

# 12. Erste Schritte in der Objektorientierte Analyse mit CRC-Karten

**Bitte selbständig studieren!**  
**Begleitmaterial zur Übung 2**

Prof. Dr. rer. nat. Uwe Aßmann  
Lehrstuhl Softwaretechnologie (ST)  
Institut für Software- und  
Multimediatechnik (SMT)  
Fakultät für Informatik  
TU Dresden  
Version 13-1.0, 13.04.13

- 1) CRC-Analyse
- 2) CRC mit Gruppen
- 3) CRRC-Analyse



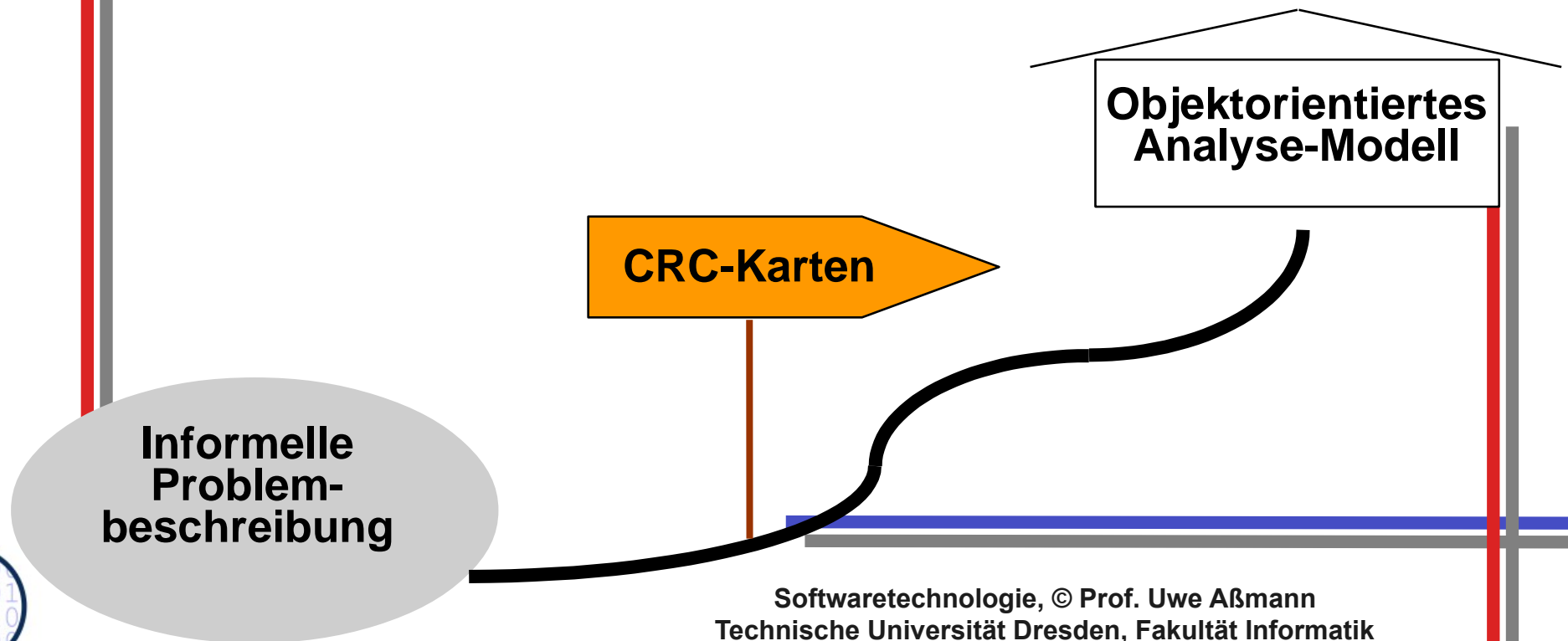
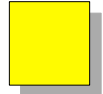
## ▶ Obligatorische Literatur

- Zuser Kap 9
- Beck, Kent; Cunningham, Ward (October 1989), "A laboratory for teaching object oriented thinking", ACM SIGPLAN Notices (New York, NY, USA: ACM) 24 (10): 1–6, <http://c2.com/doc/oopsla89/paper.html>

## ▶ Weiterführende Literatur

- HotDraw CRC cards <http://c2.com/doc/crc/draw.html>
- Scott Ambler. The Object Primer. Cambridge University Press. Gutes Kapitel über CRC

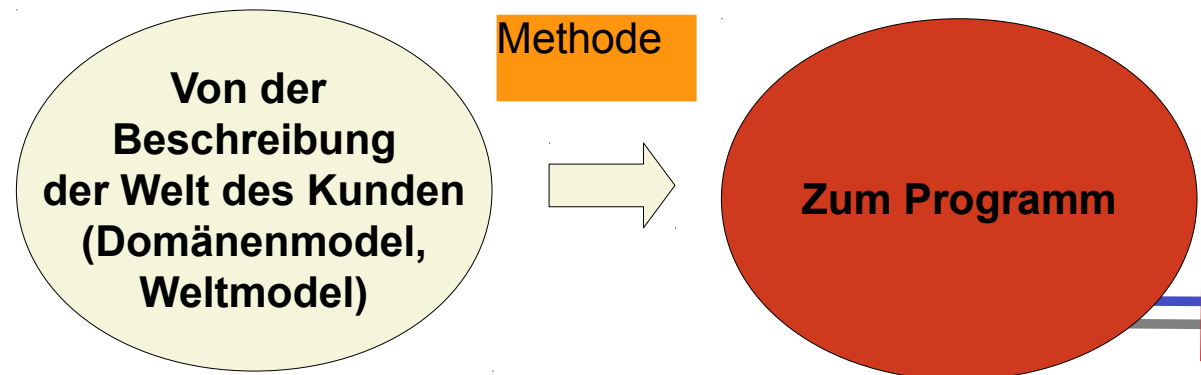
# 12.1 Analysemethoden: Analyse mit CRC-Karten



# Die Rolle von Methoden in der Softwareentwicklung

- ▶ Um ein System zu entwickeln, sollte man sich an eine *Methode* halten, die einem durch alle Schritte leitet
  - Eine Methode beruht auf einer oder mehreren zentralen Fragen, die immer wieder gestellt werden
  - Sie werden im Laufe Ihres Lebens verschiedene Methoden kennenlernen
  - Analysemethode – Entwurfsmethode - Implementierungsmethode

Wie kommen wir vom Problem des Kunden zum Programm (oder Produkt)?



