

3.2 Anwendungsfall-Sicht

- 3.2.1 Exkurs: Softwareentwicklungsprozess
- 3.2.2 Elemente der Anwendungsfall-Sicht
- 3.2.3 Beziehungen zwischen den Elementen der Anwendungsfall-Sicht

3.2.1 Exkurs:

Softwareentwicklungsprozess

Softwareentwicklungsprozess besteht im wesentlichen aus den Phasen:

- **Anforderungsanalyse** (Requirement Analysis)
 - Beschreibung des Problems und der Anforderungen
- **Bereichsanalyse** (Domain Analysis)
 - Identifikation der beteiligten Objekte

Softwareentwicklungsprozess

(Forts.)

- **Entwurf (Design)**
 - Verantwortlichkeit der einzelnen Objekte
 - Zusammenarbeit der Objekte zur Bewerkstelligung der Anforderungen
- **Konstruktion**
 - Implementation des Entwurfs
 - Ausführung auf Hardware

Softwareentwicklungsprozess

(Forts.)

- Softwareentwicklungsprozess wird **iterativ** und **inkrementell** durchlaufen
 - **Iterativ**: mehrere Durchläufe
 - **Inkrementell**: Gesamtfunktionalität des Systems wächst

Softwareentwicklungsprozess

(Forts.)

- In der OO Analyse und Design hat sich für die Anforderungsanalyse die Benutzung von **Anwendungsfällen** durchgesetzt
- **Zentraler Aspekt:**
 - **Modellierung** der Funktionalität eines Systems, Subsystems oder einer Klasse im Kontext der **Interaktion zu aussenstehenden Benutzern** und **Systemen**

3.2.2 Elemente der Anwendungsfall-Sicht

- Akteure
- Anwendungsfälle

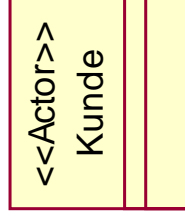
Akteure (Actors)

- *Akteur* (Actor) = feststehende Rolle, die ein ausenstehendes Objekt (Anwender, externes System) in einem Anwendungsfall einnimmt
 - derselbe Anwender kann in die Rolle mehrerer Akteure schlüpfen
 - derselbe Akteur kann durch mehrere verschiedene Anwender repräsentiert sein
- Interaktion durch Austausch von Nachrichten

Akteure (Forts.)

Graphische Darstellung:

- Strichmännchen
- Stereotypisierte Klasse



Stereotypisierte Klassennotation



Verkäufer



DBSystem

Anwendungsfall (Use Case)

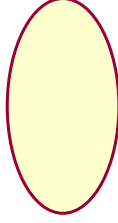
- *Szenario* (Scenario) =
Beschreibung einer Folge von Aktionen und Interaktionen zwischen Akteur und System
 - *Anwendungsfall* (Use Case) =
zusammengehörige Menge von Szenarien, die eine bestimmte nach aussen sichtbare Funktionalität bereitstellt
- ⇒ Szenario ist ein individueller Ablauf eines Anwendungsfalls

Anwendungsfall (Forts.)

- *Granularität* eines Anwendungsfalls:
 - zeitlich abgeschlossene Aktionsfolge
- *Spezifikation* eines Anwendungsfalls:
 - informaler strukturierter Text
 - strukturierter Text mit Vor- und Nachbedingungen
 - Pseudo Code
 - Veranschaulichung durch Zustandsmaschine oder Aktivitätsdiagramm

Anwendungsfall (Forts.)

Graphische Darstellung:



Überprüfe Benutzer

Anwendungsfall (Forts.)

- **Beispiel:** Geldautomat, Überprüfe Benutzer
 - **Hauptzenario:**
 - System erwartet PIN vom Benutzer
 - Benutzer kann PIN über Tastatur eingeben
 - Bestätigung der PIN mit RETURN-Taste
 - System überprüft PIN auf Gültigkeit
 - Wenn PIN gültig, erlaubt das System den Zugriff
 - Beendigung des Anwendungsfalls

Anwendungsfall (Forts.)

- **Ausnahme Szenario 1:**
 - Benutzer bricht Transaktion durch Betätigung der CANCEL-Taste ab
 - Anwendungsfall startet erneut, ohne dass Änderungen am Konto des Kunden durchgeführt werden
- **Ausnahme Szenario 2:**
 - Benutzer löscht PIN und gibt neue ein, bevor er die RETURN-Taste betätigt

Anwendungsfall (Forts.)

