

## Methoden der SE: Business Object Model

- Ein BOM erstellen können
- Den Zusammenhang von BOM und Datenbankdefinition erkennen
- Beziehungen zw. Objekten erkennen und exakt definieren können
- Den Unterschied zw. Assoziation, Aggregation und Komposition erkennen
- Navigierbarkeit und Kardinalität richtig wählen können
- Die Generalisierung erkennen und darstellen können
- Java-Sprachelemente im BOM richtig darstellen können
- Die richtige Vorgangsweise bei der Erstellung eines BOM wählen
- Zu einem gegebenen Problem das BOM erstellen können

**Bis jetzt:** Die besprochenen Diagramme beschreiben Abläufe und Funktionalitäten

**Jetzt:** Beschreibung der zu berücksichtigenden Objekte mit ihren

- *Eigenschaften*,
- *Methoden* und
- *Beziehungen*

Welche Begriffe sind im IT-System wichtig, und was versteht man genau darunter



1. Ergänzen Sie noch mind. drei Objekte die mit den Nachfolgenden in Zusammenhang gebracht werden können

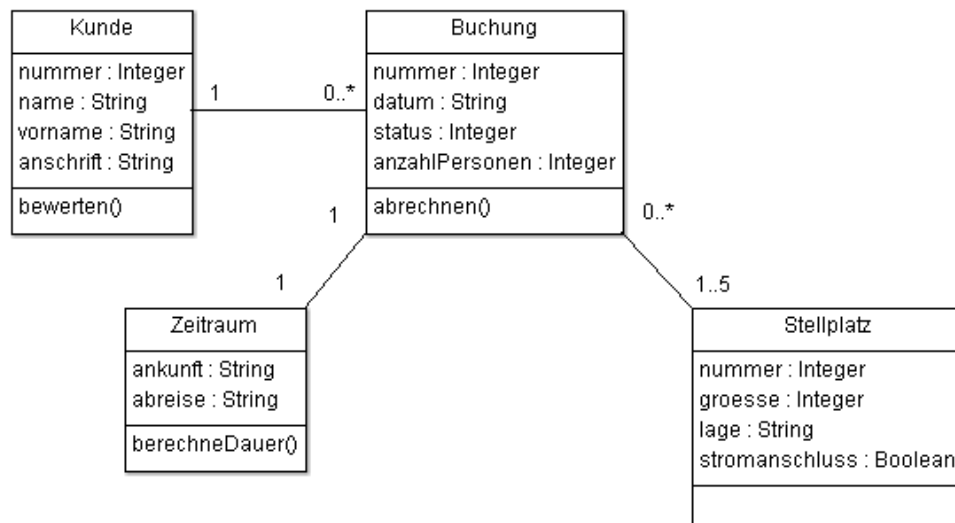
**Kunde, Auftrag, ...**

2. Welche Attribute zeichnen einen Kunden aus?

3. Beschreiben Sie die Beziehung zw. Kunde und Auftrag

- BOM wird auch als **Klassendiagramm** bezeichnet
- Aus BOM können erste Datenbankstrukturen abgeleitet werden
- Weiter Darstellungsmöglichkeit **Entity Relationship Diagramm**

### Beispiel BOM Campingplatz



Name
Attribute : Typ
Methoden()

## Eindeutiger Klassenname

### Attribute bzw. Membervariablen: „Wissen“ des Objektes

## Methoden: „Können“ des Objektes

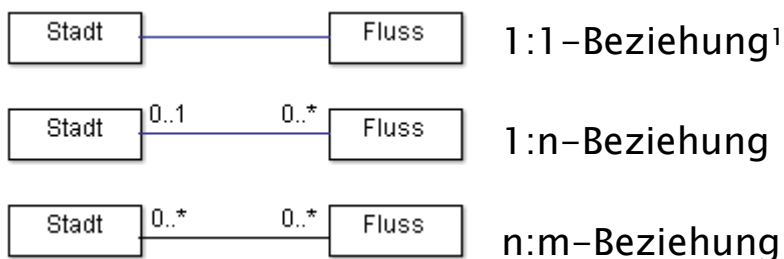
### Assoziationen mit Kardinalitäten

- Beschreiben die Beziehungen zw. Objekten
- Beziehungen zw. gleichberechtigten Klassen

