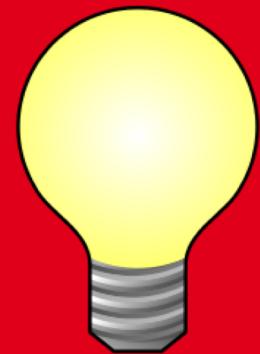


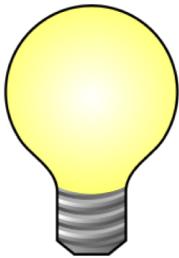


Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

# UML II - Klassendiagramm

Einführung in Klassendiagramme





Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

# AGENDA

Einführung ins Thema

Grundlagen

Weitere Anmerkungen zu Assoziationen

Vererbung

Attribute, Methoden, Sichtbarkeiten

Zwei mögliche Bedeutungsarten

Fazit

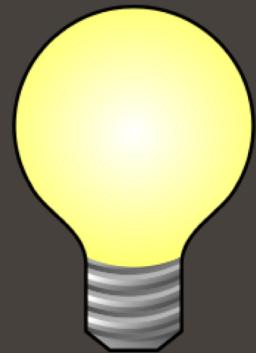


Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

# 01

## EINFÜHRUNG INS THEMA

Ziel:  
Die Eckpunkte des Themas kennenlernen



# DAS KLASSENDIAGRAMM



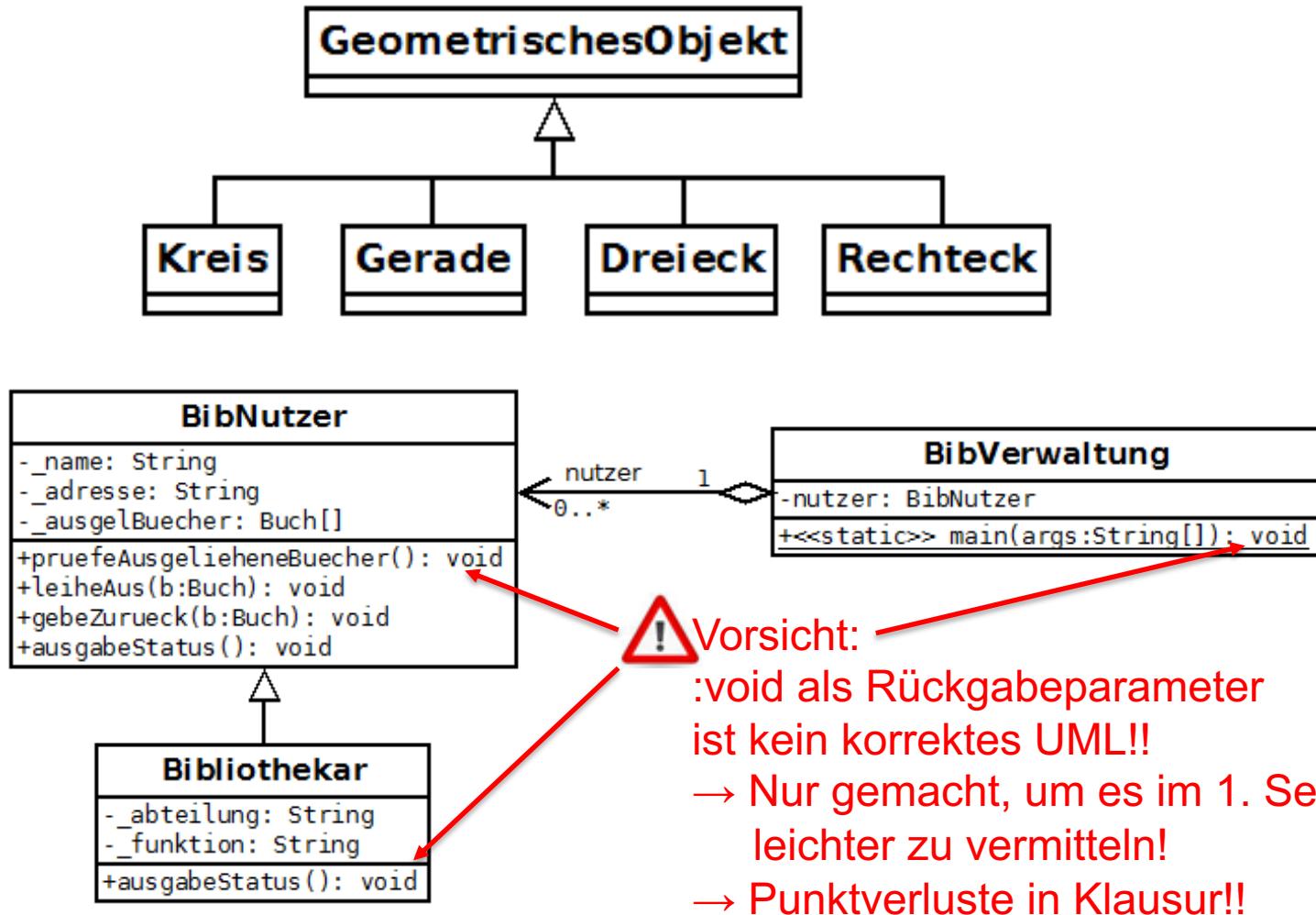
Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

- Neben dem Objektdiagramm
  - Objekte und deren Beziehungen (Links)
  - Schnappschuß einer bestimmten Situation zu bestimmter Zeit
  - dynamischen Sicht
- Gibt es das Klassendiagramm
  - Klassen und deren Beziehungen (Associations)
  - Allgemeine Definition der Klassen und aller möglichen Beziehungskonfigurationen durch Assoziationen
  - statische Sicht

# BEISPIELE AUS DER OOSE – VORLESUNG



- Das Klassendiagramm habe ich in OOSE bereits verwendet:

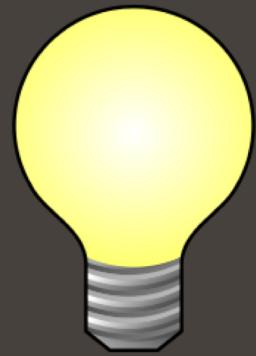




Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

## 02 LOSLEGEN

Ziel:  
Die Grundlagen kennenlernen

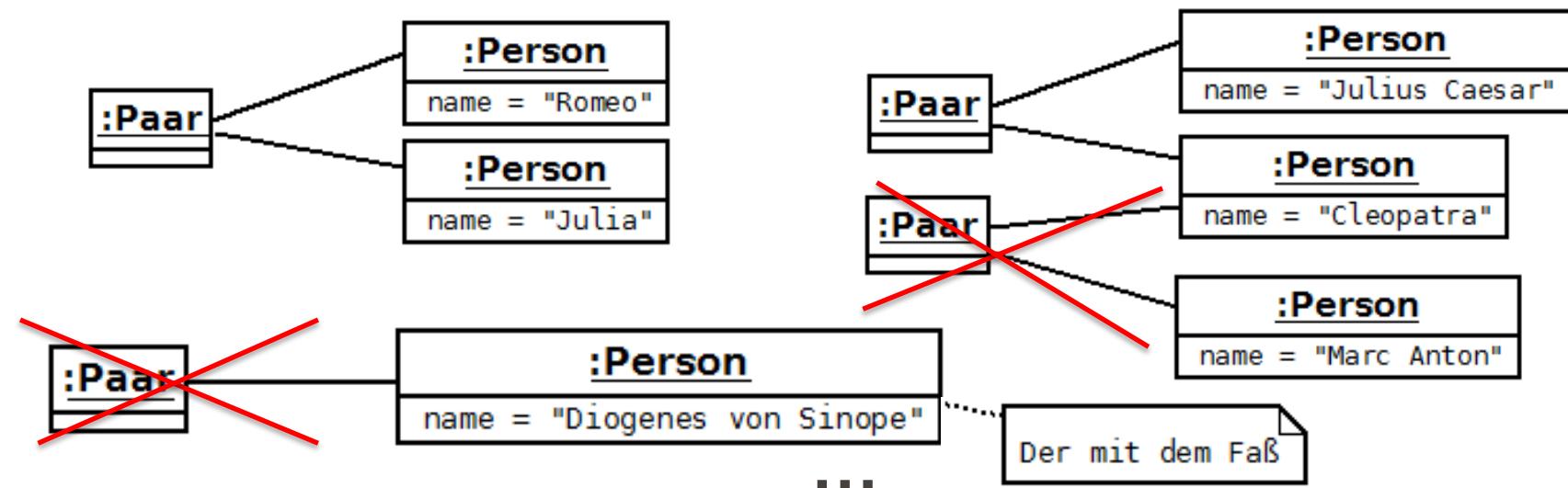


# BSP: DEFINITION PAARE UND DEREN BEZIEHUNGEN



1. Es gibt nur Paare, die mit 2 Personen verbunden sind
2. Eine Person kann höchstens mit einem Paar verbunden sein

→ Man kann nun einige Objektdiagramme zeichnen:

