




# Softwareentwurf

Kapitel II: Das Pflichtenheft  
II.2: Modell des Problembereichs





Prof. Dr. Gregor Engels  
AG Datenbank- und  
Informationssysteme





**UNIVERSITÄT PADERBORN**  
*Die Universität der Informationsgesellschaft*



## Gliederung Pflichtenheft <-> Vorlesung

- Abschnitt 1:  
Zielbestimmung 
  
- Abschnitt 2:   
Produkteinsatz   
2.1 Beschreibung des Problembereichs  
2.2 Glossar  
2.3 Modell des Problembereichs **Kapitel II.2**  
2.4 Geschäftsprozesse
  
- Abschnitt 3:  
Produktfunktionen
  
- Abschnitt 4   
Produktcharakteristiken

Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      2 



## Pflichtenheft – Abschnitt 2: Produkteinsatz


---


### Aufgaben der Abschnitte 2.1 – 2.3:

- 2.1 Beschreibung des Problembereichs:  
Erläuterung von Fachbegriffen und Zusammenhängen für den Laien
- 2.2 Glossar:  
Nachschlagewerk für Fachbegriffe
- 2.3 Modell des Problembereichs:  
Nachschlagewerk für Präzisierung der Zusammenhänge

**DURCH**  
graphisches, objektorientiertes Modell

---


Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      3 




## Unified Modeling Language (UML)

---

- graphische, objektorientierte Familie von Modellierungssprachen
- Standard, herausgegeben von der Object Management Group (OMG) seit 11/1997
- aktuelle Version: 2.0


 <http://www.omg.org/uml>

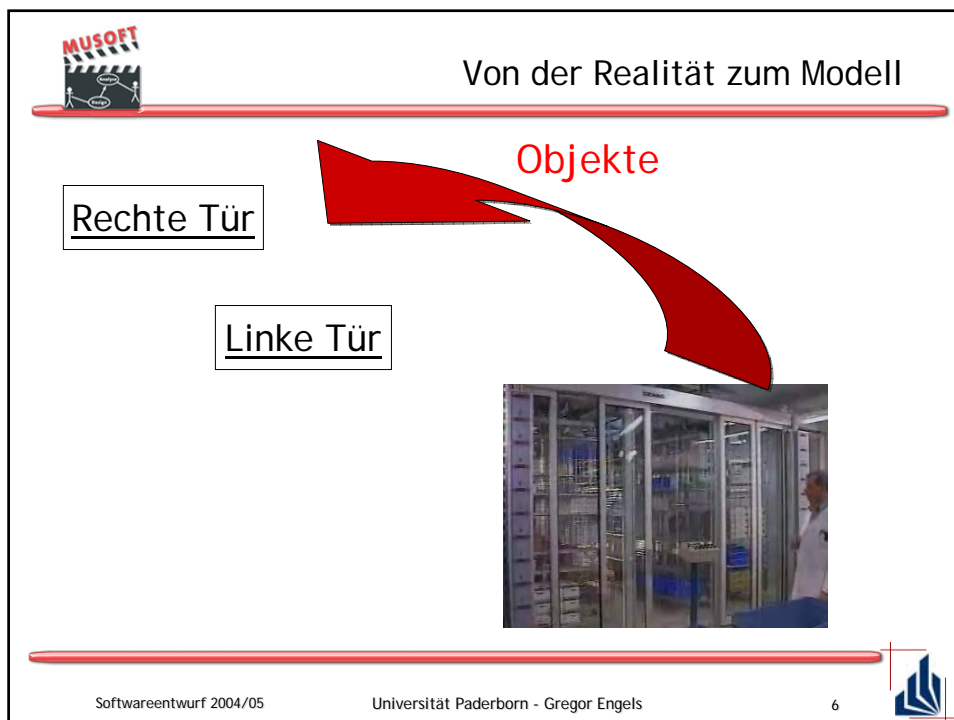
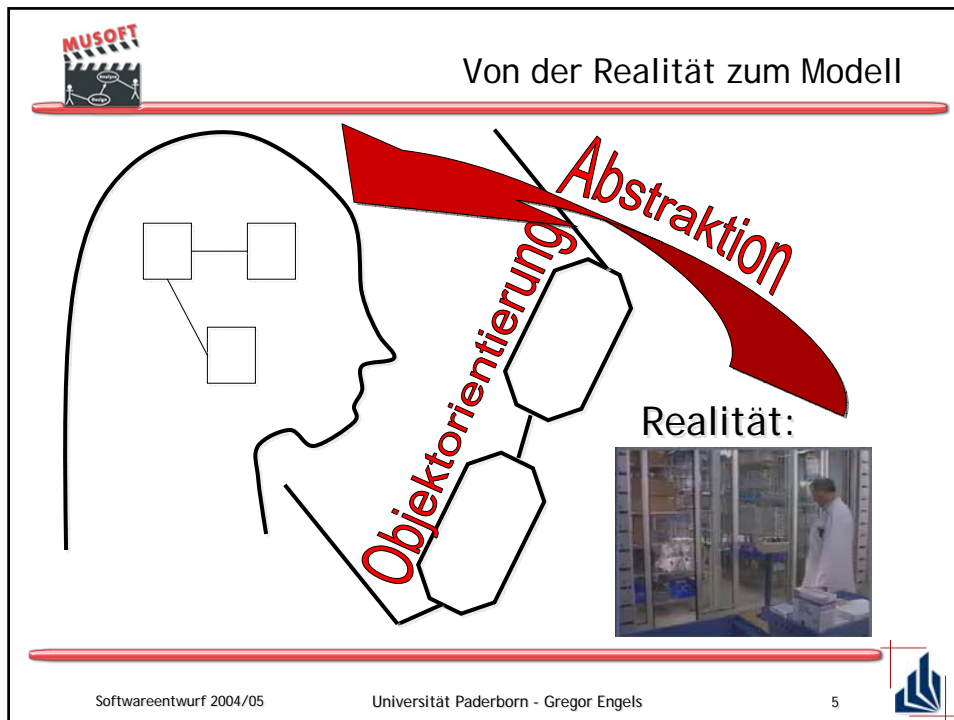
Notationsübersicht und Glossar

