

# Aufgaben der Anforderungsanalyse

## Anforderungen...

- ... erheben
- ... konsistent zusammenstellen, (priorisieren)
- ... dokumentieren
- ... überprüfen

*Elicitation*



*Analysis*

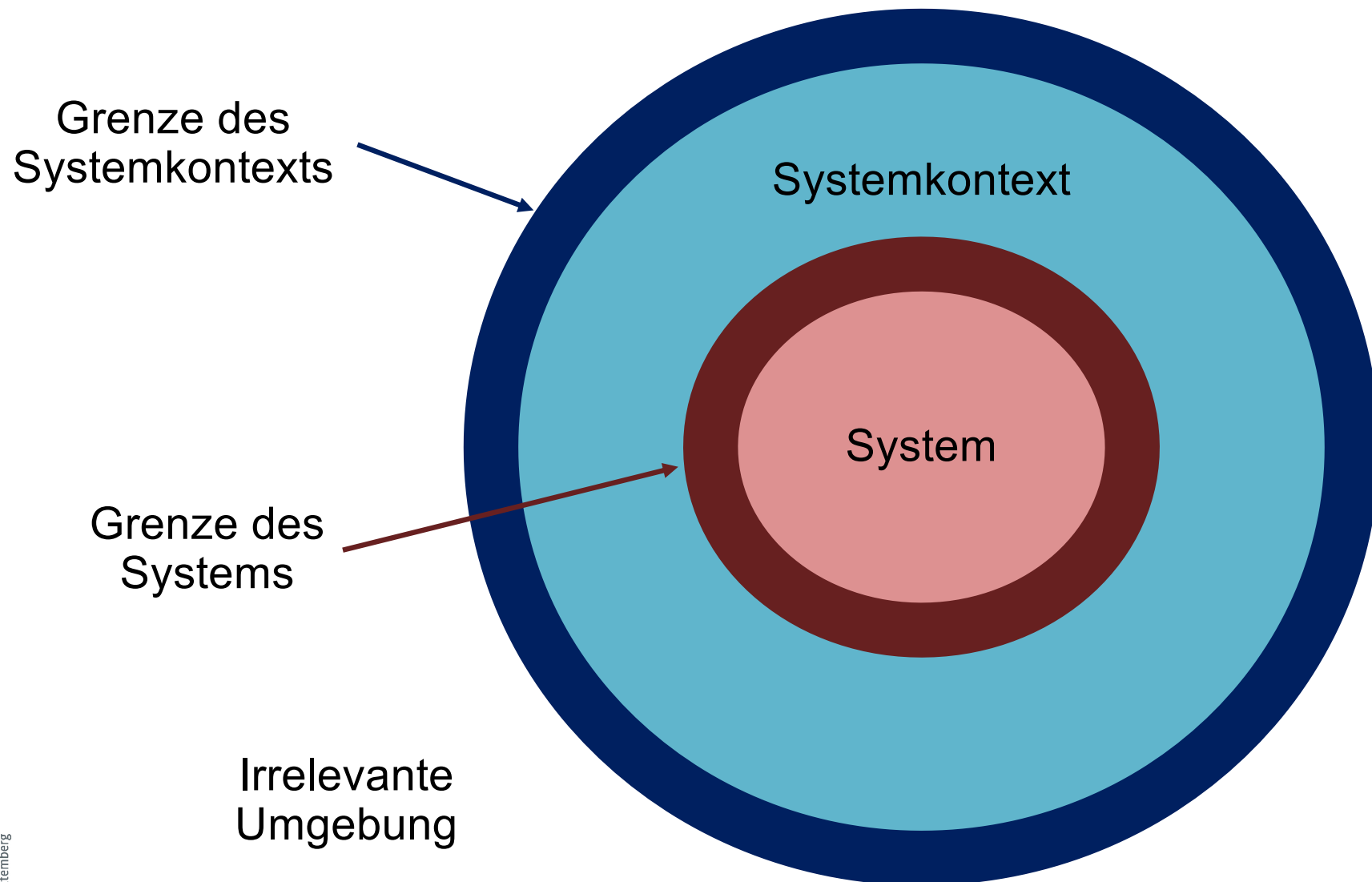


*Specification*

*Validation*

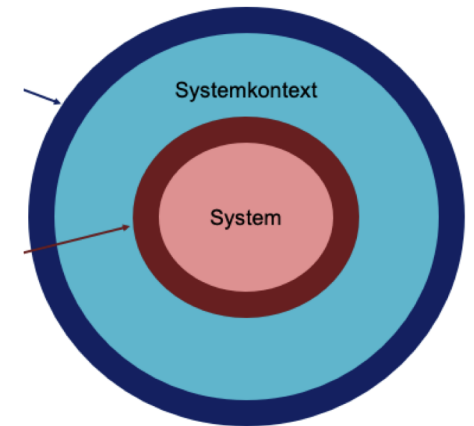
- Motivation
- Gliederung des Themas, Begriffe
- Erheben von Anforderungen
- Analysieren und Dokumentieren von Anforderungen
  - Systemkontext
    - ▶ Use Case-Diagramme
  - Aktivitätsdiagramme
  - Use Case-Beschreibungen (Tabellen)
  - User Stories, User Story Mapping
    - ▶ Priorisierung
  - Snow Cards (für Qualitätsanforderungen / NFR)
- Überprüfen von Anforderungen

# Der Systemkontext



# Systemkontext - Inhalt

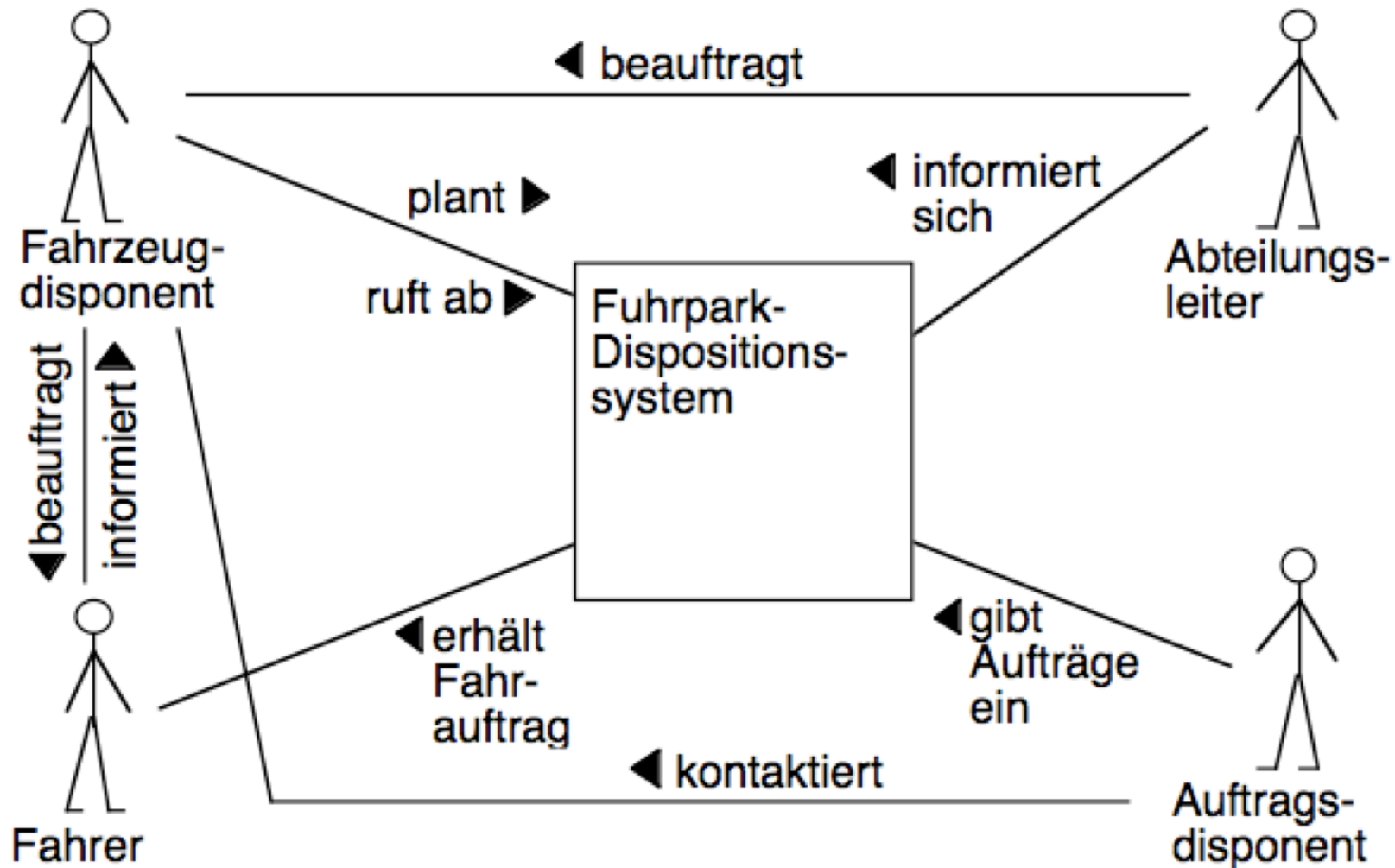
- Der Systemkontext umgibt das (zu entwickelnde) System
- Der Systemkontext besteht aus
  - Geschäftsprozessen, Technischen Prozessen
  - Personen, Organisationsstrukturen
  - IT-Infrastruktur (z.B. Datenbanken, Fremdsysteme)
  - Interaktion
- „Grauzone“ der Systemgrenze/Systemkontextgrenze
  - Die Grenzen sind zunächst oft nicht klar
  - Im Verlauf des RE wird klar (muss klar werden!),
    - ▶ welche Funktionen im SW-System realisiert werden sollen (und welche nicht)
    - ▶ welche Objekte zum Systemkontext gehören



# Bedeutung des Systemkontexts

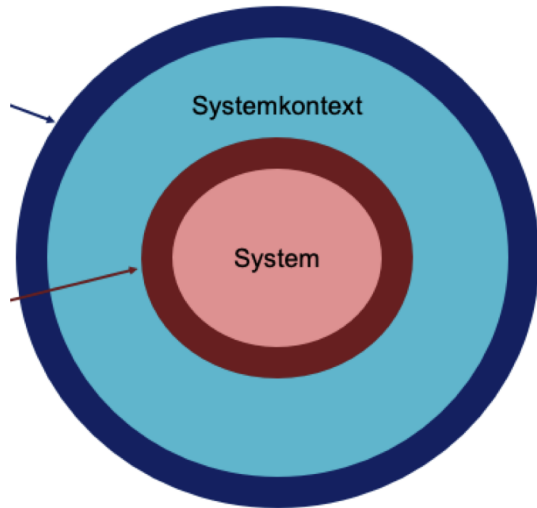
- Anforderungen werden für einen bestimmten Kontext definiert
- Erst die Kenntnis des Kontexts macht die Anforderungen interpretierbar und bestimmt deren Umfang
- Die Anforderungen müssen aus dem Systemkontext erhoben werden
  - Stakeholder haben bestimmte Anforderungen / Erwartungen
  - Geschäftsprozesse implizieren gewisse Anforderungen
  - Richtlinien der Organisation werden in Anforderungen übersetzt
  - etc.
- Je besser das Systemkontext verstanden wird, desto geringer ist das Risiko falscher oder unvollständiger Anforderungen

