

# Objektorientierte Software-Entwicklung

---

Prof. Dr. Rolf Hennicker

22.11.2007

# Kapitel 3

## Objektorientierte Analyse

## Ziele

- Eine Anwendungsfall-Analyse für ein zu entwickelndes System durchführen können.
- Schrittweise ein statisches Modell für ein Anwendungsproblem erstellen können.
- Interaktionsdiagramme für kommunizierende Objekte erstellen können.
- Zustands- und Aktivitätsdiagramme (aus dem Interaktionsmodell) herleiten können.

Die vorgestellte Methode orientiert sich an

- OMT (Rumbaugh et al.)
- "Uses Cases" nach OOSE (Jacobson et al.)

**Ziel:** Präzise und verständliche Beschreibung der Anforderungen.

### ***Schritte dazu:***

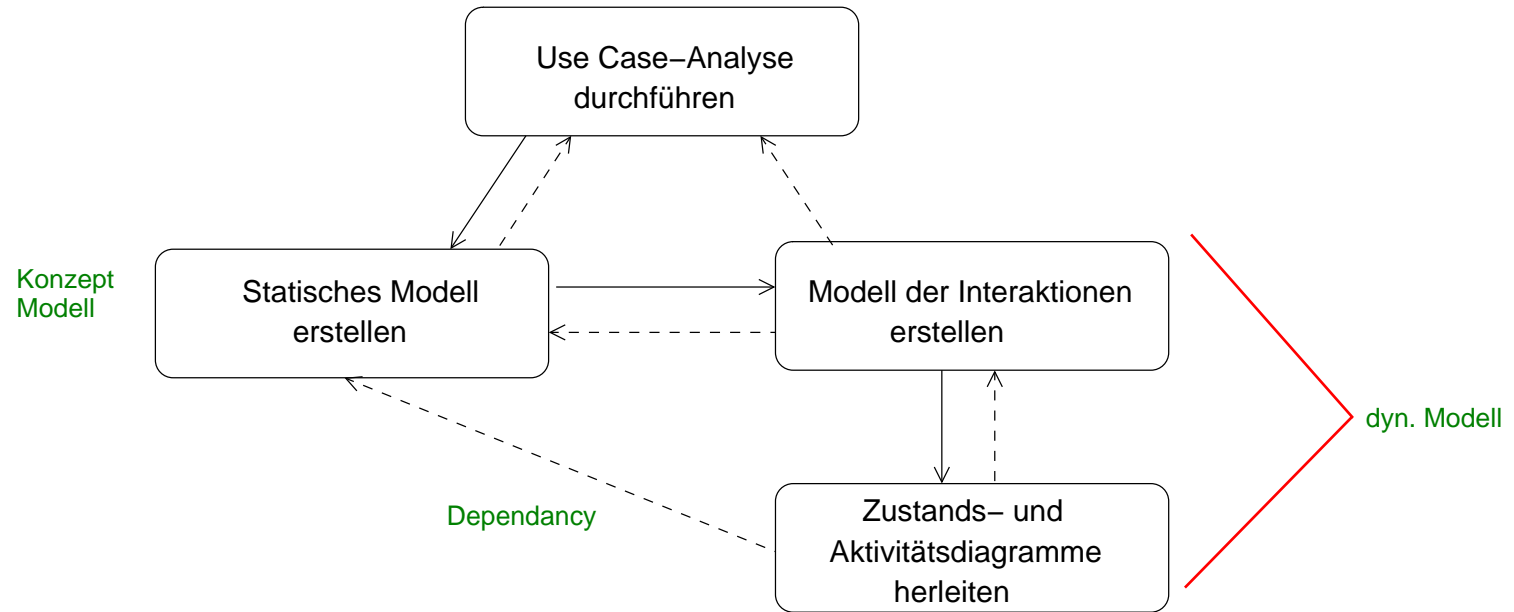
1. "Use Case"-Analyse
2. Entwicklung eines statischen Modells (in Form von Klassendiagrammen)
3. Entwicklung eines dynamischen Modells  
(in Form von Interaktionsdiagrammen, Zustands- und Aktivitätsdiagrammen)
4. Validieren, überarbeiten und erweitern der Modelle (in mehreren Iterationen)