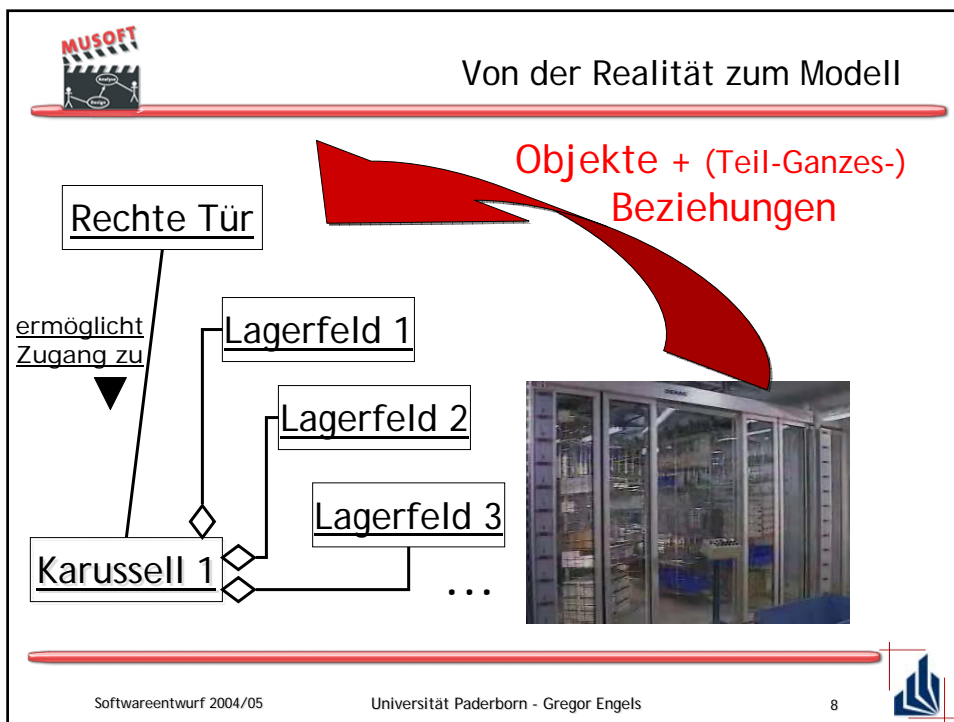
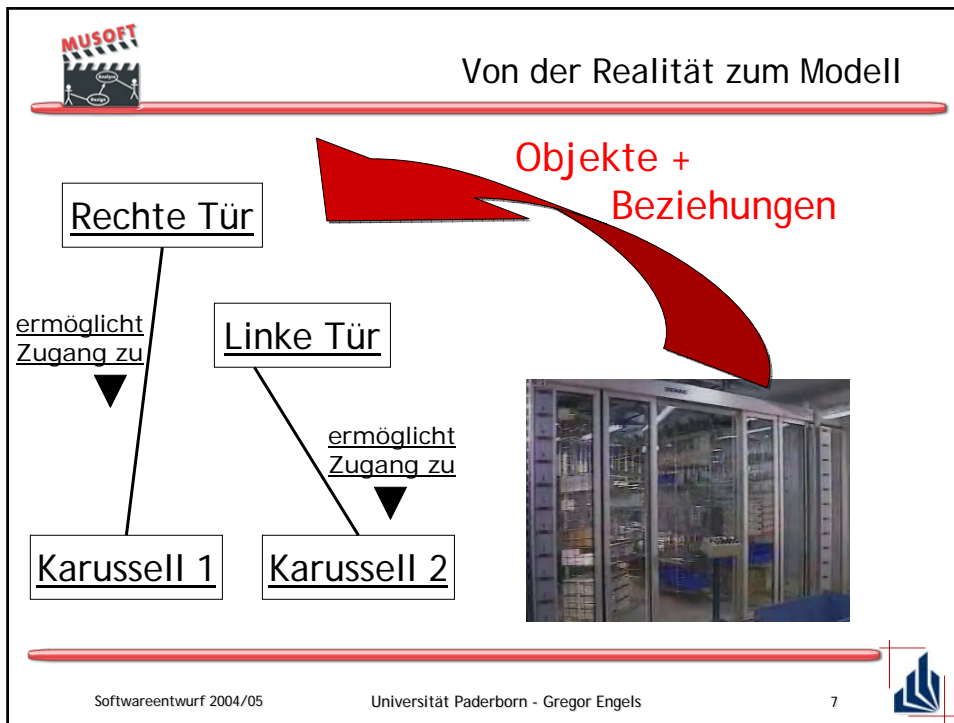
 <http://www.oose.de/uml/>