# Beispiel

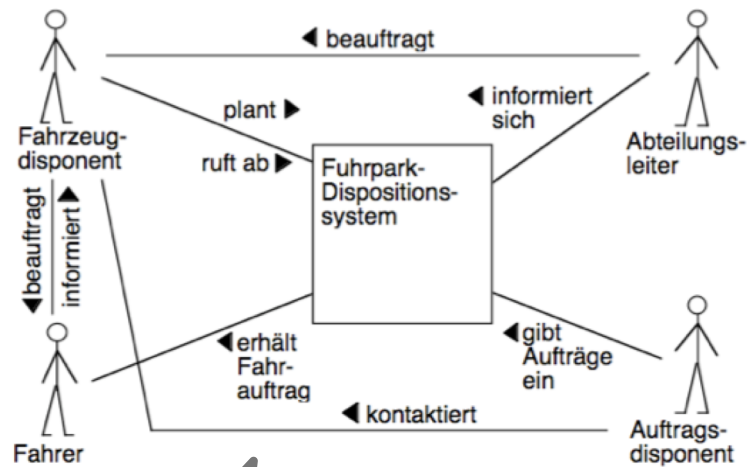


Glinz, Vorlesung RE, Uni Zürich

# Anmerkung: Es gibt nicht *die* Notation für Kontextdiagramme

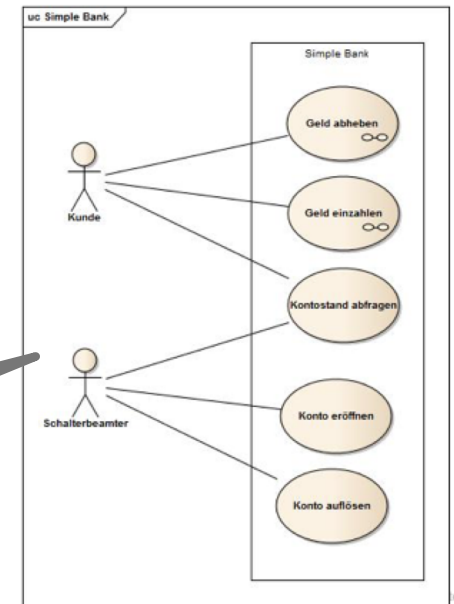


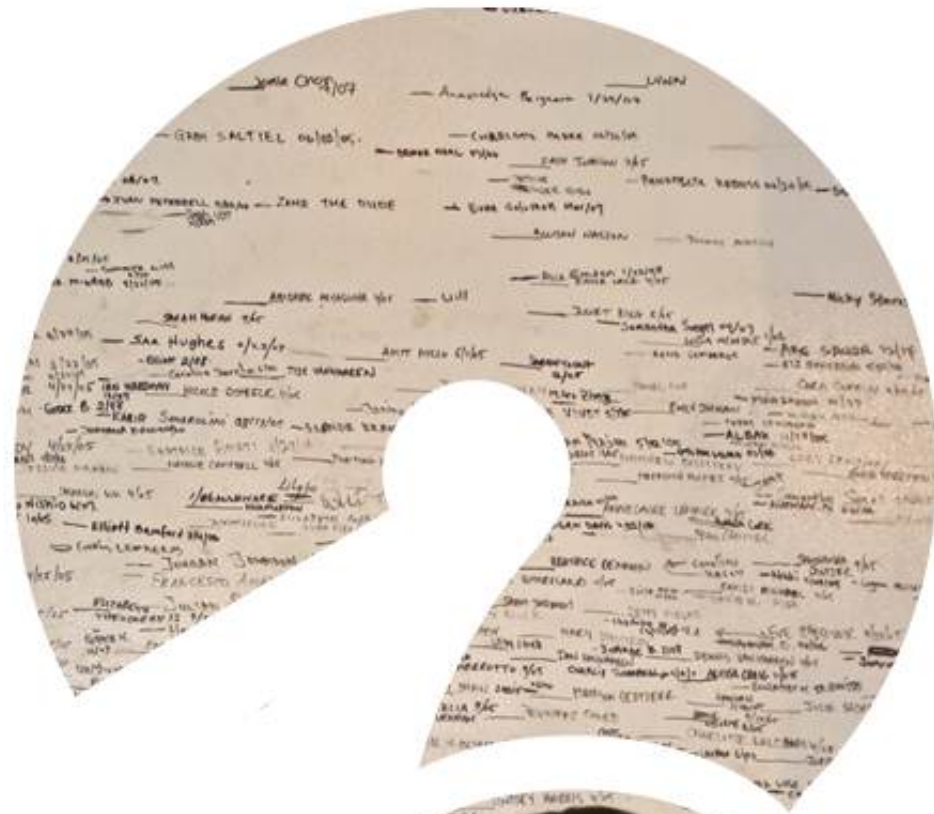
Nur für die didaktische Illustration hier in der Vorlesung



(Halbwegs) freie Notation nach Glinz

UML-Use Case-Diagramm, etwas „missbraucht“





F R A G E N

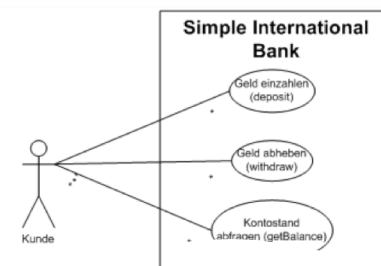


photography: woodleywonderworks  
<http://www.flickr.com/photos/wwworks/2350106729>  
art work: Peter Kaiser

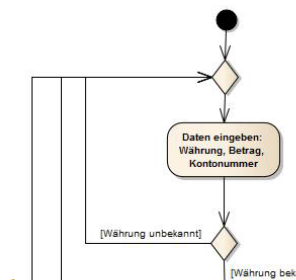
# Use Case

- *Deutsch: Anwendungsfall*
- Zur strukturierten Dokumentation existierender oder geplanter Systeme durch *Interaktionsfolgen*

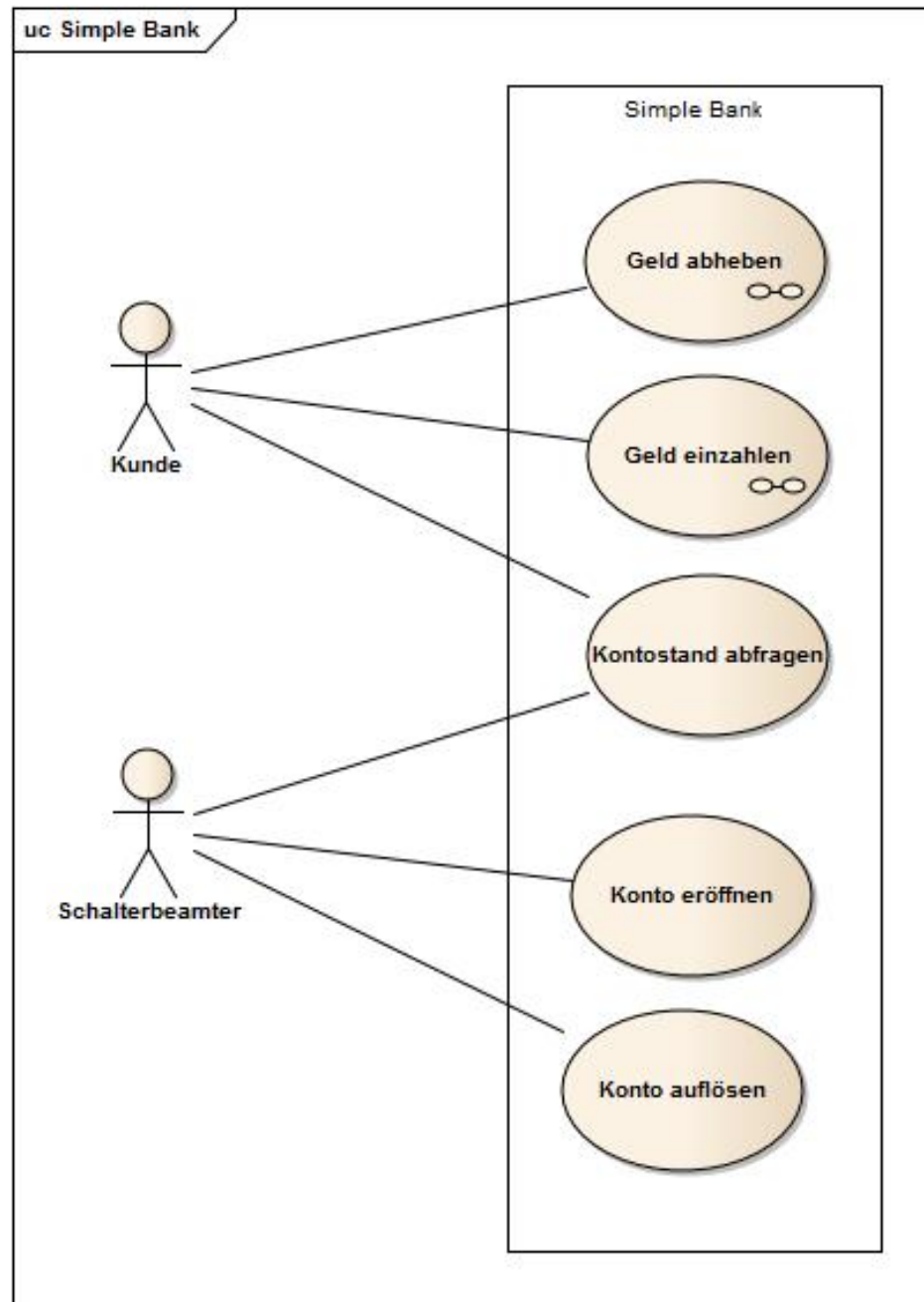
- Besteht aus
  - Use Case-Diagramm*und*
  - Use Case-Spezifikationen*oder*
  - Aktivitätsdiagrammen



Anwendungsfallname	HebeGeldAb
Akteure	Kunde
Ziel	Geld abheben
Hauptablauf	1. Der Kunde meldet sich mit seiner Kontonummer/PIN am Geldautomaten an. 2. Das System überprüft die Identität. 3. Der Kunde gibt einen Betrag ein und wählt eine Währung aus. 4. Das System überprüft, ob die Währung vorrätig und der Betrag auf dem Kontostand verfügbar ist, und es erteilt den Ausgabebetrag.

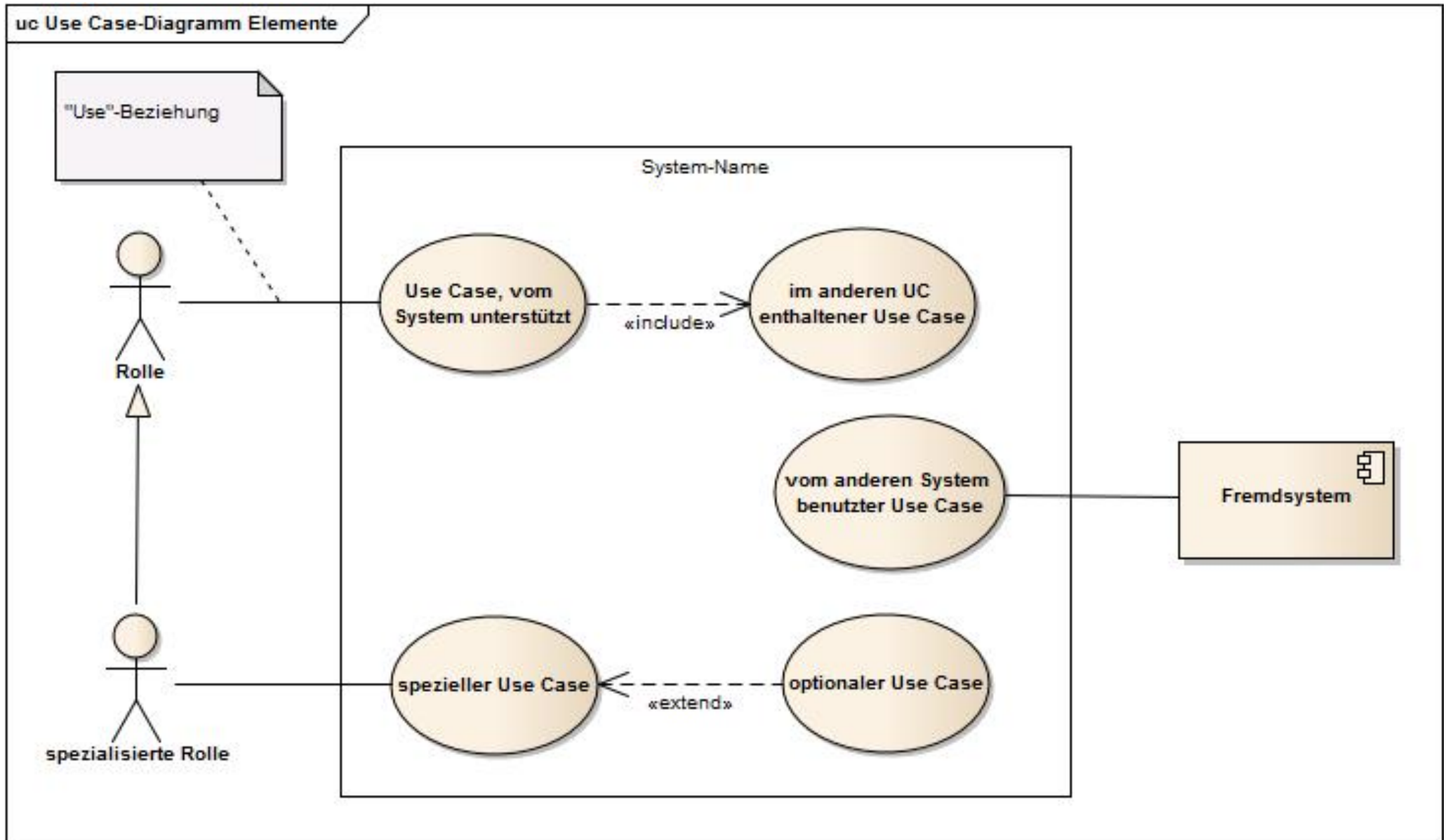


# Beispiel: Use Case „Hebe Geld Ab“



- UML Use Case-Diagramme zeigen,
  - *welche* Anwendungsfälle (= welche Funktionalität) ein System bietet und
  - *wer* diese in Anspruch nehmen kann

