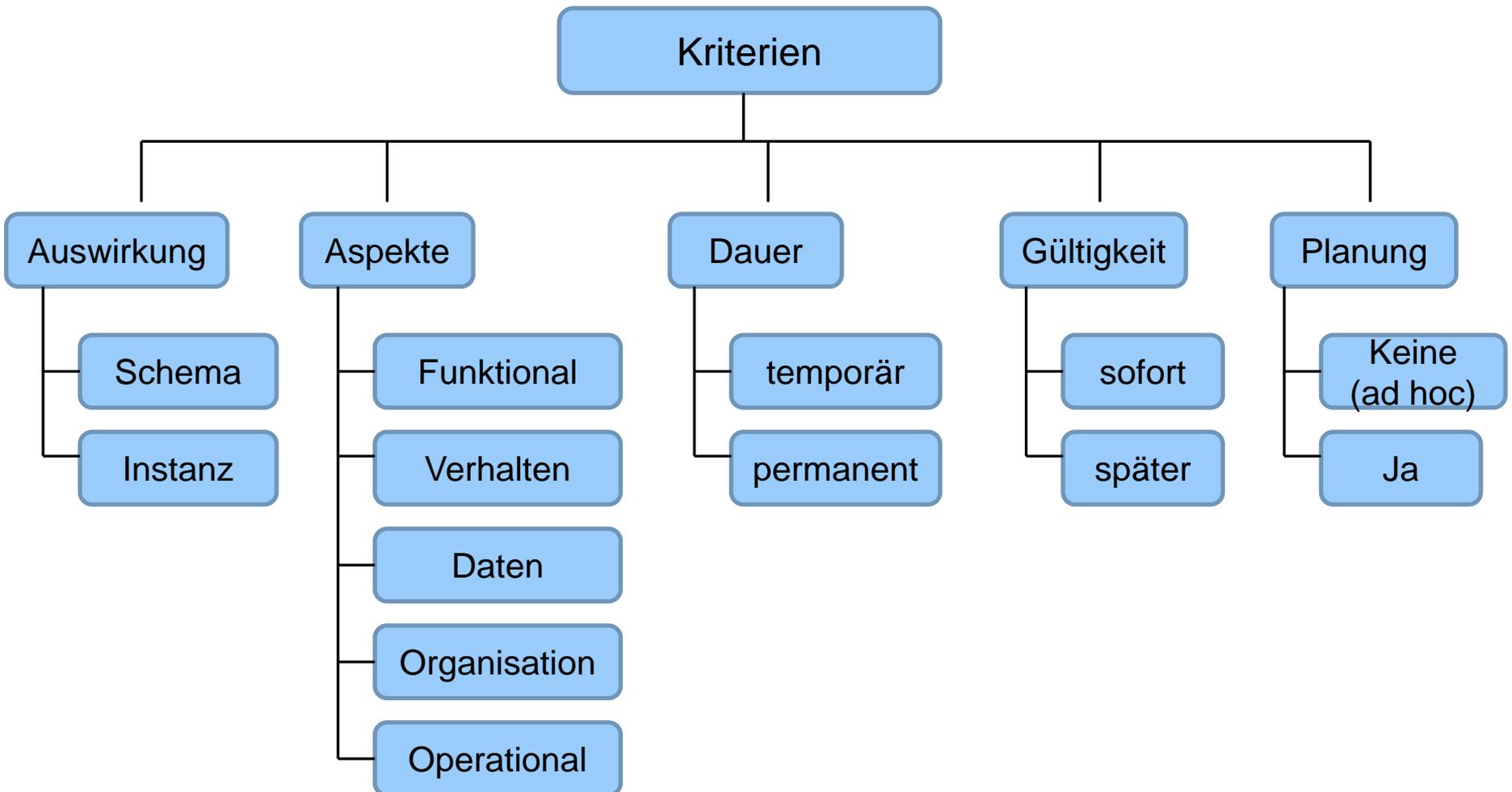
Exemplarische organisatorische Änderungen

- ◆ Unabhängig von Workflow-Schemata
 - Änderungen der Anzahl von Verarbeitungs-Entitäten (Ressourcen), die zur Ausführung einer Aktivität geeignet sind
 - z.B. Änderung von Rollenzuweisungen
 - Ereignisse allgemeinerer Art, die nicht spezifisch für einen speziellen Workflow sind, aber die Ausführung eines Workflows betreffen
 - z.B. Änderung der Zuweisungsart, Änderung von Zugriffsrechten