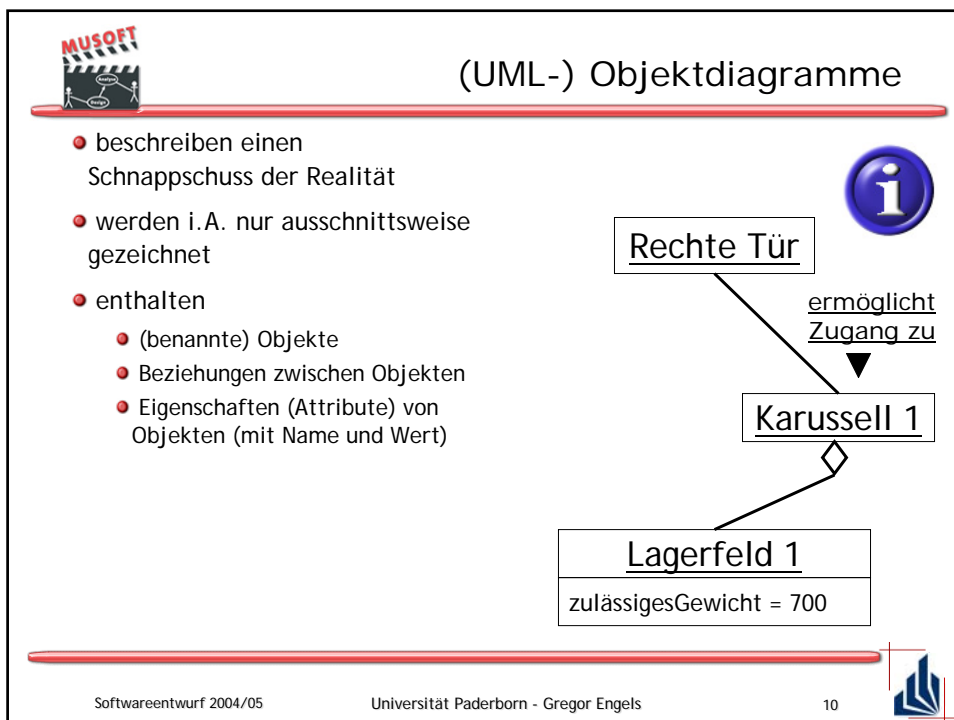
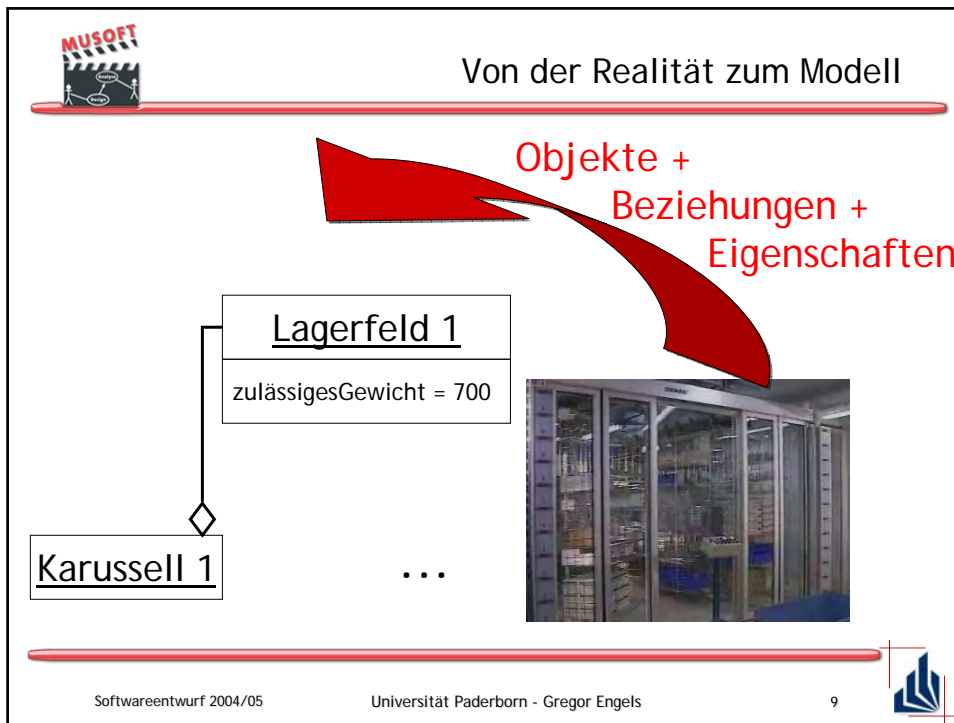
Grady Booch, James Rumbaugh, Ivar Jacobson, *The Unified Modeling Language User Guide*, Addison Wesley 1999.

---

Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      4 







MUSOFT

### Von der Realität zum Modell

Menge  
allgemein möglicher  
Situationen  
=  
Modell des  
Problembereichs

**Objektorientierung**

Realität:

Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      11

MUSOFT

### Vom konkreten zum allgemeinen Modell

Tür

Es gibt Türen.

**Klassen**

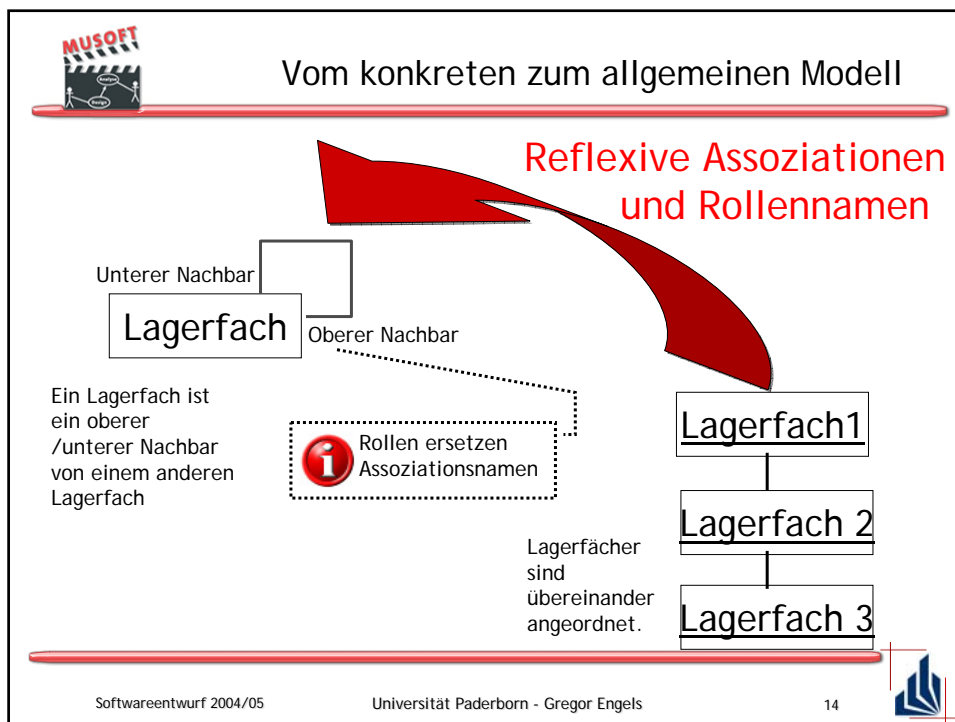
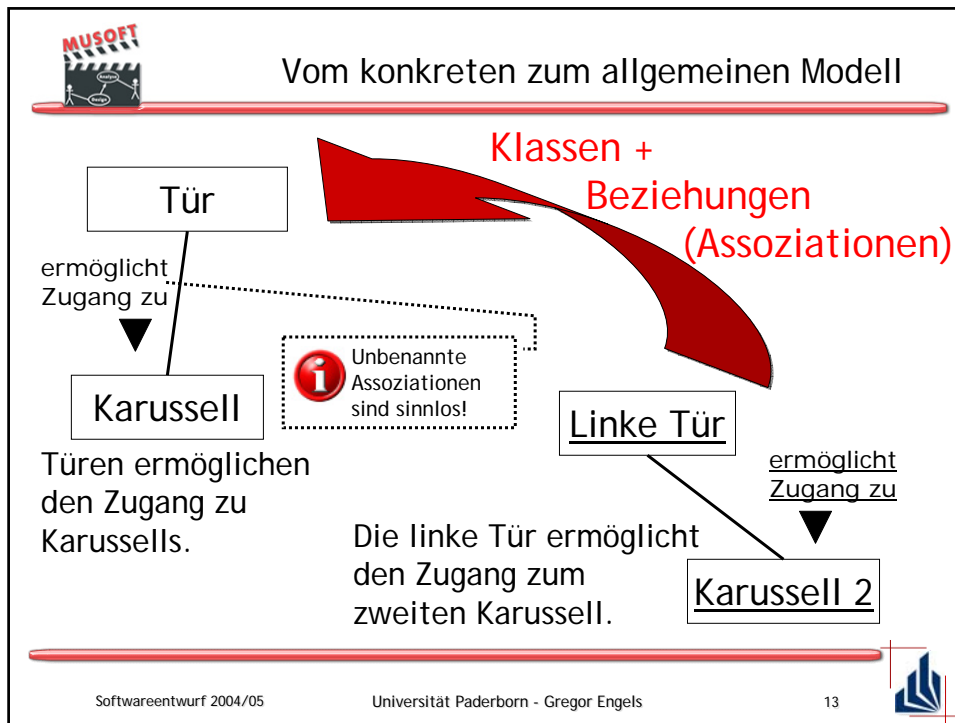
Rechte Tür

