

# Software- und Systementwicklung

Seminar: Designing for Privacy  
11.11.2009

Moritz Vossenberg

# Inhalt

- Vorgehensmodelle
  - Wasserfallmodell
  - V-Modell
  - Phasen (Pflichtenheft)
- UML
  - Klassendiagramm
  - Sequenzdiagramm
  - Use Case (Anwendungsfalldiagramm)
  - UML Erweitern
- Quellen

# **VORGEHENSMODELLE**

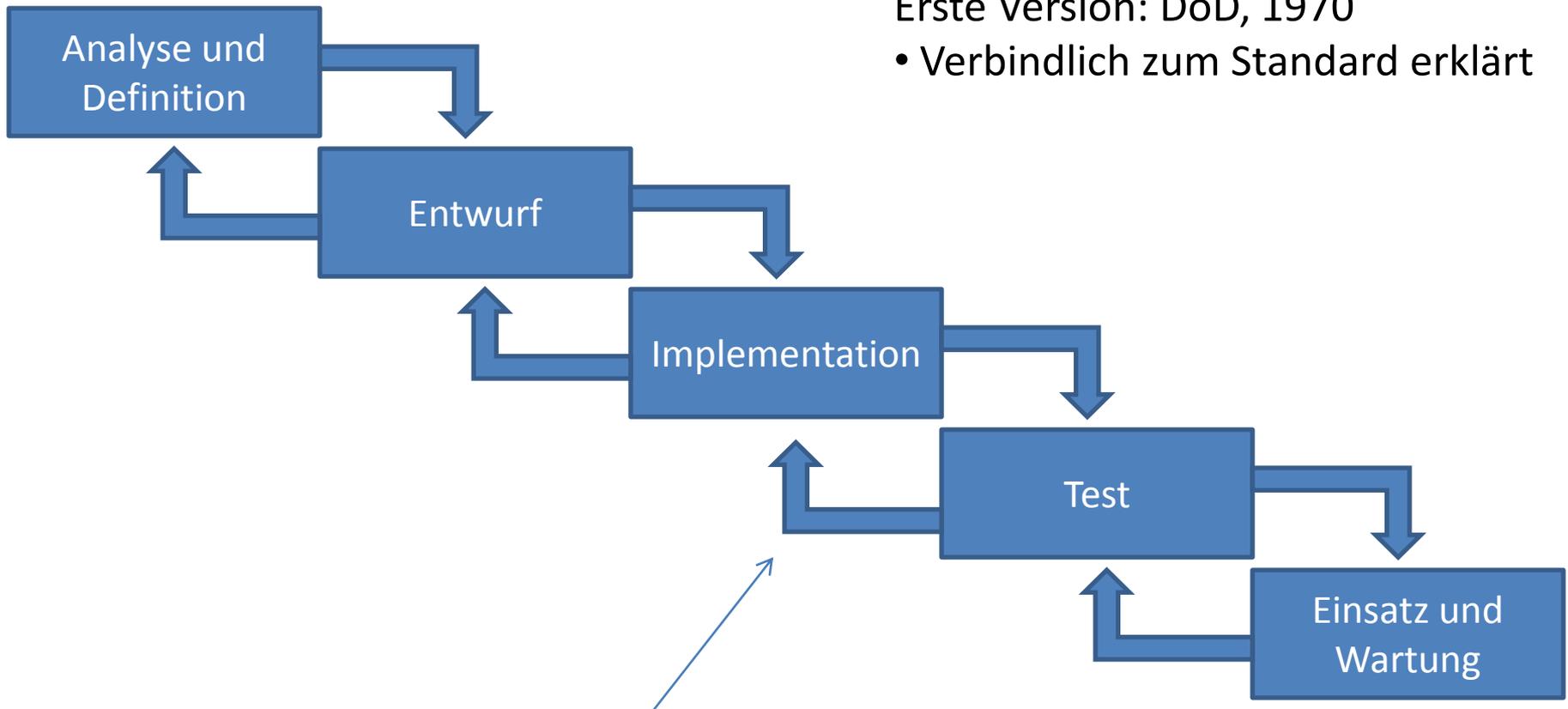
# Vorgehensmodelle (1)

- Definition:
  - „Festlegung, welche Aktivitäten in welcher Reihenfolge von welchen Personen in welchen Rollen erledigt und welche Ergebnisse dabei entstehen und wie diese überprüft werden.“

# Vorgehensmodelle (2)

- Wesentlich für ...
  - Projektorganisation
    - Systematische Herangehensweise, ...
  - Analyse von Projekten
    - Schwachstellen, ...
  - Zeit- und Aufwandsplanung
    - Aufwandsabschätzung, ...
  - Qualitative Bewertung von Software(-firmen)
    - ISO 9000, ...