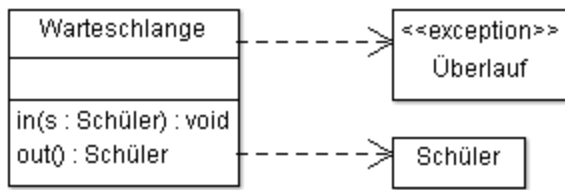
*Einer Buchung muss Kunde zugewiesen werden, Kunde kann auch ohne Buchung existieren*

n	genau n
n..m	eine Zahl zw. n und m
n, m	entweder n oder m
0..*	beliebig viele, auch Keines
1..*	beliebig viele, mindestens Eins

## Weitere Beispiele



<sup>1</sup> Keine Kardinalitätsangabe steht für 1 bzw. 0..1

**Benutztbeziehung**

Methode nutzt als Argument, Rückgabewert, Exception andere Klasse

**Navigierbarkeit**

```

public class Kunde
{
    private List buchungen = null;
}

public class Buchung
{
    private Kunde kunde = null;
}
  
```

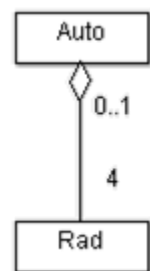
```

public class Kunde
{
    private List buchungen = null;
}

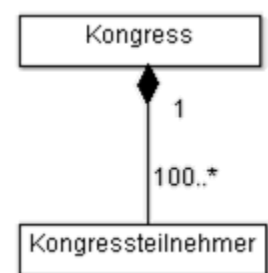
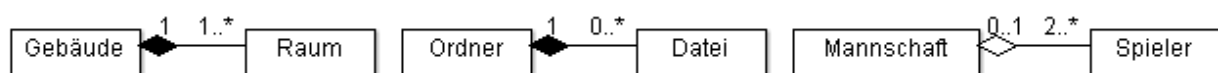
public class Buchung
{
}
  
```

**Aggregation**

- Besondere Art der Assoziation
- Beschreibt „hat-Teil“-Beziehung
- Teil kann aber auch ohne dem Ganzen existieren

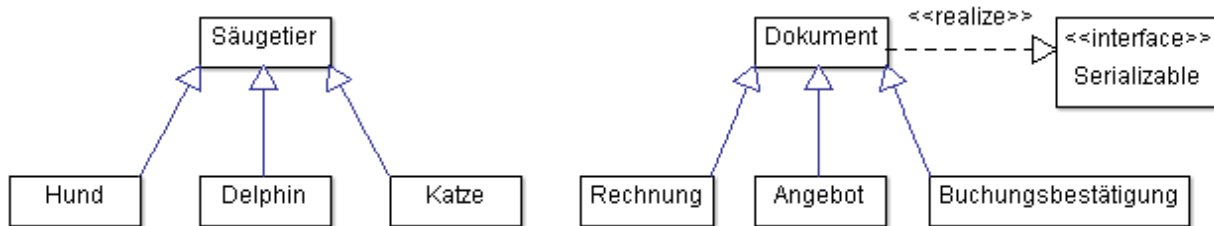
**Komposition**

- Starke Bindung der Objekte
- Objekt ist existenzabhängiger Teil eines Ganzen
- Kongressteilnehmer existiert nur wenn Kongress stattfindet
- Kongressteilnehmer stirbt wenn Kongress vorbei ist

**Weitere Beispiele**

<sup>2</sup> Darstellungsvariante verbirgt Eigenschaften und Methoden und stellt nur Beziehungen dar