## **Welche Objekte (Klassen) enthält ein System? Welche Verantwortlichkeiten haben sie?**

- ▶ Vom objektorientierten Entwurf existieren einige Spielarten, die zusätzliche Hilfsmittel einsetzen.
  - CRC-Karten
  - Strukturgetriebener Klassenentwurf (z.B. nach Balzert, später)

# Analyse mit CRC-Karten

- ▶ **CRC** = **C**lass – **R**esponsibility – **C**ollaborator  
(Klasse – Verantwortlichkeit – Mithelfer)
- ▶ [Beck, Cunningham, Wilkerson, Wirfs-Brock (ca. 1989-1995)]
- ▶ Technik zur Gruppenarbeit (Rollenspiele)
- ▶ Wichtigstes Hilfsmittel: Zu beschriftende Karteikarten

(Vorderseite)

<b>Klassenname (class)</b>	
<b>Ober- und Unterklassen (opt.)</b>	
<b>Verantwortlichkeiten</b>  (responsibilities)	<b>Mithelfer</b>  (collaborators)

(Rückseite)

<b>Klassenname (class)</b>
<b>Definition</b>
<b>Attribute</b>  (attributes)

# Class Responsibility Cards (CRC)

**Welches Objekt ist für welche Aufgaben zuständig?**

## Zuständigkeit für Aufgabe (Dienst):

- Aktion
- Auskunft (query)
- Prüfung (check)

## Kooperation mit Partner

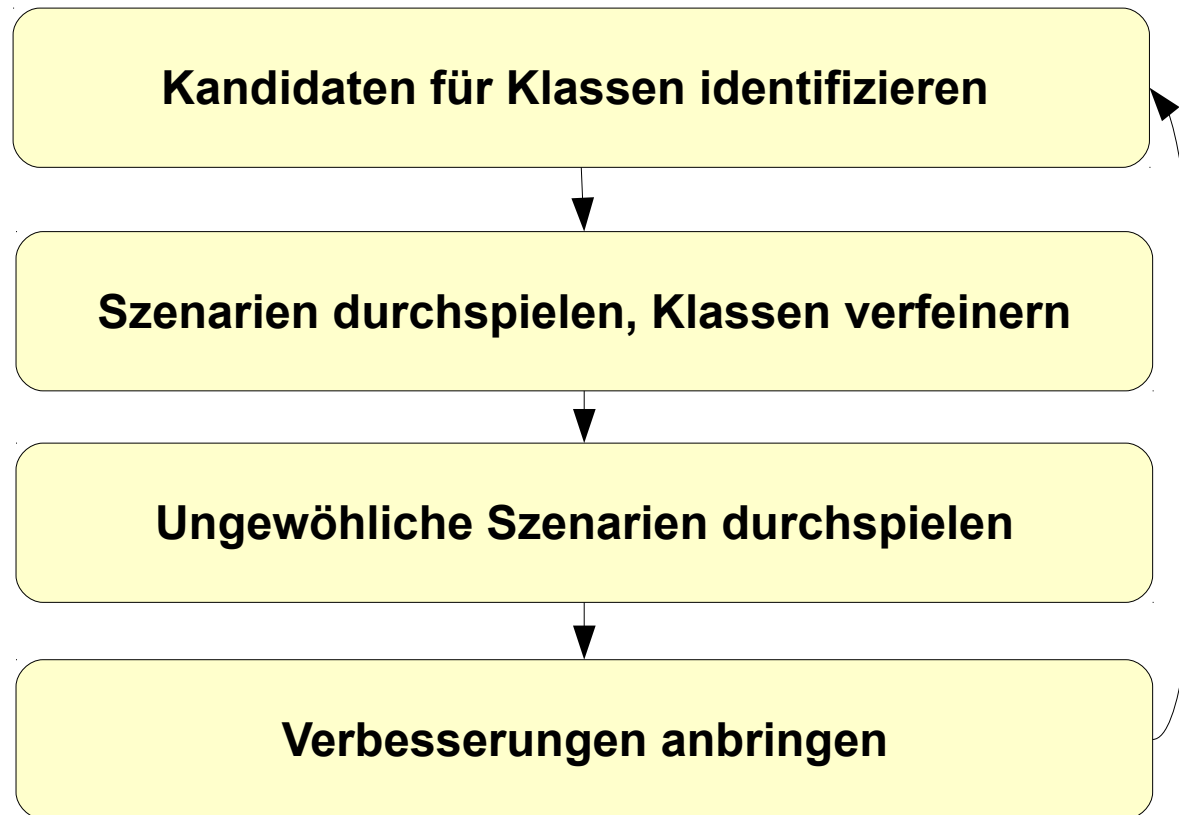
- Wenn Klasse alleine zur Bewältigung der Aufgaben nicht fähig
- "Mit wem muss ich kooperieren, um einen Dienst zu erhalten?"

(Vorderseite)

<b>Klassenname (class)</b>	
<b>Ober- und Unterklassen</b>	
<b>Verantwortlichkeiten</b> <b>(responsibilities)</b>	<b>Mithelfer</b> <b>(collaborators)</b>

# CRC-Karten-Methode: Vorgehensweise als Einzelner

- ▶ Voraussetzung: informelle Anforderungsbeschreibung (ideal: ausführliche Anforderungsspezifikation)



**Gesetz 51 (PP): Nicht Anforderungen sammeln, sondern nach ihnen schürfen**



- ▶ Ein **Szenario** ist ein typischer Ablauf von Aktionen zur Erfüllung des geplanten Systemzwecks.
  - z.B. notwendige Schritte zur Einrichtung einer neuen Teambesprechung:
    - Festlegung des Titels
    - Festlegung der Teilnehmer
    - Festlegung eines ersten Terminvorschlags und Abstimmung
    - Festlegung eines Besprechungsraums
  - Viele Szenarien zur Beschreibung eines Teilaspekts des Systemzwecks.
- ▶ Szenarien werden interaktiv "durchgespielt". Man stellt sich die Entwicklungsfragen der CRC-Methode:
  - Wer ist zuständig (Verantwortungsbereich)?
  - Welche Aufgaben sind dazu zu erfüllen? (auf Karte eintragen)
  - Welches Privatwissen ist dazu nötig? (auf Karte eintragen)

# Identifikation von Klassen-Kandidaten: Substantiv-Verb-Analyse

- ▶ Analysiere textuelle Anforderungen:
  - Finde Objekte aus Hauptwörtern
  - Finde Kooperationen aus Subjekt-Objekt-Relationen, Genitiven, Nebensätzen
  - Finde Aktivitäten aus Verben und substantivierten Verben
  - Finde Kontextklassen durch Orte und adverbiale Bestimmungen

**Finde Aufgaben aus Verben, Klassen aus Hauptwörtern**

- ▶ Beispiel
  - "When the **student** **orders** a **computer**, he has to **contact** the **computer dealer**. The **computer dealer** **ships** the **packet** *via the parcel service*."
  - "When the **driver** turns on the **lights** the **battery** is discharged. When the **engine** **runs** the **dynamo** **recharges** the **battery**..."