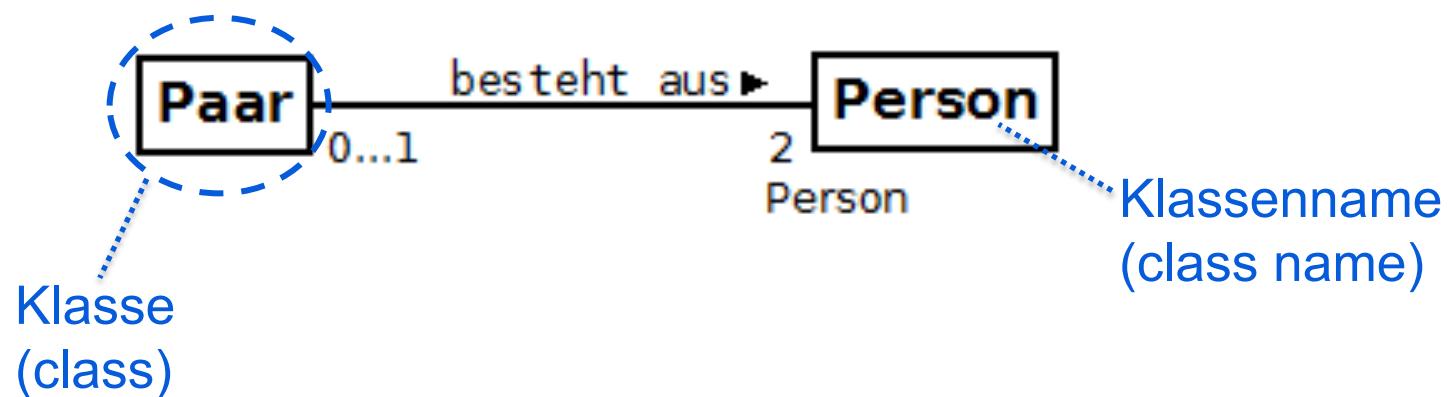


→ Nur unvollständige Auflistung von richtigen & falschen Beispielen möglich



# DAS KLASSENDIAGRAMM

- Wir brauchen eine vollständige Darstellung der Sachverhalte
- Klassendiagramm:



Vergleiche die Darstellung!

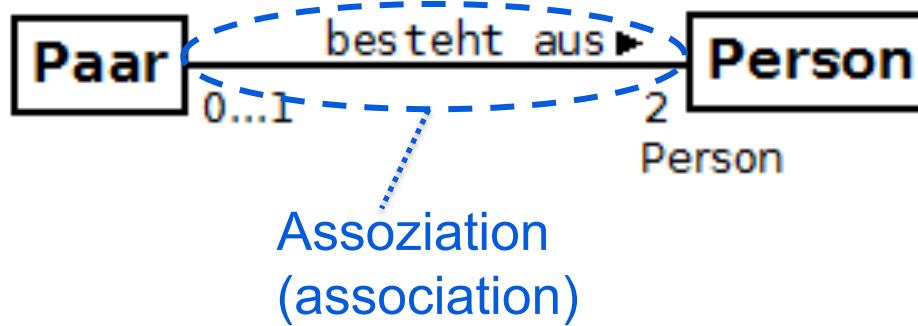
Klasse: **Person**  
Objekt: **:Person**

# DAS KLASSENDIAGRAMM – ASSOZIATIONEN



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

- Verbindungen zwischen Klassen → Assoziationen:

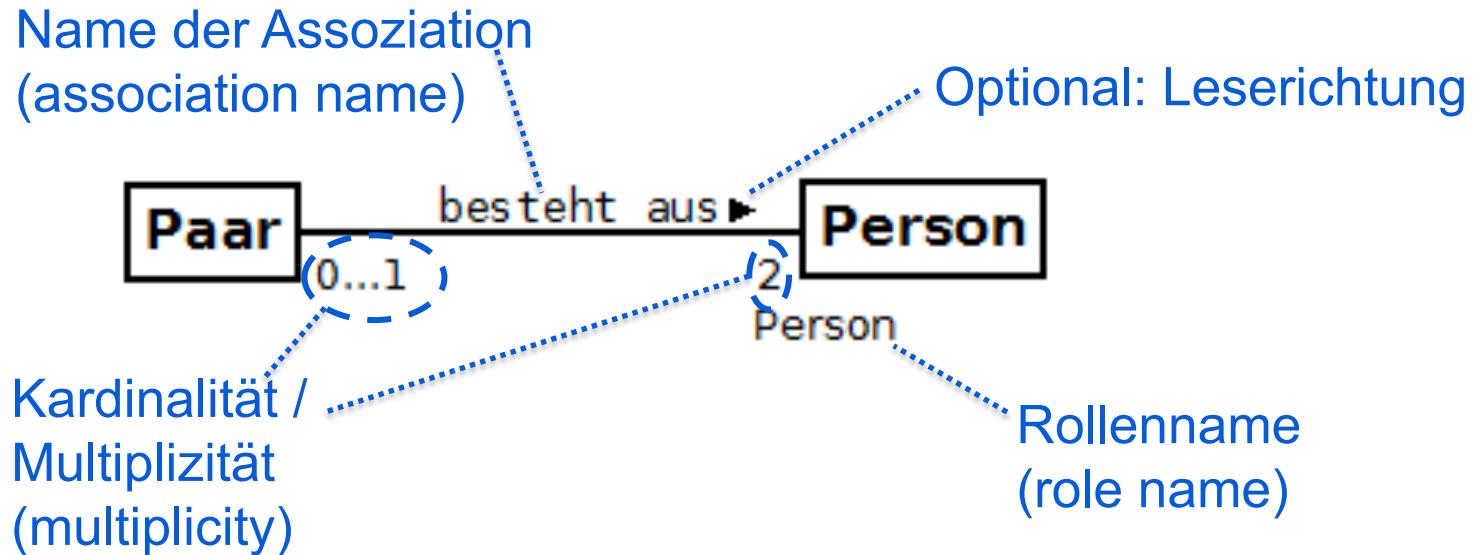


BEM: Assoziation  $\triangleq$  Relation im Entity-Relationship-Diagramm

# DAS KLASSENDIAGRAMM – ASSOZIATIONEN



- Weitere Infos zu einer Assoziation:



## Kardinalitäten:

- Allgemeine Form: Untergrenze...Obergrenze  
 $\in \mathbb{N}$        $\in \mathbb{N} \cup \{\ast\}$        $\hat{=} \infty / \text{beliebig}$

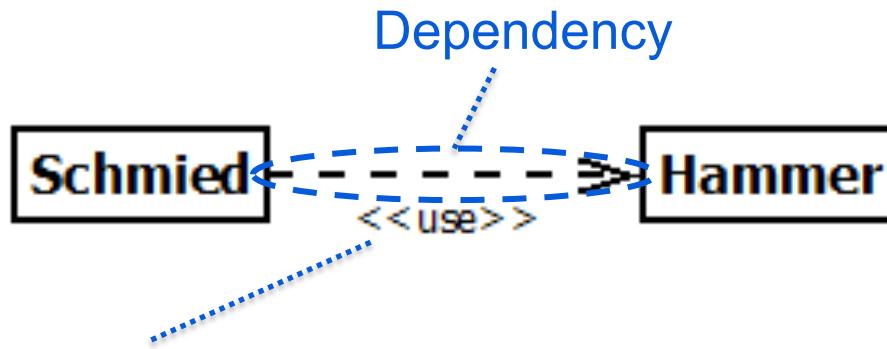
- Spezialformen:
  - a...a | a → nur/genau a (Bsp: 7...7 → genau 7)
  - <leer> → genau 1

# DAS KLASSENDIAGRAMM – DEPENDENCY\*



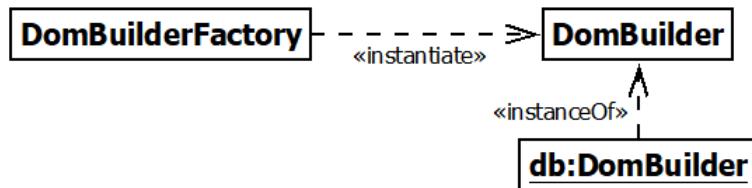
Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

- Neben Assoziationen gibt es noch Dependency-Pfeile  
→ Für einfache Abhängigkeitsbeziehungen



Stereotyp (Stereotype) «...» = Erweiterungsmechanismus der UML

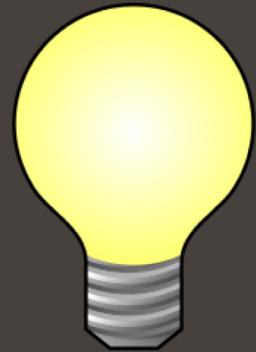
- Genauere Definition (Eingrenzung) der Abhängigkeitsbeziehung
- Typische Stereotypen für Dependencies\*: «use», «import», «create», «call», « abstraction», «instanciate», «instanceOf», ...
  - Eigene Stereotypen möglich