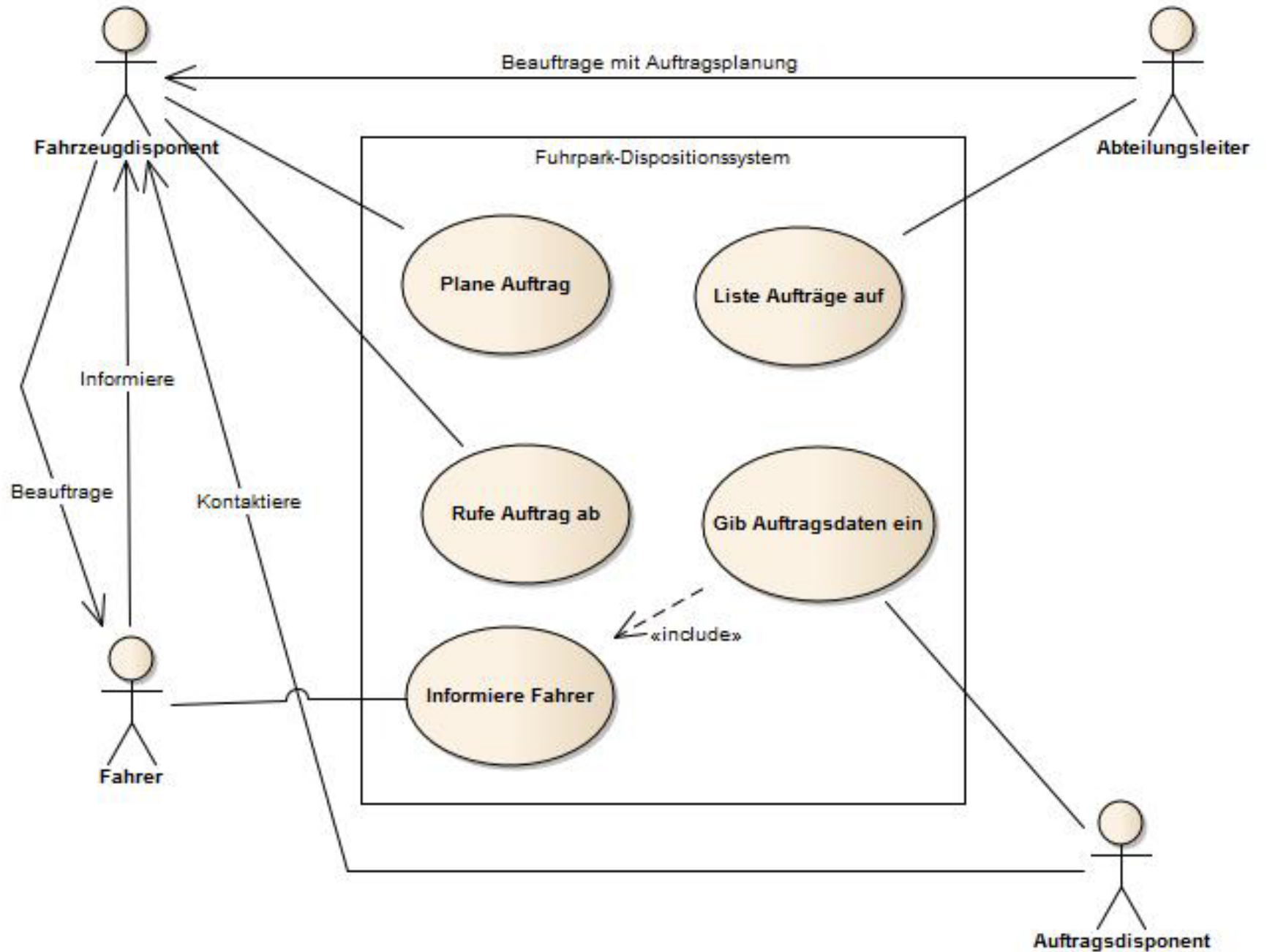
# Elemente von Use Case-Diagrammen



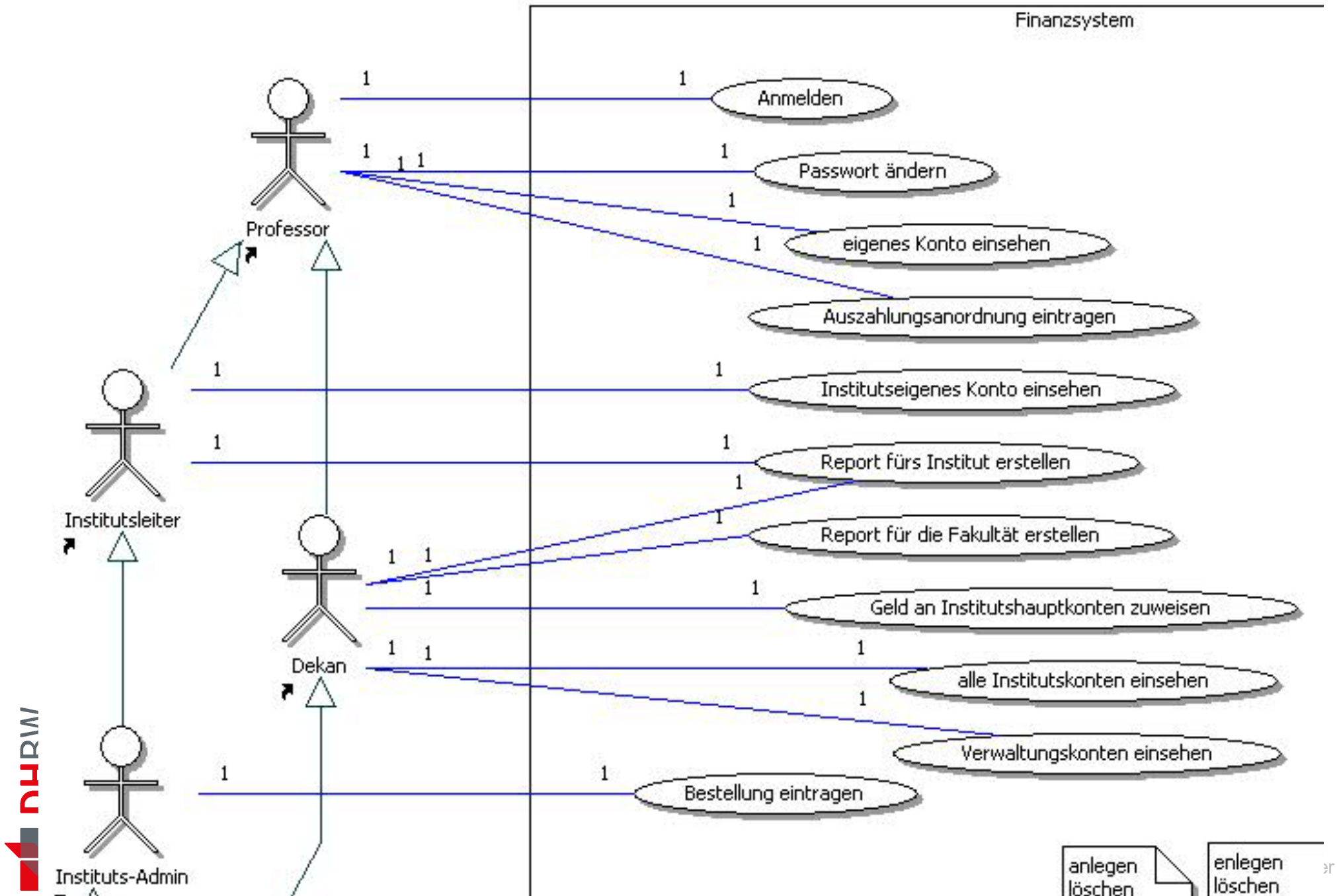
# Regeln für Use Cases

1. *Use Cases* sollen mit *Verbalphrasen* bezeichnet werden. Der Name sollte anzeigen, was der Anwender bewirken möchte.
  - z.B.           Melde Vorfall  
                  Eröffne Vorfall
2. *Rollen* sollen mit *Nominalphrasen* benannt werden.
  - z.B.           Außenbeamter  
                  Dienstleiter  
                  Zentraler DB-Server
3. Ein Use Case soll eine *vollständige Transaktion* eines Anwenders schildern.
  - z.B.           „Login“ ist *kein Use Case*

# Der Systemkontext als Use Case-Diagramm



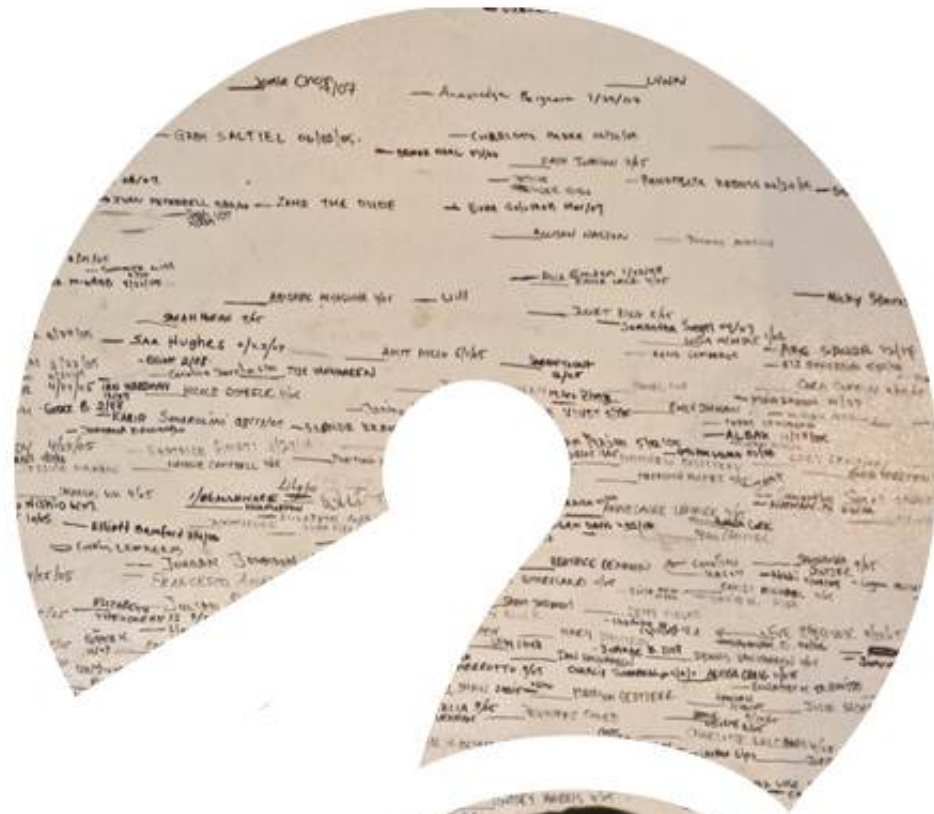
# Teil 1 des Diagramms



# Andere Darstellung

Aktivität	Details	Professor	Institutsleiter	Dekan	Instituts-Admin	Admin
Anmelden		x	x	x	x	x
Passwort ändern		x	x	x	x	x
eigenes Konto einsehen		x	x	x	x	x
Auszahlungsanordnung eintragen		x	x	x	x	x
Institutseigenes Konto einsehen			x		x	x
Report fürs Institut erstellen			x	x	x	x
Report für die Fakultät erstellen				x		x
Geld an Institutshauptkonten zuweisen				x		x
alle Institutskonten einsehen				x		x
Verwaltungskonten einsehen				x		x
Bestellung eintragen					x	x
Settings ändern	anlegen, löschen, wechseln					x
Haushaltsjahr-Verwaltung						x
Benutzer-Verwaltung	anlegen, löschen, Benutzer <-> Rolle, Benutzer <-> Institut, Benutzer <-> Konto, ändern (Name, Pwd etc.)					x
Rollen-Verwaltung	anlegen, löschen, Rolle <-> Rechte, ändern					x
Institutskonten-Verwaltung	anlegen, löschen, ändern, freigeben					x
Verwaltungskonten-Verwaltung	inkl. Gelder zuweisen					x
Buchungs-Verwaltung	ändern, löschen					x
Log einsehen						x
Lieferung eintragen						x
Instituts-Verwaltung	anlegen, löschen, ändern					x





