

Informatik II: Modellierung  
Prof. Dr. Martin Glinz

Kapitel 9

# Modellierung von Arbeitsprozessen



Universität Zürich  
Institut für Informatik

---

# Inhalt

---

9.1 Grundlagen

9.2 Ereignisgesteuerte Prozessketten (EPK)

9.3 Andere Sprachen



# 9.1 Grundlagen

---

- Informatik **unterstützt** oder **automatisiert Arbeitsprozesse**
  - von Menschen ausgeführte Prozesse
  - maschinell ausgeführte Prozesse

⇒ Die Modellierung von Arbeitsprozessen ist wichtig

**Arbeitsprozess (work process)** – eine geordnete Folge von Arbeitsschritten zur Erreichung eines geplanten Arbeitsergebnisses.

- Von besonderer Bedeutung sind **Geschäftsprozesse**, insbesondere in der Wirtschaftsinformatik

**Geschäftsprozess (business process)** – Funktions- und stellenübergreifender Arbeitsprozess in einem Unternehmen, welcher direkt oder indirekt zur Erzeugung einer Leistung für einen Kunden oder den Markt dient.

# Charakteristika von Arbeitsprozessen

---

- Elemente
  - Arbeitsschritte
  - Ereignisse, welche den Ablauf steuern
  - Beteiligte Personen / Stellen / Maschinen
  - Verwendete / erzeugte Materialien
- Eigenschaften
  - Auf ein Ziel gerichtet
  - Transformiert Eingaben (Daten, Materialien, Energie) in Ausgaben
  - Durch Ereignisse angestoßen und gesteuert
  - Durch Aktionsträger (Personen, Organisationen, Maschinen) ausgeführt

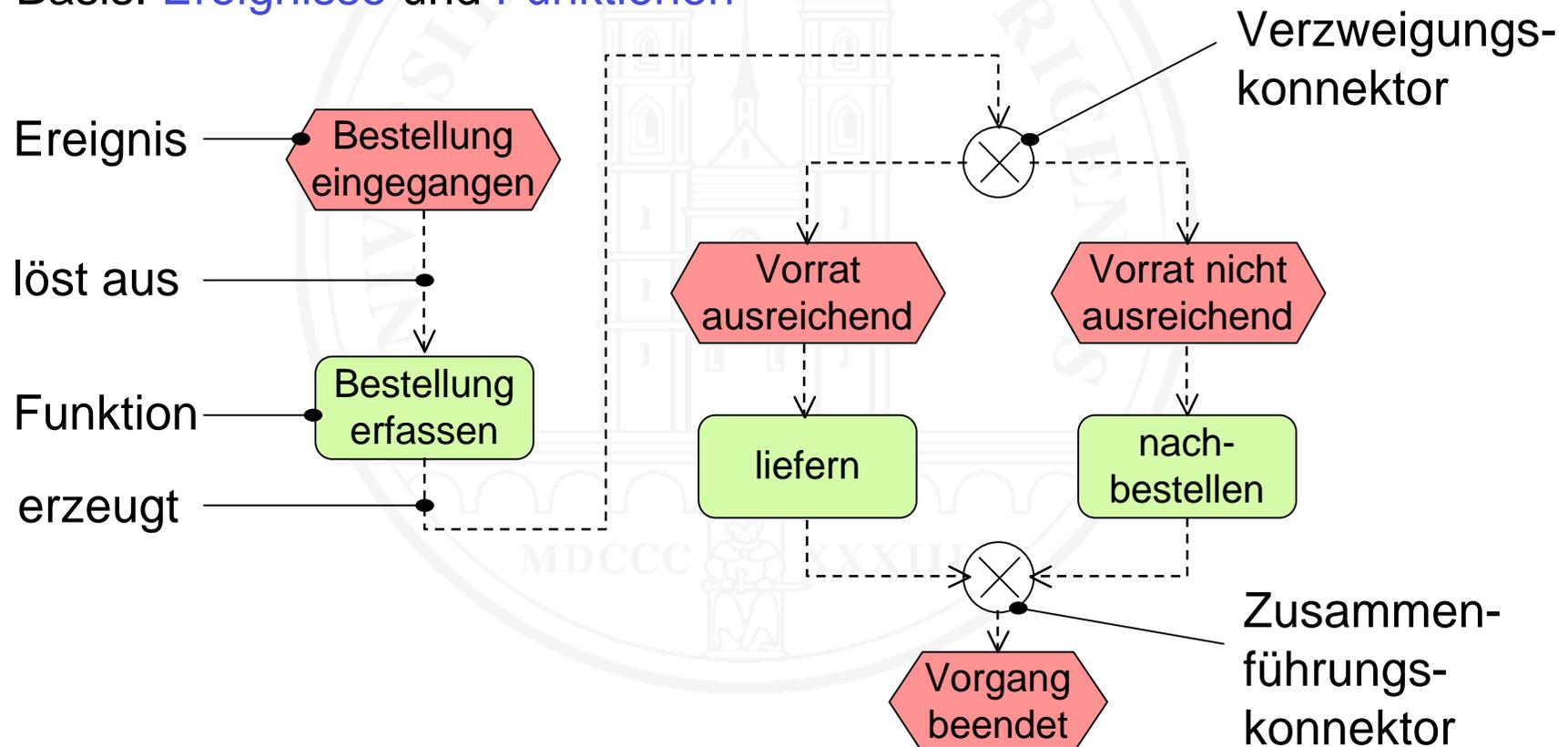
# Arbeitsprozessmodelle

---

- **Einfache Arbeitsprozessmodelle** modellieren nur eine Folge von Arbeitsschritten und Ereignissen
- **Erweiterte Arbeitsprozessmodelle** modellieren zusätzlich auch Beteiligte und Materialien
- **Sprachen** für die Modellierung von Arbeitsprozessen
  - **Ereignisgesteuerte Prozessketten (EPK)**
  - **UML-Aktivitätsdiagramme** (vgl. Kapitel 5)
  - **Sprachen zur Verhaltensmodellierung** (zum Beispiel Statecharts oder Petrinetze, vgl. Kapitel 7)
  - **Programmablaufpläne** (veraltet → nicht mehr verwenden, vgl. Kapitel 5)