# Kriterien für Klassen in der Substantiv-Verb-Analyse

- ▶ Problemrelevante Substantive auswählen
  - Allgemeine Worte weglassen (z.B. "System")
  - Auch "versteckte" Substantive betrachten (z.B. "Privattermin")
- ▶ Hat jede Klasse einen klar abgegrenzten Verantwortungsbereich?
  - Gibt es Aufgaben, die spezifisch für die Objekte der Klasse sind?
  - Passen die Aufgaben zusammen?
  - Gibt es "Privatwissen", das ein Objekt der Klasse besitzt?
- ▶ Sind Verantwortungsbereiche von mehreren Klassen abgedeckt?
- ▶ "Haben wir alles?"
  - Gibt es nicht als Substantive erwähnte wichtige "Mitspieler"?

# Weitere Regeln für das Ausfüllen von CRC-Karten

- ▶ Verantwortlichkeiten:
  - Eine Verantwortlichkeit enthält fast immer ein Zeitwort.
  - '... wissen' kann auch eine Verantwortlichkeit sein.
- ▶ Mithelfer:
  - Mithelfer-Einträge nur eintragen, wenn Kommunikation mit anderen Objekten notwendig.
  - Eine Verantwortlichkeit kann mehrere Mithelfer benötigen.
  - Die Rückgabe einer Antwort gehört zu einem normalen Kommunikationsvorgang - nicht als Verantwortlichkeit nennen.
- ▶ Karten-Rückseiten:
  - Definitionen am besten vor dem Spiel ausfüllen, später überprüfen.
  - Attribute können während des Spiels oder später ausgefüllt werden.

# Klassen finden: Beispiel Terminverwaltung

## *Problembeschreibung:*

Es ist ein Terminverwaltungssystem für Arbeitsgruppen zu entwickeln.

Das System soll alle geplanten Teambesprechungen (z.B. Projektbesprechungen) speichern und die Reservierung von Besprechungsräumen unterstützen.

Das System soll automatisch Kollisionen mit bereits bekannten Terminen vermeiden. Deshalb soll auch die Eintragung privater Termine möglich sein.

# Beispiel einer CRC-Karte

**Teambesprechung**

**Oberklassen:** Termin  
**Unterklassen:**

**Titel wissen**  
**Datum wissen**  
**Teilnehmer wissen**  
**Teilnehmer einladen**  
**Raum festlegen**

**Teammitglied**  
**Besprechungsraum**

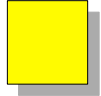
**Teambesprechung**

**Ein Objekt 'Teambesprechung' beschreibt genau einen Termin, an dem mehrere Teilnehmer der Gruppe teilnehmen sollen.**

**Titel**  
**Datum**

***Rückseite:***

## 12.2. CRC in Gruppen



# CRC-Karten-Methode: Vorgehensweise in Gruppe

- ▶ Das Kernstück der Methode sind intensive Gruppensitzungen.
- ▶ Voraussetzung: informelle Anforderungsbeschreibung (ideal: ausführliche Anforderungsspezifikation)
  - Kandidaten für Klassen (Karten) identifizieren
  - Karten auf einem Tisch oder Whiteboard plazieren.
    - Kollaborierende Karten nah zueinander plazieren, andere voneinander entfernen. Anordnung ständig ändern, je mehr Zusammenarbeiten zustandekommen
    - “Heisse” Karten in die Mitte des Tisches
  - Typische Szenarien identifizieren und durchspielen (dabei: Karten schrittweise ausfüllen)
  - Iteration: Verbesserungen, mehrfache Wiederholung
  - Ungewöhnliche Szenarien durchspielen

**Gesetz 17 (PP): Programmieren Sie nahe am Problem  
(an der Sprache der Anwender)**





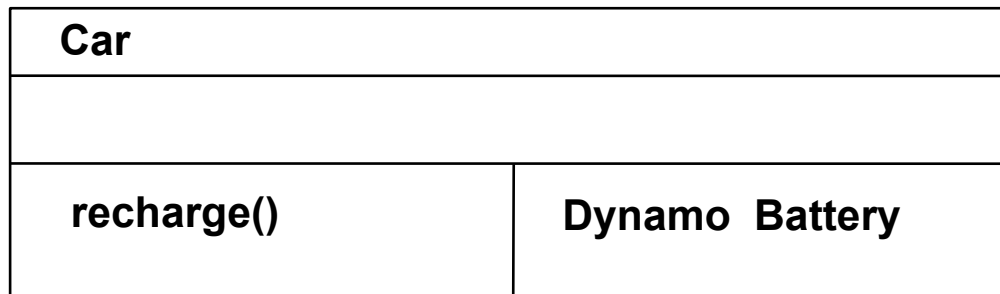
# Gruppenspiel

- ▶ Ideale Gruppengröße: 5 bis 6 aktive TeilnehmerInnen
- ▶ Teilnehmer(Innen):
  - Fachspezialisten, ev. Kunden
  - Systemanalytiker
  - Systementwickler
  - Manager (?)
  - Moderator, 'Facilitator'
- ▶ Gruppendynamik:
  - CRC-Karten-Sitzungen können Teamgeist stärken
  - Vorhandene Gruppen-Probleme können aufbrechen
  - **Kein** Mittel zur Klärung und Lösung von Problemen im Team !

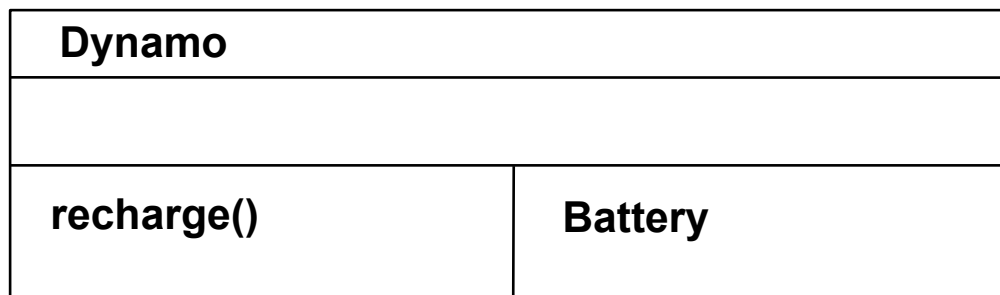
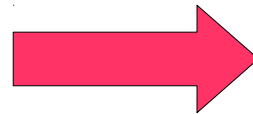
**Gesetz 52 (PP): Arbeiten Sie mit Anwendern zusammen, damit Sie denken wie ein Anwender**

# Vorsicht: Klassen des Kontexts

- ▶ Oft werden Verantwortlichkeiten fälschlicherweise einer Klasse zugeordnet, die eigentlich eine Klasse im *Kontext* ist
- ▶ Achte auf Ortsbestimmungen und Urheberschaften
- ▶ “the car recharges the battery *via the dynamo*”