03

## Weitere Anmerkungen zu Assoziationen



Ziel:

Weitere Details zu Assoziationen besprechen

# KLASSENDIAGR.: ASSOZIATION ↔ OBJEKTDIAGR.: LINKS

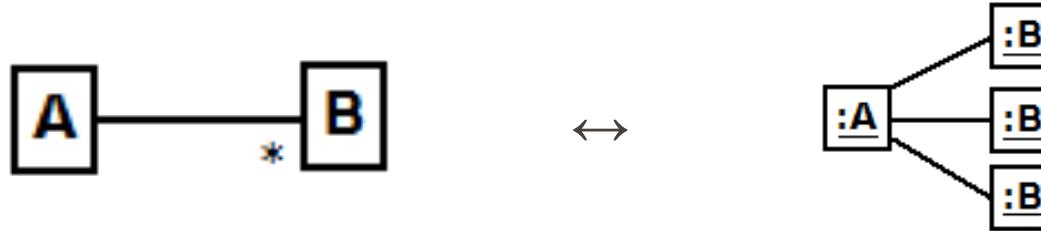


Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

- Weitere Bemerkungen zu Assoziationen:

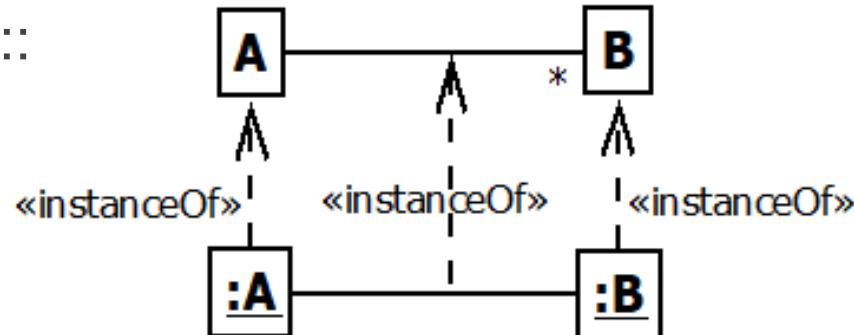
- Eine **Assoziation** im Klassendiag. (KD):

entspricht **evtl. n Links** in einem Objektdiag. (OD):



- Beziehung in der UML:

Klassendiagr.::



Objektdiagr.::

# KLASSENDIAGR.: ASSOZIATION ↔ OBJEKTDIAGR.: LINKS

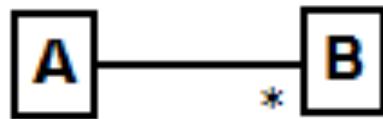


Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

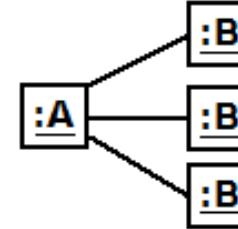
- Weitere Bemerkungen zu Assoziationen:

- Eine **Assoziation** im Klassendiag. (KD):

entspricht **evtl. n Links** in einem Objektdiag. (OD):



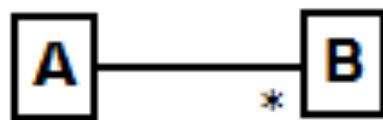
↔



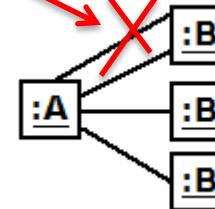
- 1 “normale” Assoziation ist auch nur 1 Link zwischen 2 Objekten



Es kann dann **nicht 2** geben!

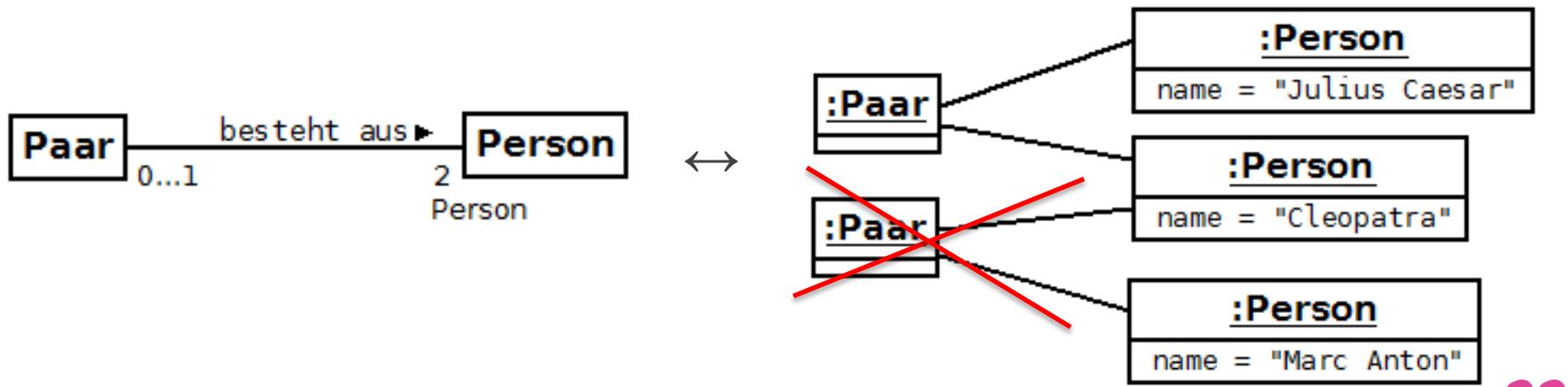


↔





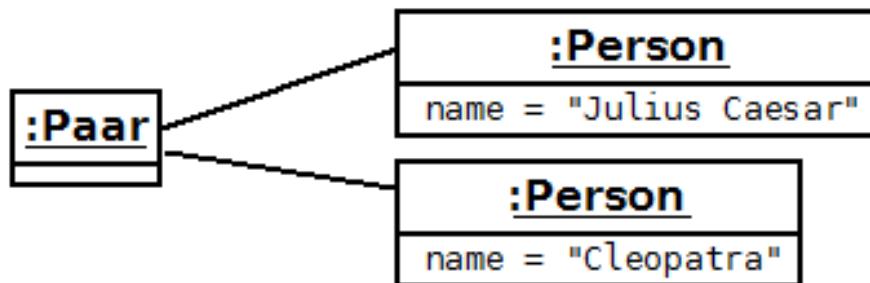
# WEITERE BEMERKUNG:



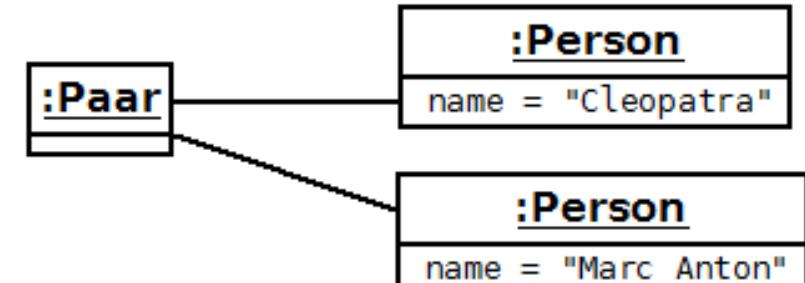
- Info. ist iwie richtig. → Wie gehört das zusammen?



Situation vor Cäsars Tod:



Einige Jahre nach Cäsars Tod:



→ Schnappschußcharakter der Objektdiagramme!

# WAS IST ZUSAMMENHANG ZW. ASSOZIATIONEN UND CODE?

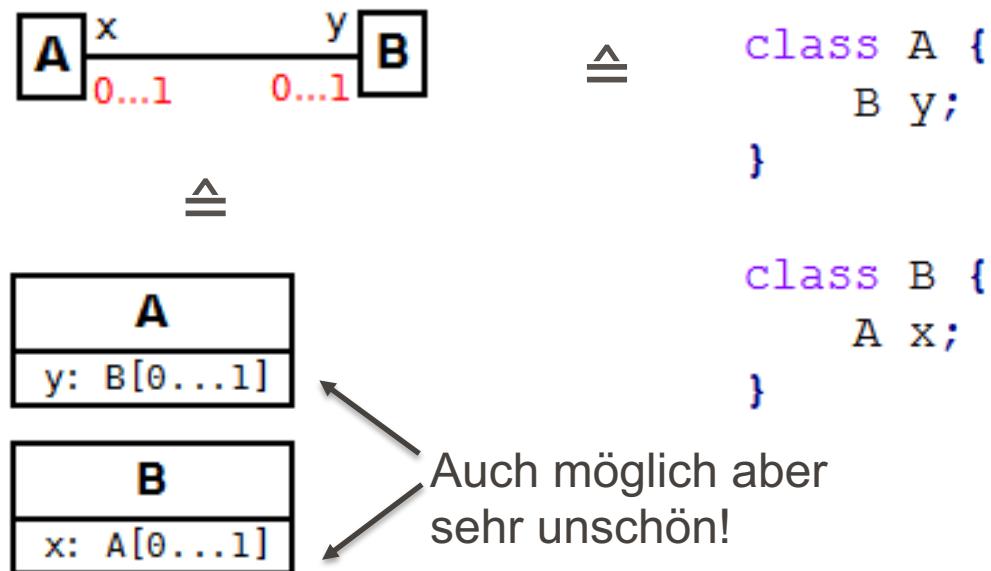


Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

In Programmiersprachen (Java, ...) gibt es keine direkte Entsprechung für das Konzept “Assoziation”

→ Wie umsetzen?