F R A G E N



photography: woodleywonderworks  
<http://www.flickr.com/photos/wwworks/2350106729>  
art work: Peter Kaiser

## Nächster Schritt

Für (fast) jeden Anwendungsfall aus dem *Use Case*-Diagramm beschreiben, wie er abläuft

Es gibt zwei Alternativen:

1. Beschreibung als Aktivitätsdiagramm

2. Beschreibung in Tabellenform

Anwendungsbeispiel

- Ein Kunde geht zur Bank und weist sich mit seinem Namen aus, um ein Konto zu eröffnen. Er bekommt dann eine (neue) Kontonummer zugewiesen. Für die Kontoeröffnung ist eine Mindesteinzahlung von € 50 verpflichtend. Die Einzahlung kann in allen bekannten Währungen erfolgen.

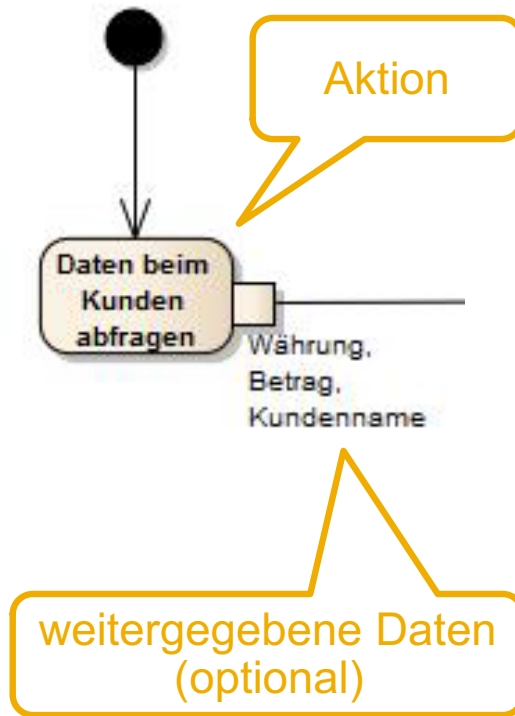
:Schalterbeamter



verantwortliche  
Rolle

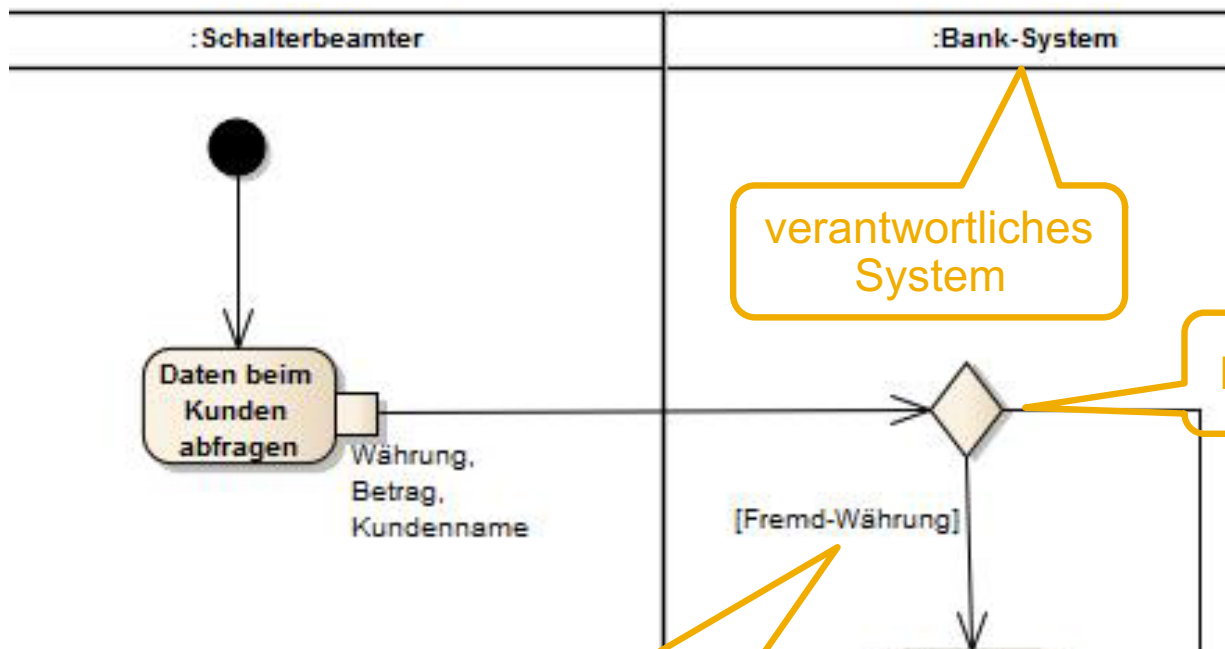
Start der Aktivität