Exemplarische organisatorische Änderungen

- ◆ Beispiel: Ändern der Zuweisung einer Aktivität „Patienten-Service“ an einen Pool von Krankenschwestern
 - Option 1: neuen Fall der kleinsten Arbeitsliste zuweisen
 - Option 2: Aufnahme in eine allgemeine Fall-Liste und Zuweisung, wenn eine Krankenschwester verfügbar ist
 - Option 3: Überprüfung, welche Krankenschwestern für diesen Fall geeignet sind.
 - Auswahl durch Nutzen von Zusatzwissen aus einer Wissensbasis mit Informationen über Erfahrungswerte, Fähigkeiten der Krankenschwestern etc.
 - Änderung/Anpassung der Wissensbasis
- ◆ Keine Änderung des Workflow-Schemas durch flexible Zuweisung der Aktivitäten zu Rollen/Akteuren

Klassifikation von Änderungen



Beispiele für Veränderungen in Workflows

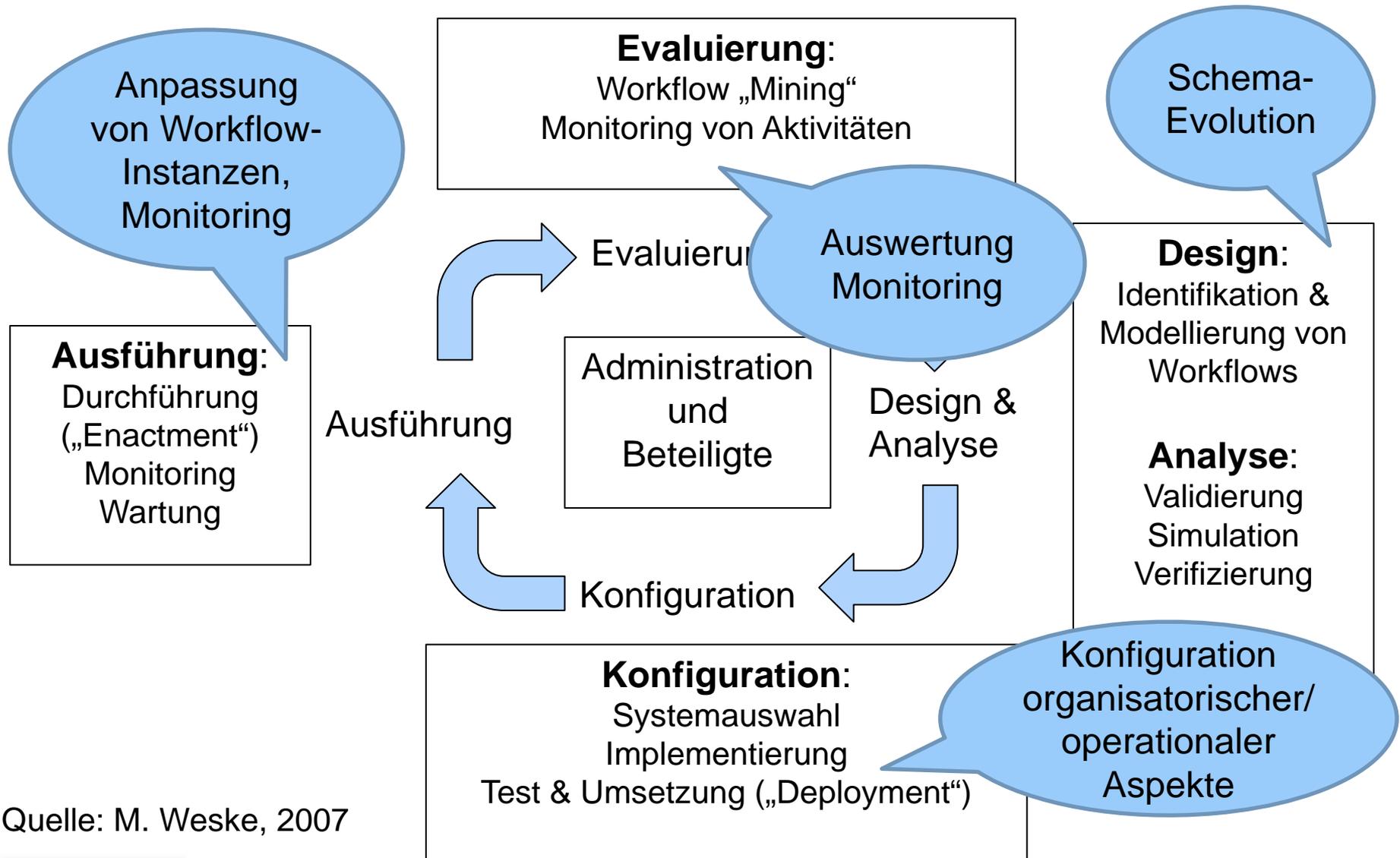
◆ Schema

- Hinzufügen von neuen Aktivitäten/ Löschen von Aktivitäten
- Austausch von Aktivitäten (z.B. durch Subprozesse)
- Re-organisation des Kontrollflusses (z.B. Sequenz -> Parallel)
- Einfügen/Verändern/Löschen von Bedingungen
- Modifikation des Datenflusses

◆ Instanz (Ad-hoc)

- Auslassen von Aktivitäten, Vertauschen von Aktivitäten
- Vorziehen einer Aktivität

Lebenszyklus von Workflows (Wdh. Kap. 2)



Quelle: M. Weske, 2007

Definition Prozessflexibilität

- ◆ Unter Prozessflexibilität wird die Fähigkeit verstanden, mit vorhersehbaren als auch unvorhersehbaren Änderungen umzugehen, indem betroffene Teile des Prozesses **verändert** oder angepasst werden, während die nicht betroffenen Teile **beibehalten** werden [Schonenberg et. al.].