## 9.2 Ereignisgesteuerte Prozessketten (EPK)

- Von Scheer und Mitarbeitern entwickelte Sprache zur **Modellierung von Geschäftsprozessen**
- Basis: **Ereignisse** und **Funktionen**



# Ereignisgesteuerte Prozessketten (EPK) – 2

---

- Abgeleitet von **Petrinetzen**: auf ein Ereignis folgt immer eine Funktion und umgekehrt
- Im deutschsprachigen Raum weit **verbreitet**
- Wird insbesondere beim Einsatz von **SAP** zur Modellierung der Geschäftsprozesse eines Unternehmens eingesetzt
- **Erweiterung** mit **Informationsobjekten** und **Organisationseinheiten** möglich (ECPK)

# Ereignisgesteuerte Prozessketten: Notation



**Ereignis:** der Zustand eines Geschäftsprozesses verändert sich



**Funktion:** Transformation von Geschäftsobjekten



**Steuerfluss:** Kausal-sachlogische Abhängigkeiten zwischen Ereignissen und Funktionen



**Konnektor:** Verzweigung oder Zusammenführung von Steuerflüssen

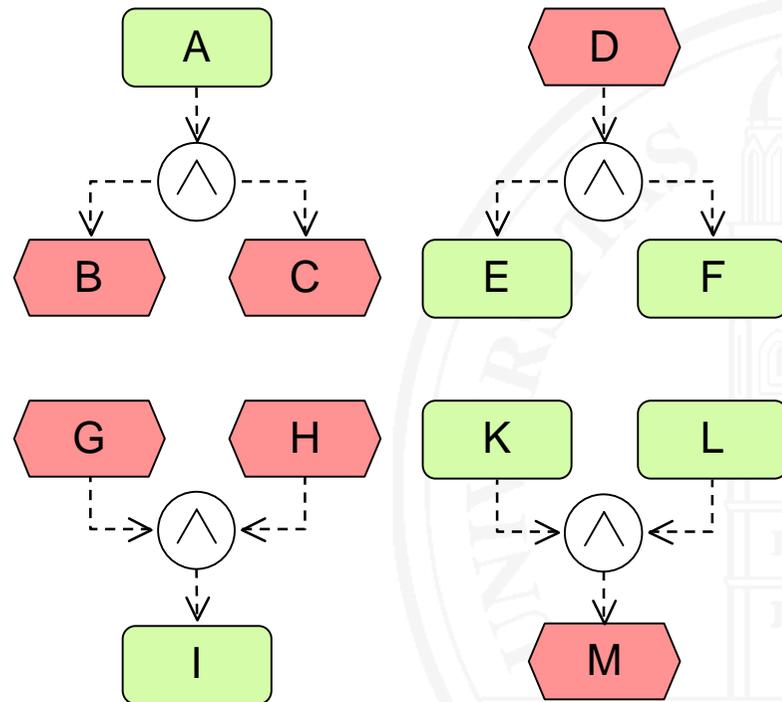


**Prozesswegweiser:** dient zur Gliederung in Teilmodelle

Notationsvarianten



# Konnektoren



## UND-Verzweigung (Split)

- A erzeugt die Ereignisse B **und** C
- D löst die Funktionen E **und** F aus

## UND-Zusammenführung (Join)

- I wird ausgelöst, wenn G **und** H eingetreten sind
- M wird erzeugt, wenn K **und** L abgeschlossen sind

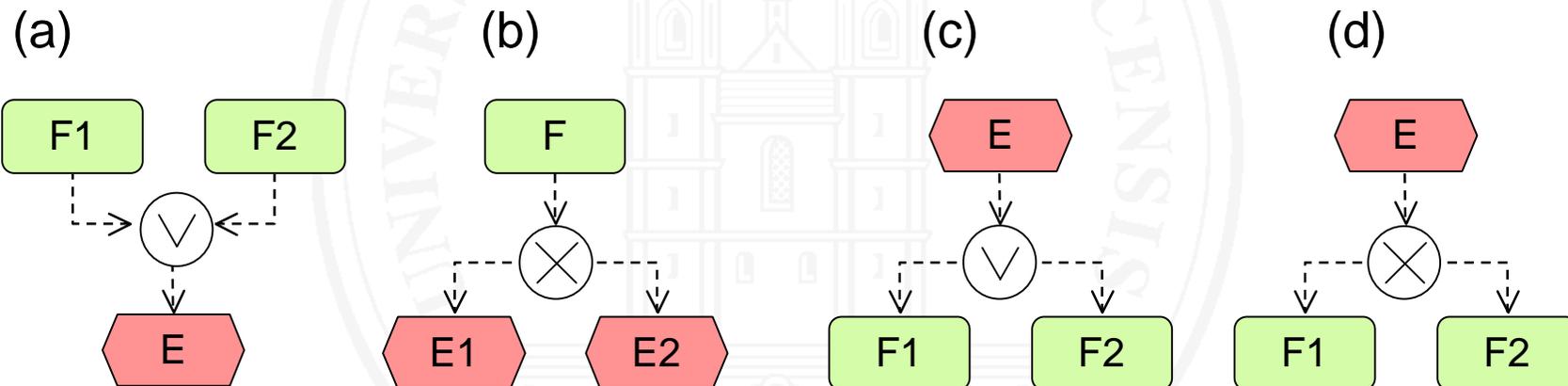
Analog:

- **ODER-Verzweigung / Zusammenführung** (V)
- **Exklusiv-ODER (XOR)-Verzweigung / Zusammenführung** (X)



# Aufgabe 9.1

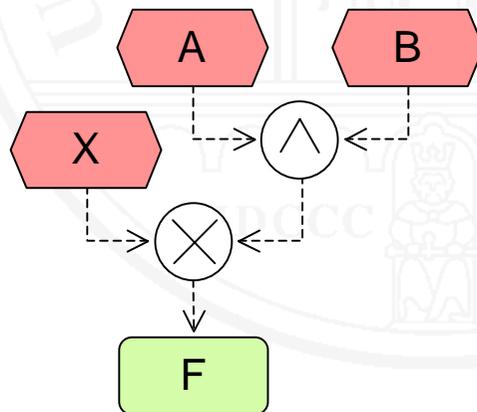
- a) Interpretieren Sie die EPK-Fragmente (a) und (b)
- b) Nach Keller, Nüttgens und Scheer (1992) sind die EPK-Fragmente (c) und (d) nicht erlaubt. Begründen Sie, warum.



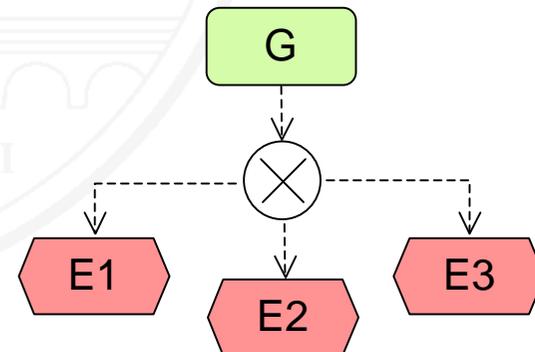
# Konnektoren – 2

- Komplexe, inhomogene Bedingungen können durch eine **Kaskade von Konnektoren** modelliert werden
- **Mehrere homogene Bedingungen** können wahlweise auf einem Konnektor zusammengeführt oder kaskadiert werden
- **Beispiel:** A. Es gibt drei Ereignisse A, B, X. Für die Ausführung der Funktion F müssen entweder A und B gemeinsam oder X eingetreten sein. B. Nach der Ausführung von G gilt entweder E1 oder E2 oder E3