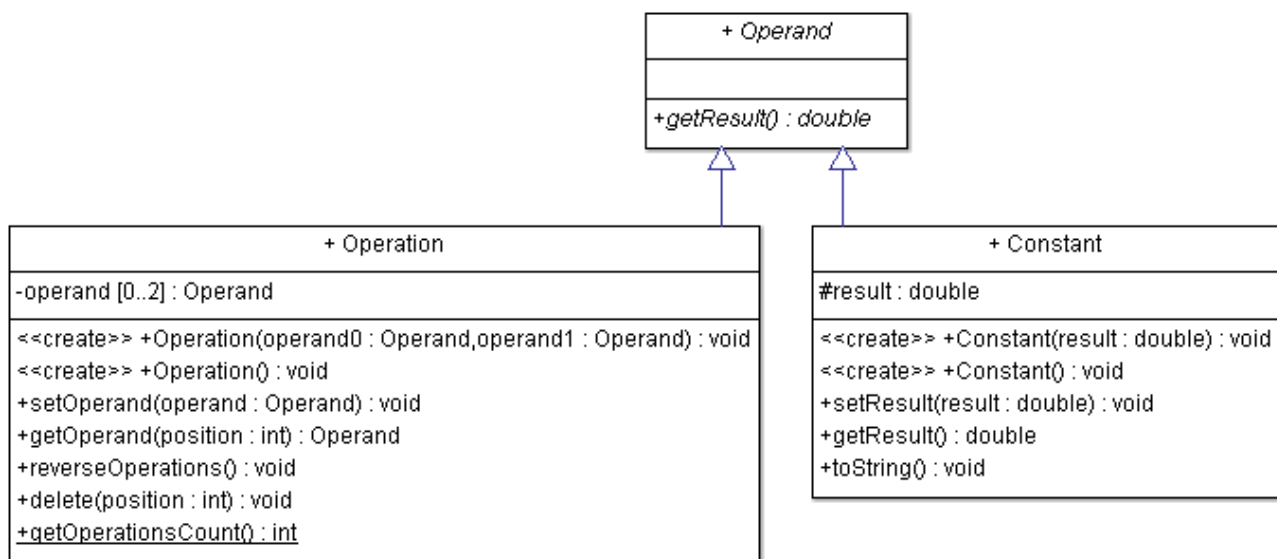
## Generalisierung und Realisierung



```
public class Dokument implements Serializable { ... }
```

```
public class Rechnung extends Dokument { ... }
```

## Darstellung anderer objektorientierter Konzepte



Sichtbarkeitsstufen – **private** **#** **protected** **+** **public** ~ **package**<sup>3</sup>

Konstruktoren<sup>4</sup> *Abstrakte Klassen*<sup>5</sup> Klassenmethoden<sup>6</sup>

## HINWEISE

- Assoziationen können auch über Attribute dargestellt werden (siehe operand[0..2]<sup>7</sup>: operand)
- ArgoUML kann aus Klassendiagramm automatisch C++, C#, Java, PHP generieren

<sup>3</sup> ArgoUML: Einstellbar bei den *Einstellungen* durch Option *Sichtbarkeit zeigen*

<sup>4</sup> ArgoUML: Einstellbar bei den *Stereotypen*

<sup>5</sup> ArgoUML: Einstellbar bei den *Eigenschaften* unter *modifiers* bei *isAbstract*

<sup>6</sup> ArgoUML: Einstellbar bei den *Eigenschaften* unter *modifiers* bei *ownerScope*

<sup>7</sup> ArgoUML: Einstellbar bei *Einstellungen* durch Option *Kardinalitäten von Attributen anzeigen*

## Vorgangsweise bei der Erstellung eines BOM

### Objekte entdecken

- Aus anderen Diagrammen erkennbar (z.B. Schnittstellen-, Use Case-Diagramm)
- Welche Begriffe aus dem Geschäftsleben sollen im IT-System wiederzufinden sein?

### Attribute u. Methoden zuordnen

- Jedes Objekt wird durch Eigenschaften und Methoden genauer beschrieben

### Assoziationen festlegen

- Wie stehen Objekte zueinander?
- Nur jene Assoziationen einzeichnen, die unbedingt zum klaren Verständnis benötigt werden
- Kardinalitäten nicht vergessen!