Methoden zur Unterstützung von Prozessflexibilität

◆ Klassifikationen

- [Heinl et. al]:
 - Flexibilität durch Auswahl („Flexibility by Selection“); z.B. alternative Pfade bei Modellierung, spätes Binden
 - Flexibilität durch Anpassung („Flexibility by Adaption“) für Schema und Instanz
 - Ad-hoc Modellierung explizit ausgeklammert
- [Aalst/Jablonski]
 - Flexibilität durch Konfiguration
 - Flexibilität durch Adaption
- [Weber et. al.]: Systematische Klassifikation von „Change Patterns“
 - Adaptions-Pattern (z.B. insert, update, delete von Prozessfragmenten)
 - Patterns für vordefinierte Änderungen (späte Auswahl; spätes Modellieren von Prozessfragmenten)
- [Schonenberg et. al]
 - Folgt (nächste Folien)

Klassifikation von Workflow-Flexibilität nach Schonenberg et. al.

Unterscheidung

- ◆ Flexibilität im Design („Flexibility by Design“)
- ◆ Flexibilität durch Abweichungen („Flexibility by Deviation“)
- ◆ Flexibilität durch Grobfestlegungen („Flexibility by Underspecification“)
- ◆ Flexibilität durch Veränderungen („Flexibility by Change“)

- ◆ Alle Methoden verbessern die Flexibilität
 - Kein Re-design des Prozessmodells notwendig
 - unterscheiden sich in Zeit und Art, wie sie angewendet werden
 - unabhängig voneinander einsetzbar

Flexibility by Design

- ◆ Motivation
 - Realisierung von Ausführungsalternativen, die im Prozessmodell spezifiziert sind
- ◆ Definition
 - Fähigkeit, **Ausführungsalternativen** im Prozessmodell zu berücksichtigen, die während der Laufzeit von Prozessinstanzen ausgewählt werden können
- ◆ Gültigkeitsbereich
 - Jedes Prozessmodell, das Ausführungsalternativen zur Verfügung stellt
- ◆ Realisierungsoptionen
 - z.B. Parallelität, Auswahl, Iteration, multiple Instanzen, Abbruch
- ◆ Diskussion
 - (eigentlich) trivial, allerdings wichtiges Instrument zur Unterstützung von Flexibilität
 - Vorteil: keine Behandlung von Ausnahmefällen zur Laufzeit notwendig
 - Nachteile: Vermischung von Normalfall und Ausnahmefall; Komplexität der Modelle

Flexibility by Deviation (I)

◆ Motivation

- Bedarf an temporären Abweichung von Prozessinstanzen vom Prozessmodell (**Ad-hoc Prozesse**)
- Beispiele Krankenhaus
 - Bei Lebensgefahr werden keine Patientendaten erfasst, obwohl dies im Prozessmodell vorgesehen ist.
 - Die erste Laborüberprüfung macht einen zusätzlichen Bluttest notwendig

◆ Definition

- Fähigkeit einer Prozessinstanz, während der Laufzeit vom Prozessmodell abzuweichen, **ohne** dass das Prozessmodell verändert wird.

◆ Anwendungsbereich

- Prozesse mit generellen Richtlinien für die Ausführungssequenz, die jedoch in Ausnahmefällen verletzt werden können

Flexibility by Deviation (II)

- ◆ Realisierungsoptionen: Variationen der Aktivitäten, die als nächstes ausgeführt werden sollen; Beispiele:
 - Wieder-Ausführung einer derzeit nicht aktiven Aktivität („Redo activity A“)
 - z.B. Wiederholtes Eingeben von fehlerhaften bzw. unvollständigen Daten
 - Auslassen einer Aktivität („Skip activity A“)
 - Auslassen einer (aktiven) Aktivität, z.B. Notfallaufnahme
 - Erzeugen einer zusätzlichen Instanz einer Aktivität
 - Paralleles Ausführen weiterer Aktivitäten
 - Anzahl (z.B. Bestellpositionen) im Vorfeld nicht bekannt, bzw. während Laufzeit der Prozessinstanz modifiziert
 - Aktivieren einer nicht aktiven Aktivität („Invoke activity A“)
 - Beispiel Notaufnahme: Vorziehen einer Routine-Untersuchung im Verdachtsfall
 - Nach Beendigung normaler Prozessablauf