# Wasserfallmodell



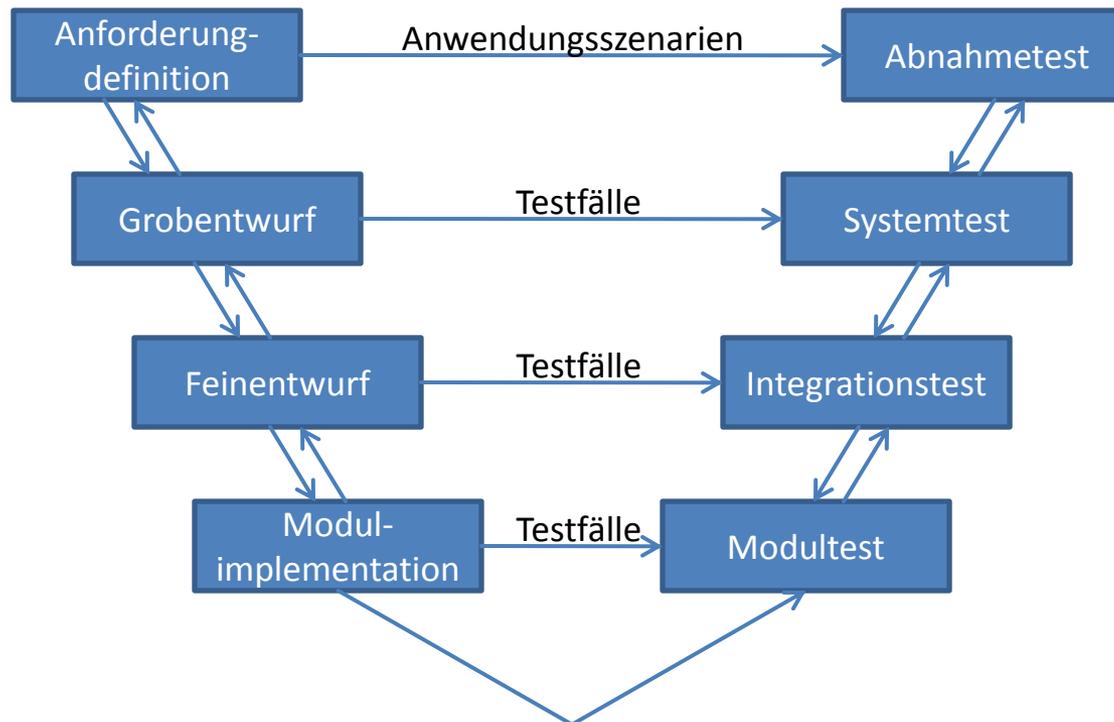
Inkrementelles Wasserfallmodell

# Wasserfallmodell

- Probleme
  - Strenge Reihenfolge
  - Nur zurück zur letzten Stufe
  - Keine Parallelarbeit
  - Keine inkrementelle Arbeit
- Grundlage der meisten anderen Modelle

# V-Model

- Erweitertes Wasserfallmodell
  - Qualitätssicherung / Tests in allen Phasen
- Standard für gesamten öffentlichen Bereich

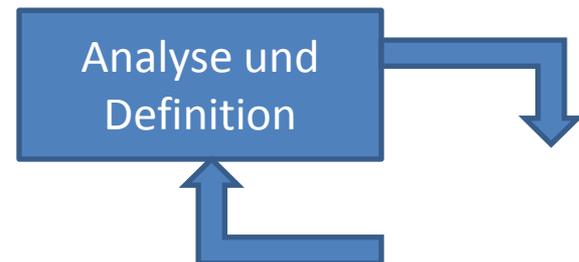


# Die Phasen (Wasserfall)

- Analyse und Definition
  - Analyse des Problems und Definition der Anforderungen
- Entwurf (Design)
  - Struktur, UML
- Implementation
  - Programmierung
- Test
  - Komponenten-, System-, Integrationstest, ...