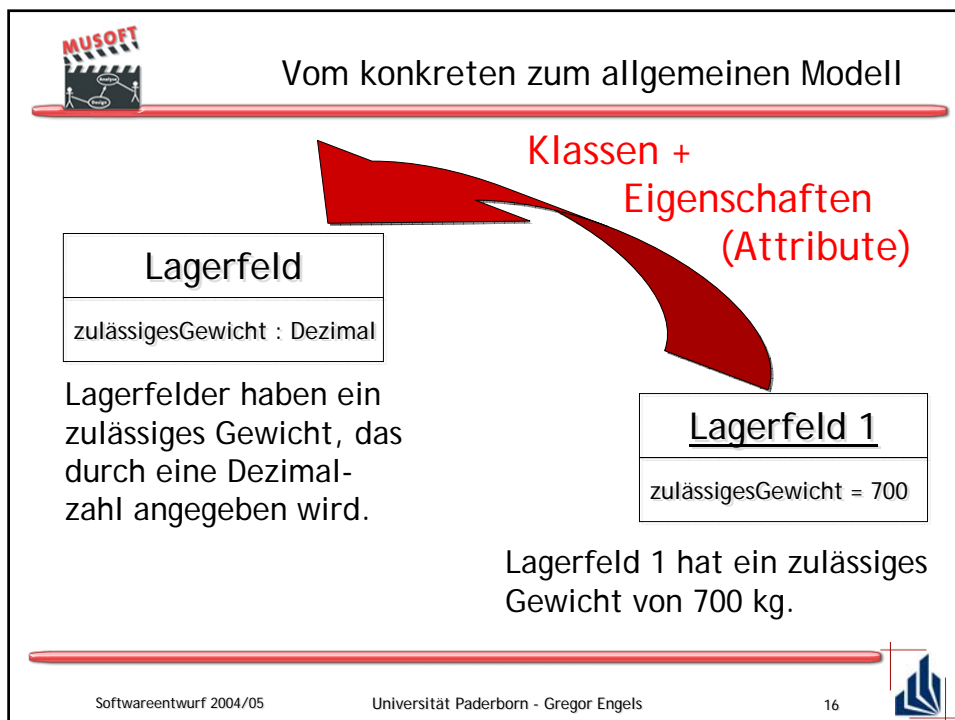
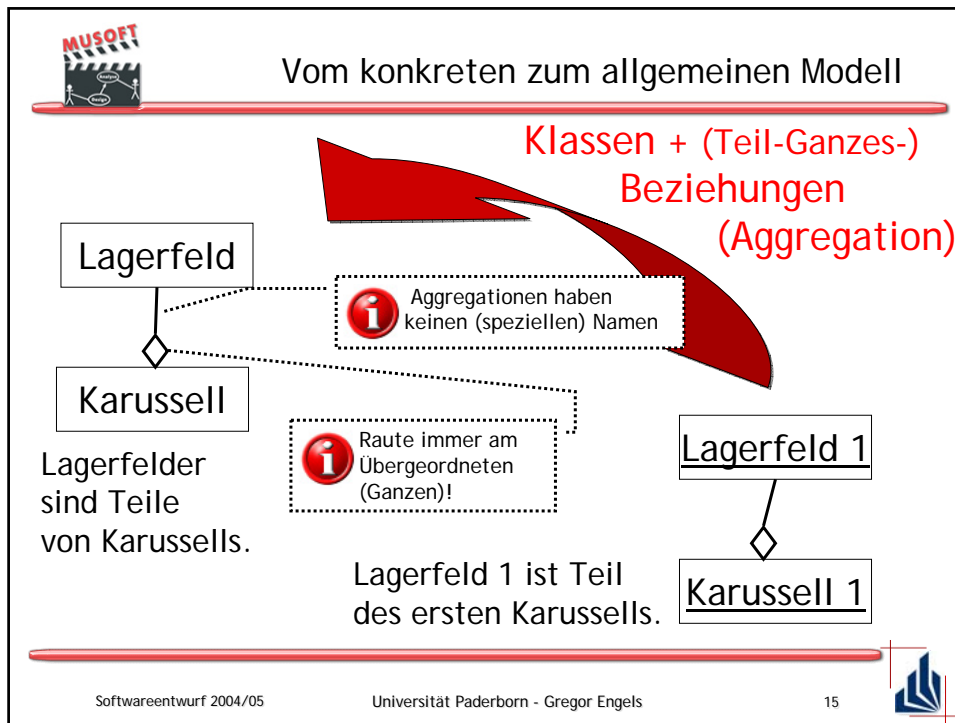
Linke Tür

Es gibt eine rechte  
und eine linke Tür.

**i** Klassennamen  
im Singular!

Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      12







**MUSOFT** (UML-) Klassendiagramme

- beschreiben allgemein mögliche Situationen
- werden zur Beschreibung des Modells des Problembereichs verwendet
- enthalten
  - Klassen (Mengen gleichartiger Objekte)
  - Beziehungen zwischen Klassen (Assoziationen)
  - Eigenschaften (Attribute) von Klassen (mit Name und Typ)

```

classDiagram
    class Tür
    class Karussell
    class Lagerfeld {
        zulässigesGewicht : Dezimal
    }
    Tür --> Karussell : ermöglicht Zugang zu
    Karussell o-- Lagerfeld
    
```