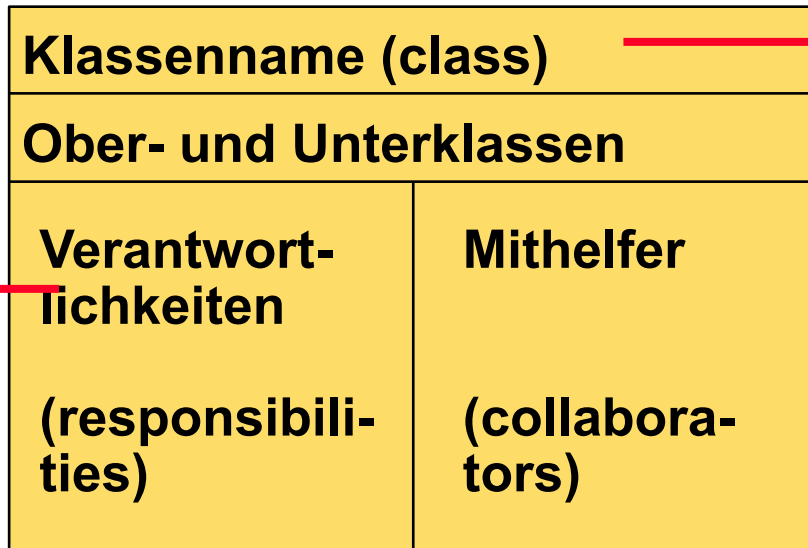


**Car ist eine Klasse im  
*Kontext* von Dynamo**

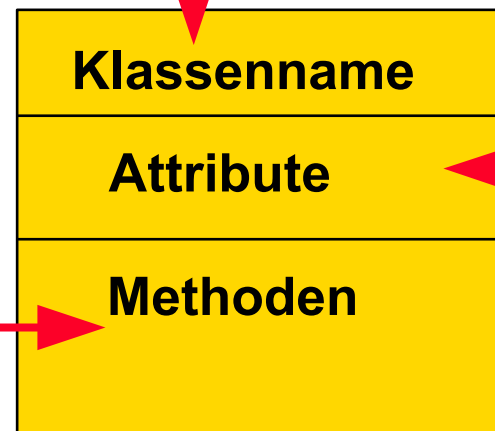
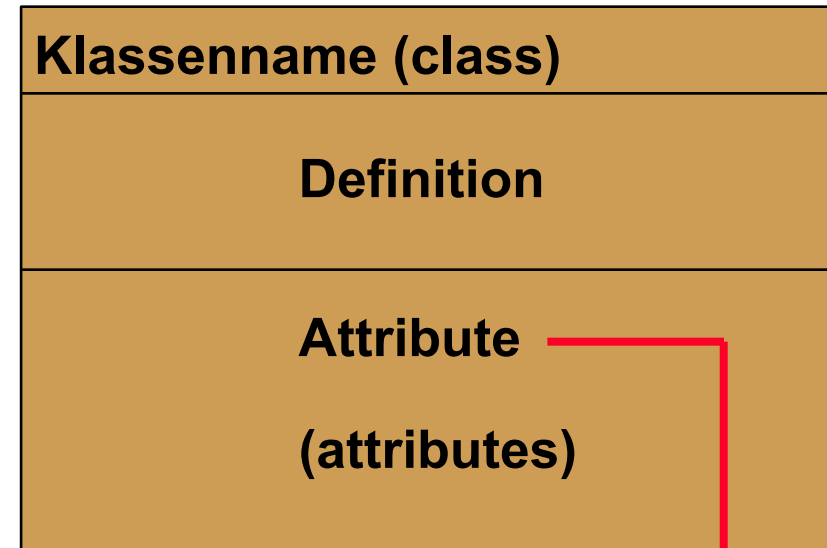


# Von CRC-Karten zum UML-Klassenmodell (1)

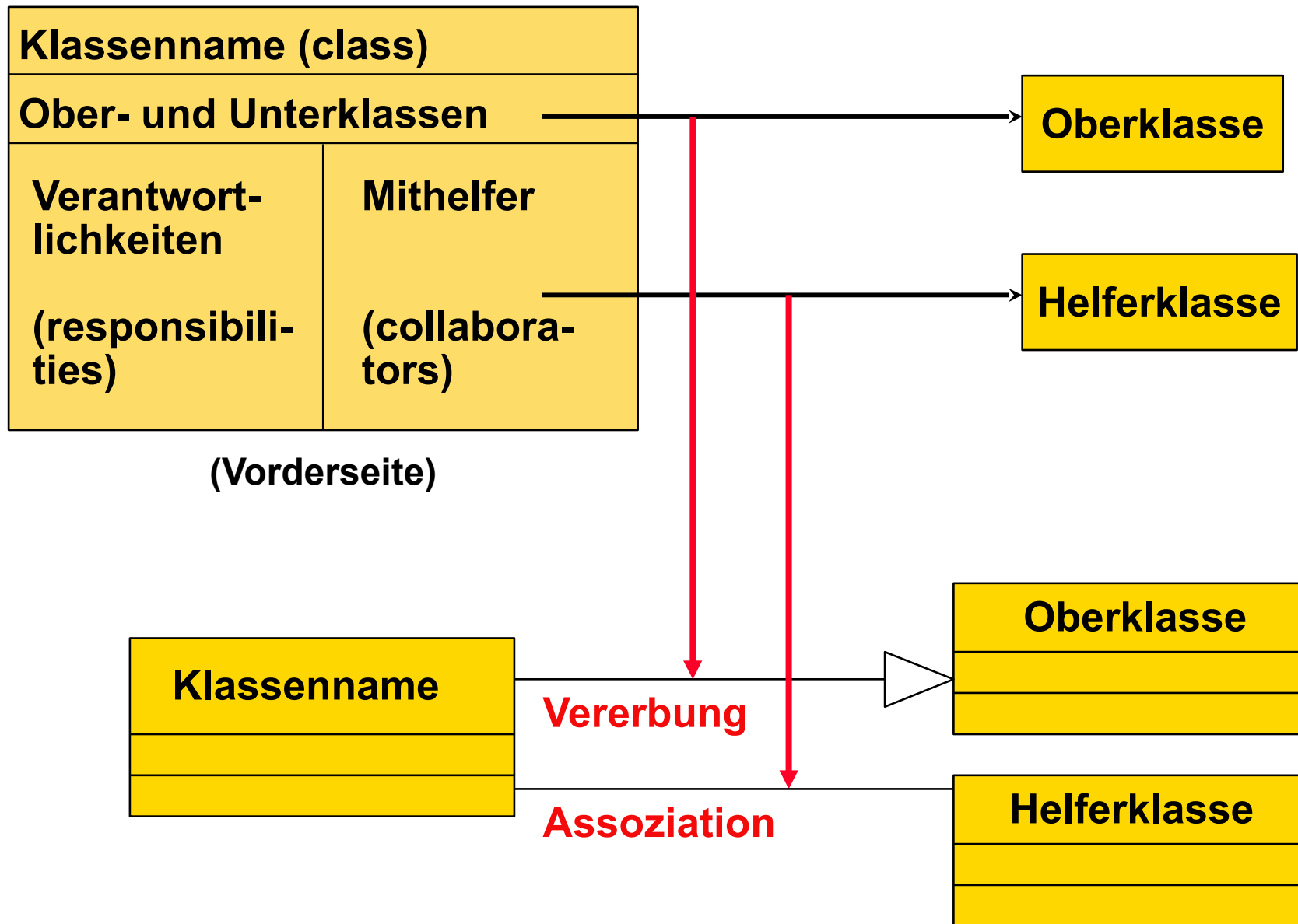
(Vorderseite)