### ***Bemerkung***

Während der Analysephase kann zusätzlich auch

- eine Grobarchitektur des Systems erstellt werden
- die Benutzerschnittstelle skizziert werden

# Prozessmodell der objektorientierten Analyse



## 3.1 Anwendungsfall-Analyse

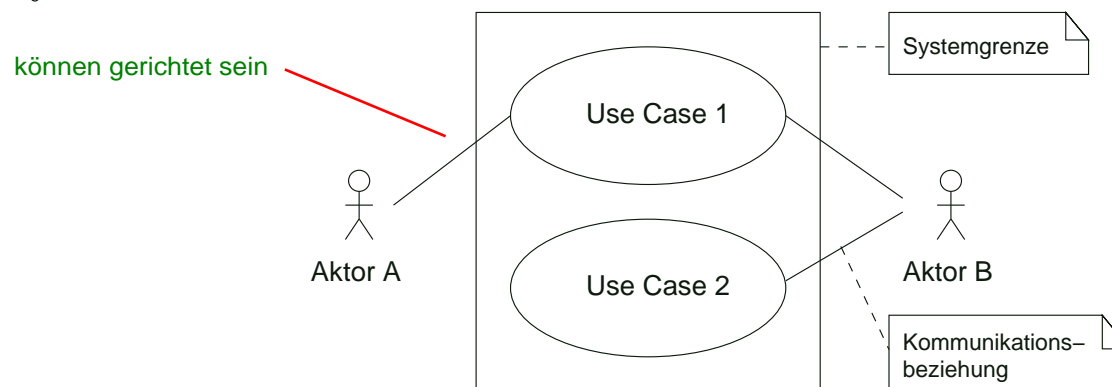
**Ausgangspunkt:** informelle, knappe Problembeschreibung

**Ziel:** Beschreibung der gewünschten Funktionalität des zu entwickelnden Systems.

### 3.1.1 Use Case-Modell

- Besteht aus *Aktoren* und *Use Cases* (*Anwendungsfällen*).
- Beschreibt eine **externe** Sicht auf das System.

*Darstellungsform:*



## Aktoren

- Tauschen von außen Informationen mit dem System aus (Benutzer, andere Systeme, Geräte).
- Aktoren werden durch die *Rolle*, die ein Benutzer gegenüber dem System einnimmt, charakterisiert.