## Darstellung als Aktivitätsdiagramm



# Darstellung als Aktivitätsdiagramm

# Darstellung als Aktivitätsdiagramm

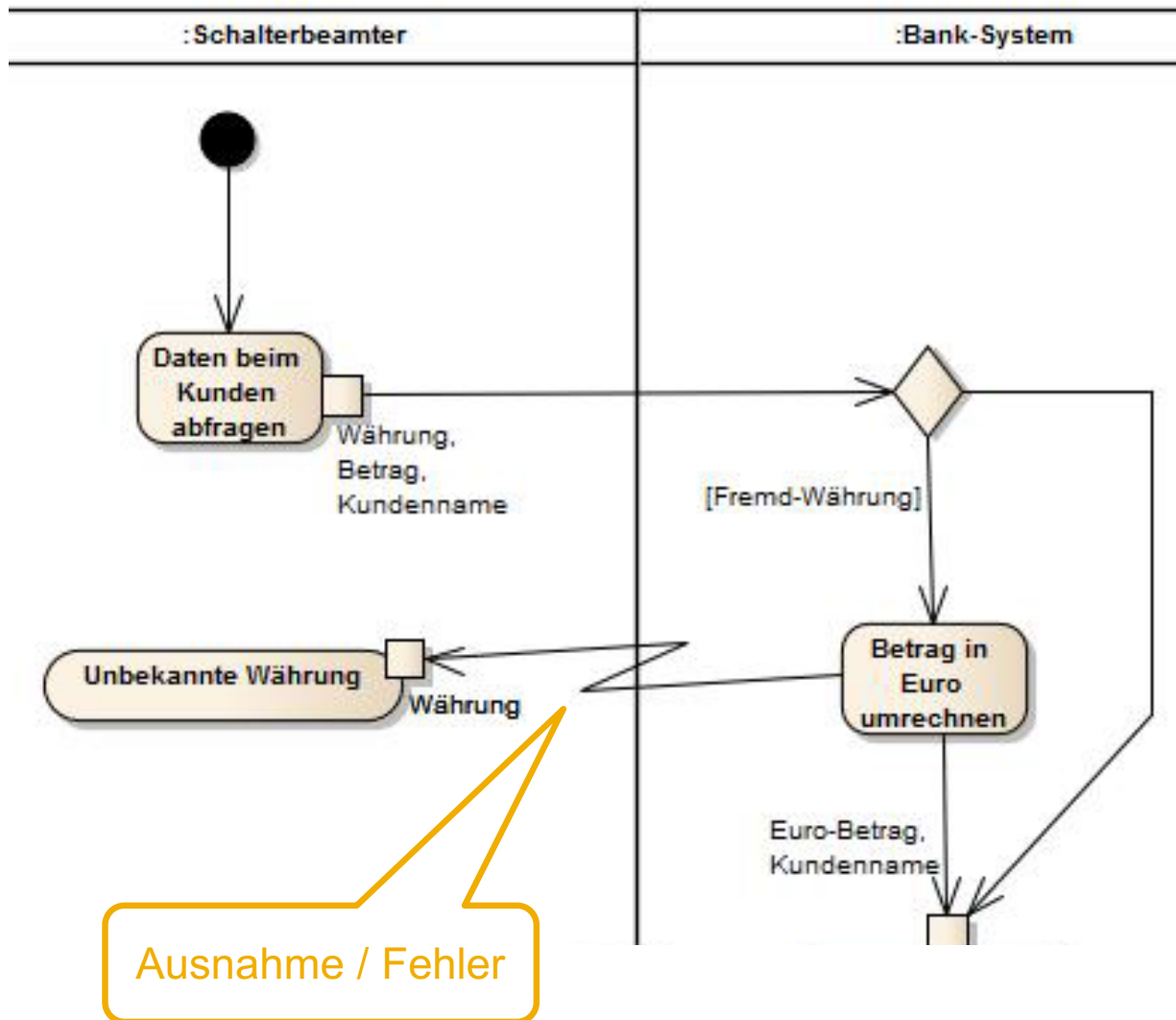


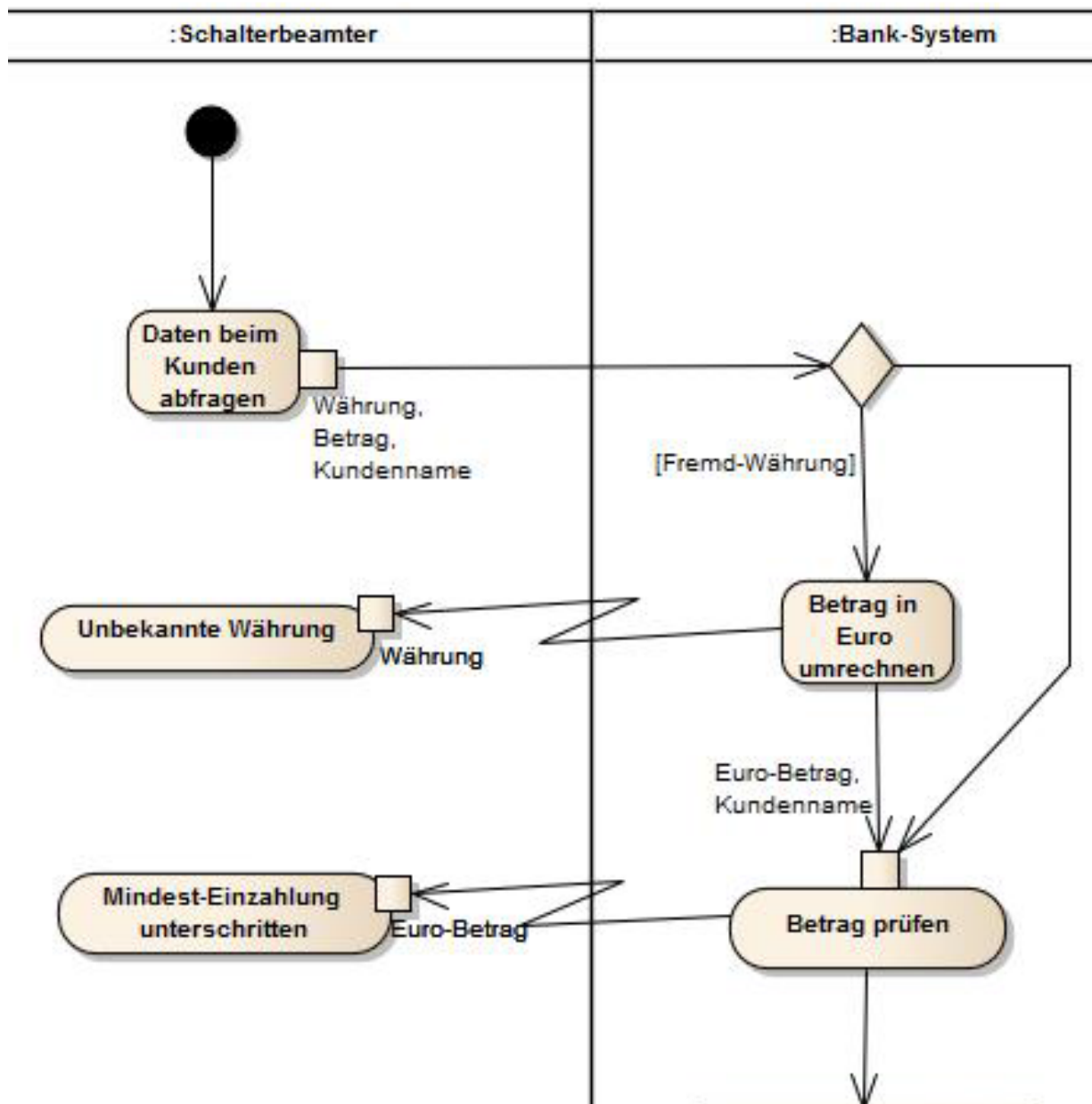
verantwortliches System

Fallunterscheidung

„Guard“ gibt an, wann dieser Zweig verfolgt wird

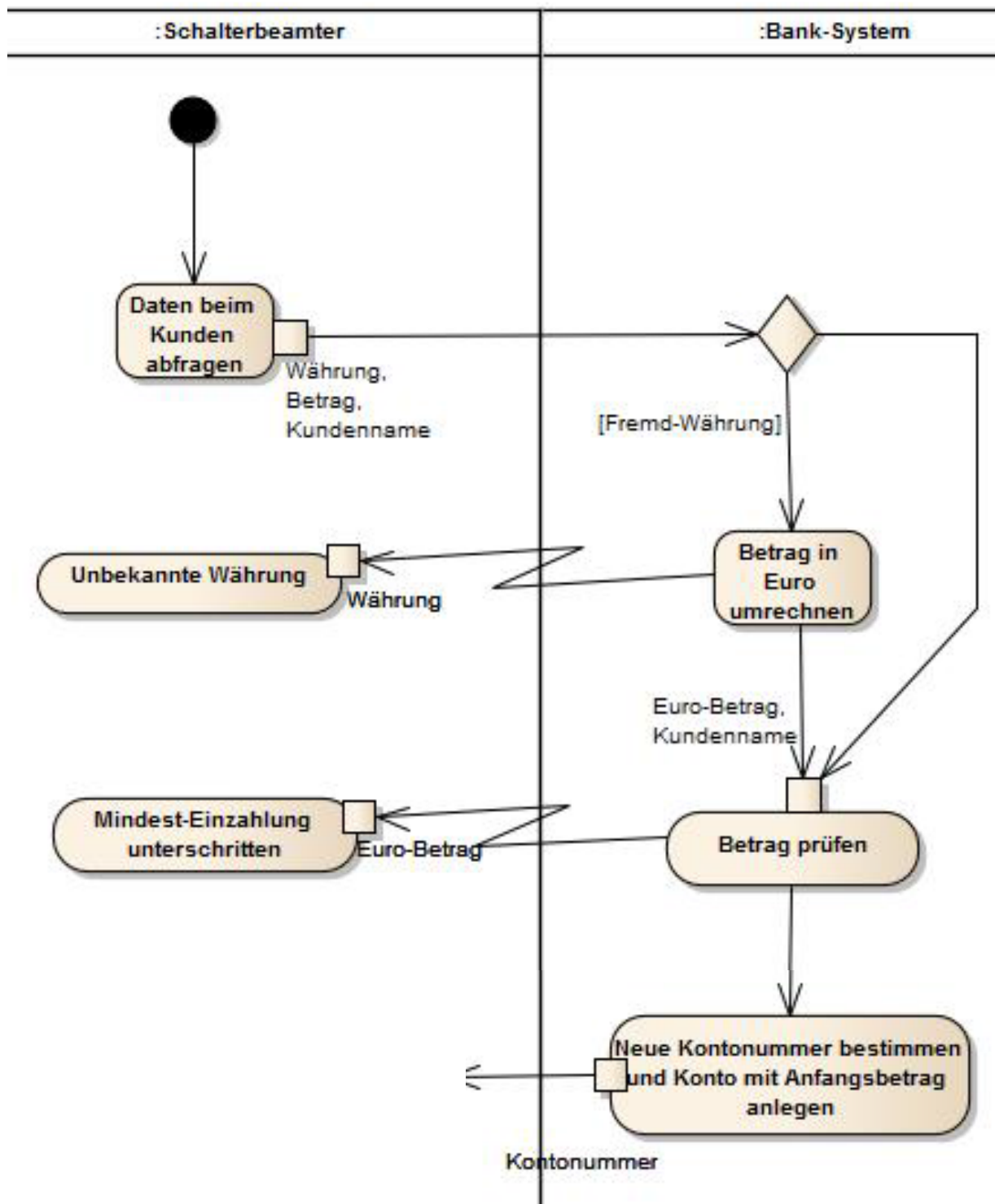
# Darstellung als Aktivitätsdiagramm



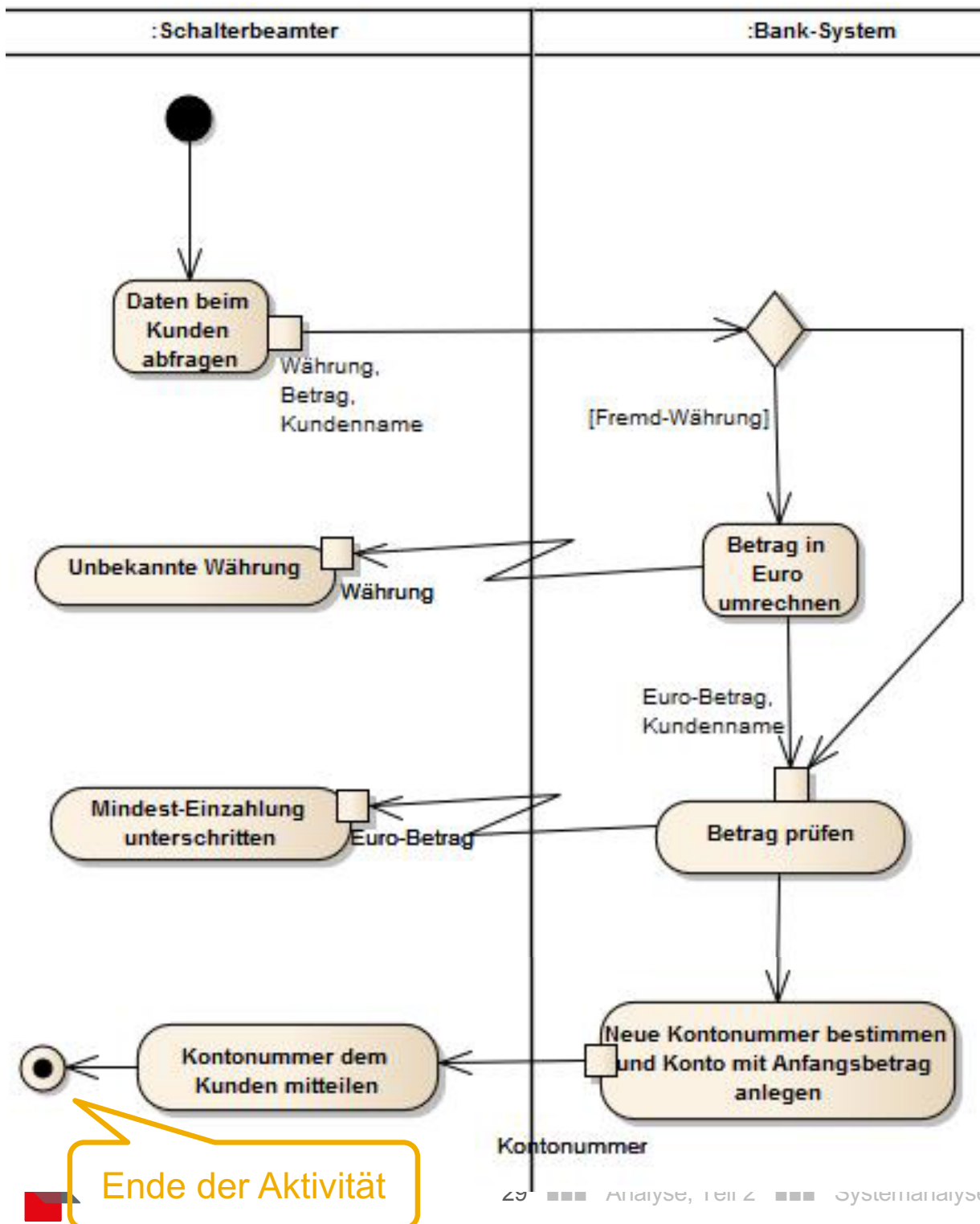


# Darstellung als Aktivitätsdiagramm

# Darstellung als Aktivitätsdiagramm



# Darstellung als Aktivitätsdiagramm



Ende der Aktivität

# Ziel der Erstellung von Aktivitätsdiagrammen

*Alle* Unklarheiten finden und  
beseitigen

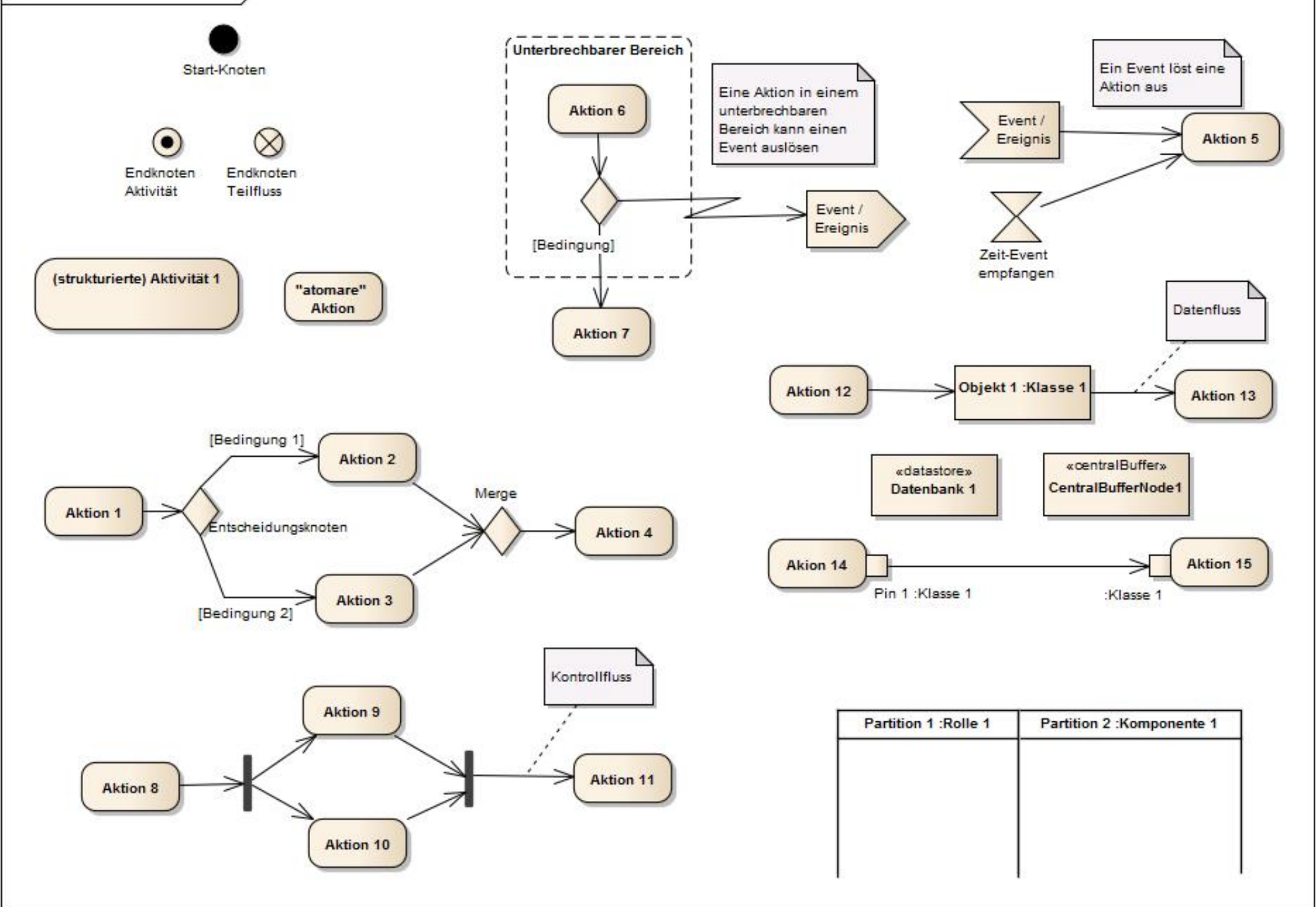
*bevor* es ans Design,  
die Implementierung  
und das Testen geht  
(spätere Korrekturen sind viel teurer!)