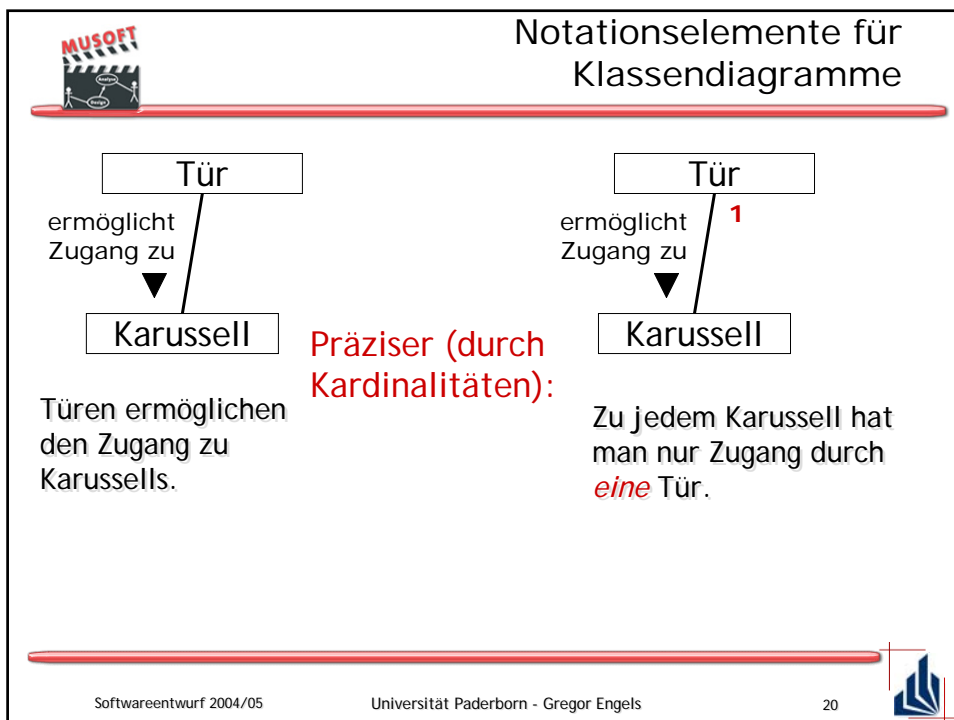
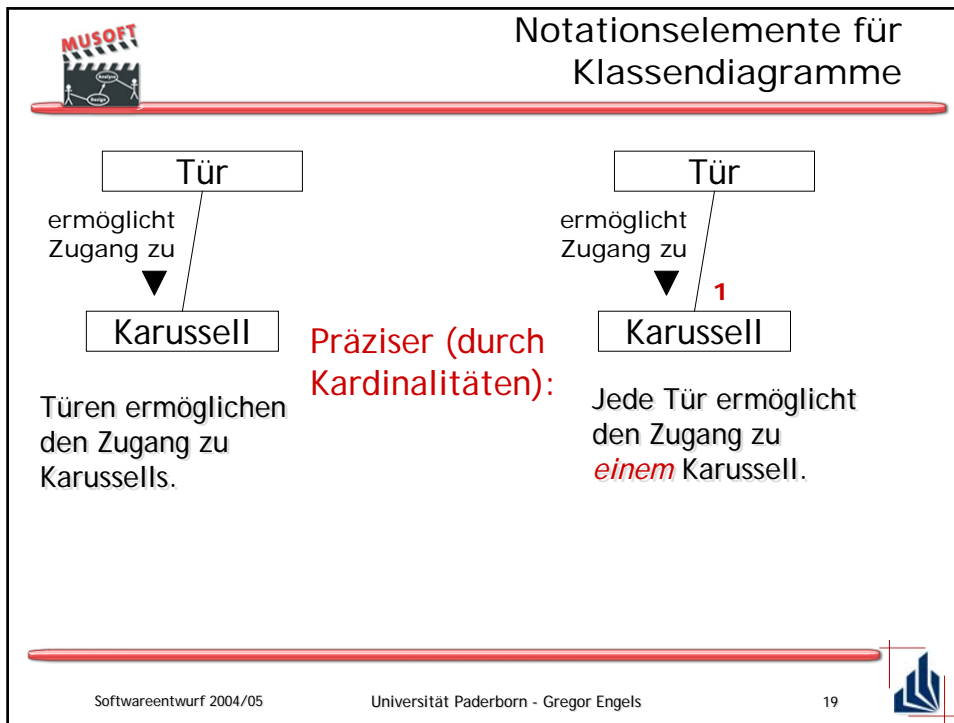
Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      17

**MUSOFT** Notationselemente für Klassendiagramme

**Klassen auf- und zuklappen:**

Klasse Karussell hat keine Attribute.  
 Diagramm macht keine Aussage, ob Klasse Karussell und Klasse Lagerfeld Attribute haben.

Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      18



MUSOFT

## Notationselemente für Klassendiagramme

---

**Kardinalitäten für Beziehungen:**

```

classDiagram
    class Karussell
    class Lagerfeld
    class Lagerfach
    class Ware
    Karussell "1" -- "26" Lagerfeld
    Lagerfeld "1" -- "4..12" Lagerfach
    Lagerfach "0..1" -- "0..*" Ware : liegt in
    
```

- 0..1 ein oder kein
- 1 genau ein
- 0..\* oder \* beliebig viele (Default)
- 4..12 zwischen vier und zwölf
- 2, 4, 6 zwei, vier oder sechs

Unbestimmte Kardinalitäten sind „offiziell“ \*, inoffiziell Schlamperie!

---

Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      21

MUSOFT

## Notationselemente für Klassendiagramme

---

**Leserichtung von Kardinalitäten für Beziehungen:**

```

classDiagram
    class Ware
    class Lagerfach
    Ware --> "0..1" Lagerfach : liegt in
    
```

Jede Ware      liegt in      keinem oder einem Lagerfach

**Ausgangsklasse      Beziehungsname      Kardinalität      Zielklasse**

---

Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      22

MUSOFT

## Notationselemente für Klassendiagramme

---

### Geordnete Assoziationen

```

classDiagram
    class Karussell
    class Lagerfeld
    class Lagerfach
    Karussell "1" o-- "26" Lagerfeld : {ordered}
    Lagerfeld "1" o-- "4..12" Lagerfach : {ordered}
    
```

Lagerfelder eines Karussells besitzen eine Sortierung.  
 Lagerfächer eines Lagerfeldes besitzen eine Sortierung.