A. Kaskadierte Konnektoren:

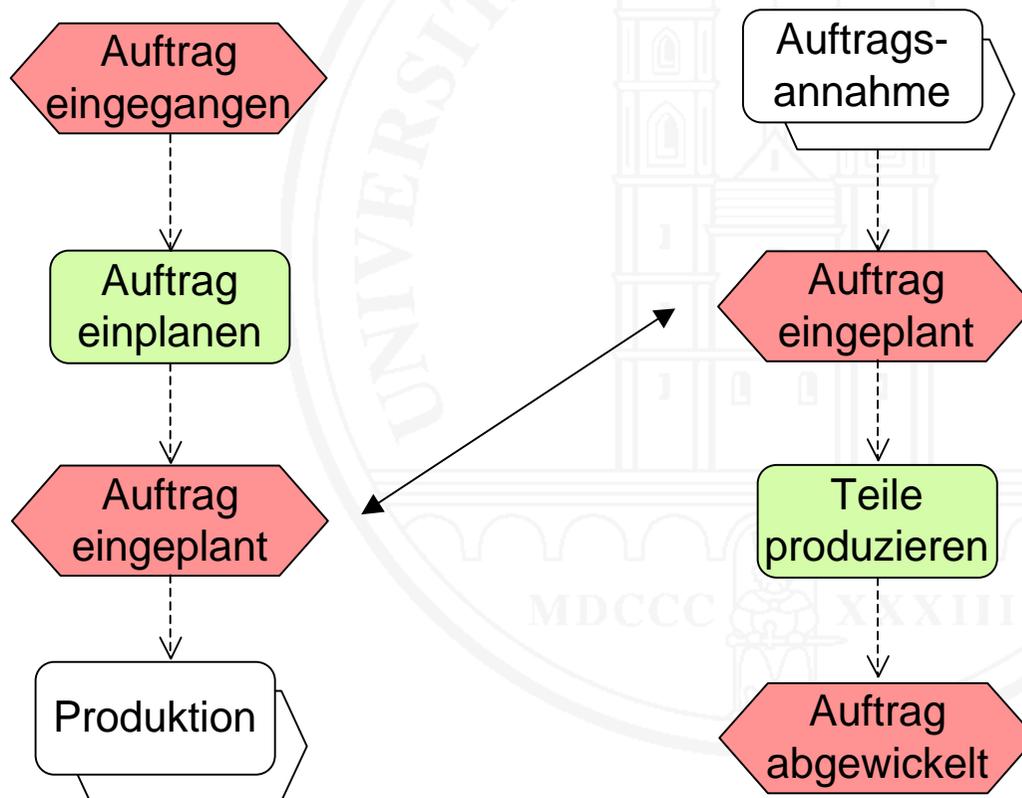


B. Mehrfach-Konnektor:



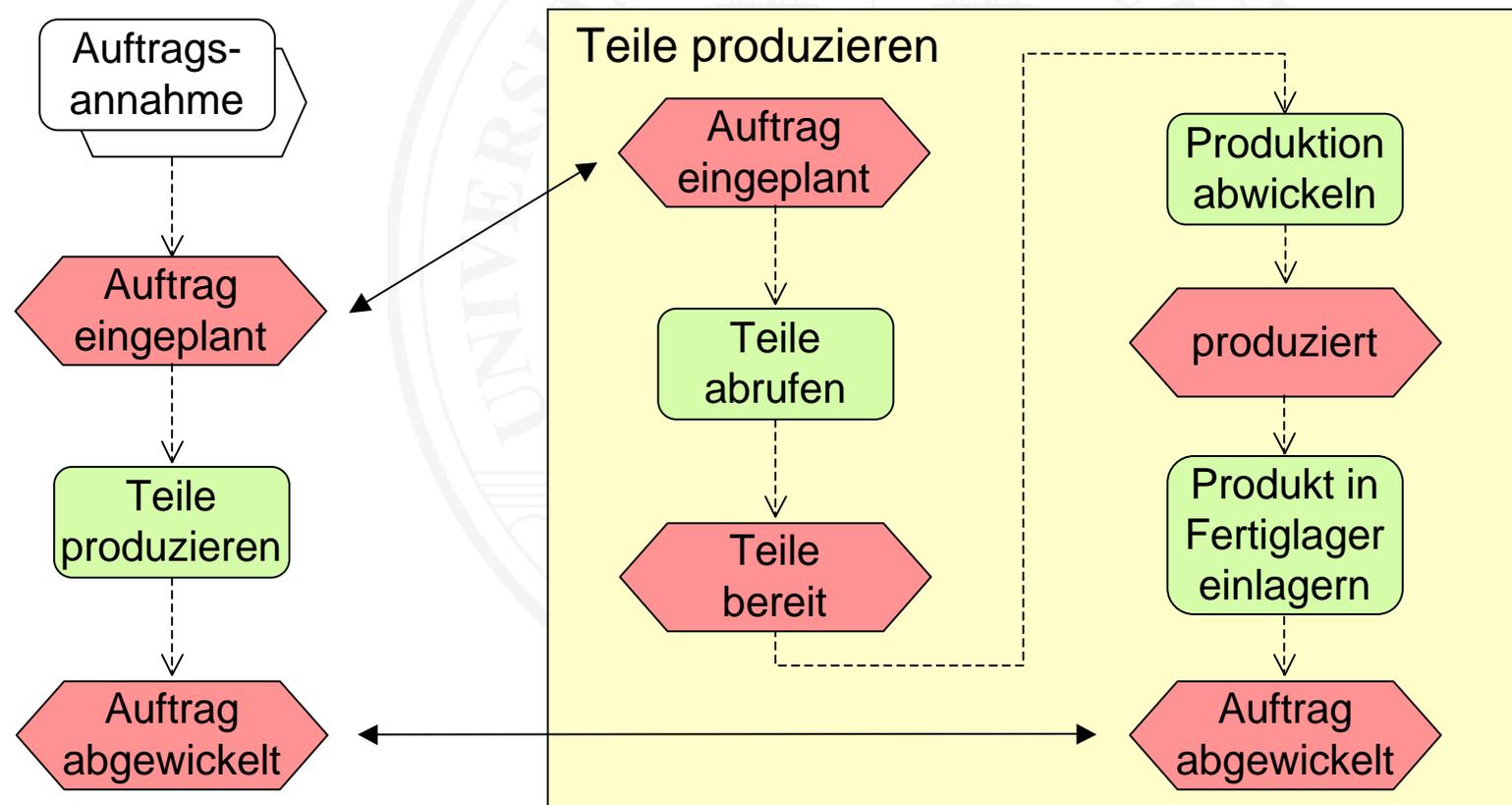
# Prozesszusammenhänge – 1: Prozesswegweiser

- Verbindung gleichgeordneter Prozesse miteinander
- Die Zuordnung erfolgt über gemeinsame Ereignisse



# Prozesszusammenhänge – 2: Subprozesse

- Einbinden untergeordneter Prozesse
- Die Zuordnung erfolgt wiederum über gemeinsame Ereignisse



# Modellbildung mit EPKs

---

- Wie **beginne** ich?
  - Immer mit einem **Ereignis** → Feststellen, welches Ereignis den Prozess anstößt und dieses Ereignis modellieren
- Ich habe ein **Ereignis** modelliert – wie weiter?  
Nach Eintreten dieses Ereignisses ...
  - ... ist **genau eine Transformation** auszuführen → Diese Transformation als Funktion modellieren und direkt an das Ereignis anschließen
  - ... sind **mehrere Transformationen auszuführen, die voneinander unabhängig** ausgeführt werden können → Mehrere Funktionen modellieren und mit UND-Konnektor an auslösendes Ereignis anschließen

# Modellbildung – 2

---

- Ich habe eine **Funktion** modelliert – wie weiter?
  - **Genau ein Ereignis** markiert den Abschluss **genau dieser einen Funktion** → dieses Ereignis modellieren und an die Funktion anschließen
  - **Genau ein Ereignis** markiert den **gemeinsamen Abschluss mehrerer Funktionen** → diese Funktionen mit einem UND-Konnektor an das Ereignis anschließen
  - **Genau ein Ereignis** markiert den Abschluss **einer beliebigen Funktion aus einer Gruppe** von Funktionen → diese Funktionen mit einem ODER-Konnektor an das Ereignis anschließen
  - Die Funktion wird durch **genau eines von mehreren möglichen Ereignissen** abgeschlossen → Ereignisse alle modellieren und die Funktion mit einem Exklusiv-ODER-Konnektor an diese Ereignisse anschließen