### Generalisierungen finden

- Gibt es Objekte die in ein gemeinsames generalisiertes Objekt zusammengefasst werden können?

- Schritte sollen nicht nur einmal durchlaufen werden
- BOMs zeigen meist nur wichtige Ausschnitte des IT-Systems
- BOM sollte nur die wesentlichen Attribute und Methoden enthalten (Getter- und Settermethoden)



In einem *Online-Shop* sollen beliebig viele Artikel angeboten werden, die von Kunden aus verschiedenen europäischen Ländern bestellt werden können.

Erstellen Sie anhand der folgenden Beschreibung das *Klassendiagramm*:

- Der Online-Shop besitzt in seinem Angebot eine Vielzahl von Artikeln. Der Kunde informiert sich über das Angebot des Shops und erhält eine Liste aller verfügbaren Artikel.
- Alle Artikel besitzen eine eindeutige Artikelnummer, einen Namen und einen Preis. Ein Artikel kann aus dem Angebot entfernt werden, ohne seine Daten aus dem System zu löschen.
- Ein Warenkorb kann leer sein, Artikel können in den Warenkorb geworfen und von diesem entfernt werden.
- Wenn der Wert eines Warenkorbes bzw. einer Bestellung eine festgelegte Grenze überschreitet, werden keine Verpackungs- und Versandkosten in Rechnung gestellt.
- Wenn ein Kunde es verlangt, kann er unterschiedliche Adressen für Rechnung und Lieferung angeben.
- Die Rechnungs- und Lieferanschrift bestehen aus Anrede, Vor- und Nachnamen, Straße mit Hausnummer, Postleitzahl und Ort. Da Bestellungen europaweit ausgeliefert werden, wählen Kunden ihr Land aus einer Liste. Für die Zustellung über einen Paketdienst ist es notwendig, dass Kunden auch ihre Telefonnummer angeben. Darüber hinaus müssen sie ihre Email-Adresse angeben, an die am Ende des Bestellvorgangs eine Bestätigungsmail geschickt wird.
- Verpackungs- und Versandkosten fallen je nach Land unterschiedlich hoch aus.

- Kundendaten müssen bei der Eingabe auf Vollständigkeit und Richtigkeit überprüft werden, soweit dies technisch möglich ist. Fehlen Daten oder sind Daten falsch angegeben, müssen verständliche Fehlermeldungen ausgegeben werden, die gleichzeitig erklären, wie der Fehler behoben werden kann.
- Als Zahlungsmethoden stehen Nachnahme, Bankeinzug und Vorkasse zur Verfügung. Je nach gewählter Zahlungsmethode fallen zusätzliche Kosten an. Außerdem muss beachtet werden, dass je nach Land, in das die Bestellung ausgeliefert werden soll, unterschiedlich hohe Kosten für Zahlungsmethoden anfallen: Zum Beispiel ist die Nachnahmegebühr im Inland geringer als die Nachnahmegebühr bei einem Versand in ein bestimmtes Ausland.
- Es dürfen nur Bestellungen aus Ländern entgegengenommen werden, bei denen die Kosten von mindestens einer Zahlungsmethode bekannt sind. Ist für ein Land nicht bekannt, was ein Versand per Nachnahme, Bankeinzug oder Vorkasse an Kosten verursacht, darf für das Land keine Bestellung entgegengenommen werden.
- In vielen Ländern, in die geliefert wird, gibt es einen Distributor, an den Bestellungen im Online-Shop weitergeleitet werden. Das muss auf sicherem Weg geschehen, da mit Kunden- und Bestelldaten verantwortungsvoll umgegangen wird. Gleichzeitig soll diese Weitergabe jedoch automatisiert sein, um den Aufwand zur Auslieferung von Bestellungen niedrig zu halten. Da diese Distributoren mit unterschiedlichen Softwaresystemen arbeiten, müssen Kunden- und Bestelldaten im jeweils richtigen Format weitergegeben werden.
- Da es nicht für jedes Land, in das geliefert wird, einen dort ansässigen Distributor gibt, sind einige Distributoren für mehrere Länder verantwortlich.