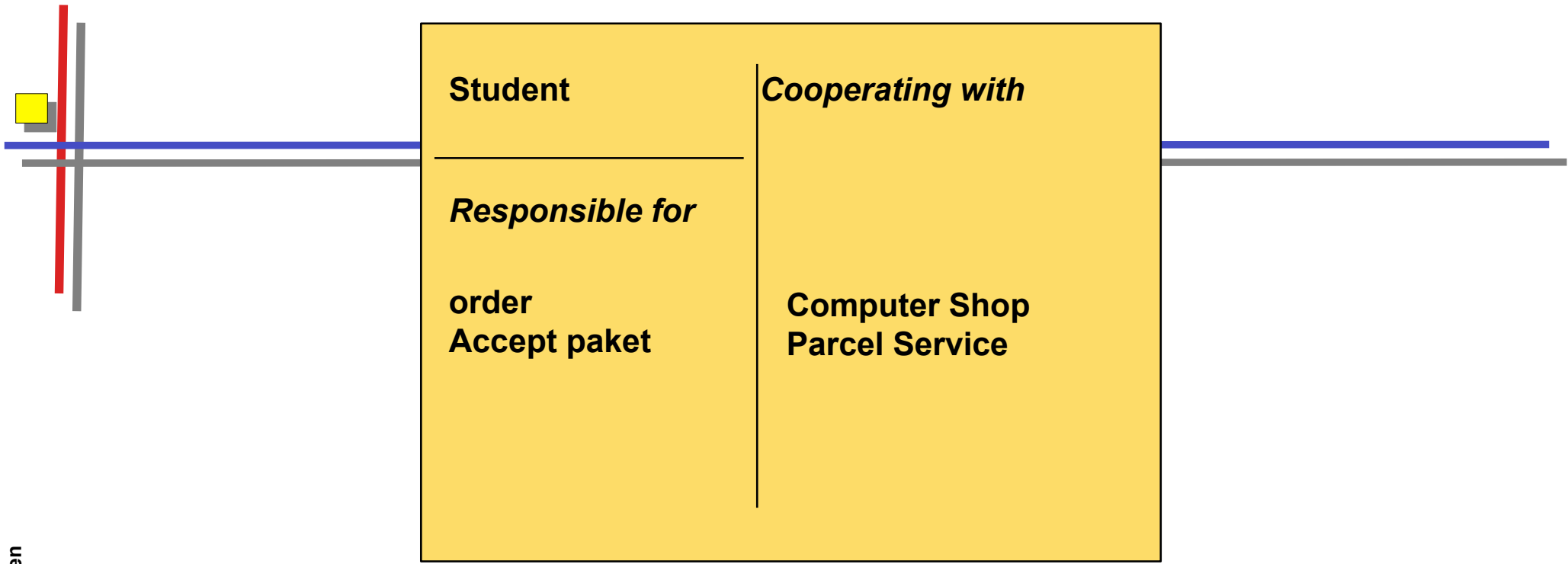


(Rückseite)

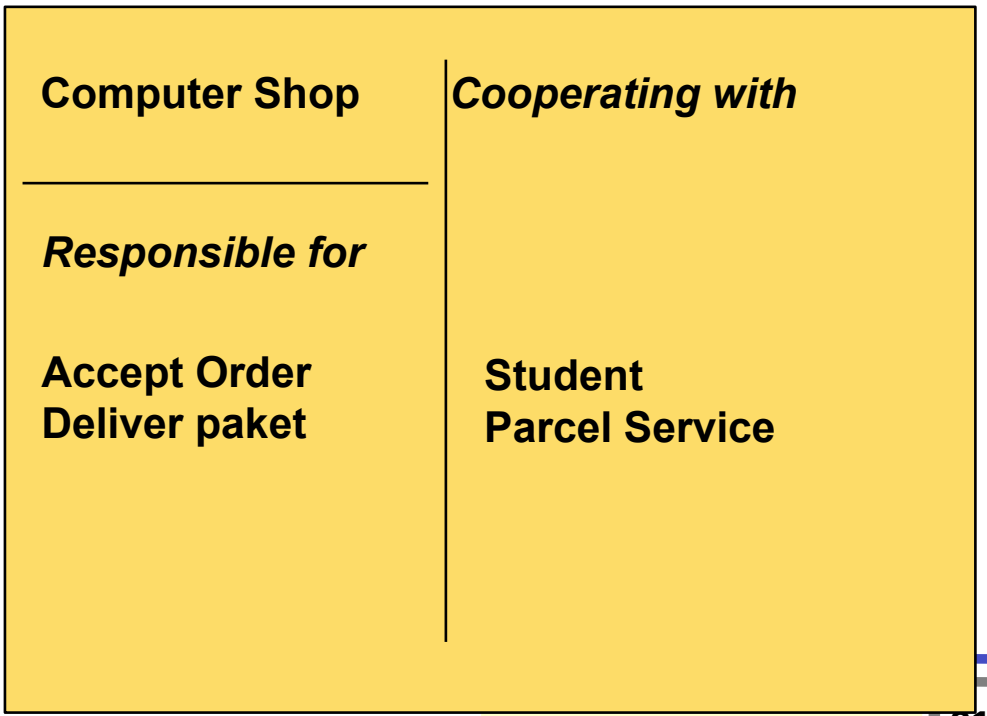
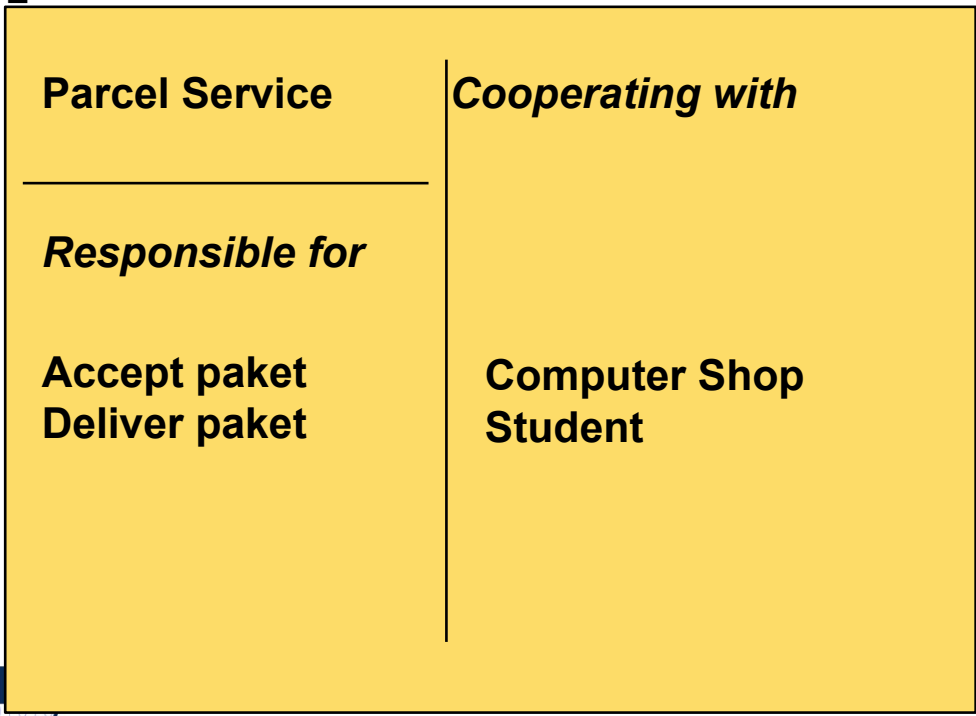