# Modellbildung – 3

---

- Die Funktion wird durch **mehrere Ereignisse, die einzeln oder gemeinsam auftreten können**, abgeschlossen → alle Ereignisse modellieren und die Funktion mit einem ODER-Konnektor an diese Ereignisse anschließen
- Auf **mehrere Ereignisse folgt eine gemeinsame Funktion** – wie weiter?
  - Die Ereignisse über einen Konnektor an die Funktion anschließen
    - Die Ereignisse **schließen sich gegenseitig aus**: Exklusiv-ODER
    - Die Ereignisse müssen **alle eingetreten** sein: UND
    - **Mindestens eines** der Ereignisse muss eingetreten sein: ODER
    - Es liegt eine **Mischform** vor: Konnektoren passend kaskadieren
- Mein **Prozess wird zu groß** – was tun? →
  - **Subprozesse** bilden
  - In mehrere Prozesse **aufteilen** und über **Wegweiser** verbinden

## Aufgabe 9.2

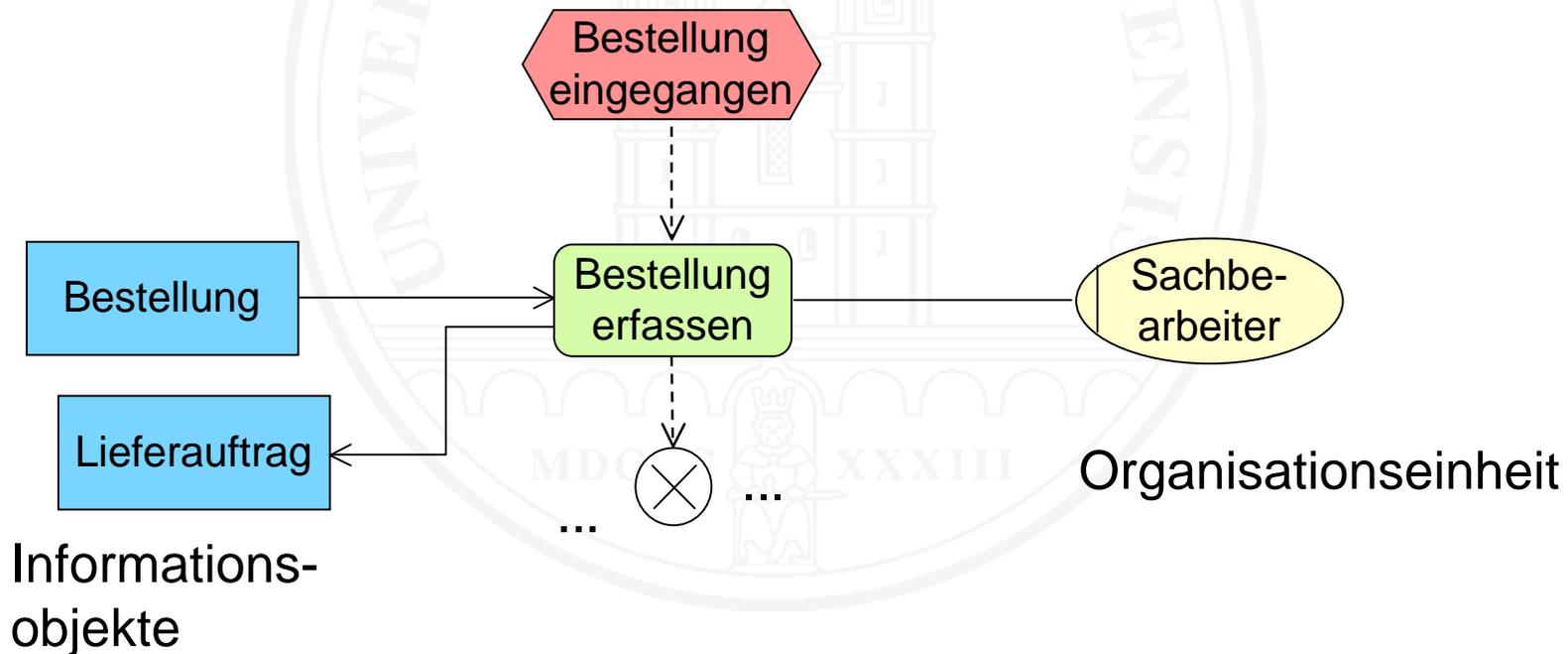
In einer Regionalbank erfolgt der Prozess der Kreditvergabe wie folgt: Zunächst wird der Antrag erfasst. Dann prüft der Kreditsachbearbeiter das Kreditrisiko. Ist das Risiko hoch, lehnt er den Antrag ab. Ist das Risiko gering, bewilligt er den Kredit und erstellt den Kreditvertrag. Bei mittlerem Risiko wird der Antrag durch die Filialleiterin begutachtet. Entscheidet sie positiv, wird der Antrag bewilligt, anderenfalls wird er abgelehnt.

Zum Schluss unterschreibt die Filialleiterin den Bescheid an den Antragsteller und der Kreditsachbearbeiter legt die Unterlagen in den Kreditakten ab.

Modellieren Sie diesen Geschäftsprozess mit einem EPK.