Nutzfall

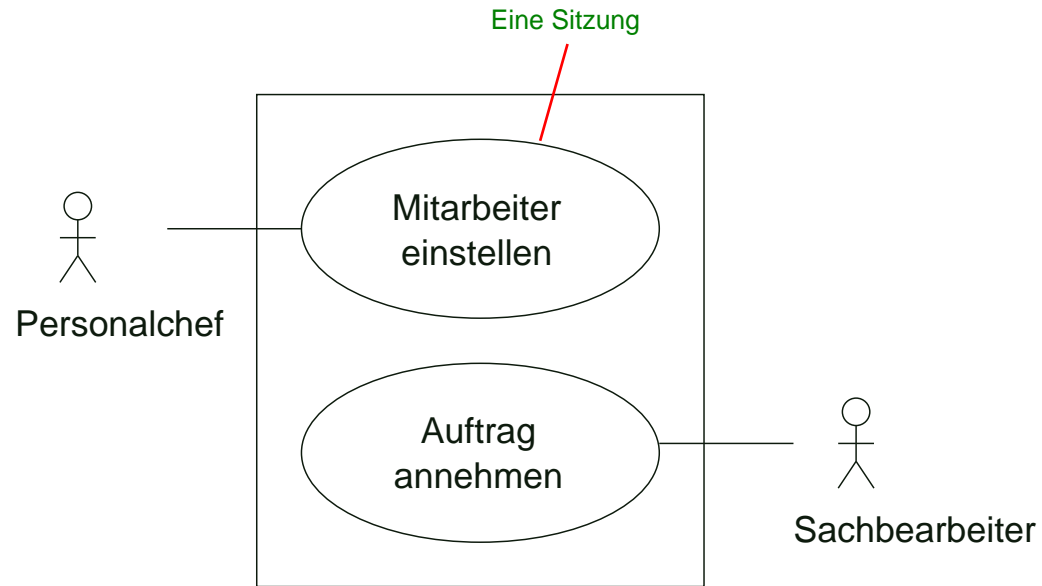
## Anwendungsfall Use Case

- Beschreibt eine funktionale Anforderung an ein System.
- Beschreibt die Interaktionen zwischen einem (oder mehreren) Aktoren und dem System bei der Bearbeitung einer bestimmten, abgegrenzten Aufgabe.

*Definition nach Jacobson:*

Ein Anwendungsfall ist umfasst Alternativen eine Menge von verhaltensverwandten Sequenzen von Interaktion Transaktionen, die durch ein System ausgeführt werden und ein zielgerichtet messbares Ergebnis liefern.

*Beispiel:*



*Beachte:*

Häufig sind Use Cases die computergestützten Teile von Geschäftsprozessen.

## 3.1.2 Vorgehensweise bei der Erstellung eines Use Case-Modells

1. Bestimmung der *Aktoren*, die mit dem System interagieren.  
(Wer benützt das System? Wer holt/liefert Informationen von dem/für das System?)
2. Bestimmung der *Anwendungsfälle* aufgrund der Aufgaben, die das System für jeden einzelnen Aktor erledigen soll (z.B. durch Interviews).  
*Schwierigkeit*: richtige Granularität finden
3. Erstellung eines *Anwendungsfall-Diagramms*, ggf. mit kurzer Beschreibung der Aktoren und Use Cases.
4. Beschreibung der Anwendungsfälle (iterativ).

Alternative: neuer Anwendungsfall oder nur dazu zum aktuellen Anwendungsfall ?!

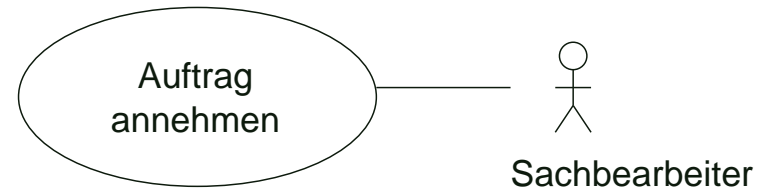
Jacobson: 10 Personen Projekt ~ 10 Anwendungsfälle  
Fauber: ~ 100 Anwendungsfälle

Eine *Anwendungsfall-Beschreibung* besteht aus:

- Name des Anwendungsfalls
- Kurzbeschreibung
- Vorbedingung  
(Voraussetzung für eine erfolgreiche Ausführung des Anwendungsfalls)
- Nachbedingung  
(Zustand nach erfolgreicher Ausführung)
- einem Standardablauf (Primärszenario)  
(Schritte bzw. Interaktionen, die im Normalfall bei Ausführung des Anwendungsfalls durchlaufen werden)
- mehreren Alternativabläufen (Sekundärszenarien)  
(bei Fehlerfällen und Optionen)

Zusätzlich kann ein Aktivitätsdiagramm für den Anwendungsfall angegeben werden.

*Beispiel:*



**Anwendungsfall:** Auftrag annehmen

**Kurzbeschreibung:**

Ein Sachbearbeiter nimmt für einen Kunden eine Bestellung von Artikeln auf.

**Vorbedingung:**

Das System ist bereit einen neuen Auftrag anzunehmen.

**Nachbedingung:**

Die Bestellung ist aufgenommen.