**Aktivitätsdiagramme** sind eine der vielen Diagrammformen der UML

- Sie beschreiben Folgen von Aktivitäten inklusive Verzweigungen und Parallelität
- Aktivitätsdiagramme in der Systemanalyse sind *immer* assoziiert mit einem *Use Case*: Sie beschreiben diesen Anwendungsfall, eventuell unter Benutzung anderer (Aktivitäts-) Diagramme  
Umgekehrt wird jeder (wichtige, typische, ...) *Use Case* durch ein Aktivitäts- (oder ein Sequenz-) Diagramm beschrieben
- Aktivitätsdiagramme ersetzen früher gebräuchliche Kontroll- und Datenfluss-Diagramme, Struktogramme etc.

# Elemente von Aktivitätsdiagrammen

## act Aktivitätsdiagramm Elemente



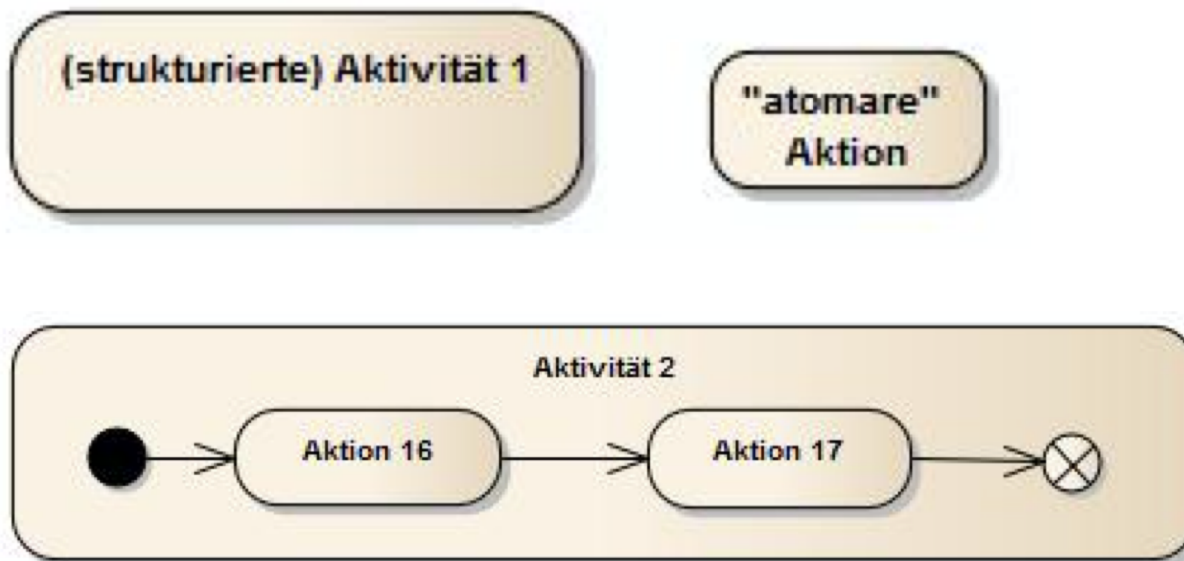
# Anfangs- und Endknoten



- Anfangsknoten bestehen aus einem dicken schwarzen Punkt
- Endknoten
  - Wird der Endknoten einer (gesamten) Aktivität erreicht, enden alle (möglicherweise parallelen) Teilflüsse
  - Der Endknoten eines Teilflusses beendet nur diesen Teilfluss

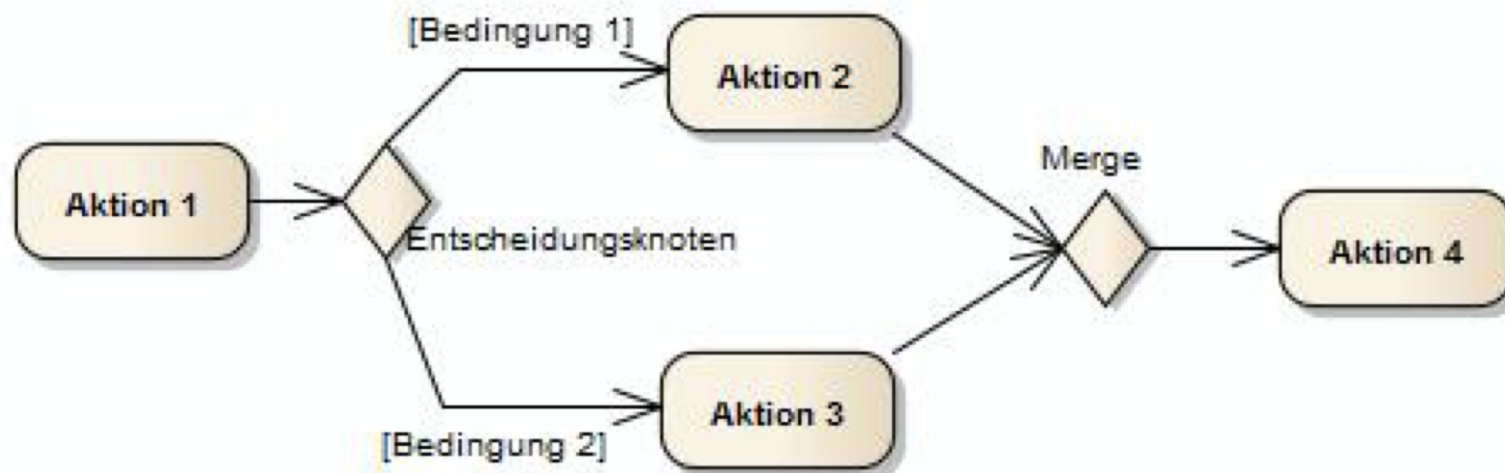
# Aktivitäten und Aktionen

- Eine Aktivität kann (meist) weiter unterteilt werden in „feinere“ Aktivitäten oder Aktionen
- Eine Aktion beschreibt einen einzelnen (atomaren) Schritt innerhalb einer Aktivität
- Der Unterschied ist graphisch nicht unbedingt sichtbar



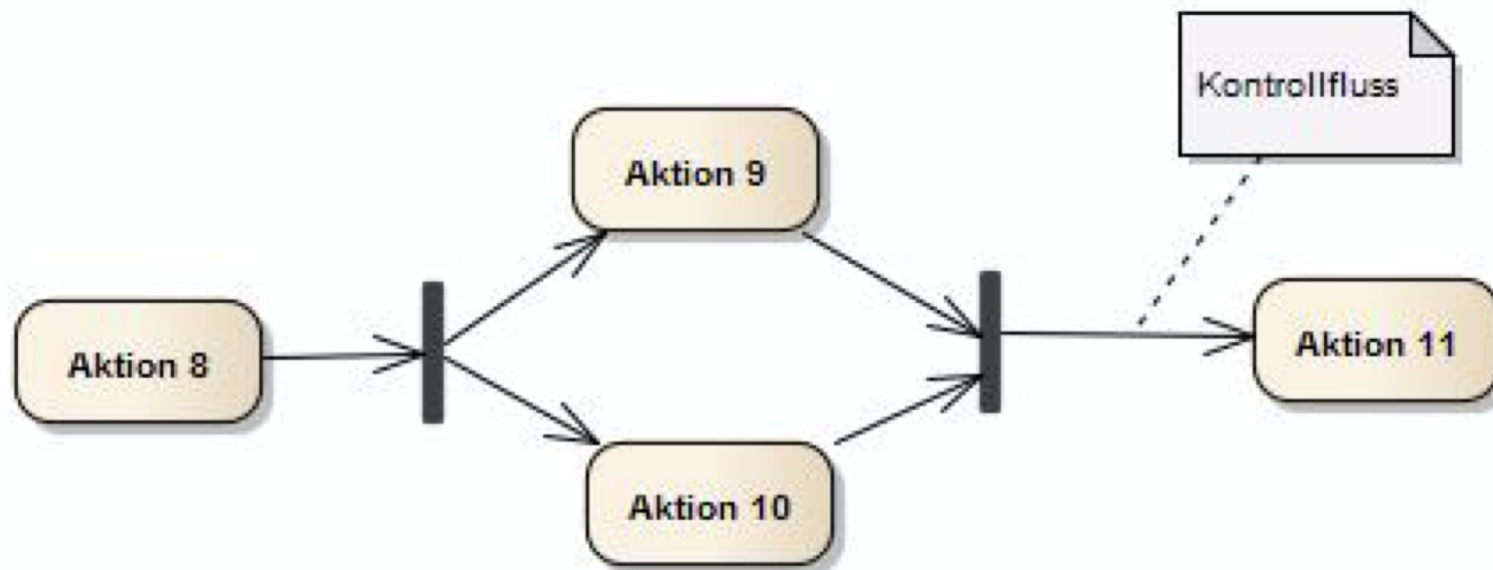
# Verzweigungs- und Vereinigungsknoten

- Verzweigungsknoten (decision) und Vereinigungsknoten (merge) werden jeweils durch eine Raute dargestellt
  - Beide können (zur Klarstellung) mit einem Namen versehen werden
  - Verzweigungsknoten haben einen eingehenden und mindestens zwei ausgehende Pfeile
  - Die davon ausgehenden Pfeile müssen (mit Ausnahme von einem) mit Bedingungen (guards) versehen werden
  - Merge-Knoten können oft entfallen



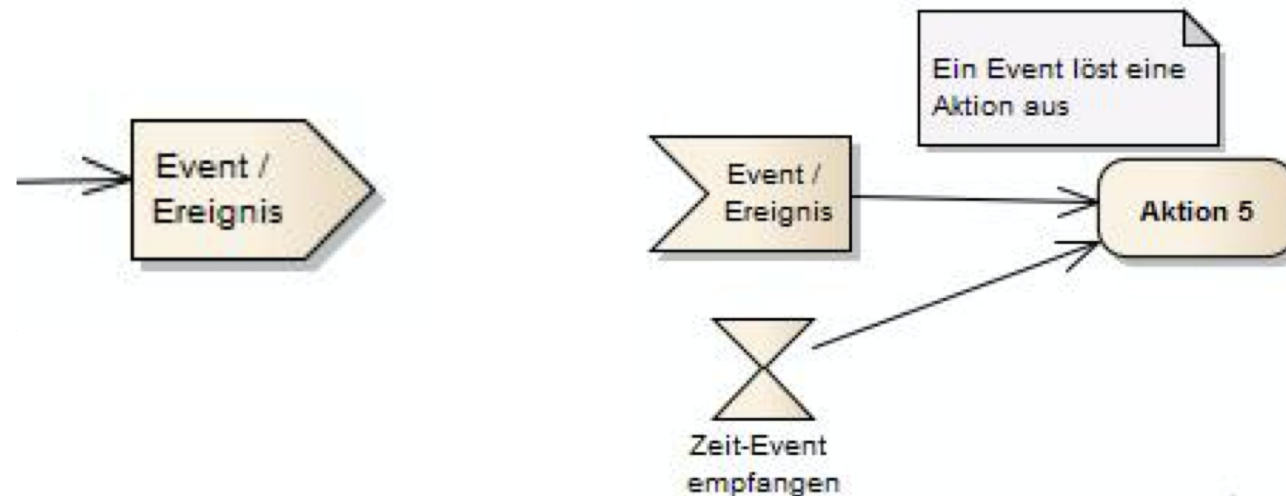
## Nebenläufigkeit (fork und join)