- Möglichkeit A:



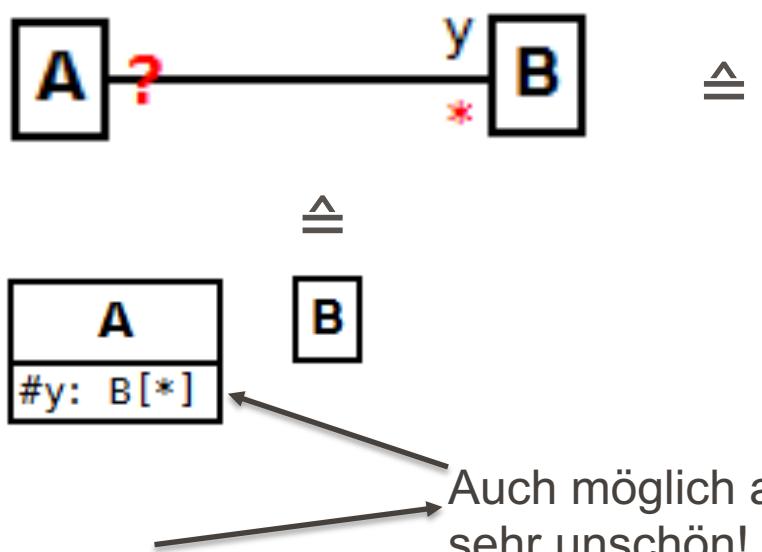
# WAS IST ZUSAMMENHANG ZW. ASSOZIATIONEN UND CODE?



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

In Programmiersprachen (Java, ...) gibt es keine direkte  
Entsprechung für das Konzept “Assoziation”

- Wie Umsetzen?  
– Möglichkeit A:



```
class A {  
    B[] y;  
//ODER:  
    Collection<B> y;  
//ODER:  
    List<B> y;  
//ODER:  
    ...  
}
```

Auch möglich aber  
sehr unschön!

## → Empfehlung:

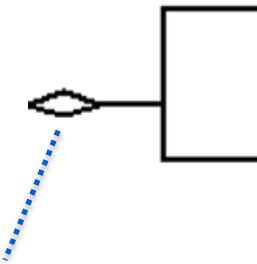
- Primitive Datentypen (Zahl, Bool, String, Enum) → Attribut
- Sonst → Assoziation

# AGGREGATION & KOMPOSITION IM KLASSEN- & OBJEKTDIAGRAMM

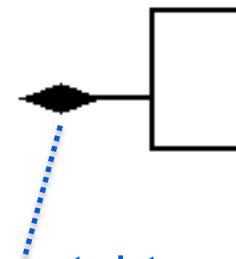


Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

- Betonung des Aspekts „besteht aus“:  
(v.a. in Analysediagrammen\* nützlich)



„besteht aus“, „enthält“  
(Aggregation)



„besteht aus“, „enthält“  
(Komposition)  
Stärker als



Klassendiagr.: Aggregation & Komposition sind Untertypen von  
Assoziation

Objektdiagr.: Aggregation & Komposition sind Untertypen von  
Link

\* Diagramme, die im Zuge der Analysephase gemacht werden  
→ Siehe Fachmodell in Vorl. 07 Anforderungsanalyse

# (WIEDERHOLUNG) AGGREGATION & KOMPOSITION



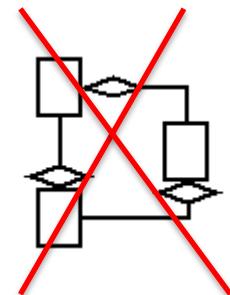
Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim



Vorsicht: UML definiert keine präzise Bedeutung für



- Lediglich:
  - Für beide sind keine zyklischen Abhängigkeiten erlaubt
  -  ist irgendwie stärker als 
    - Inoffizielle Unterscheidung:  
  $\triangleq$  Gesamtteil kann nicht ohne Einzelteile existieren



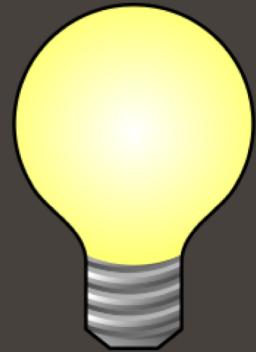
→ TIPP: Nur verwenden, wenn WIRKLICH notwendig



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

## 04 Vererbung

Ziel:  
Wie kann Vererbung dargestellt werden?



# VERERBUNG



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

Notation Vererbung in UML:



→ Drück aus: B erbt von A

- Hier gibt es auch eine extensionale und eine intensionale Bedeutungsart

# VERERBUNG – AUSWIRKUNGEN AUF OBJEKTE & OBJEKTDIAGRAMM



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

Mögliche Notation für B-Objekte:



→ Typ-Info wird unspezifischer



Auch möglich:



⚠ Vererbung ist in der Regel im Objektdiagramm  
nicht (direkt) sichtbar!

**Typregel:** Die B-Objekte müssen auch die gleichen

- Eigenschaften
- Fähigkeiten

wie die A-Objekte haben

Mit anderen Worten: Jedes B-Objekt ist auch A-Objekt  
(ABER: Nicht jedes A-Objekt ist auch B-Objekt)

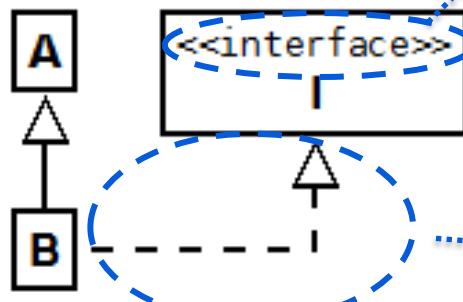


# MEHRFACHVERERBUNG



## UML erlaubt echte Mehrfachvererbung (wie C++)

- Damit UML auch kompatibel zu Sprachen wie C++ ist
- Man sollte Mehrfachvererbung in Designsichten vermeiden, wenn die Sprache das nicht unterstützt
  - Oder zumindest dann eine Beschreibung angeben wie diese realisiert wird
- “Erben” von Interfaces:



Stereotyp (Stereotype) «...»  
= Erweiterungsmechanismus  
der UML

Hier: gleiches Modellelement  
(rechteckiger Kasten) wie für Klasse,  
aber „Kasten“ hat mit «interface»  
andere Bedeutung: Interface

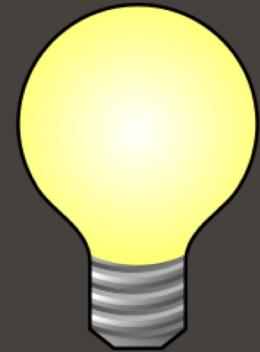
Implementierungspfeil /  
Realisierungspfeil





05

## Attribute, Methoden, Sichtbarkeiten

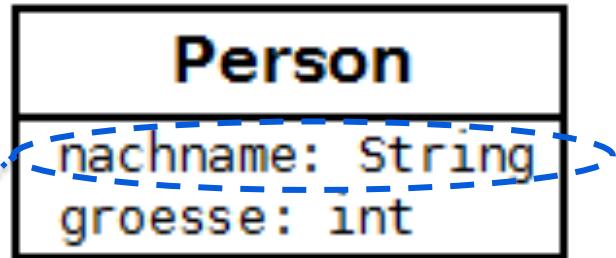


Ziel:

Den inneren Aufbau von Klassen detaillierter spezifizieren

# (MÖGLICHE) DARSTELLUNG FÜR BEIDE SICHTEN

- Analog wie in Objektdiagramm dargestellt:



Konkrete Eigenschaft (Attribut)

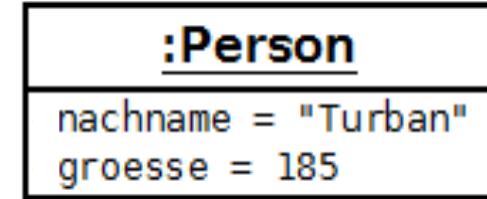
→ Besteht aus:

- Name der Eigenschaft (feature name)
- optional: Art der Eigenschaft (z.B. :String)
- optional: Initiale Belegung / Wert  
(Vorgabewert  $\triangleq$  default value specification)



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

Vgl. Objekt:



# WEITERE BEMERKUNG ZUR DARSTELLUNG



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

- Eine Klasse kann in der Regel\* maximal drei Abschnitte (Compartments) haben:
  - (vgl. auch Objektdiagramm vorletzte Woche)