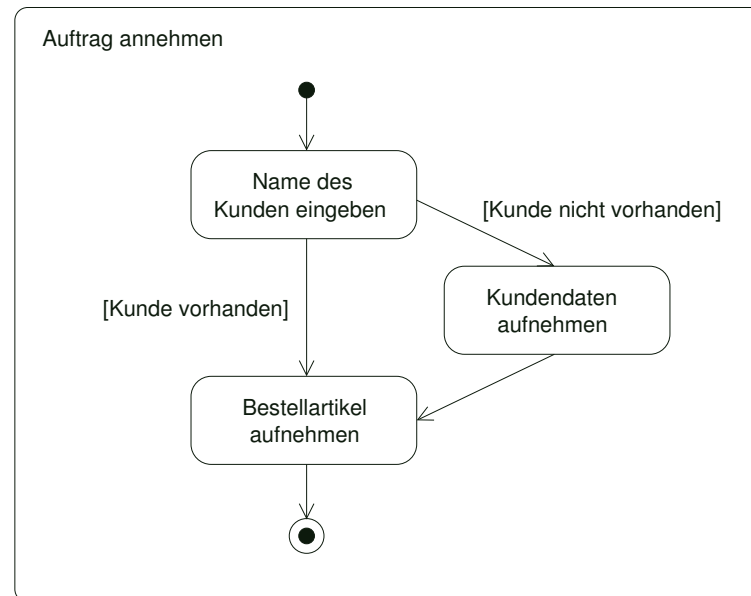
## Primärszenario:

1. Der Sachbearbeiter gibt den Namen des Kunden ein.
2. Das System zeigt die Kundendaten an.
3. Der Sachbearbeiter gibt die Daten für die zu bestellenden Artikel ein.

## Sekundärszenarien:

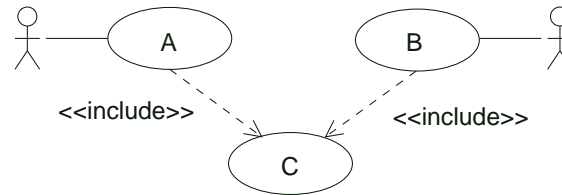
Kunde nicht vorhanden

## Aktivitätsdiagramm:



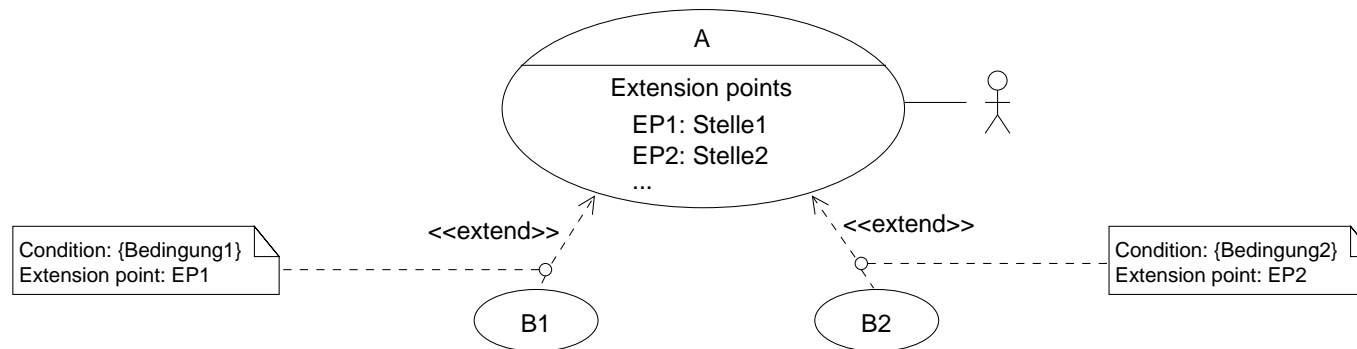
### 3.1.3 Beziehungen zwischen Anwendungsfällen

#### 1. Enthält-Beziehung



Jeder Ablauf von A bzw. B beinhaltet als Teilablauf (notwendigerweise) einen Ablauf von C.