---

Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      23

MUSOFT

## Notationselemente für Klassendiagramme

---

### Finden von Generalisierungen: Schritt 1

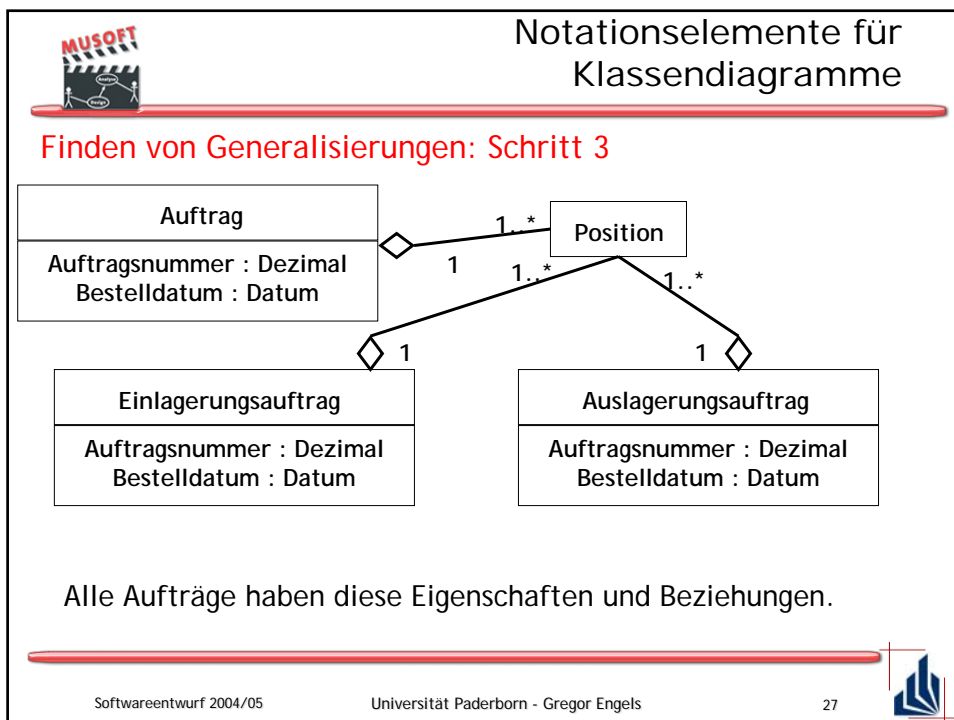
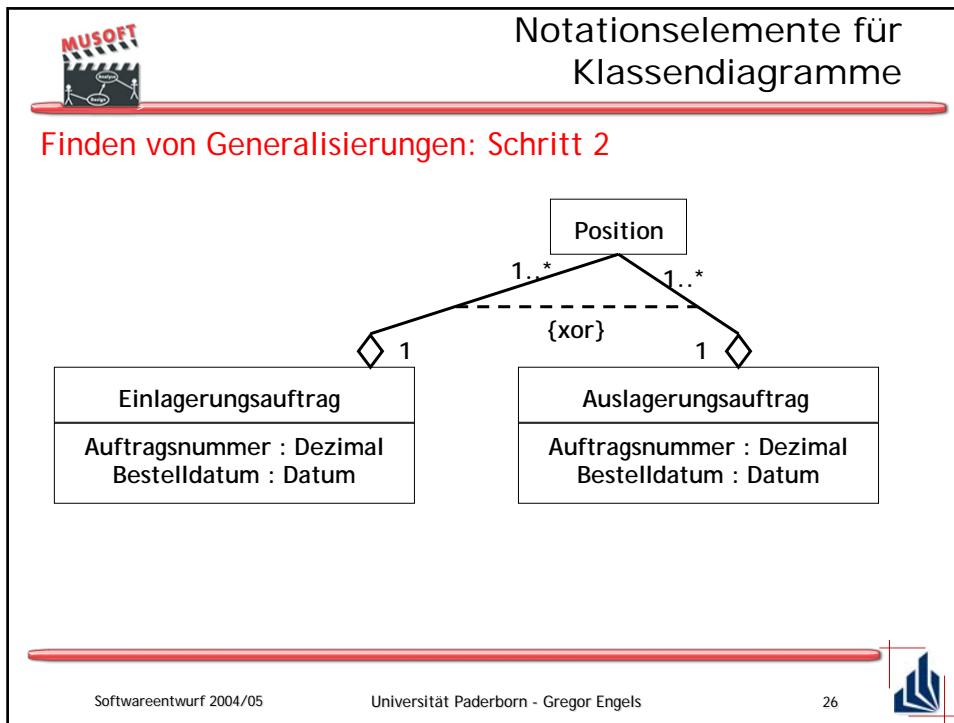
```

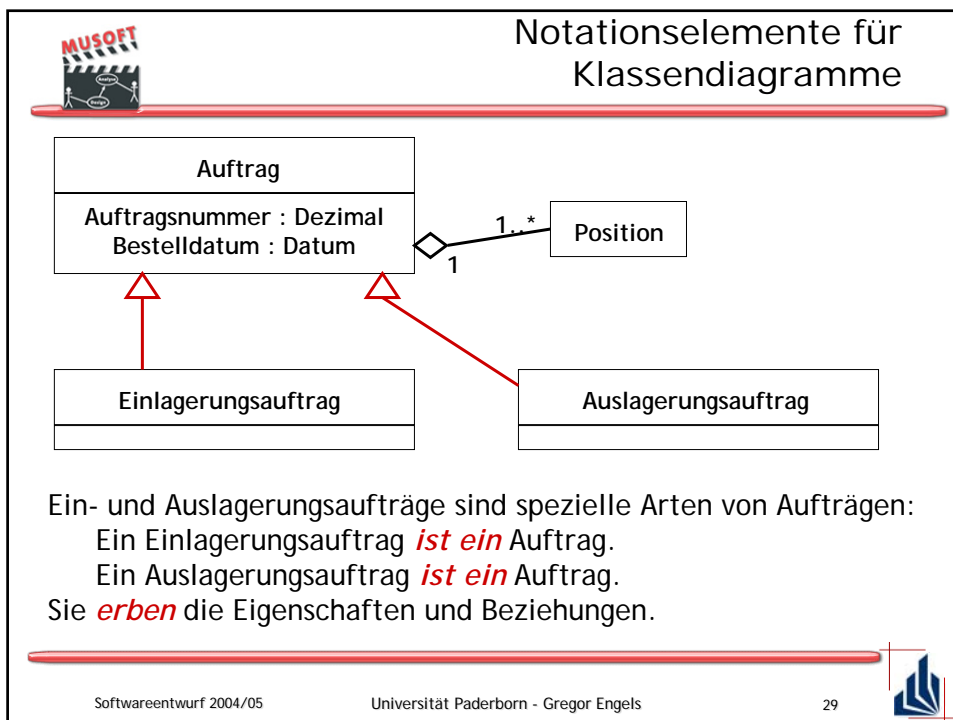
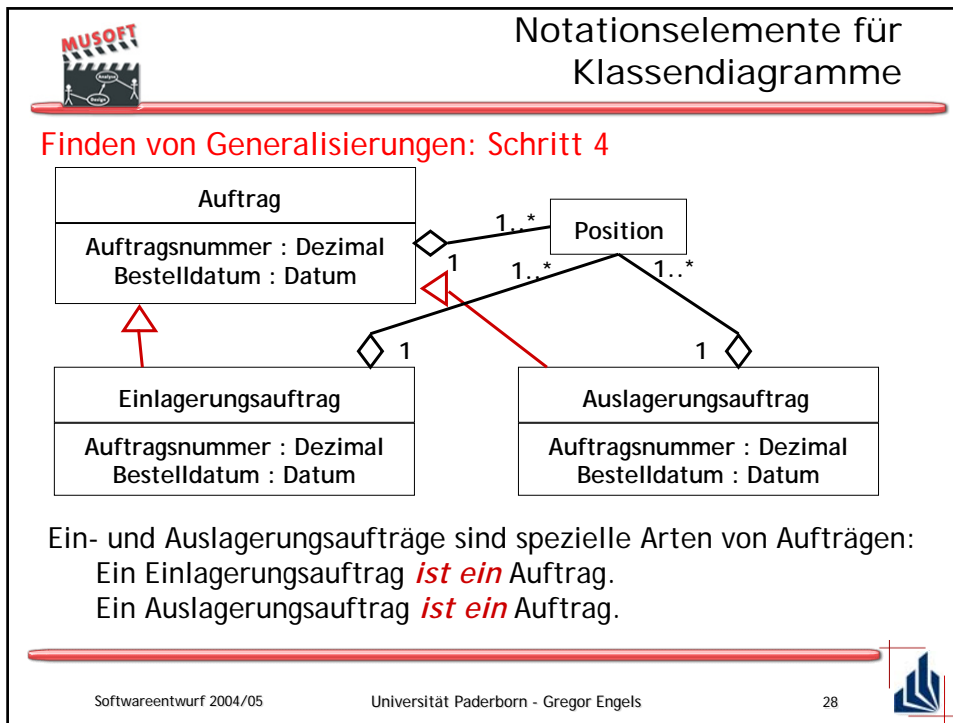
classDiagram
    class Position
    class Einlagerungsauftrag {
        Auftragsnummer : Dezimal
        Bestelldatum : Datum
    }
    class Auslagerungsauftrag {
        Auftragsnummer : Dezimal
        Bestelldatum : Datum
    }
    Position "1..*" o-- "1" Einlagerungsauftrag
    Position "1..*" o-- "1" Auslagerungsauftrag
    
```