Flexibility by Deviation (III)

◆ Diskussion

- Akteure müssen in der Lage sein (Kompetenz, Kenntnisse, Fähigkeiten, Berechtigungen)
- Vorteil: Trennung von Normalfall und Ausnahmesituation
- Nachteile: Gewährleistung der Korrektheit

Flexibility by Underspecification (I)

- ◆ Motivation
 - Flexibilität hinsichtlich weiterer Ausführungspfade und konkreter Umsetzung von Aufgaben (da z.B im Vorfeld nicht bekannt)
- ◆ Definition
 - Fähigkeit, ein wohlgeformtes, aber **unvollständiges** (d.h. zu Ausführungszwecken nicht hinreichend spezifiziertes) Prozessmodell auszuführen; keine Veränderung, sondern Vervollständigung des Prozessmodells zur Laufzeit
- ◆ Anwendungsbereich
 - Im Vorfeld bekannt, dass das Prozessmodell an speziellen Stellen verfeinert werden muss
 - Beispiele:
 - Grobstruktur bekannt, aber Prozessbeteiligte unterschiedlicher Arbeitsgruppen möchten bestimmten Grad an Autonomie
 - Die erste Laborüberprüfung macht einen zusätzlichen Bluttest notwendig, diese Aktivität kann aus dem Repository entnommen werden und im Ablauf eingefügt werden (Mischfall „Flexibility by Deviation/ Underspecification“)

Flexibility by Underspecification (II)

- ◆ Realisierung mit Hilfe von **Platzhaltern**, die als „underspezifiziert“ gekennzeichnet sind
 - Kann auch als Ad-hoc Prozess interpretiert werden
- ◆ Optionen zur Festlegung der notwendigen Spezifikation
 - Spätes Binden („**Late Binding**“)
 - Auswahl aus vordefinierten Prozessfragmenten (z.B. in Repository verwaltet)
 - Spätes Modellieren („**Late Modelling**“)
 - Explizite Erzeugung von Prozessfragmenten
 - Vollständige Modellierung und/oder Auswahl (und ggf. Modifikation) von vordefinierten Fragmenten
- ◆ Festlegung
 - Statisch: einmalig für alle zukünftigen Ausführungen des Platzhalters
 - Dynamisch: Spezifikation bei jeder Ausführung des Platzhalters (z.B. Ad-hoc Prozesse)

Flexibility by Change (I)

◆ Motivation

- Schema-Evolution: Umsetzung von (zur Modellierungszeit) nicht vorhersehbaren Anforderungen, die eine **permanente** Veränderung (z.B. Hinzufügen, Löschen von Tasks) des Prozessmodells mit sich bringen
- Notwendigkeit einer Re-Modellierung eines Workflows beispielsweise basierend auf
 - Analyse von Monitoringergebnissen
 - Ergebnis einer Simulationsstudie
 - Änderungen in der Organisation, den Rollen, der Rechnerinfrastruktur, etc.
 - Feedback von Systemverwalterseite

◆ Definition

- Fähigkeit zur Veränderung eines Prozessmodells zur Laufzeit. Die laufenden Prozessinstanzen müssen vom alten Prozessmodell zum neuen Prozessmodell überführt werden (**Migration**)