#### 2. Erweiterungsbeziehung



Erweitert A um zusätzlich mögliches Verhalten, falls die angegebene Bedingung erfüllt ist.

### 3.1.4 Beispiel ATM (Automatic Teller Machine)

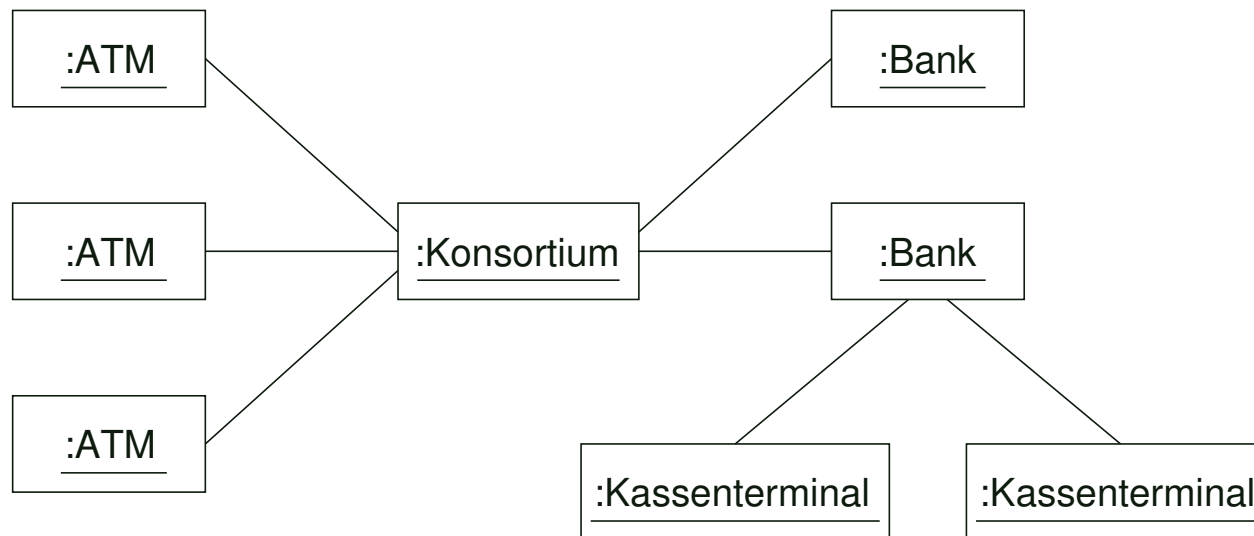
*Problembeschreibung: Netzwerk von Bankautomaten*  
(aus Rumbaugh et al., 1993)

Entwickelt werden soll Software zur Unterstützung eines rechnergesteuerten Bankennetzwerks einschließlich Kassierern und Bankautomaten (ATMs), das sich ein Bankenkonsortium teilt. Jede Bank besitzt einen eigenen Computer, auf dem sie ihre Konten verwaltet und die Transaktionen auf Konten durchführt. Die Banken besitzen Kassenterminals, die direkt mit dem bankeigenen Computer kommunizieren.

Kassierer geben Konto- und Transaktionsdaten ein. ATMs kommunizieren mit einem Zentralrechner, der Transaktionen mit den jeweiligen Banken abklärt. Ein ATM akzeptiert eine Scheckkarte, interagiert mit dem Benutzer, kommuniziert mit dem zentralen System, um die Transaktion auszuführen, gibt Bargeld aus und druckt Belege.

Das System erfordert geeignete Aufzeichnungsmöglichkeiten und Sicherheitsmaßnahmen. Das System muss parallele Zugriffe auf das gleiche Konto korrekt abwickeln. Die Banken stellen die SW für ihre eigenen Computer selbst bereit.

Sie sollen die Software für die ATMs und das Netzwerk entwickeln. Die Kosten des gemeinsamen Systems werden nach Zahl der Scheckkarteninhaber auf die Banken umgelegt.



## *Vorgehensweise*

1. *Aktoren*: Kunde, Kassierer, Manager, ...