# Erweiterte EPKs (EEPK)

- Zu jeder Funktion wird zusätzlich erfasst
  - die beteiligten **Informationsobjekte**
  - die ausführende **Einheit** in der **Organisation**
- Die Darstellung erfolgt in der Regel **tabellarisch** (siehe SAP Beispiel)



# Erweiterte EPK: Notation

---

Filialeiterin

Organisationseinheit

Lieferauftrag

Informationsobjekt

Teilweise wird noch unterschieden zwischen:

Lieferposition

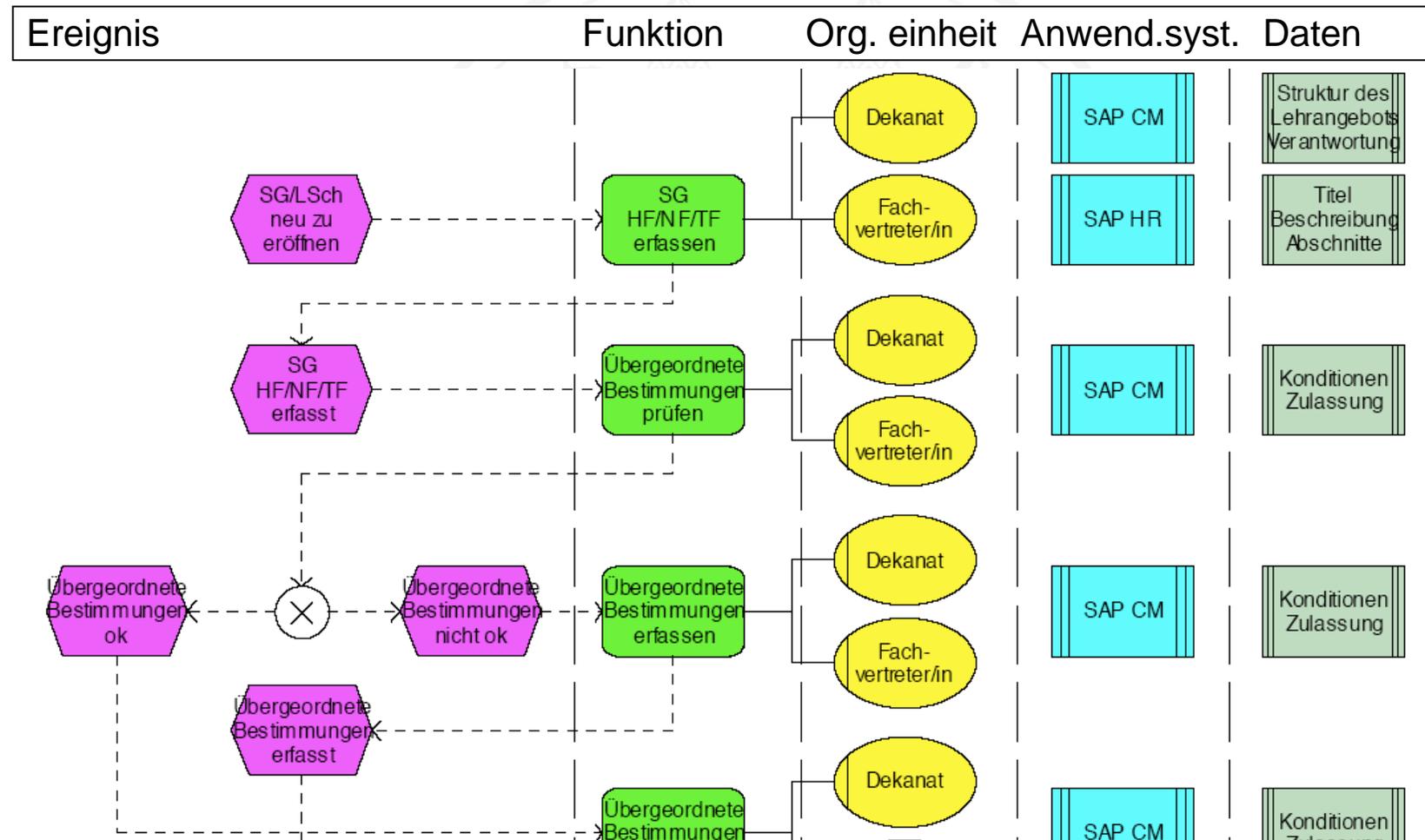
Datenobjekt (im Datenmodell)

Lieferung

Informationsobjekt (auf Geschäftsebene)

# Beispiel: SAP CM Universität Zürich

## Erfassung Studiengang/Studienschwerpunkt



## 9.3 Andere Sprachen

---

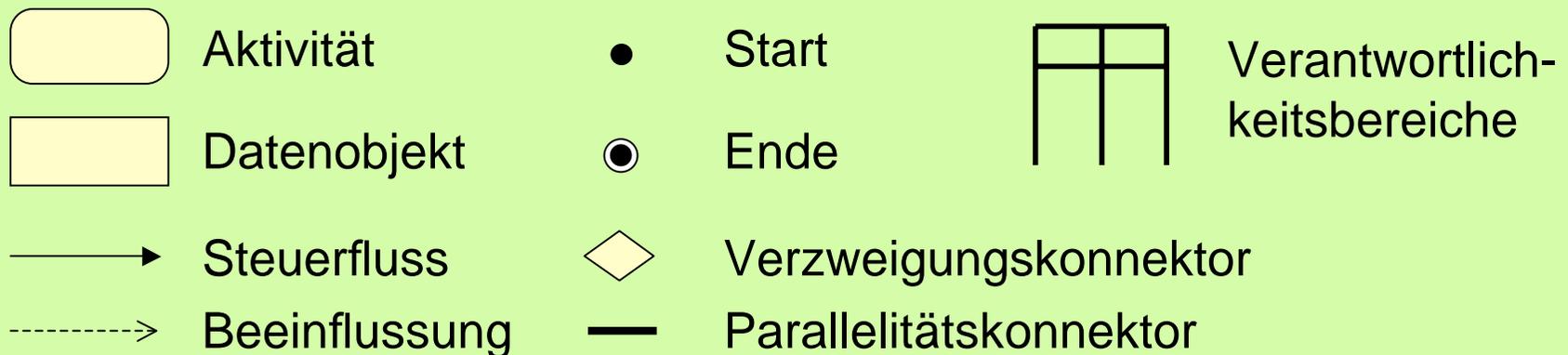
- Jede Modellierungssprache, welche durch **äußere Ereignisse gesteuerte Abläufe** modellieren kann, **eignet sich** grundsätzlich zur Modellierung von Arbeits- bzw. Geschäftsprozessen.
- **Gebräuchlich** sind
  - UML-Aktivitätsdiagramme
- **Möglich** sind ferner
  - Statecharts
  - Petrinetze
  - Programmablaufpläne