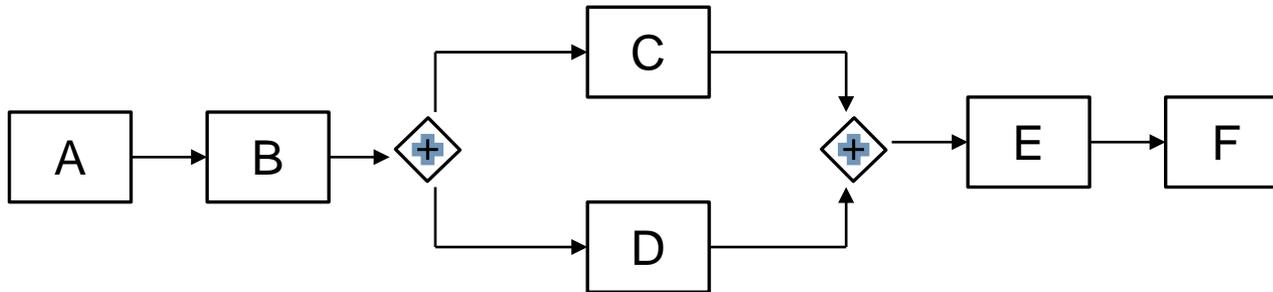
◆ Anwendungsbereich

- Generisch; Umsetzung beliebiger Arten von Änderungen

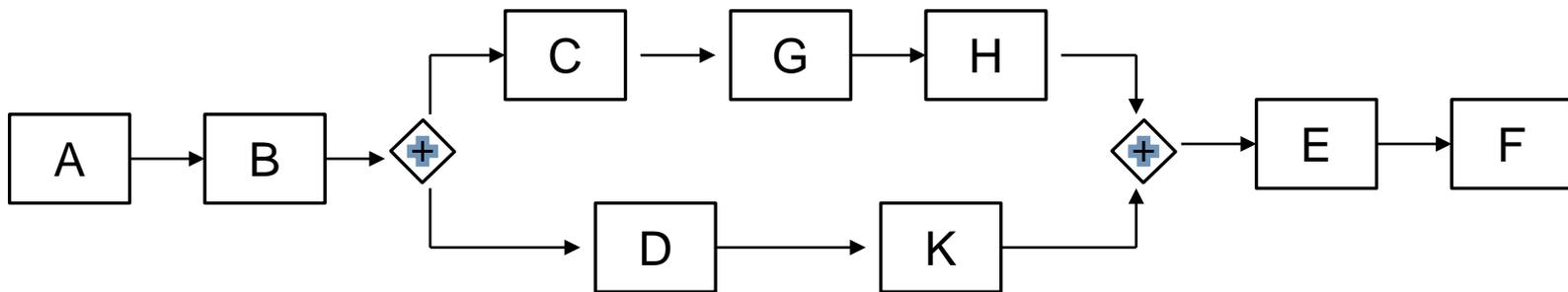
Flexibility by Change (II)

- ◆ Zeitpunkte für erlaubte Veränderungen der Instanzen
 - Entry Time: Änderungen nur vor Erzeugung der Prozessinstanz möglich
 - On-the-fly: Änderungen sind jederzeit möglich
- ◆ Migrationsstrategien:
 - „Forward Recovery“: Abbruch von betroffenen Prozessinstanzen
 - „Backward Recovery“: Abbruch, ggf. Kompensation, und Neustart von betroffenen Prozessinstanzen
 - „Proceed“: laufende Prozessinstanzen verfahren wie bisher, Änderungen werden ignoriert; neue Prozessinstanzen sind änderungs-konform
 - „Transfer“ („propagation“): Überführung der laufenden Prozessinstanzen gemäß des neuen Prozessmodells
- ◆ Diskussion
 - Migration oft nicht trivial
 - Terminologie auch teilweise (adaptive,) dynamische Workflows

Beispiel Migration (I)