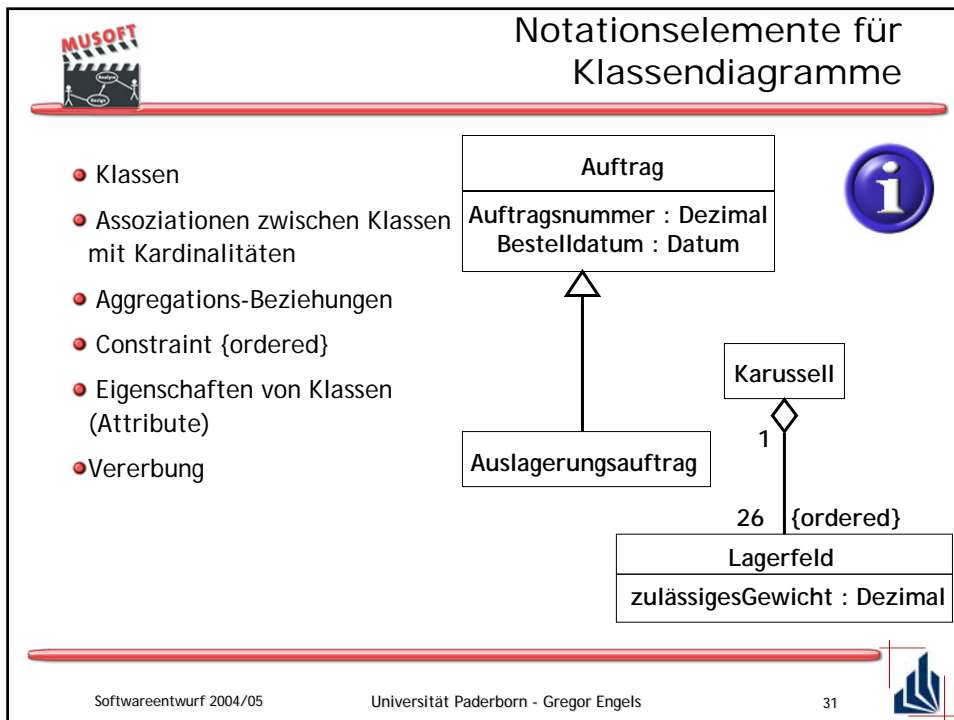
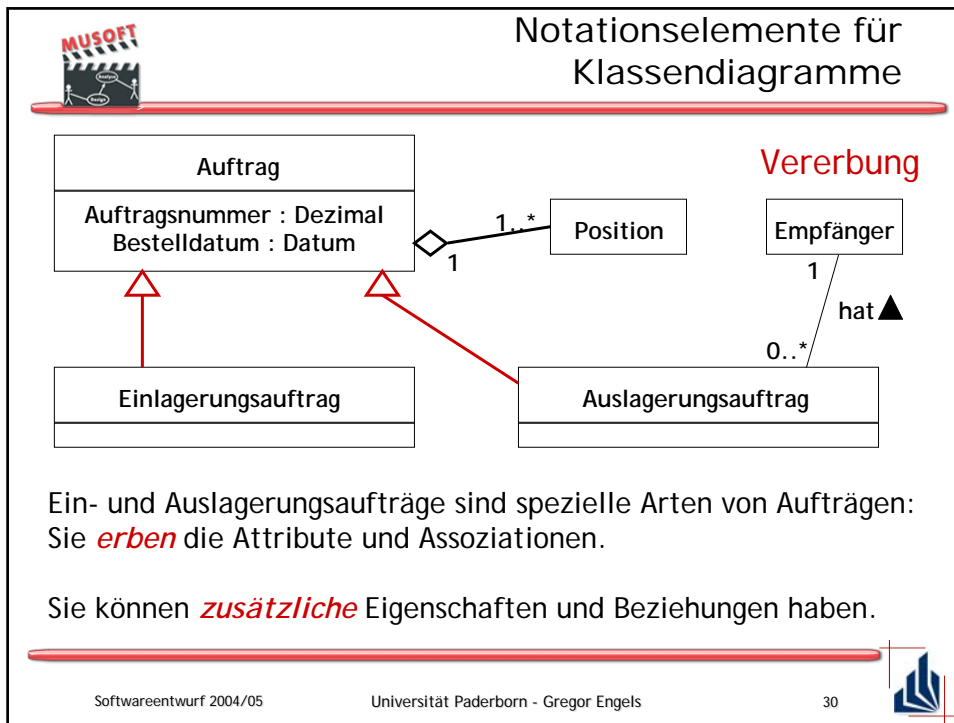
Ein- und Auslagerungsaufträge haben gemeinsame Eigenschaften und Beziehungen.