# Von CRC-Karten zum UML-Klassenmodell (2)

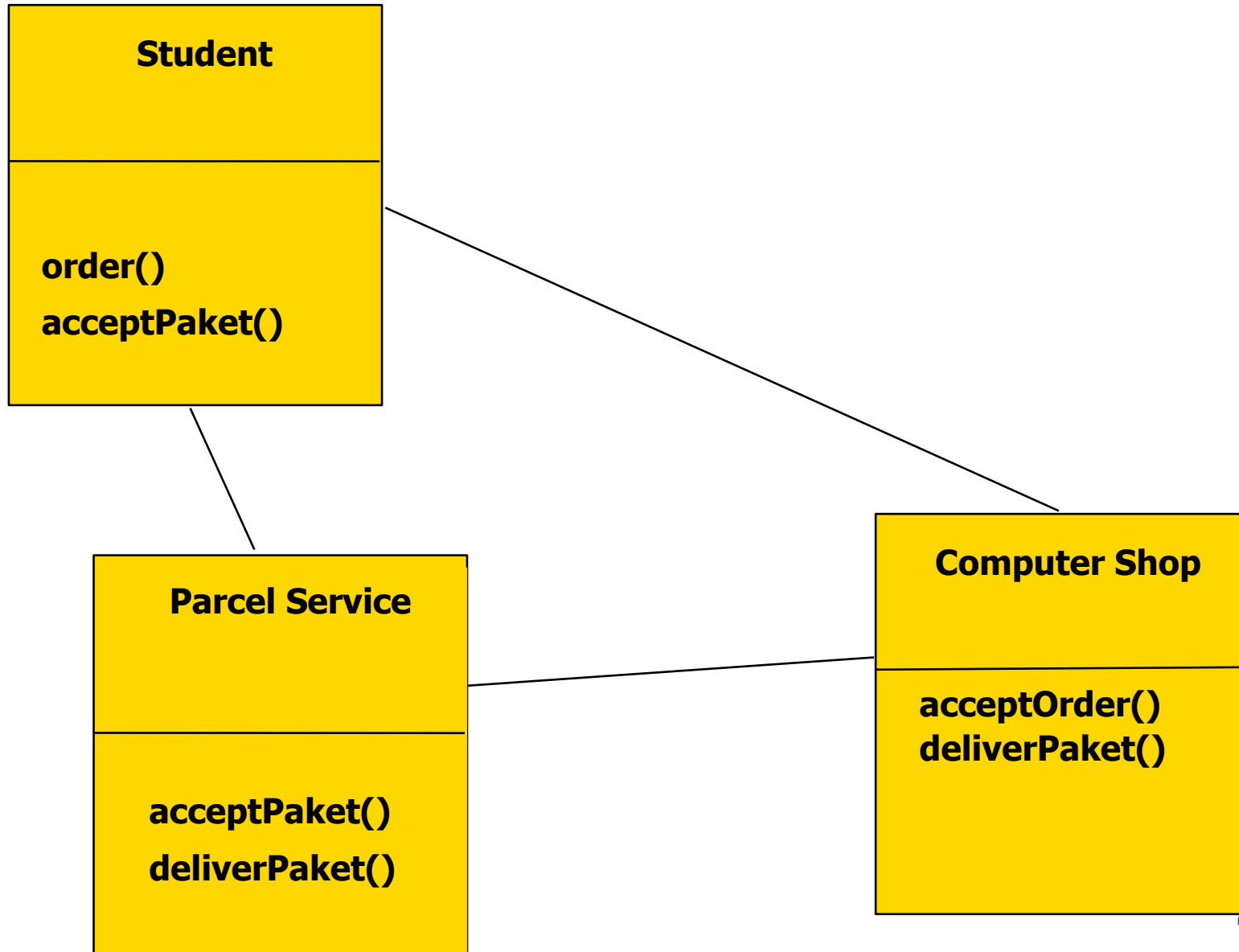




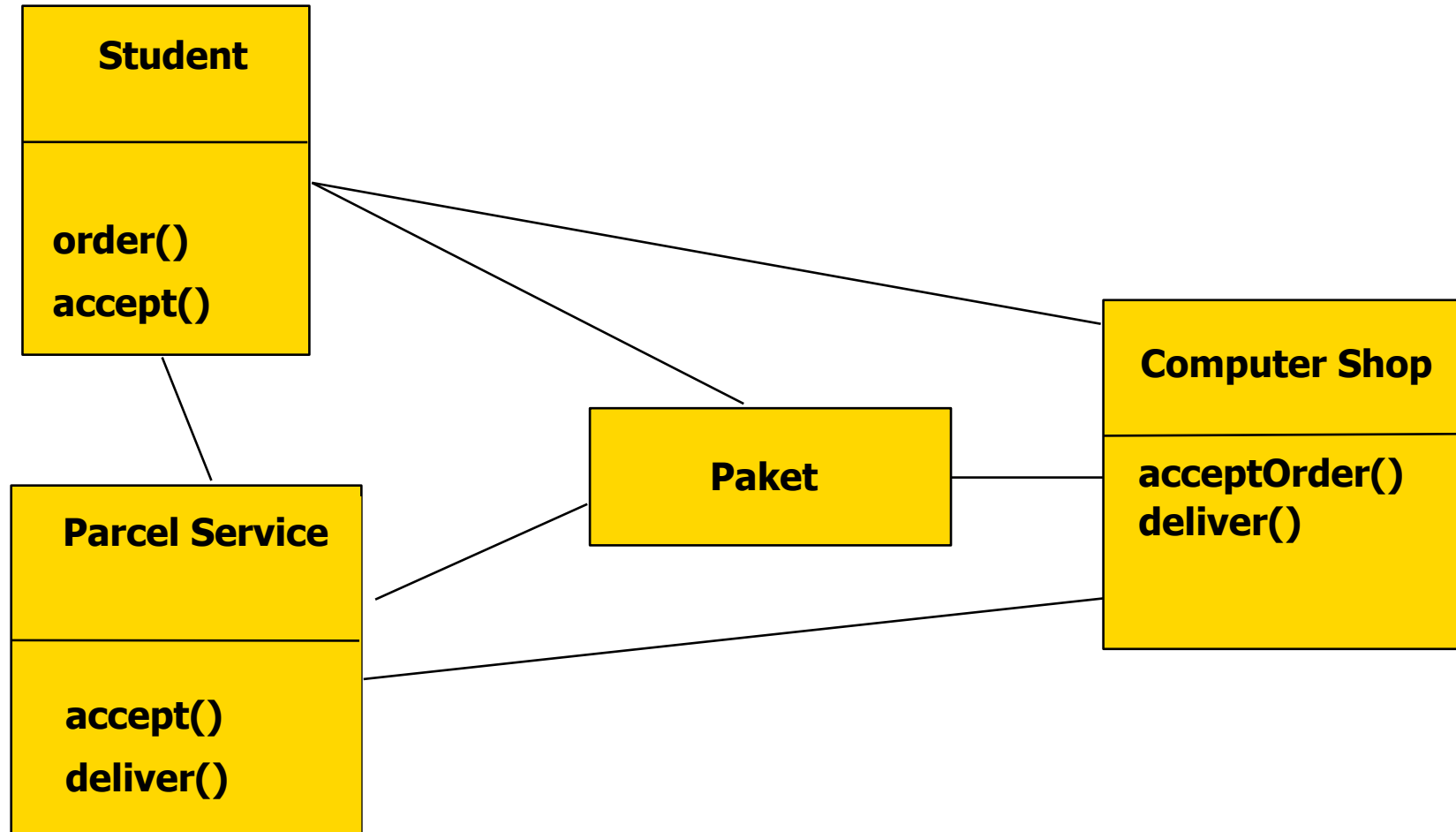
Dresden



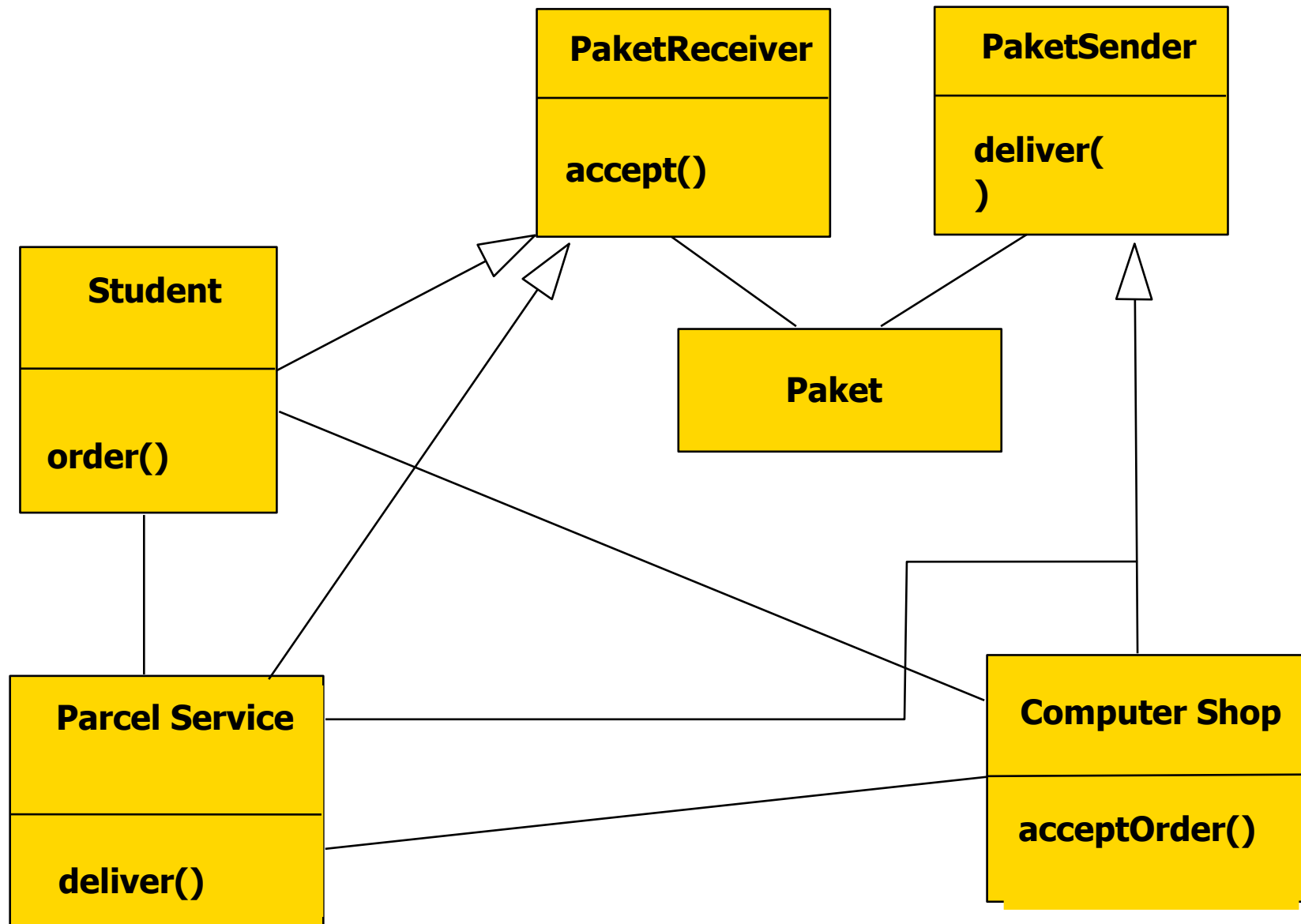
# Ein erstes Klassendiagramm



# Verfeinerung: Pakete als Objekte



# Verfeinerung: Ausfaktorisieren von Gemeinsamkeiten





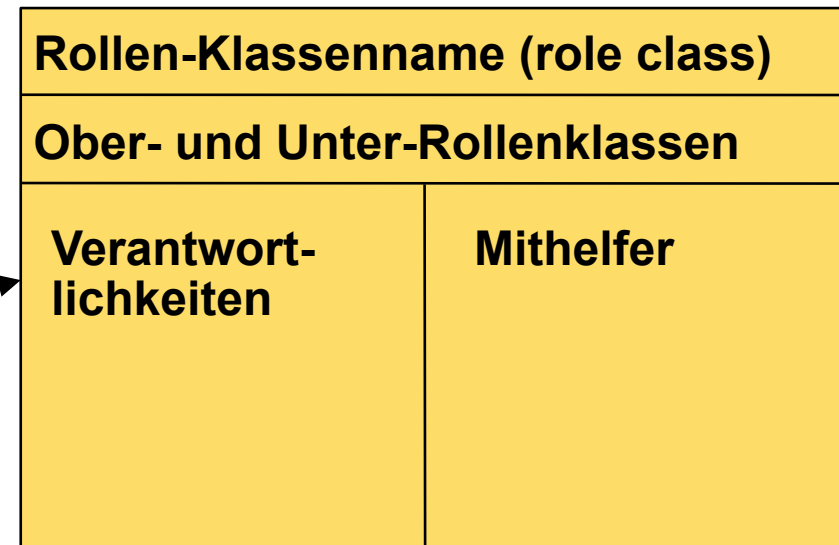