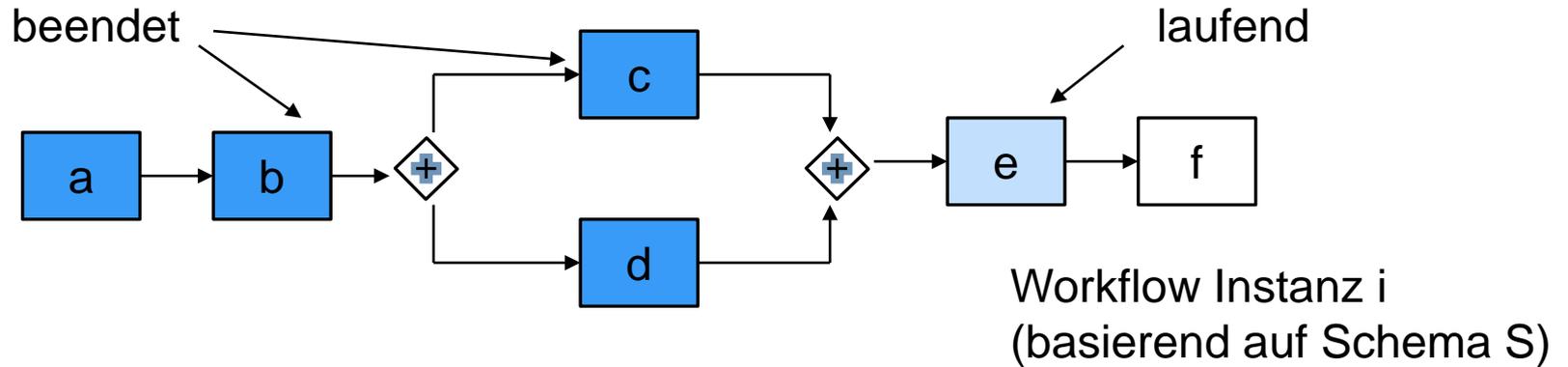
Schema S



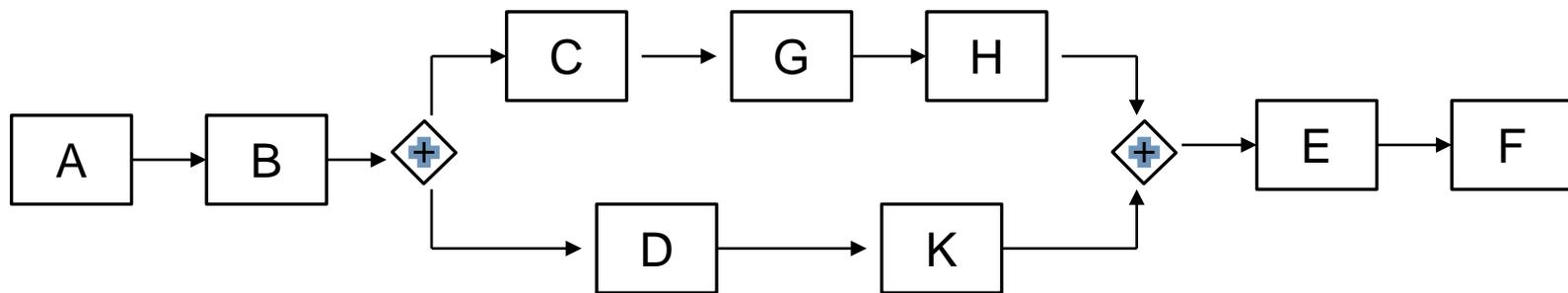
Schema S'

Quelle: Weske 2007

Beispiel Migration (II)



Migration ?

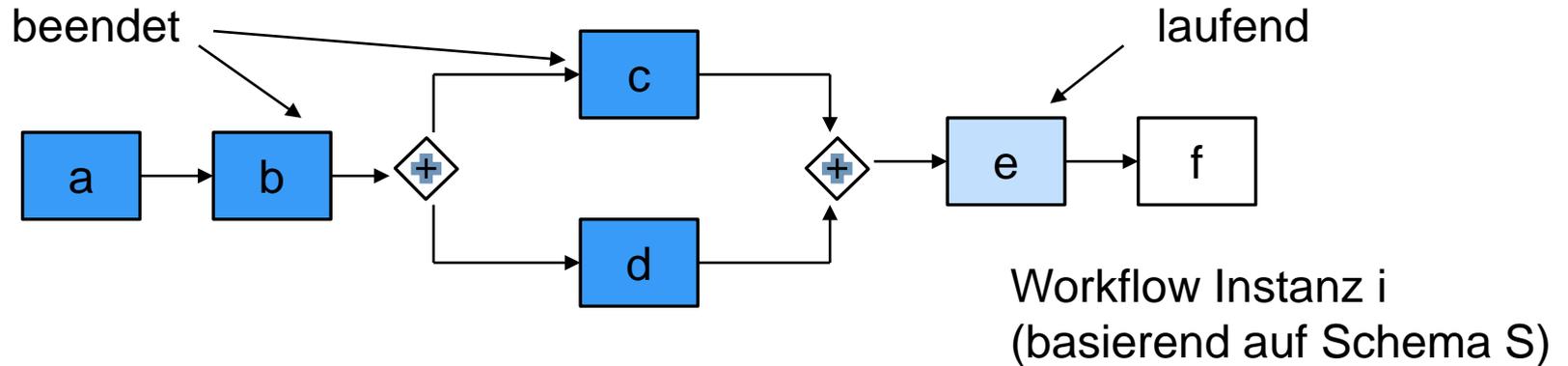


Schema S'

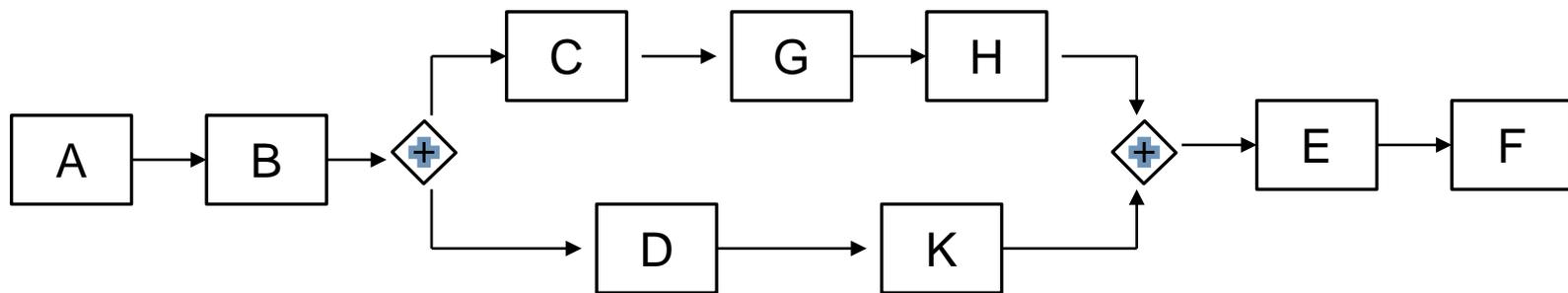
Kriterien für Migration

- ◆ Eine Workflow-Instanz i ist migrierbar (adaptierbar) auf ein Workflow Schema S' :
 - wenn eine Fortsetzung von i möglich ist, so dass i mit S' konform ist (nach [Weske 2007])
 - Definition von Weber: wenn die Ausführungshistorie von i basierend auf S' erzeugt werden kann [Weber et al. 2007]
- ◆ (Potentielle), weiterführende Kriterien
 - Berücksichtigung des Prozesszustandes, Rollback von Instanzen, Korrektheit des Datenflusses