# Dokumente der Phasen

- Analyse und Definition
  - Durchführbarkeitsstudie
    - Glossar, Lastenheft, Aufwandsabschätzung, Projektplan
  - Produktdefinition
    - Pflichtenheft, Produktmodell, Benutzeroberfläche, Benutzerhandbuch



# Das Pflichtenheft

- Funktionale Anforderungen
  - Funktionalität, Daten (logische Sicht), Benutzeroberfläche
- Anforderungen an die Einsatzumgebung
  - Einsatzsituation, Anwenderprofil
- Technische Anforderungen
  - Programmiersprache, Betriebssystem, Hardware
- Leistungsanforderungen
  - Effizienz, Datenumfang
- Validitätsanforderungen
  - Vorbereitung von Tests, Spezifikation von Testfällen
- Qualitätsanforderungen
  - Benutzerfreundlichkeit, Zuverlässigkeit
- Realisierungsanforderungen
  - Vorgehensmodell, Dokumentation, Vorschriften, Termine, Kosten, Entwickler

# UML

# UML - Einführung

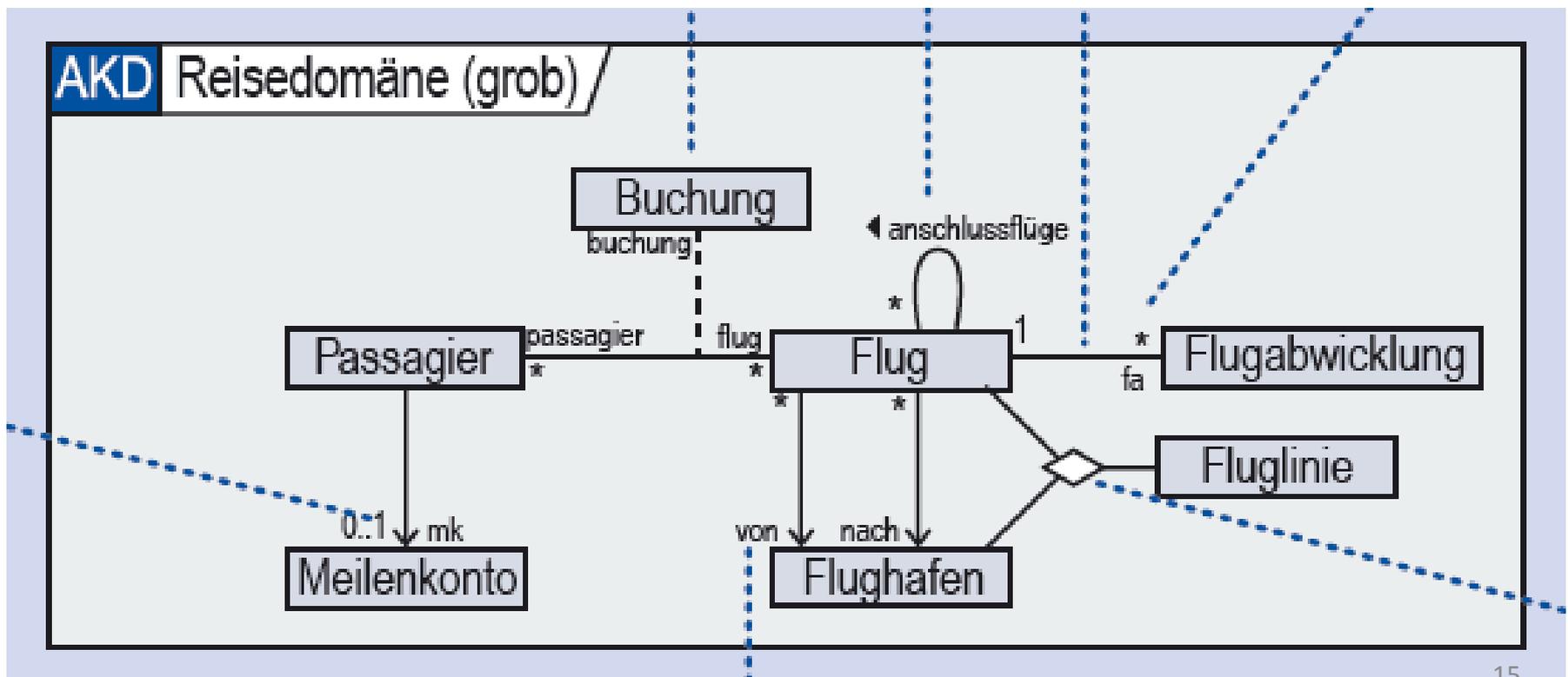
- UML entstand in den 1990ern aus dem Bedürfnis nach Standardisierung in der Methodik des Softwareentwurfs
- 1994 wird UML als Standard vorgeschlagen
- UML ist keine Methode, sondern Modellierungssprache
- Beitrag/Ziel der UML
  - Integration
    - Gemeinsame Infrastruktur der Teilsprachen (UML-Metamodell)
  - Kanonisierung
    - Palette von bewährten, aufeinander abgestimmten Konzepten
  - Standardisierung
    - Modelle und Wissen über Modelle können ausgetauscht werden
- Kann in allen Phasen des Softwareentwurfs benutzt werden