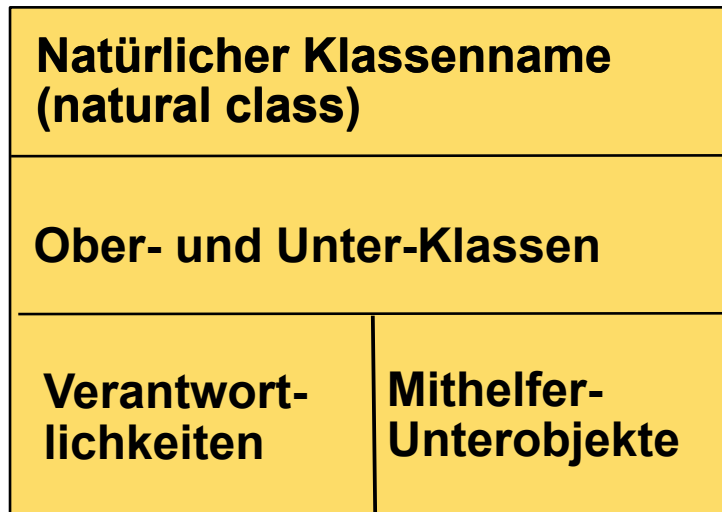
# 12.3 CRRC-Analyse



# Class-Role Responsibility Cards (CRRC)

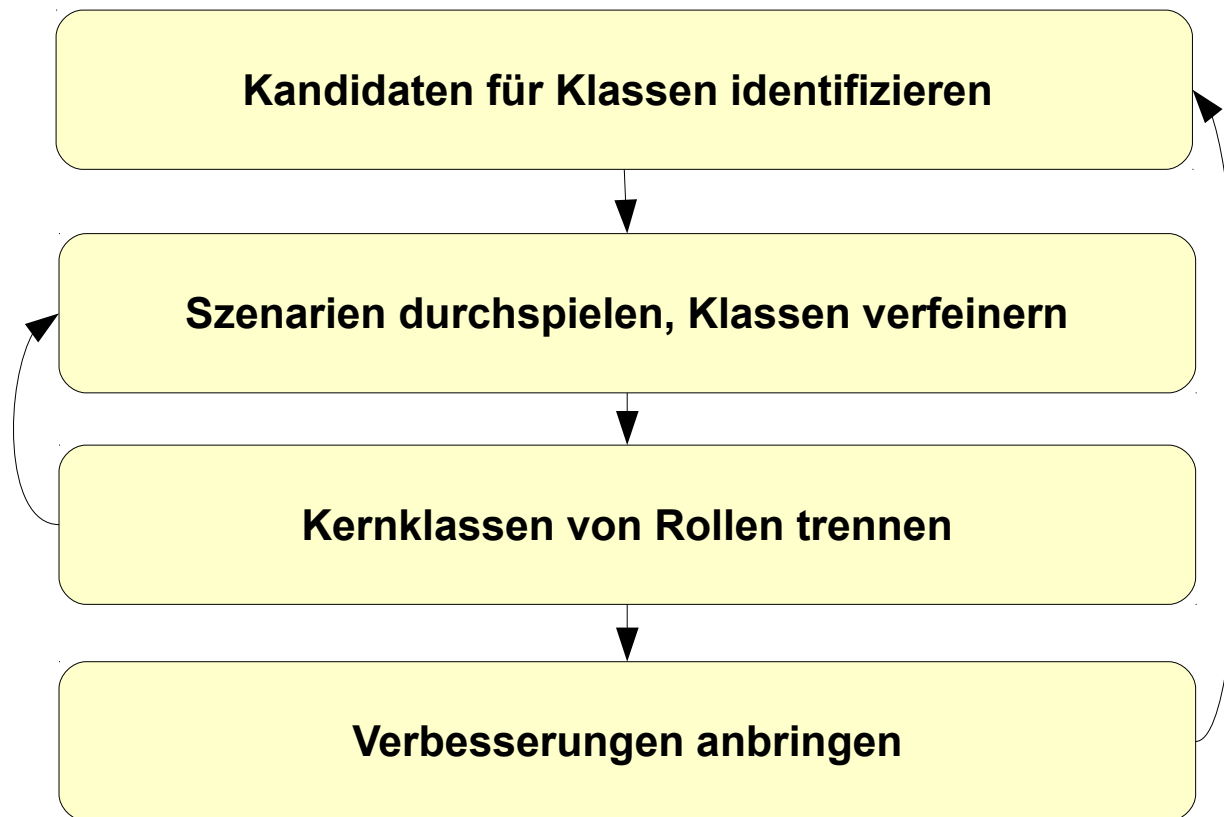
**Welches Objekt ist in welchen Rollen für welche Aufgaben zuständig?**

- ▶ Unterscheide natürliche und Rollen-Typen
- ▶ Kernklassen und Rollen-Klassen



# CRRC-Karten-Methode: Vorgehensweise

- ▶ Wichtig: Kerne von Unterobjekten durch Delegation trennen



# Was haben wir gelernt?

- ▶ CRC-Karten dienen als Mittel, mit Gedankensturm (brainstorming) die Klassen und ihre Zuständigkeiten herauszufinden.
- ▶ Achte auf den korrekten Kontext von Klassen
- ▶ Gruppenspiele dienen zum iterativen, reflektiven Finden von Klassen und Zuständigkeiten
- ▶ CRRC-Karten helfen Kerne von Rollen zu trennen

# The End