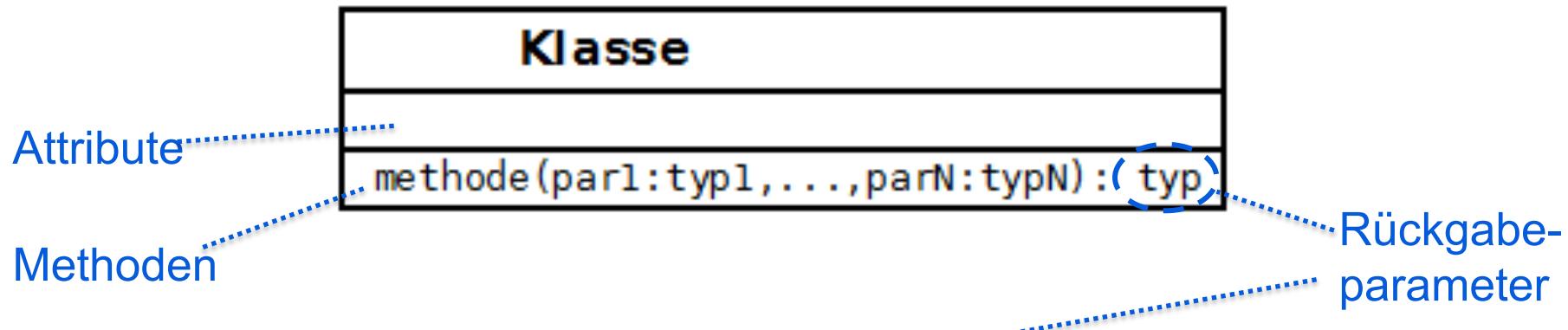


\* Theoretisch gibt die UML keine Vorgaben über Reihenfolge und Anzahl außer, dass oben der Name stehen muss → beliebige Anzahl und Reihenfolge  
Praktisch werden eigentlich nur diese 3 und eigentlich immer in dieser Reihenfolge genutzt



# METHODEN

- Übliche Namen: Methoden, Operationen, Member Functions
- Notation:



- BEM: void  $\notin$  UML  
typ == <leer> hat zwei Bedeutungen:  
void unbekannt
- BEM: Meth. v.a. bei Entwurfs-/Implementierungsdiagramm interessant



# SICHTBARKEITEN

- Für Attribute und Methoden können die Sichtbarkeiten definiert werden:

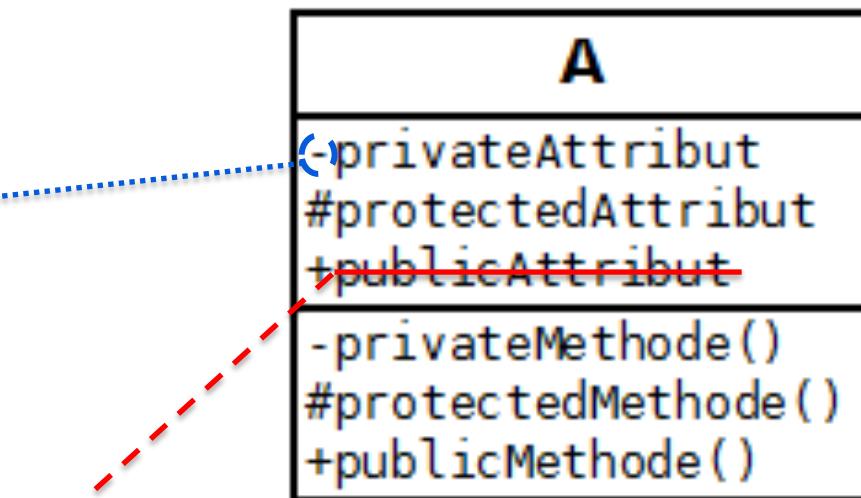
Sichtbarkeiten:

+ == public

# == protected

- == private

~ == package



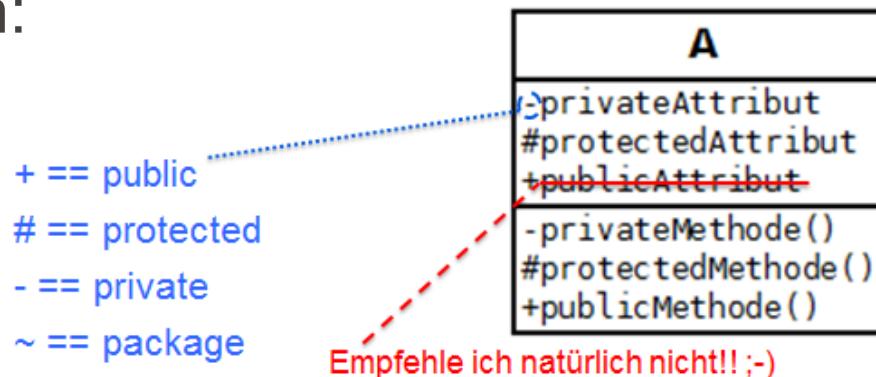
Empfehle ich natürlich nicht!! ;-)

- Ein unterstrichenes Attribut oder Methode bedeutet “static”



# SICHTBARKEITEN

- Für Attribute und Methoden können die Sichtbarkeiten definiert werden:



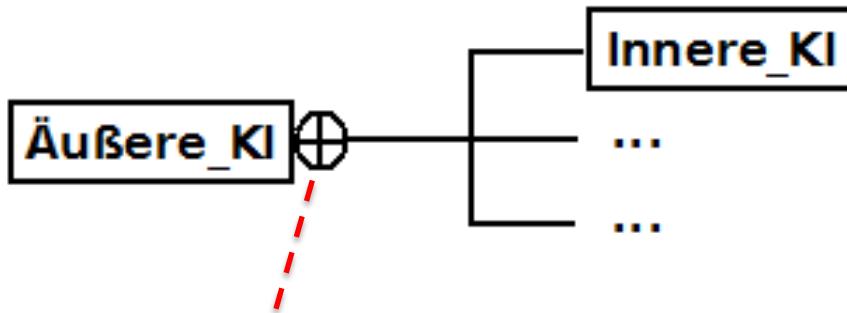
- Sichtbarkeiten können auch weggelassen werden
  - Wird in Objektdiagrammen üblicherweise weggelassen
  - Kann in Analysemodellen / Fachmodell auch weggelassen werden
  - In der Regel nur im Entwurfs-/Implementierungsdiagramm und im Zusammenhang mit Methoden interessant

**BEM:** Leider gibt es Werkzeuge, die Sichtbarkeiten immer anzeigen → Muss man dann einfach ignorieren ☹

# DARSTELLUNG INNERE KLASSEN



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

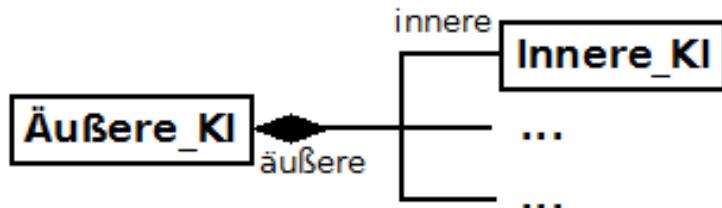


Nicht offiziell  
UML 2.x!

→ war Teil von UML 1.4, aber  
wird teilweise noch verwendet (auch im Standard)

```
class Äußere_Kl {  
    ...  
    class Innere_Kl {  
        ...  
    }  
}
```

- Ansonsten immer möglich (aber mehrdeutig!):

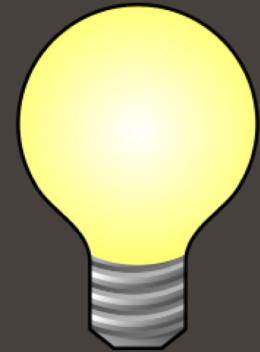


⚠ Komposition bedeutet nicht autom. innere Klasse!!  
(kann aber auf mögliche Lösung als innere Klasse hindeuten)



06

## Zwei mögliche Bedeutungsarten



Ziel:

Verstehen lernen, dass das Klassendiagramm verschiedene Bedeutungsarten annehmen kann

# GRUNDSÄTZLICH MÖGLICHE BEDEUTUNGSSARTEN



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

- Auf der Bedeutungsebene (Semantik) kann man mit einem Klassendiagramm zwei Bedeutungsarten abbilden:
  - I. Extensionale Bedeutungsart
    - △ Mengenorientiert
      - (Menge aller Objekte mit gleichen Eigenschaften)
  - II. Intensionale Bedeutungsart
    - △ Bauplanorientierte Bedeutung
      - (Gleicher Aufbau)

---

Hinweis: Beide Bedeutungsarten wurde von Aristoteles schon in der griech. Antike zur Definition von Begriffen eingeführt und unterschieden.  
Hierzu eine nähere Beschreibung:

[https://de.wikipedia.org/wiki/Extension\\_und\\_Intension](https://de.wikipedia.org/wiki/Extension_und_Intension)

# I. EXTENSIONALE BEDEUTUNGS-ART



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

Extensional:

Darstellung der

- Gemeinsamen Eigenschaften &
- Gemeinsamen Fähigkeiten

} = Gemeinsamkeiten

einer Menge an Objekten

→ 1 Klasse  $\hat{=}$  Menge gleichartiger Objekte

→ Mengenorientierte Darstellung

# I. EXTENSIONALE BEDART – BSP

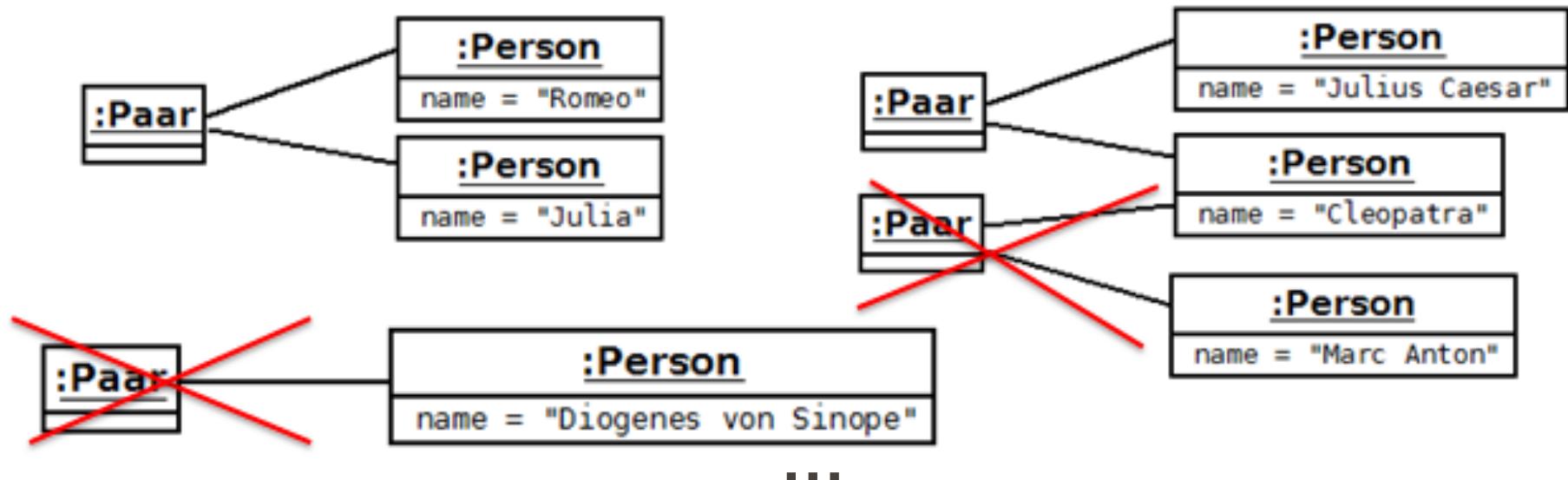
→ SIEHE BEISPIEL VON OBEN!



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

1. Es gibt nur Paare, die mit 2 Personen verbunden sind
2. Eine Person kann höchstens mit einem Paar verbunden sein

→ Man kann nun einige Objektdiagramme zeichnen:



→ Nur unvollständige Auflistung von richtigen & falschen Beispielen möglich

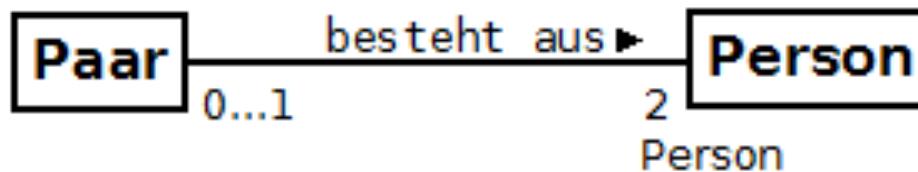
# I. EXTENSIONALE BEDART – BSP

## → DAS KLASSENDIAGRAMM



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

- Wir brauchen eine vollständige Darstellung der Sachverhalte
  - Klassendiagramm:



## II. INTENSIONALE BEDEUTUNGS- ART



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

Intensional:

Darstellung des Bauplans für ein(e) Art/Sorte/Typ/Klasse von Objekten

→ 1 Klasse  $\triangleq$  Bauplan für gleichartige Objekte

→ Bauplanorientierte Darstellung

## II. INTENSIONALE BEDEUTUNGS- ART – BSP



```
public class Person { ... }
```

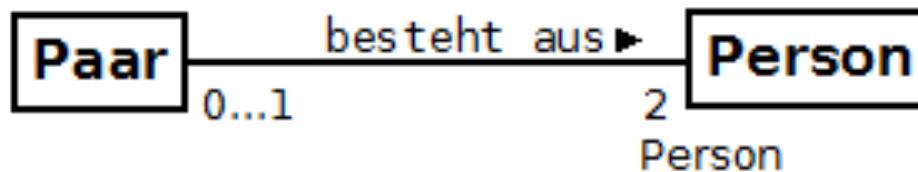
```
public class Paar { ... }
```

Gegebenenfalls weiterer Code

Quellcode stellt sicher, dass:

- Jedes Paar-Objekt genau mit 2 Person-Objekten verbunden ist.
- Jedes Person-Objekt mit höchstens einem Paar-Objekt verbunden ist

→ Klassendiagramm:



→ Selbes Diagramm wie in I. !!

Wie passt das jetzt zusammen?



# EXT. & INT. BEDEUTUNGSART – WIE PASST DAS JETZT ZUSAMMEN? \*

Hochschule RheinMain  
University of Applied Sciences  
Wiesbaden Rüsselsheim

- UML legt nicht eindeutig fest welche Bedeutungsart ein Klassendiagramm hat
  - Die Bedeutungsarten können zusammenfallen
  - Müssen aber nicht
- Man kann sich zu jedem KD fragen
  - Ist es Extensional oder Intensional gemeint?
  - Oder beides?
- **Grundsätzliche Regel:**
  - Ein Diagramm sagt mehr als 1000 Worte
  - Kann aber auch tausenderlei verschieden interpretiert werden  
→ Auch textuelle Beschreibung eines Diagramms nötig

# VERERBUNG – I. EXTENSIONAL

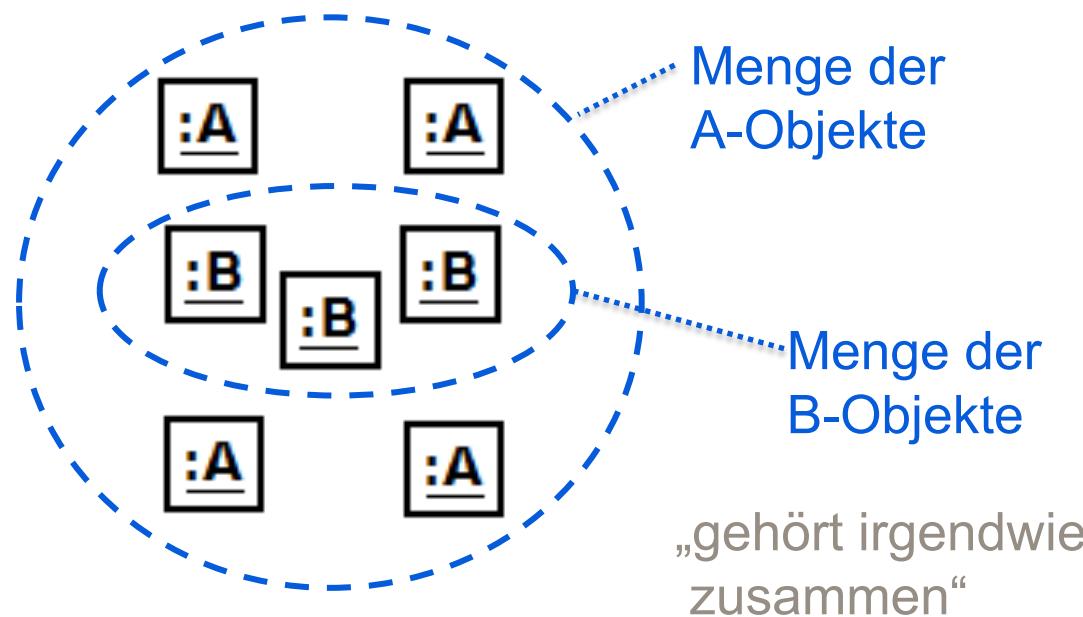
Meist im Analysediagramm

Bedeutungsart I. Extensional → Mengenorientiert:

A ist Oberklasse von B bedeutet:

Die Menge der B-Objekte ist eine  
Teilmenge der Menge der A-Objekte

Schematisch:

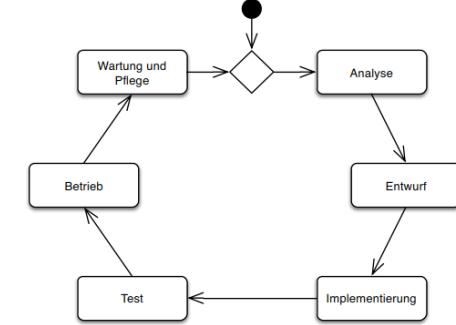


Notation  
Vererbung  
in UML:



# VERERBUNG – II. INTENSIONAL

Im Designdiagramm

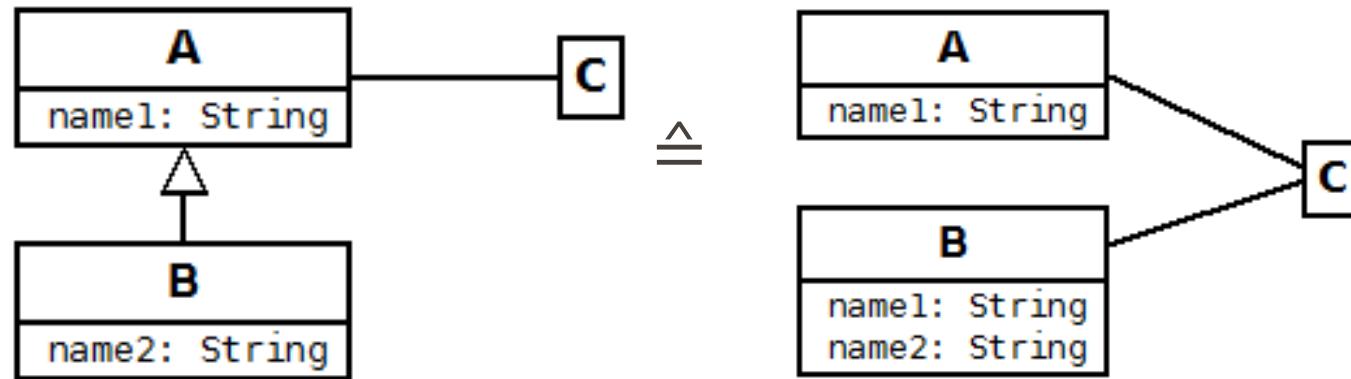


Bedeutungsart II. Intensional → Bauplanorientiert

A ist Oberklasse von B bedeutet:

Der Bauplan für B-Objekte umfasst auch den Bauplan  
für A-Objekte

Schematisch:



→ Nutzen von  $\uparrow$ :

Kurzschreibweise für die Wiederverwendung von Bauplänen

# EXT. & INT. BEDEUTUNGSART – BEISPIELE AUS DER REALEN WELT



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

1. Analyse: Aal und Schlange könnten beispielsw. als Beinlose zusammengefasst werden

 ABER: Aal ist ein Knochenfisch ↔ Schlange: ist ein Reptil!  
→ Interne Baupläne sind anders

→ Besseres Design: Aal erbt von Knochenfisch, Schlange von Reptil

2. Analyse: Delphine und Haie könnte man zu den Fischartigen zählen

 ABER: Delphine sind Säugetiere (näher mit Robben als mit Haien verwandt!)

→ Besseres Design: Delphin erbt von Säugetier, Hai von Knorpelfisch

(Biologie: Beispiele für Konvergente Evolution)

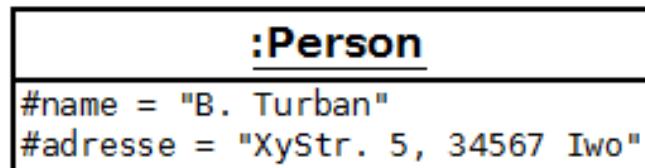
# LETZTE WOCHE: FACHLICHE ↔ TECHNISCHE SICHT



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

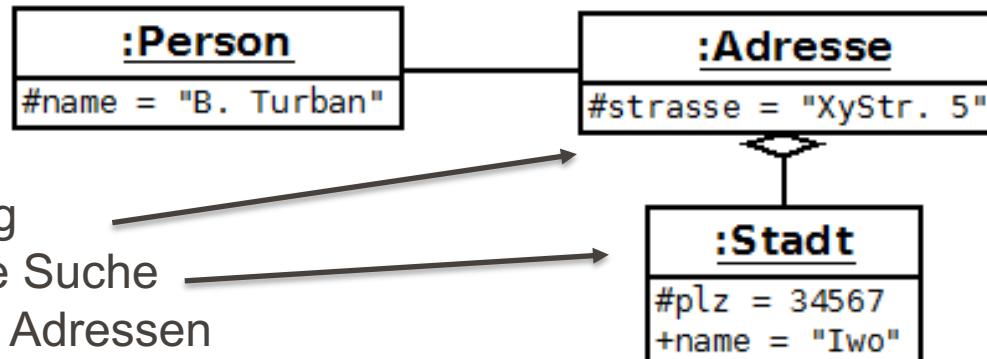
→ **Fachliche Sicht betrachtet alles ohne technische Details**

- Wird in der Analyse verwendet, um **NUR** die fachlichen Anforderungen zu ermitteln



→ **Technische Sichten**

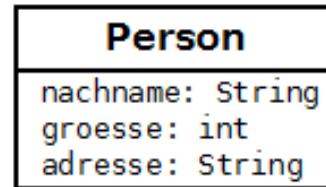
- In Architektur, Detailed Design, Implementierungsdoku verw.
- Es werden auch die für die Lösung **relevanten** technischen Details dargestellt:





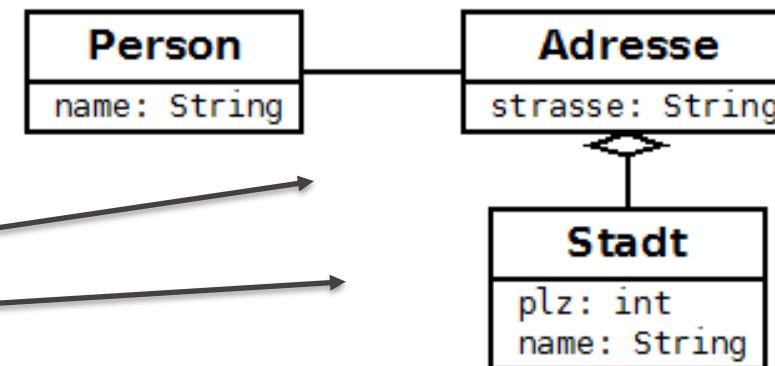
→ ***Fachliche Sicht betrachtet alles ohne technische Details***

- Wird in der Analyse verwendet, um **NUR** die fachlichen Anforderungen zu ermitteln



→ **Technische Sichten**

- In Architektur, Detailed Design, Implementierungsdoku verw.
- Es werden auch die für die Lösung **relevanten** technischen Details dargestellt:



Technische Lösung  
z.B. für spezifische Suche  
nach Städten oder Adressen

# EXTENSIONALE & INTENSIONALE SICHTWEISE



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

→ **Fachliche Sicht betrachtet alles ohne technische Details**

- Wird in der Analyse verwendet, um **NUR** die fachlichen Anforderungen zu ermitteln
- Eigentlich immer **Extensionale Bedeutungsart**

→ **Technische Sichten**

- Architektur (High-Level)
- Detailed Design
- Code / Diagramm mit exakter Darstellung des Codes
- “**Sollten**” beide **Bedeutungsarten** spätestens beim Implementierungsnahen Diagramm **zusammenfallen**
  - Nicht Zusammenfallen deutet auf möglichen Designfehler oder Architekturerosion hin

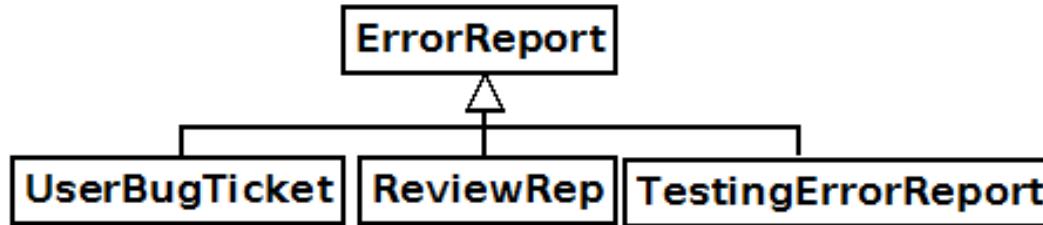
# EXTENSIONALE & INTENSIONALE BEDEUTUNGSARTEN