- Nebenläufigkeitsverzweigungen werden jeweils durch einen horizontalen oder vertikalen Balken dargestellt
  - Es gibt keinen Bedeutungsunterschied zwischen horizontalen und vertikalen Balken
- Diese Verzweigungen drücken aus, dass die Aktivitäten / Aktionen zwischen ihnen parallel ablaufen können



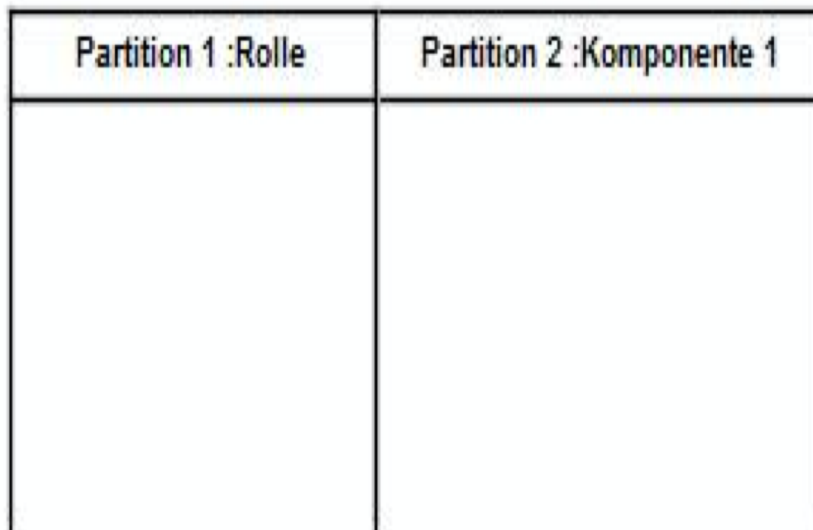
# Events / (Fehler-)Ereignisse

- Es gibt spezielle Darstellungsformen für Aktionen, die Events auslösen oder auf sie reagieren
  - Das Auslösen von Events wird durch ein konvexes Fünfeck mit dem Namen des Events dargestellt
  - Das Empfangen von Events wird durch ein konkaves Fünfeck mit dem Namen des Events dargestellt
  - Das zeitlich geplante Empfangen eines Events wird durch eine Sanduhr mit dem Namen des Events dargestellt



## Partitionen (Swim Lanes)

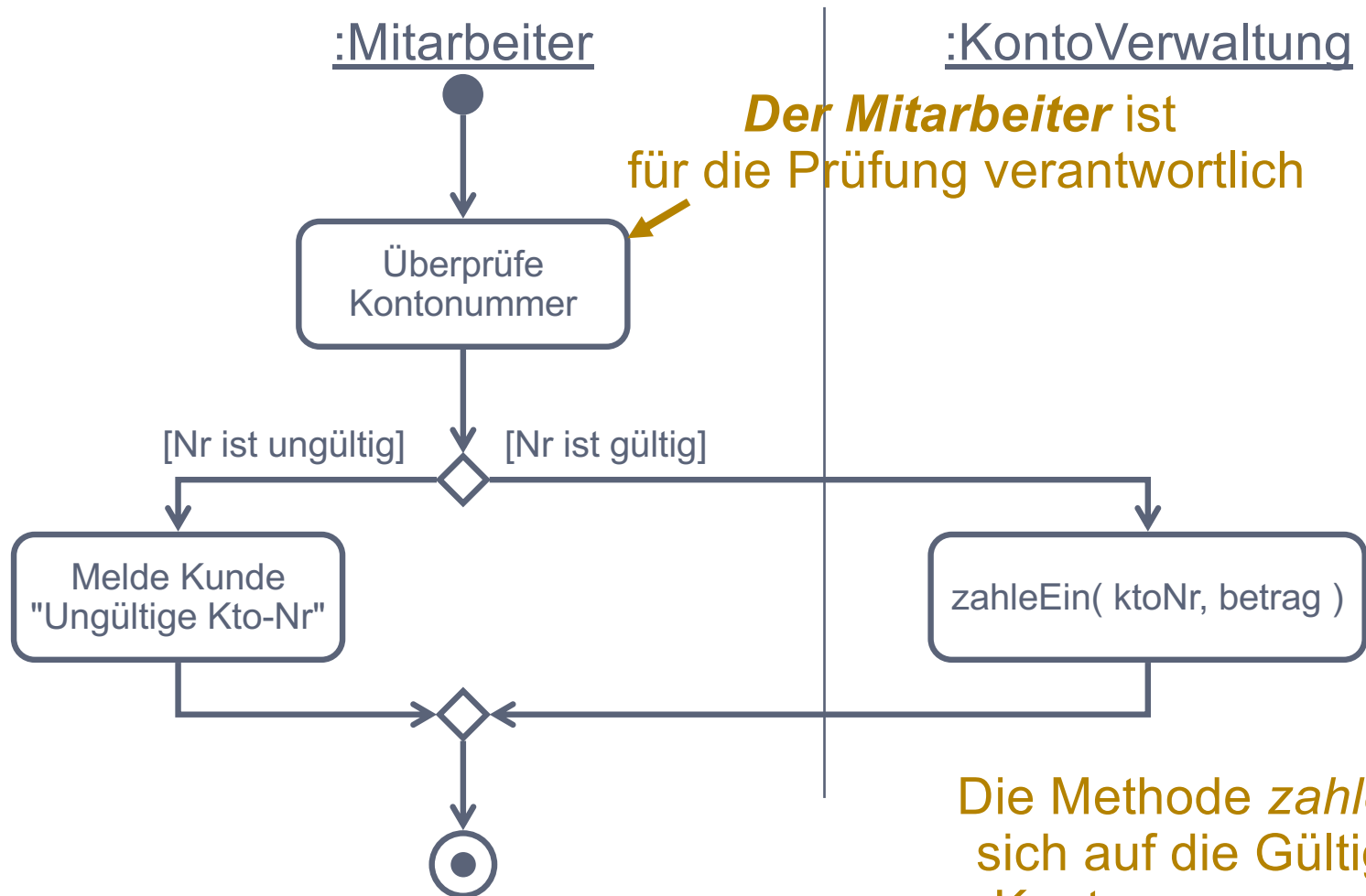
- Partitionen (hießen früher *swim lanes*) kennzeichnen Bereiche, für die bestimmte Rollen oder Systeme / Systemteile verantwortlich sind  
Die darin enthaltenen Aktivitäten / Aktionen werden von diesen Rollen / Systemen ausgeführt
- Partitionen können waagrecht oder senkrecht dargestellt werden



# Beispiel für die Anwendung von Partitionen: Lösung 1

Was passiert, wenn beim Einzahlen auf ein Konto eine falsche Kontonummer angegeben wird?

Wer überprüft das?



**Der Mitarbeiter ist für die Prüfung verantwortlich**

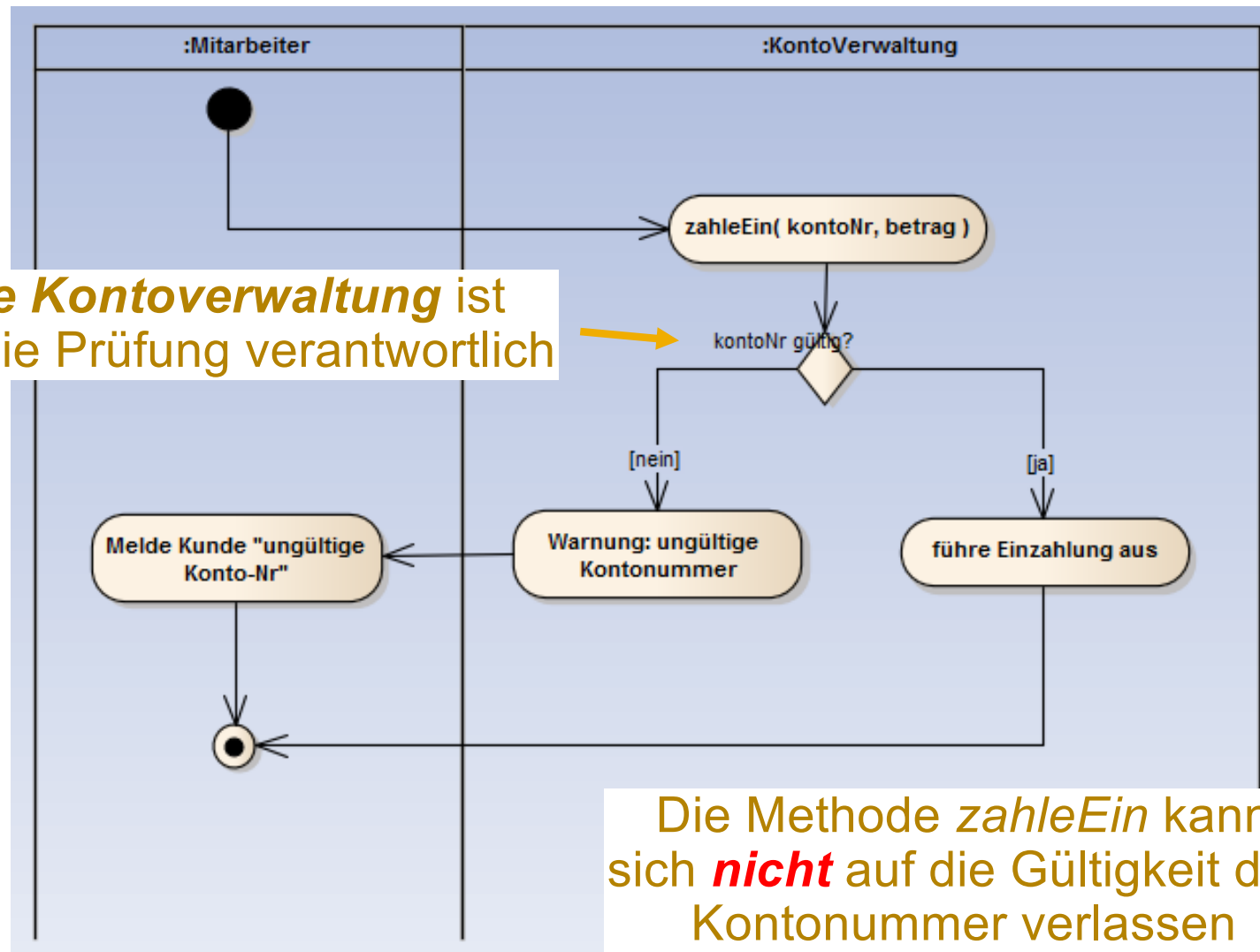
**Die Methode *zahleEin* kann sich auf die Gültigkeit der Kontonummer verlassen**

# Beispiel für die Anwendung von Partitionen: Lösung 2

Was passiert, wenn beim Einzahlen auf ein Konto eine falsche Kontonummer angegeben wird?

Wer überprüft das?