2. *Use Cases*:

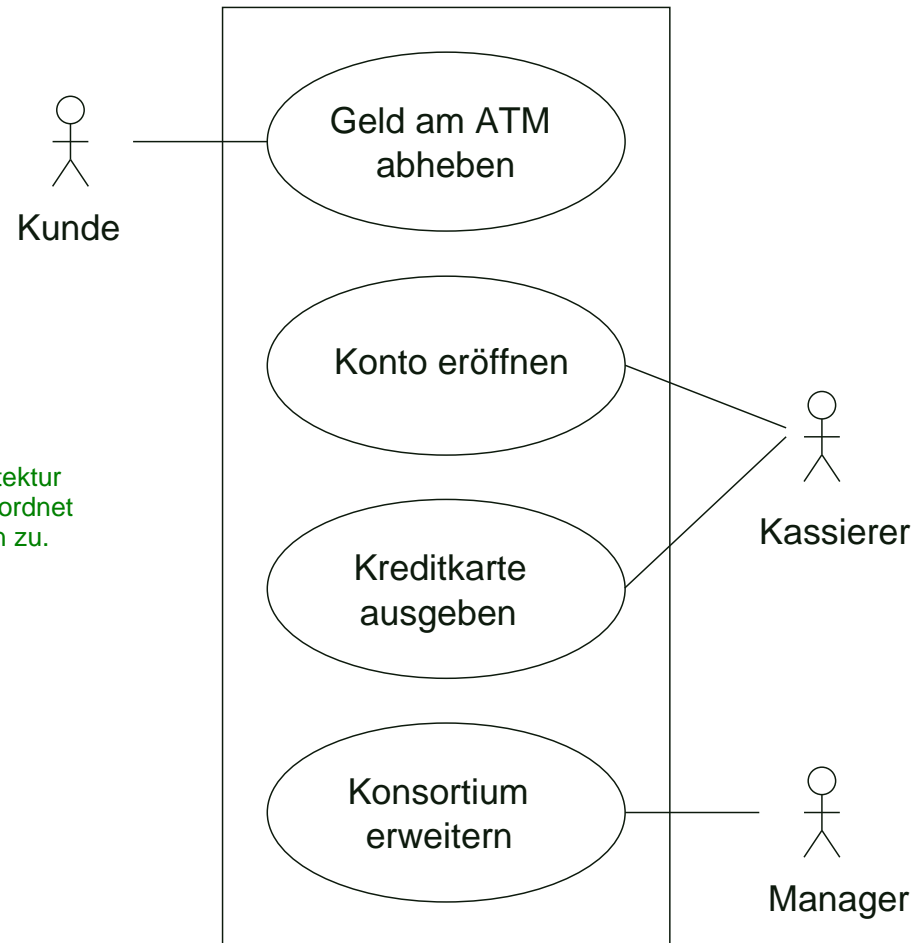
- Geld am ATM abheben:  
Ein Benutzer hebt am Geldautomaten mit Hilfe seiner Kreditkarte Geld von seinem Konto ab.
- Konto eröffnen:  
Ein Kassierer richtet ein neues Konto für einen Kunden ein.
- Neue Kreditkarte ausgeben:  
Ein Kassierer gibt eine neue Kreditkarte für einen Kunden aus.
- Konsortium erweitern:  
Ein Manager des Konsortiums nimmt eine Bank in das Konsortium auf.

etc. für weitere Anwendungsfälle wie z.B. "Geld einzahlen", "Überweisung durchführen".

### 3. Use Case-Diagramm (Ausschnitt):

**Bemerkung:**

Häufig erstellt man zunächst eine Grobarchitektur (z.B. in Form eines Kontextdiagramms) und ordnet die Use Cases den passenden Subsystemen zu.



#### 4. *Use Case Beschreibungen:*

**Anwendungsfall:**

Geld am ATM abheben.

**Kurzbeschreibung:**

Ein Benutzer hebt am Geldautomaten mit Hilfe seiner Kreditkarte Geld von seinem Konto ab.