# Klassendiagramm

- Darstellung von Strukturen (im Ggs. zu Verhalten)
- Wichtige fachliche Konzepte werden als Klasse dargestellt
- Phasen im Software-Lebenszyklus
  - Analyse
    - Klassen als Konzepte der fachlichen Anwendungsdomäne
  - Entwurf
    - Klassen noch fachlich motiviert, mit technischen Aspekten
  - Implementation
    - Klassen als Äquivalente zu Klassen-Konstrukten der Implementierungssprache

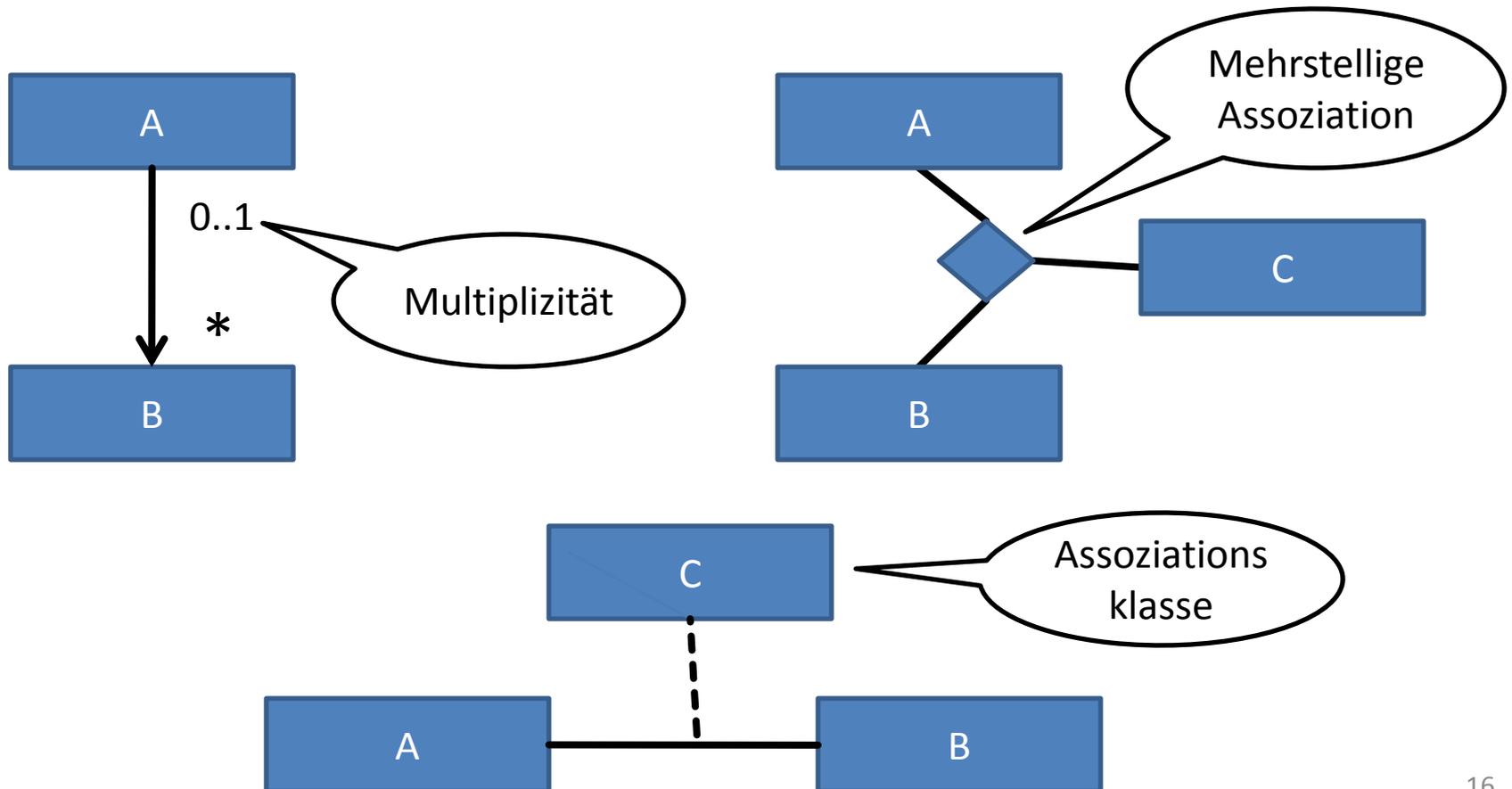
# Klassendiagramm: Beispiel

- Domäne: Ausschnitt der Realen Welt, welcher von der Anwendung betrachtet wird



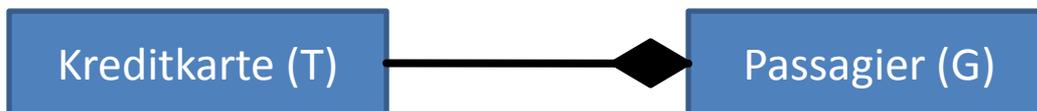
# Klassendiagramm: Assoziation

- Zeiger auf ein zweites Objekt