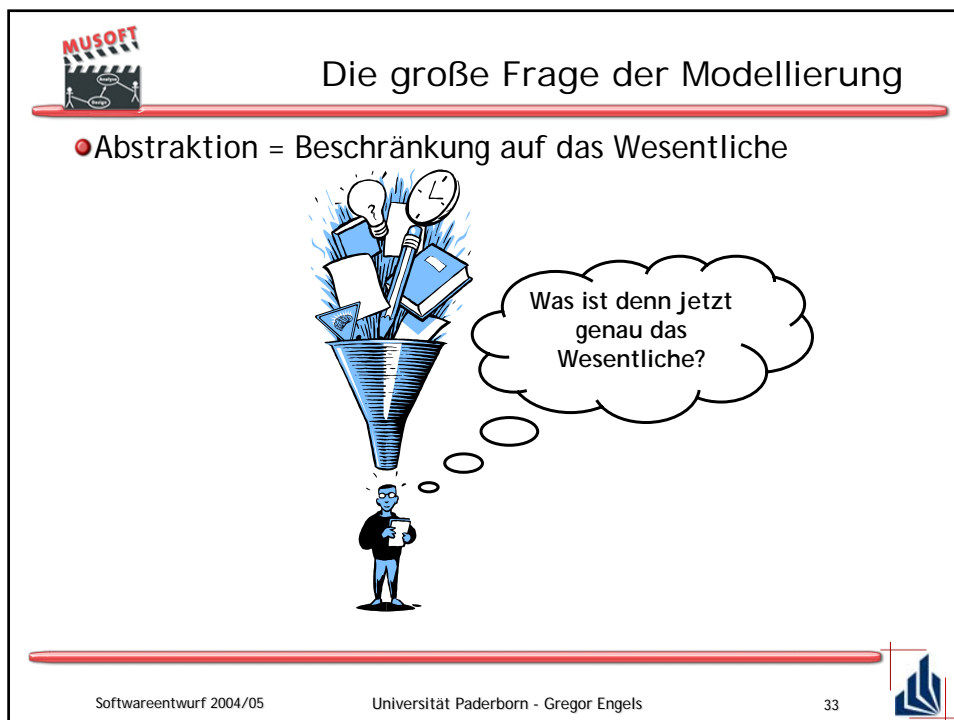
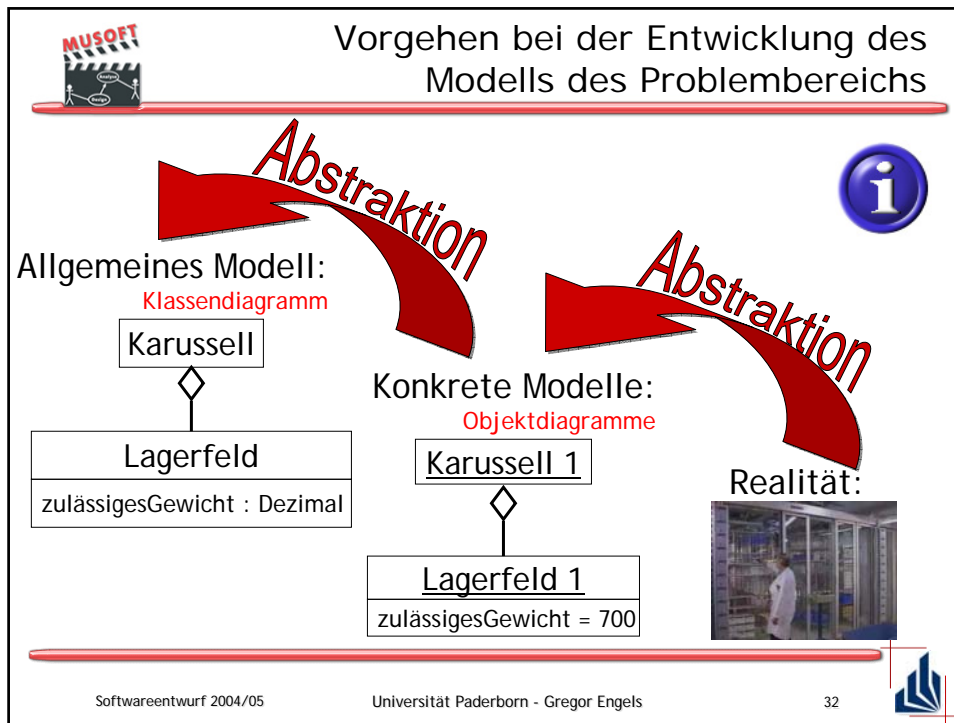
---

Softwareentwurf 2004/05      Universität Paderborn - Gregor Engels      25














## Qualitäten des MdPs

---


Ein gutes Modell des Problembereichs ist


- **Vollständig**  
Zwei Situationen in der Realität, die sich relevant unterscheiden, ergeben auch zwei unterschiedliche Objektdiagramme
- **Präzise**  
Es werden nur die relevanten Situationen beschrieben

Was für einen Problembereich relevant ist, ist nicht leicht zu erkennen!

---


Softwareentwurf 2004/05Universität Paderborn - Gregor Engels34






## Modell des Problembereichs und die Realität


---






**Pflichtenheft**


```
graph TD
  Tür -- 1 --> Karussell
  Tür -- 1 --> Auslagerungsauftrag
  Auslagerungsauftrag -- 1 --> Empfänger
```





**Code**

```
class Auslagerungsauftrag
  private myEmpf Empfänger
```



Warum kann ich jetzt nicht schnell für die Innere was mitbestellen?

---

Softwareentwurf 2004/05Universität Paderborn - Gregor Engels35