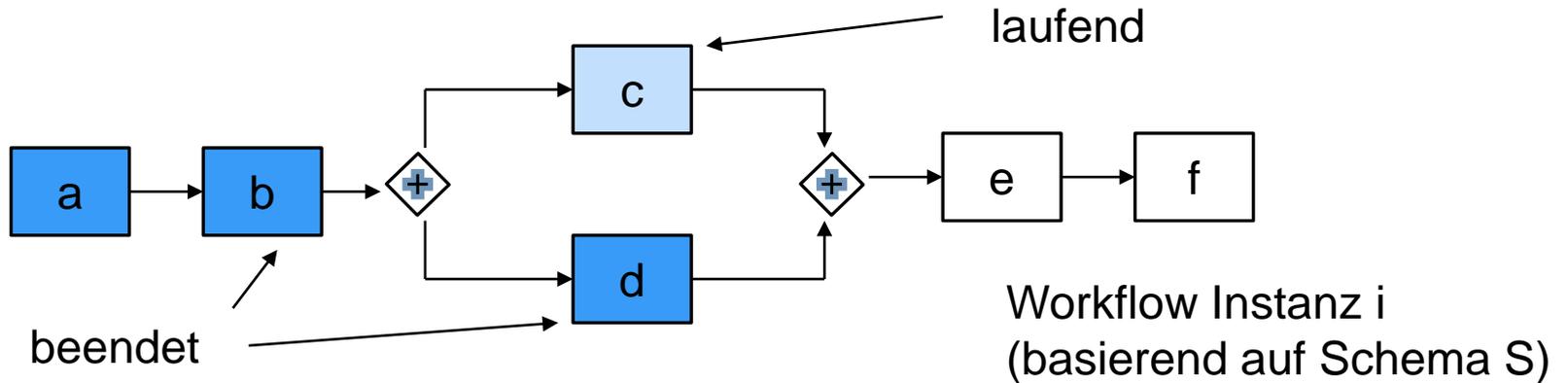
Wdh. Beispiel Migration (II)



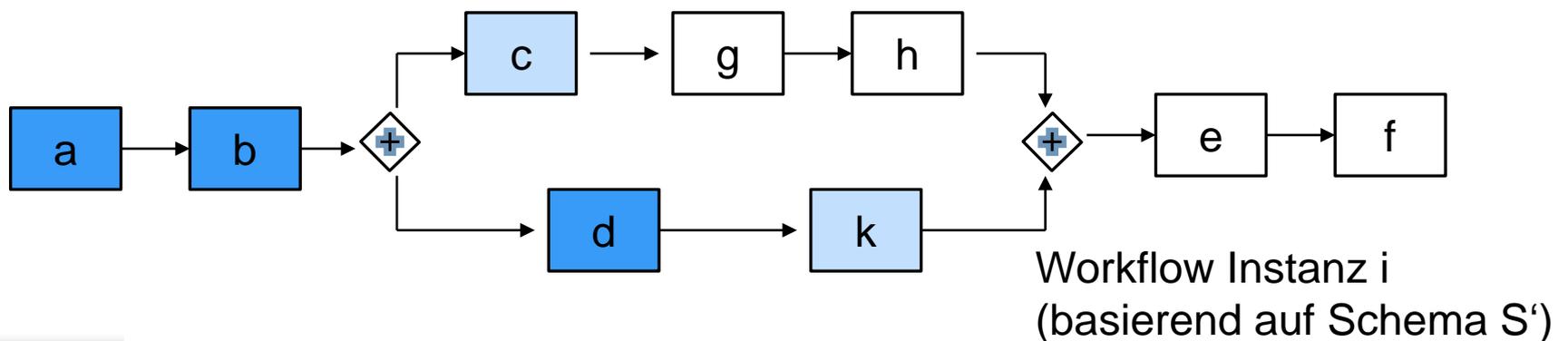
Migration nicht möglich !



Beispiel Migration (III)



Migration möglich !



Diskussion Migration (I)

- ◆ Ursprüngliche Trennung von Build-Time und Run Time aufgehoben
- ◆ Alternativer Begriff: dynamische Adaptionen („dynamic adaptations“)
- ◆ Fragestellungen
 - Wie kann Migration von laufenden Workflow-Instanzen konzeptioniert und implementiert werden?
 - Wie wird Migration kontrolliert, welche Korrektheitskriterien sind möglich? Welche Regeln legen fest, ob eine Workflow-Instanz migriert werden kann?
 - Oft viel komplexere Regeln als Beispiel Vorlesung!
 - Wie ist der Gültigkeitsbereich der Migration? Welche Instanzen sollen migriert werden, welche nicht?
 - Wer hat unter welchen Bedingungen die Erlaubnis, diese Veränderungen durchzuführen?