**Vorbedingung:**

Das ATM ist bereit für einen Benutzer eine Transaktion durchzuführen.

**Nachbedingung:**

Der Benutzer hat die Kreditkarte, das Geld und den Beleg entnommen. Das ATM ist erneut bereit eine Transaktion durchzuführen.

## Primärszenario (Standardablauf):

1. Der Kunde gibt seine Kreditkarte ein.
2. Das ATM liest die Kreditkarte und fordert daraufhin die Geheimzahl an.
3. Der Benutzer gibt die Geheimzahl ein.
4. Das ATM liest die Geheimzahl, überprüft sie und lässt dann die BLZ und die Kartenummer beim Konsortium überprüfen.
5. Das Konsortium überprüft die BLZ, gleicht die Kartenummer mit der Bank des Kunden ab und gibt dem ATM sein OK.
6. Das ATM fordert den Benutzer auf die Transaktionsform (Abhebung, Einzahlung, Überweisung, Kontoauszug) zu wählen.
7. Der Benutzer wählt "Abhebung", woraufhin das ATM den Betrag erfragt.
8. Der Benutzer gibt den gewünschten Betrag ein.
9. Das ATM liest den Betrag, überprüft, ob er innerhalb vordefinierter Grenzen liegt, und fordert dann das Konsortium auf die Transaktion zu verarbeiten.
10. Das Konsortium leitet die Anforderung an die Bank weiter, die den Kontostand aktualisiert und die Ausführung bestätigt.
11. Das Konsortium teilt dem ATM den erfolgreichen Abschluss der Transaktion mit.
12. Das ATM gibt den gewünschten Betrag Bargeld aus und fordert den Benutzer auf es zu entnehmen.

13. Der Benutzer entnimmt das Bargeld, woraufhin das ATM fragt, ob der Benutzer eine weitere Transaktion durchführen will.
14. Der Benutzer verneint.
15. Das ATM druckt einen Beleg, gibt die Karte aus und fordert den Benutzer auf sie zu entnehmen.
16. Der Benutzer entnimmt den Beleg und die Karte.
17. Das ATM fordert den nächsten Benutzer auf eine Kreditkarte einzugeben.

## **Sekundärszenarien (abweichende Fälle):**

### **Karte gesperrt**

In Schritt 5, wenn die Bank feststellt, dass die Karte gesperrt ist, teilt sie dies dem Konsortium mit. Das Konsortium leitet die Nachricht an das ATM weiter. Das ATM meldet dem Benutzer den Fehler. Der Use Case wird dann bei Schritt 15 fortgesetzt.

### **Transaktion gescheitert**

In Schritt 10, wenn die Bank feststellt, dass der Kreditrahmen überschritten wird, teilt sie dem Konsortium mit, dass die Banktransaktion gescheitert ist. Das Konsortium leitet die Nachricht an das ATM weiter. Das ATM meldet dem Benutzer das Scheitern der Transaktion. Der Use Case wird ab Schritt 6 wiederholt.

## **Abbruch durch den Benutzer**

**Karte nicht lesbar**

**Falsche Geheimzahl**

**Falsche Bankleitzahl**

**Grenzen überschritten**

**Karte abgelaufen**

**Kein Geld im ATM**

**Netzwerk unterbrochen**

## Zusammenfassung von Abschnitt 3.1

- Ein Use Case-Modell besteht aus Anwendungsfällen und Aktoren.
- Ein Aktor tauscht Informationen mit dem System aus.
- Ein Anwendungsfall beschreibt eine Aufgabe, die das System für einen oder mehrere Aktoren durchführen soll.
- Eine Anwendungsfall-Beschreibung beinhaltet u.a. ein Primärszenario und mehrere Sekundärszenarien.
- Anwendungsfälle können mit <<include>> und <<extend>>-Beziehungen strukturiert werden.
- Ein Use Case-Modell wird schrittweise erstellt.