# UML-Aktivitätsdiagramme

---

- Funktionen werden als **Aktivitäten** modelliert
- Ereignisse werden nur bei **Fallunterscheidungen** explizit modelliert
- **Organisationseinheiten** und **Informationsobjekte** sind **modellierbar**
- **Verzweigungskonnektor** entspricht dem **XOR-Konnektor** in EPKs
- **Parallelitätskonnektor** entspricht dem **UND-Konnektor** in EPKs

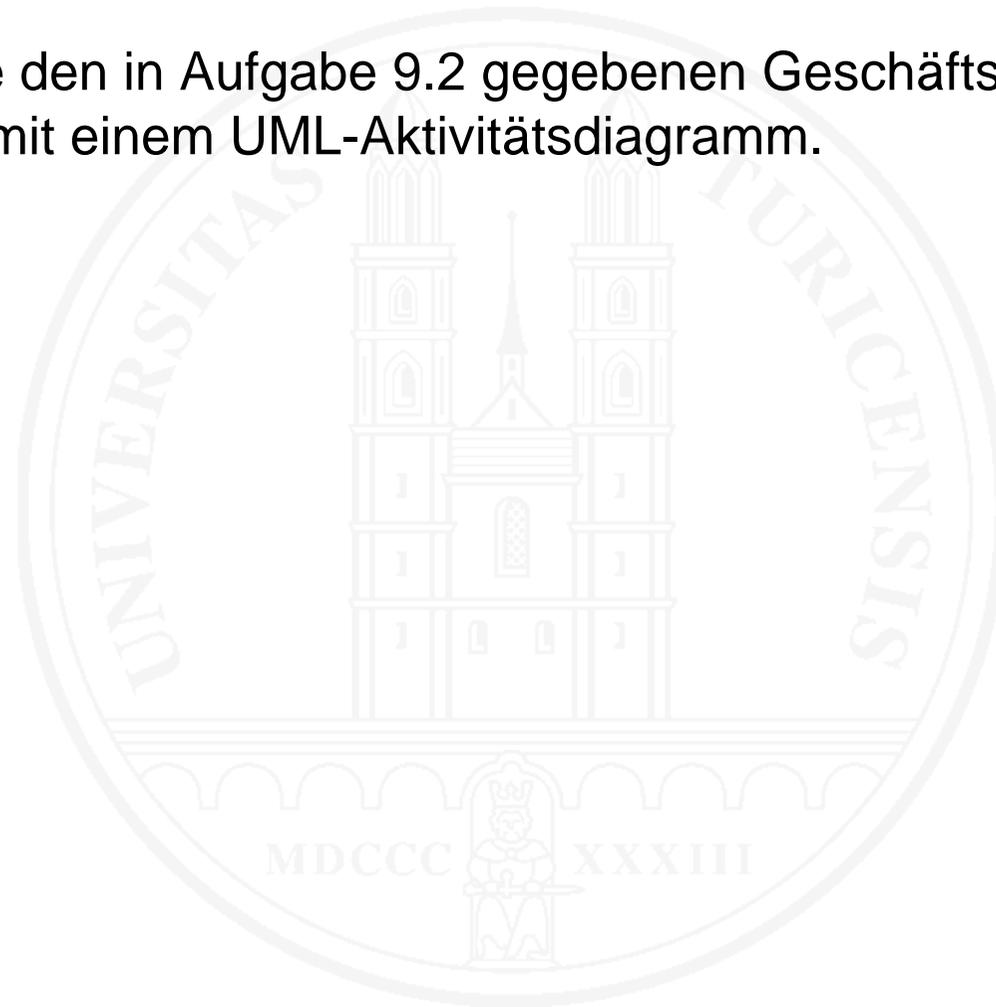
Notation (siehe Kapitel 5):





## Aufgabe 9.3

Modellieren Sie den in Aufgabe 9.2 gegebenen Geschäftsprozess für die Kreditvergabe mit einem UML-Aktivitätsdiagramm.



# Petrinetze

---

- **Naheliegend:** Modellierung von Arbeitsprozessen als **Prädikat-Transitionsnetze** (vgl. Kapitel 7)
  - Arbeitsschritte → **Stellen**
  - auslösende oder terminierende Ereignisse → **Transitionen**
  - Bedingungen → **Prädikate auf Transitionen**
- **Alternativ** auch möglich:
  - Funktionen (aktiv, transformierend) → **Transitionen**
  - Situationen, eingetretene Ereignisse (passiv, speichernd) → **Stellen**
  - In dieser Form zur Semantikdefinition von EPK verwendet
- **Nicht modellierbar**
  - Organisationseinheiten
  - Verwendete Daten