Diskussion Migration (II)

- ◆ Insbesondere für langdauernde Prozessinstanzen von Bedeutung
- ◆ Funktionalität in kommerziellen Systemen nicht bzw. nur eingeschränkt unterstützt
 - Anwender greifen auf workflow-externe Anwendungen zu (Fehleranfälligkeit z.B. durch Integration von Daten)
 - Negative Auswirkung auf Zufriedenheit und Akzeptanz der Anwender
- ◆ Flexible Workflow-Management Systeme aus dem Forschungsbereich
 - z.B. ADEPT 2, YAWL, Declare

Change Patterns [Weber et. al]

- ◆ Ziel: Erweiterung der „Workflow Patterns“ um Patterns, die Flexibilität in Workflows repräsentieren
- ◆ 17 Patterns
- ◆ Strukturierung
 - Adaption-Pattern (z.B. insert, update, delete von Prozessfragmenten)
 - Patterns für vordefinierte Änderungen
 - späte Auswahl
 - spätes Modellieren von Prozessfragmenten
- ◆ Bewertung von akademischen und kommerziellen Systemen hinsichtlich ihrer Flexibilität anhand der Change Patterns

Strategien zur Unterstützung flexibler WF

schwieriger
flexibler

- 
- 1) Keine Änderung des Workflow-Schemas erforderlich; Passe Ausführungsumgebung und Verarbeitungs-Entitäten an
 - 2) Modifiziere Workflow-Schema für zukünftige Instanzen durch Versionierung
 - 3) Instantiiere ein neues Workflow-Schema (oder ein anderes, aber vordefiniertes Schema)
 - 4) Ändere dynamisch die Workflow-Spezifikation für aktive Workflow-Instanzen

Qualitätskriterien für Änderbarkeit von Workflows

- ◆ Korrektheit
 - Syntaktische Korrektheit
 - Z.B. Knoten, die nicht verbunden sind
 - Semantische Korrektheit
 - Korrektes Abarbeiten eines Falles
- ◆ Scheduling / Ausführbarkeit
- ◆ Zuverlässigkeit

Forschungsfragen und –herausforderungen (1)

- ◆ Modellierungskonzepte zur Unterstützung flexibler Workflows
- ◆ Spezifikation, welche Änderungen Workflows beeinflussen; wie kann man solche Monitore hierfür aufbauen?
- ◆ (Teil-)automatisierung von Änderungen, die durch andere Aspekte als die operationalen und Verhaltensaspekte hervorgerufen werden
- ◆ Semantische Beschreibung von Prozessen und Spezifikation der Komposition von Diensten und Prozessen
- ◆ Korrektheitsanalyse des Workflows (Schema und Instanz) nach der Änderung

Forschungsfragen und –herausforderungen (2)

- ◆ Korrekte Prozesstransformationen und ihre Auswirkungen auf Ressourcen-Zustände und Datenkonsistenz
- ◆ Wiederverwendung von vordefinierten Workflow-Komponenten und Vorhalten von Änderungen für zukünftige Nutzung, Versionierung
- ◆ Laufzeitunterstützung für dynamische Workflows
- ◆ Benutzerunterstützung für die Änderung von Workflows

Exemplarische Fragen zu Kapitel 10

- ◆ Beschreiben Sie CPI und BPR und stellen Sie die Unterschiede heraus.
- ◆ Beschreiben Sie die 4 Bewertungskriterien für BPR.
- ◆ Auf welche Gegebenheiten sollte ein WfMS flexibel agieren können?
- ◆ Geben Sie für verschiedene Aspekte von WfMS Beispiele hinsichtlich der Flexibilität und diskutieren Sie (kurz) Umsetzungsmöglichkeiten.
- ◆ Klassifizieren Sie Flexibilität in Workflow-Systemen (z.B. nach Schonendorf) und geben Sie jeweils ein Beispiel an.
- ◆ Was versteht man unter Migration? Geben Sie ein Beispiel für eine mögliche Migration an.
- ◆ Was versteht man unter Ad-hoc Prozessen? Welche Realisierungsmöglichkeiten gibt es?

Ergänzende Literatur zu Kapitel 10

- ◆ Helen Schonenberg, Ronny Mans, Nick Russell, Nataliya Mulyar, and Wil van der Aalst: Process Flexibility: a Survey of Contemporary Approaches. In: CIAO! / EOMAS 2008, pages 16-30.
- ◆ Barbara Weber, Stefanie Rinderle, and Manfred Reichert: Change Patterns and Change Support Features in Process-Aware Information Systems, In: CAiSE 2007, pages 574-588.