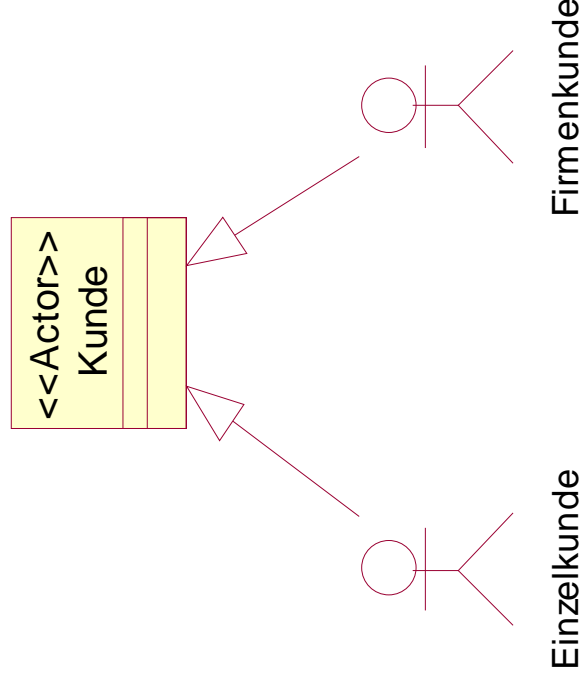
- **Ausnahme Szenario 3:**
 - Falls ungültige PIN eingegeben wurde, startet der Anwendungsfall erneut
 - Falls sich dies dreimal wiederholt, wird der Zugriff gesperrt

3.2.3 Beziehungen zwischen Elementen der Anwendungsfall-Sicht

- Es gibt Beziehungen zwischen
 - Akteur - Akteur
 - Akteur - Anwendungsfall
 - Anwendungsfall - Anwendungsfall

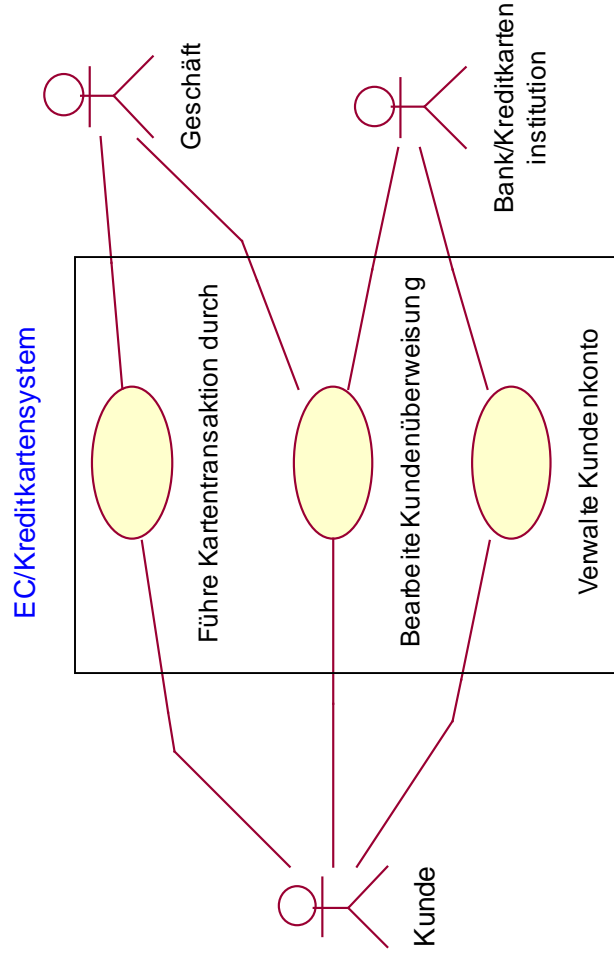
Beziehungen - Generalisierung von Akteuren

- Bei Akteuren mit Gemeinsamkeiten lassen sich **Generalisierungsbeziehungen** modellieren



Beziehungen - Assoziation

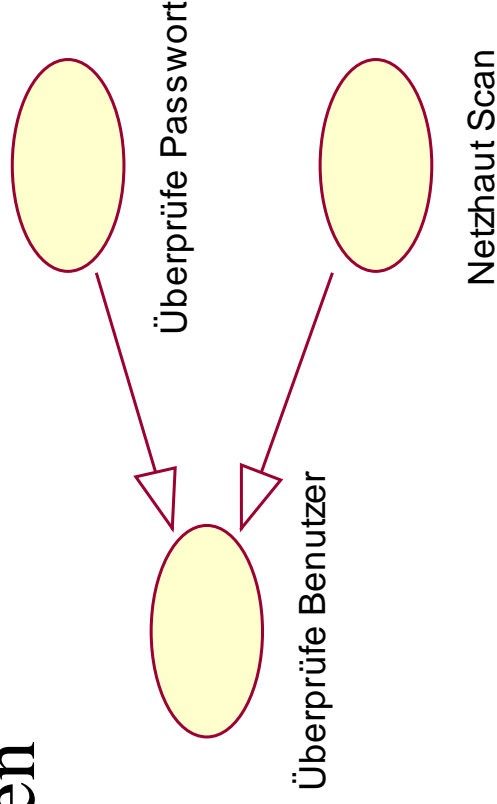
- Interaktion zwischen Akteuren und Anwendungsfällen wird durch **Assoziation** dargestellt