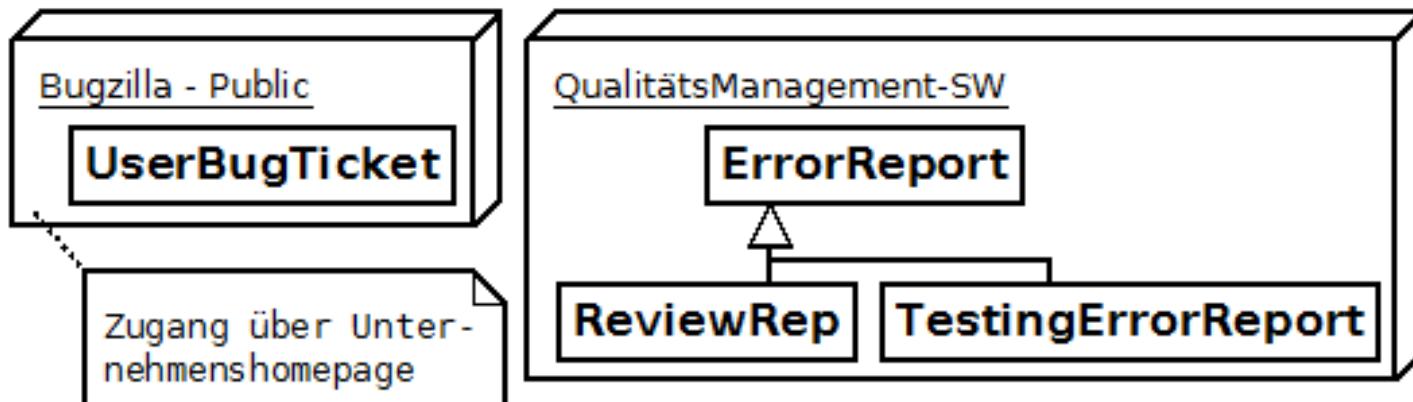
Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

## BSP - Mögliche Architekturerosion:

- Ursprünglich analysiert (Fachliche Sicht):



- Mapping auf IT-Systeme zunächst (Deploymentdiagramm):



Weil User UserBugTickets über die Homepage erstellen müssen  
wird Bugzilla dafür verwendet

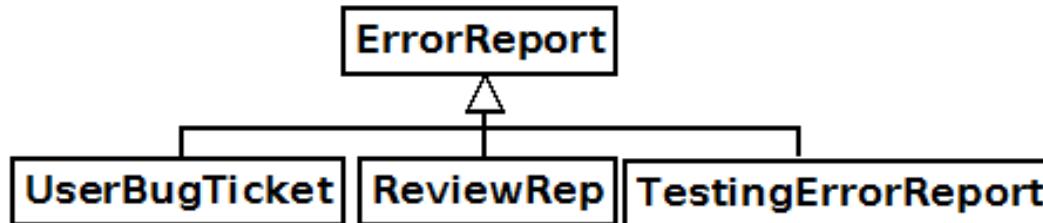
# EXTENSIONALE & INTENSIONALE BEDEUTUNGSARTEN



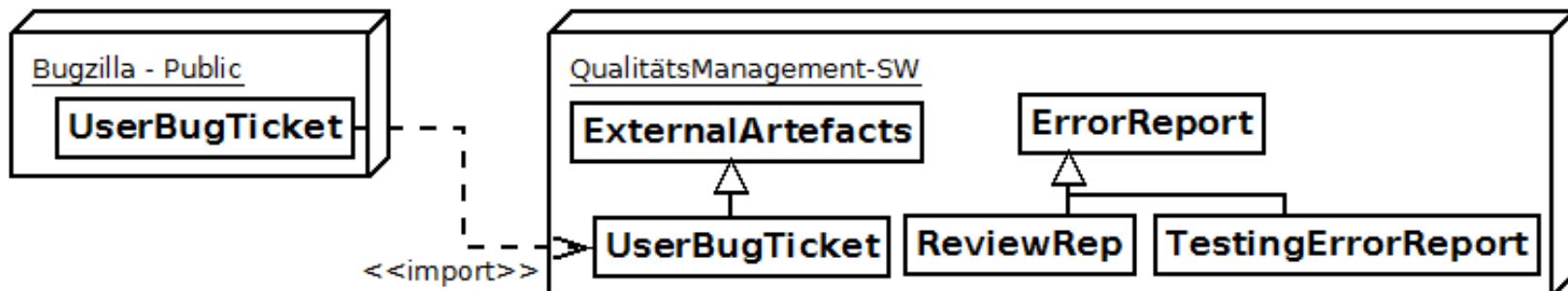
Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

## BSP Mögliche Architekturerosion:

- Eigentlich gilt:



- Einige Jahre später – import in das QM-System:



Aus anderen Gründen scheint es besser UserBugTicket anders zu behandeln  
ABER: Diese „Sonderlocke“ muss dann in Zukunft muss immer extra behandelt  
werden → macht dann später immer mal wieder Probleme

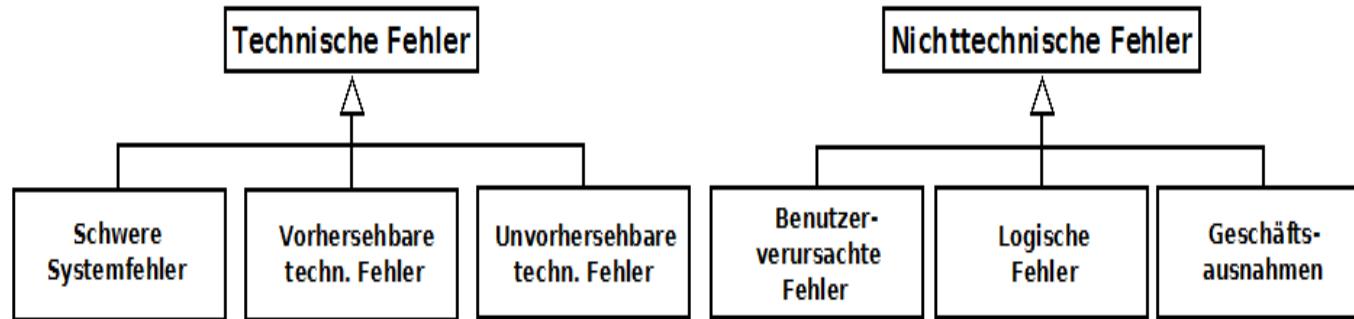
# EXTENSIONALE & INTENSIONALE BEDEUTUNGSARTEN



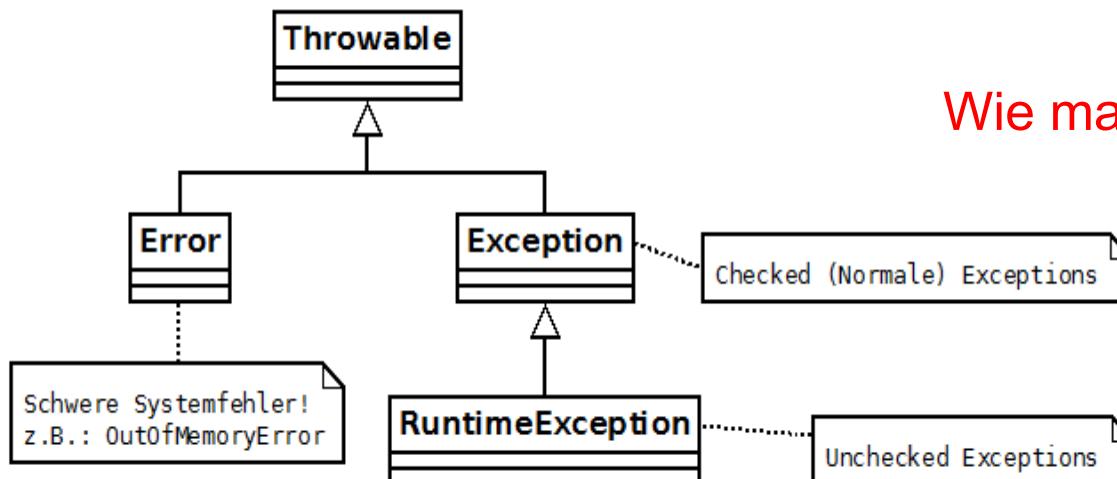
Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

## BSP Fehlerbehandlung:

- Analyse (→ Extensional):



- Technische Ebene JAVA Typsystem (→ Intensional):



Wie mappt sich das?

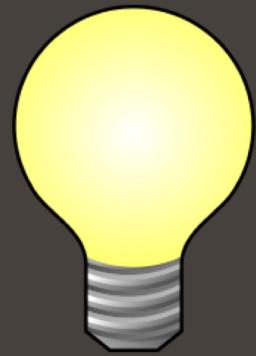




# 07

## Fazit

Ziel:  
Was haben wir damit gewonnen?





# WAS HABEN WIR GELERNT?

- Das Klassendiagramm
  - Klassen und Assoziationen, Dependency-Beziehungen
  - Vererbung, Attribute, Methoden, Sichtbarkeiten
  - Stereotypen dienen zur Bedeutungseinschränkung
- Verschiedene Bedeutungsarten möglich
  - Extensional vs. Intensional
    - Vielleicht eher eine akademische Unterscheidung
  - ABER: Wichtig zu Unterscheiden bei Analyse- und Designphase
  - ABER: Wichtig zu verstehen, dass die Bedeutung nicht vollständig eindeutig ist und variieren kann
    - Leider, die Welt ist halt nicht perfekt
    - Man sollte die Bedeutung auch textuell beschreiben!



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

AUF GEHT'S!!

SELBER MACHEN UND LERNEN!!