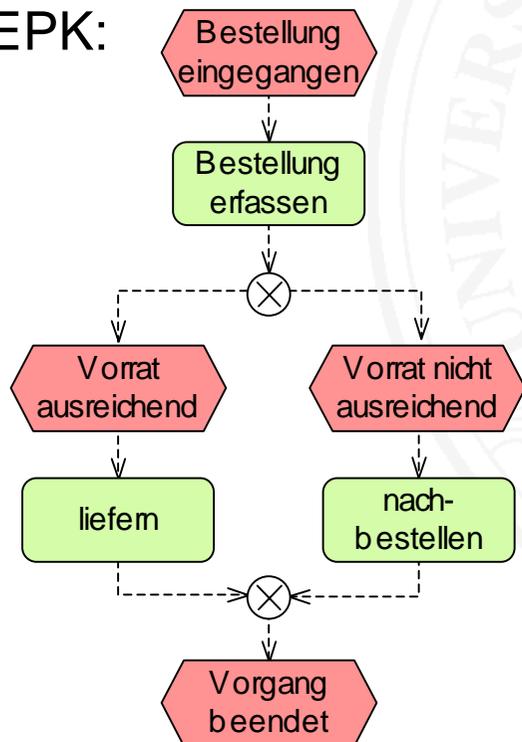
# Petrinetze und Ereignisgesteuerte Prozessketten

○ Definition der **Semantik** von EPK durch **Stellen-Transitionsnetze**:

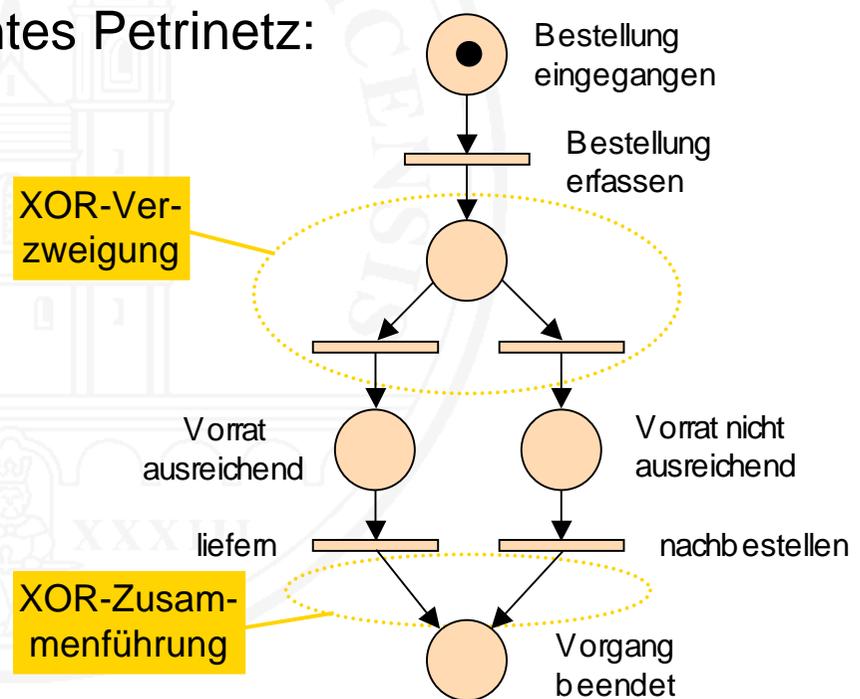
- Funktionen in EPK → **Transitionen**
- Ereignisse in EPK → **Stellen**

[van der Aalst 1999]

EPK:

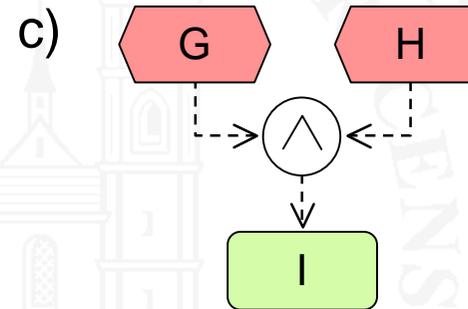
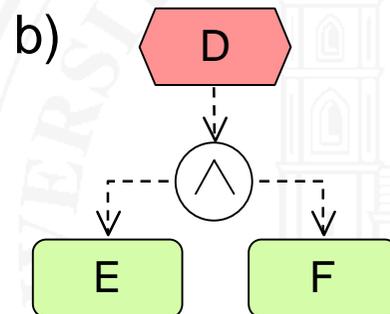
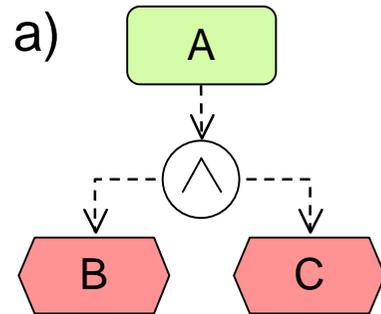


Äquivalentes Petrinetz:



# Aufgabe 9.4

Bilden Sie die folgenden Konnektoren von ereignisgesteuerten Prozessketten auf entsprechende Petrinetz-Konstrukte ab.



# Literatur

---

Keller, G., M. Nüttgens, A.-W. Scheer (1992). Semantische Prozessmodellierung auf der Grundlage „Ereignisgesteuerter Prozessketten (EPK)“. In: A.-W. Scheer (Hrsg.): *Veröffentlichungen des Instituts für Wirtschaftsinformatik*, Heft 89, Saarbrücken.

OMG (2007). *Unified Modeling Language: Superstructure*, version 2.1.1. OMG document formal/2007-02-05

Scheer, A.-W. (2002). *ARIS – vom Geschäftsprozess zum Anwendungssystem*, 4. Auflage, Berlin: Springer.

van der Aalst, W.M.P. (1999). Formalization and Verification of Event-driven Process Chains. *Information and Software Technology* 41(10): 639-650.