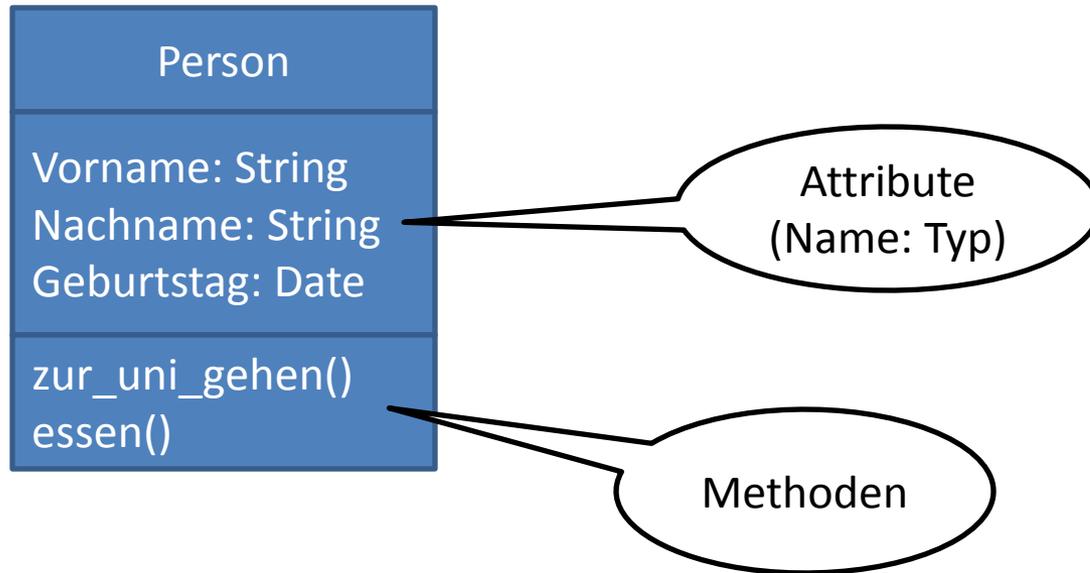


# Klassendiagramm: Komposition

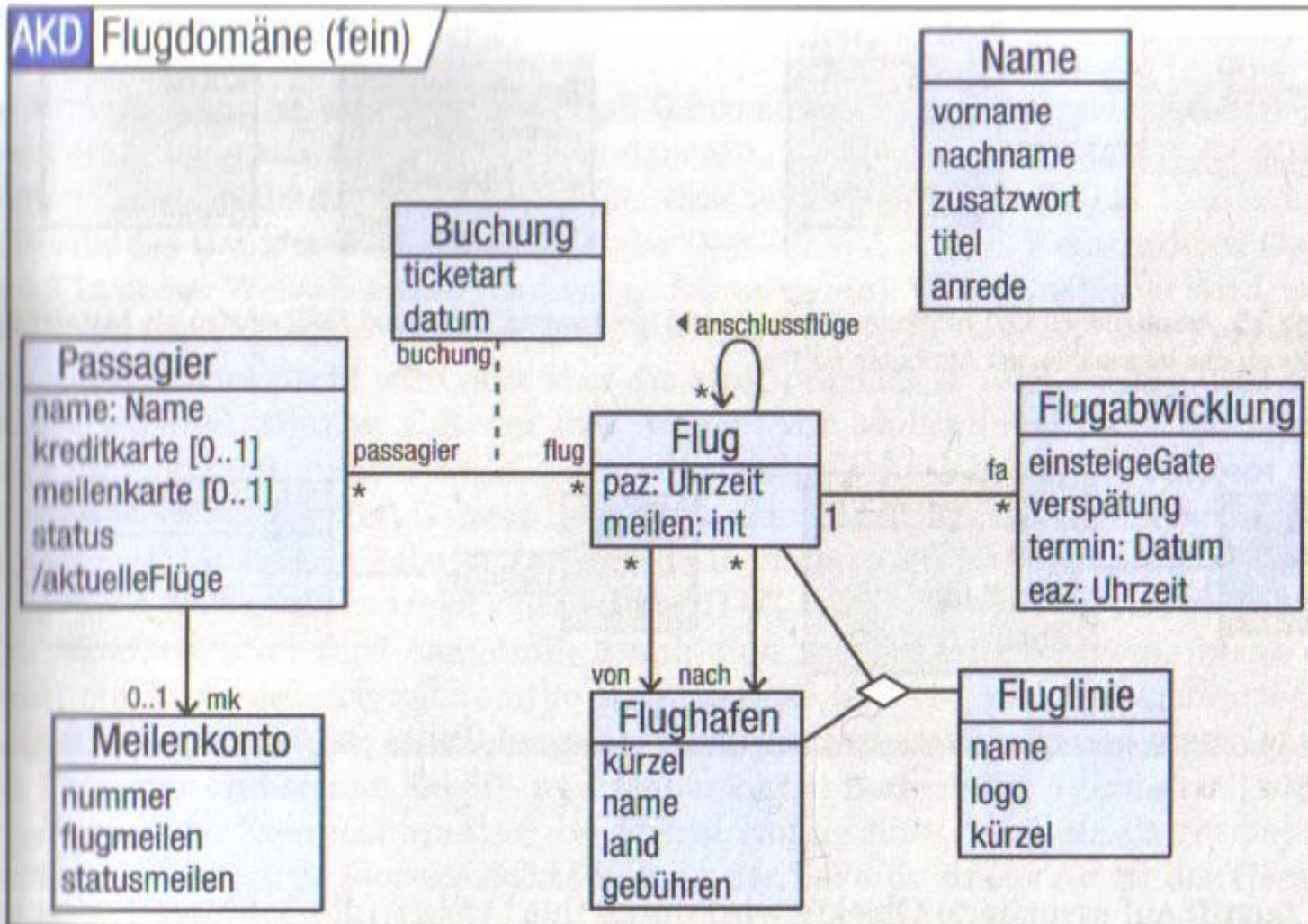
- Teil-Ganzes-Beziehung
  - Teil-Objekt T ist vollständig in Besitz des Ganzen-Objektes G
  - Andere Objekte können T nicht mehr besitzen
  - Wird G gelöscht, kann auch T nicht mehr existieren