Beziehungen - Anwendungsfall

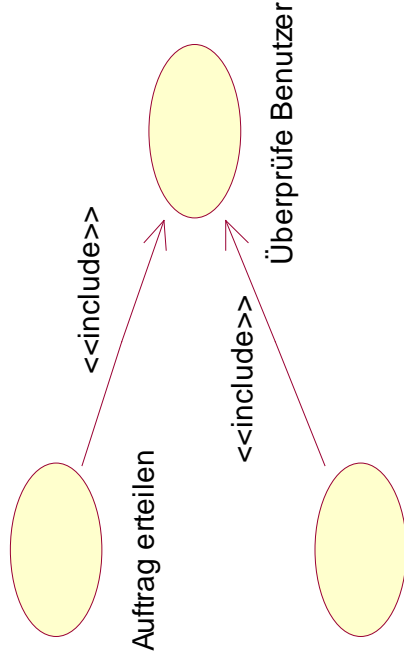
Generalisierung

- Spezialisierungen eines Anwendungsfalls lassen sich durch Generalisierungen modellieren



Beziehungen - include

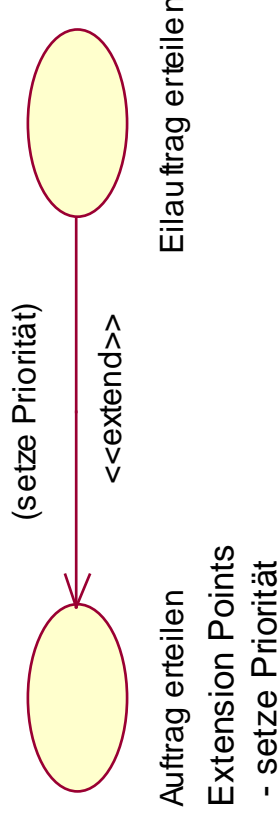
- Einfügen eines Anwendungsfalls in einen anderen
- Modellierung durch Abhängigkeitsbeziehung mit Stereotyp `include`



Auftragsstatus abfragen

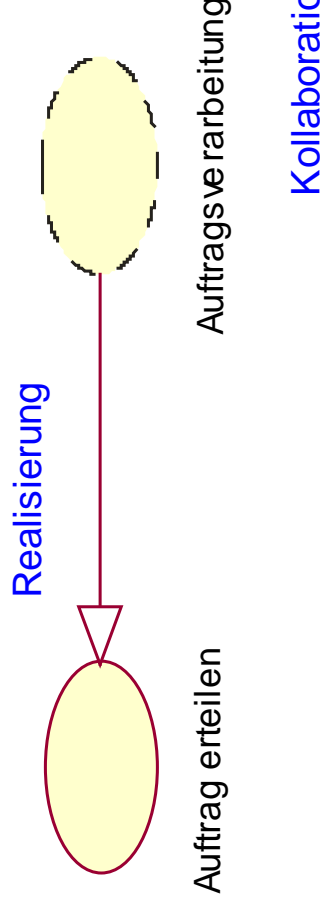
Beziehungen - extend

- Erweiterung eines Anwendungsfalls um einen anderen (optionales Verhalten)
 - Basis-Anwendungsfall besitzt *Erweiterungspunkte* (Extension Points)
 - Erweiterungs-Anwendungsfall spezifiziert die Erweiterungspunkte, an denen er eingefügt wird



Beziehungen - Realisierung

- Anwendungsfall ist strikt getrennt von seiner Implementierung
- Letztendlich erfolgt seine Implementierung i.a. durch eine *Kollaboration*, eine Zusammenarbeit mehrerer Klassen und anderer Elemente



Anwendungsfall-Sicht

Zusammenfassung

- Darstellungsmöglichkeiten relativ einfach
- Graphische Darstellung eher zweitrangig
- Wichtig:
 - Identifizierung und genaue Ausformulierung der Anwendungsfälle, die das System ausmachen
 - Grundlage für die Modellierung von geeigneten Klassen und Interaktionen zwischen diesen