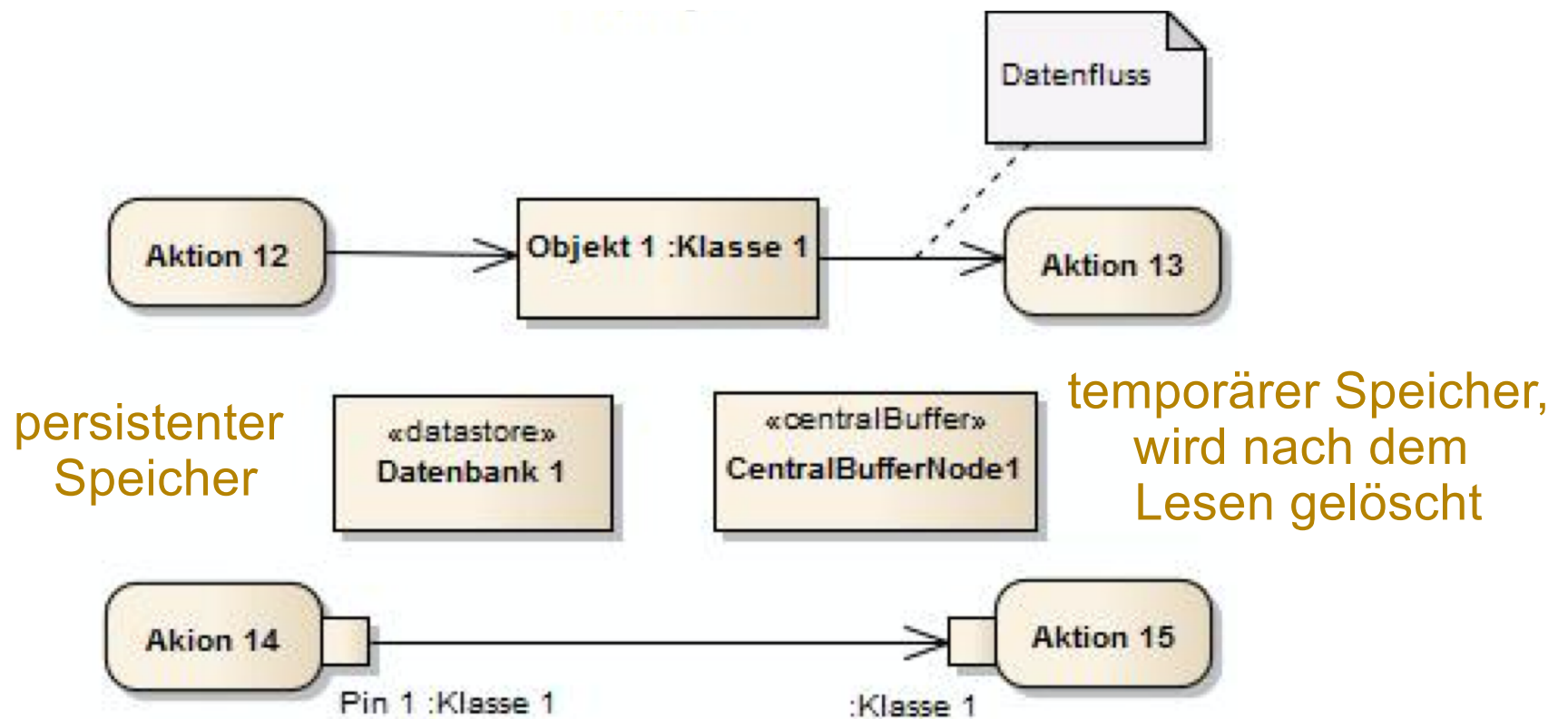
**Die Kontoverwaltung ist für die Prüfung verantwortlich**



Die Methode *zahleEin* kann sich **nicht** auf die Gültigkeit der Kontonummer verlassen

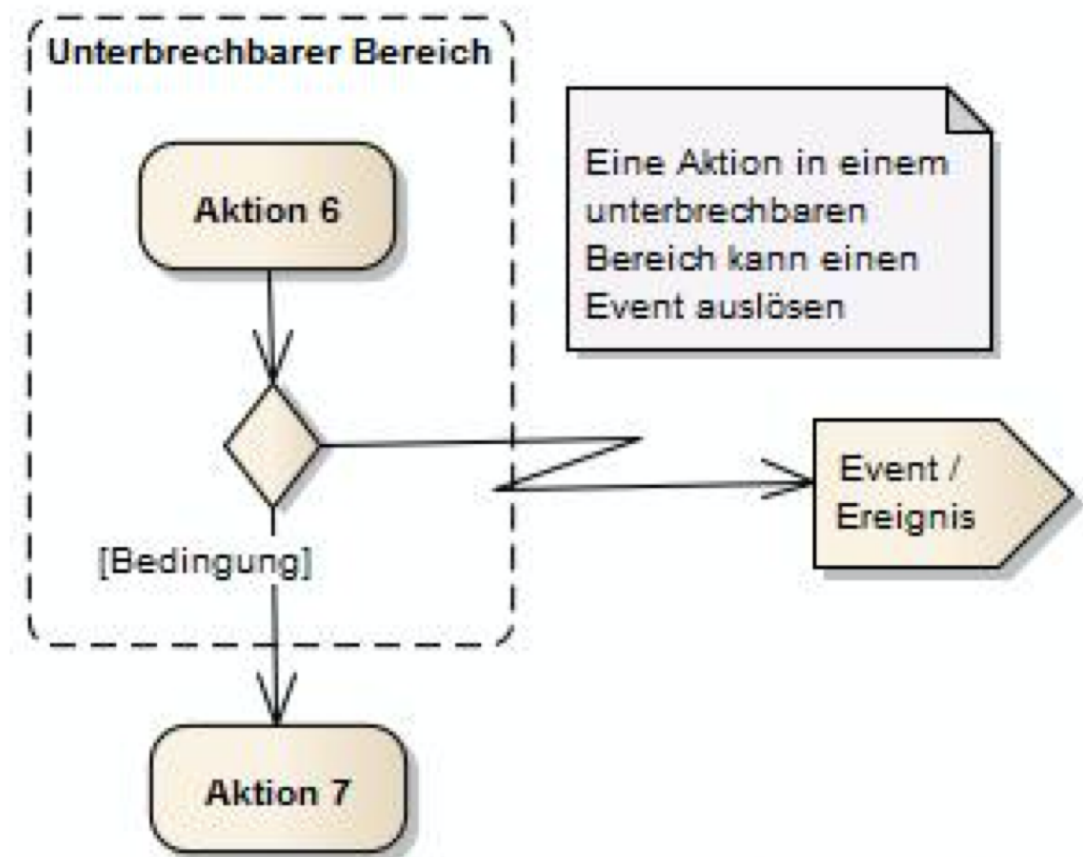
# Objektfluss

- Mittels *Objektfluss* wird dargestellt, welche Ein- oder Ausgaben bestimmte Aktivitäten / Aktionen haben
- Es gibt unterschiedliche, aber gleichbedeutende Darstellungsformen

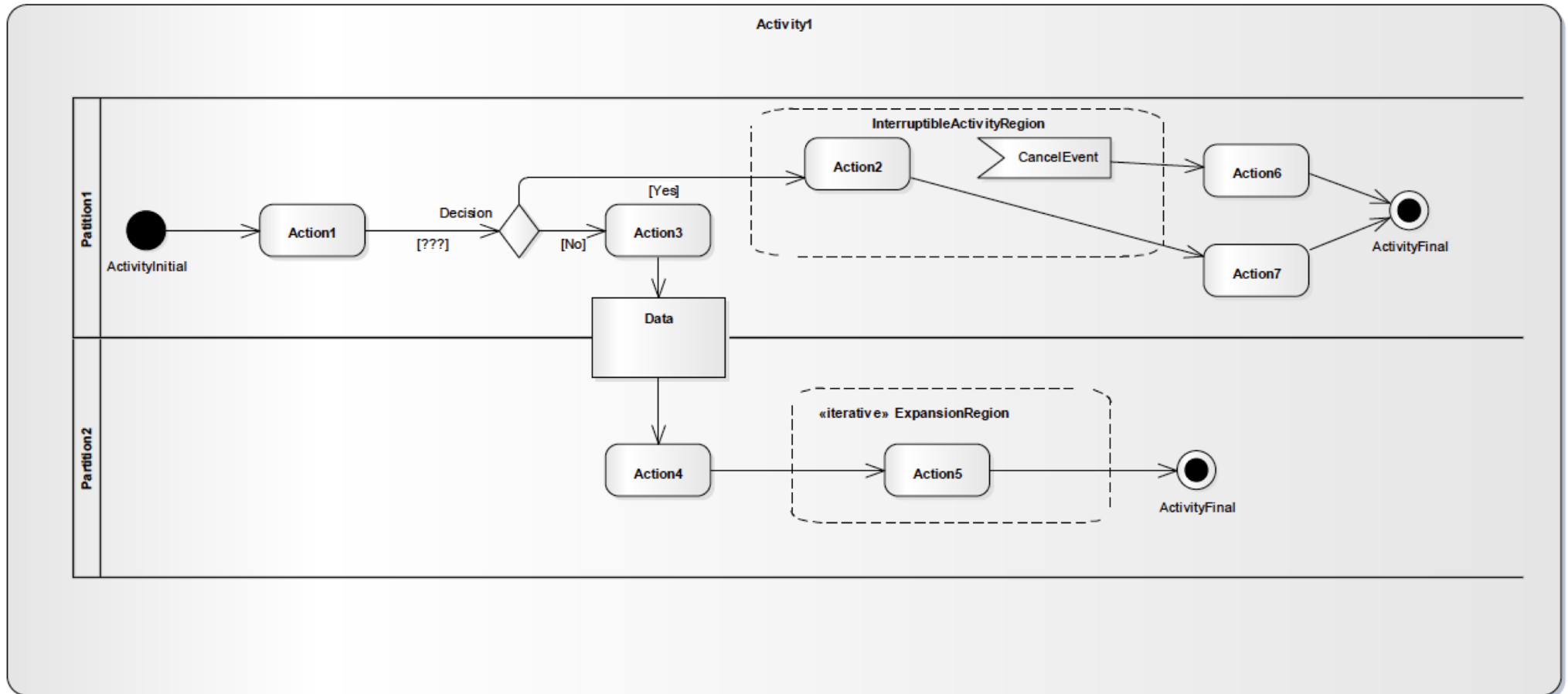


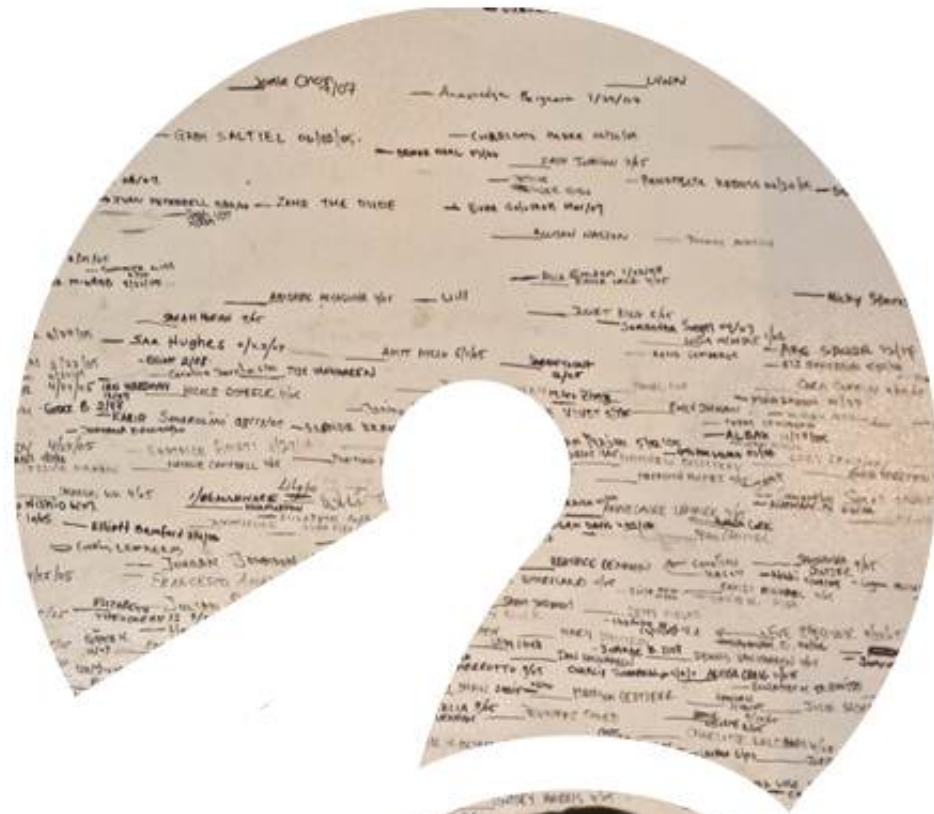
## Bereiche (*regions*)

- Bereiche dienen dazu, zusammengehörige Aktivitäten zu kennzeichnen
- Es gibt
  - iterative,
  - parallele,
  - stream- und
  - unterbrechbare Bereiche



# Komplett-Illustration





F R A G E N



photography: woodleywonderworks  
<http://www.flickr.com/photos/wwworks/2350106729>  
art work: Peter Kaiser