# Klassendiagramm: Attribute & Operationen



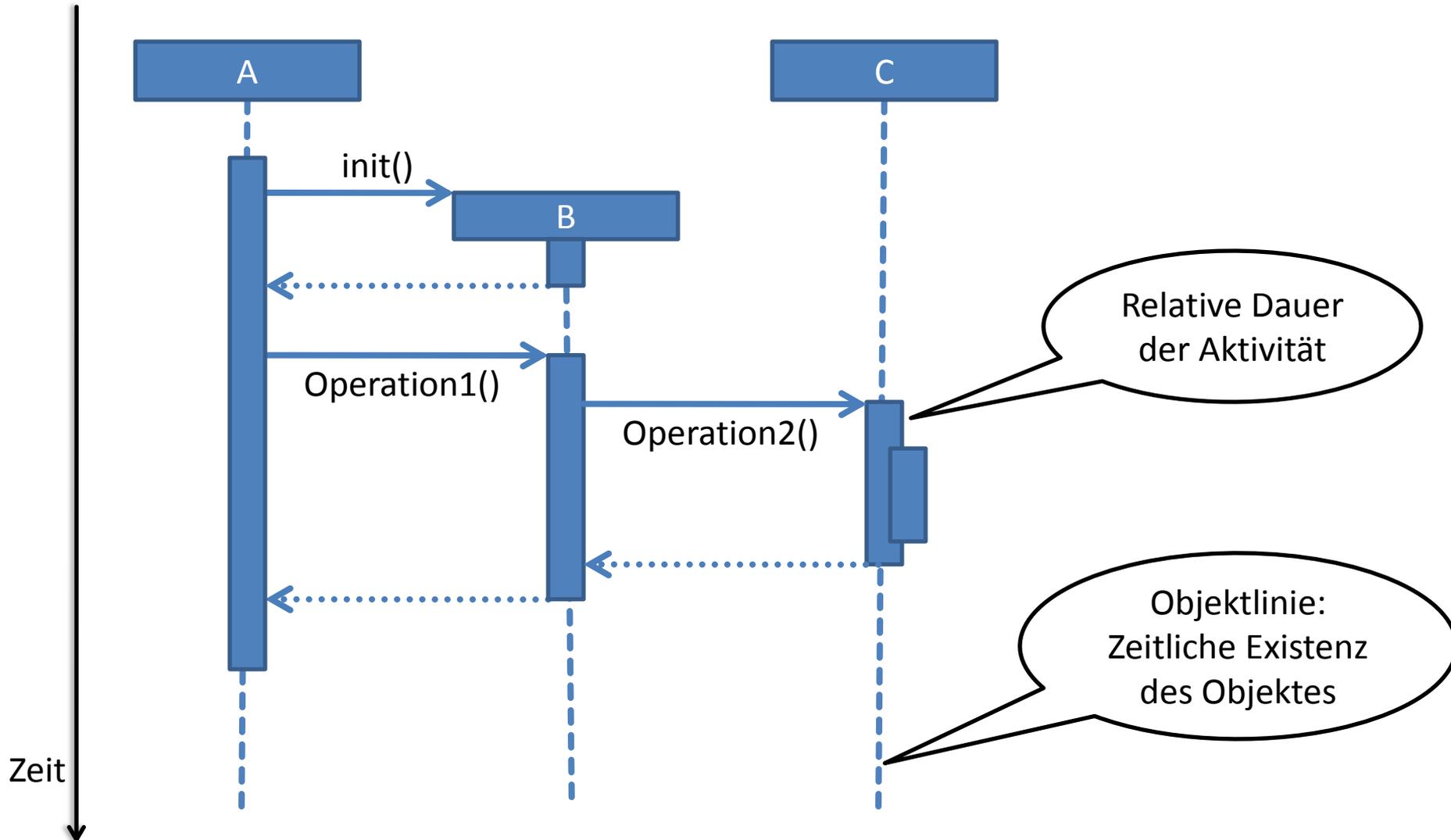
# Klassendiagramm: Beispiel 2



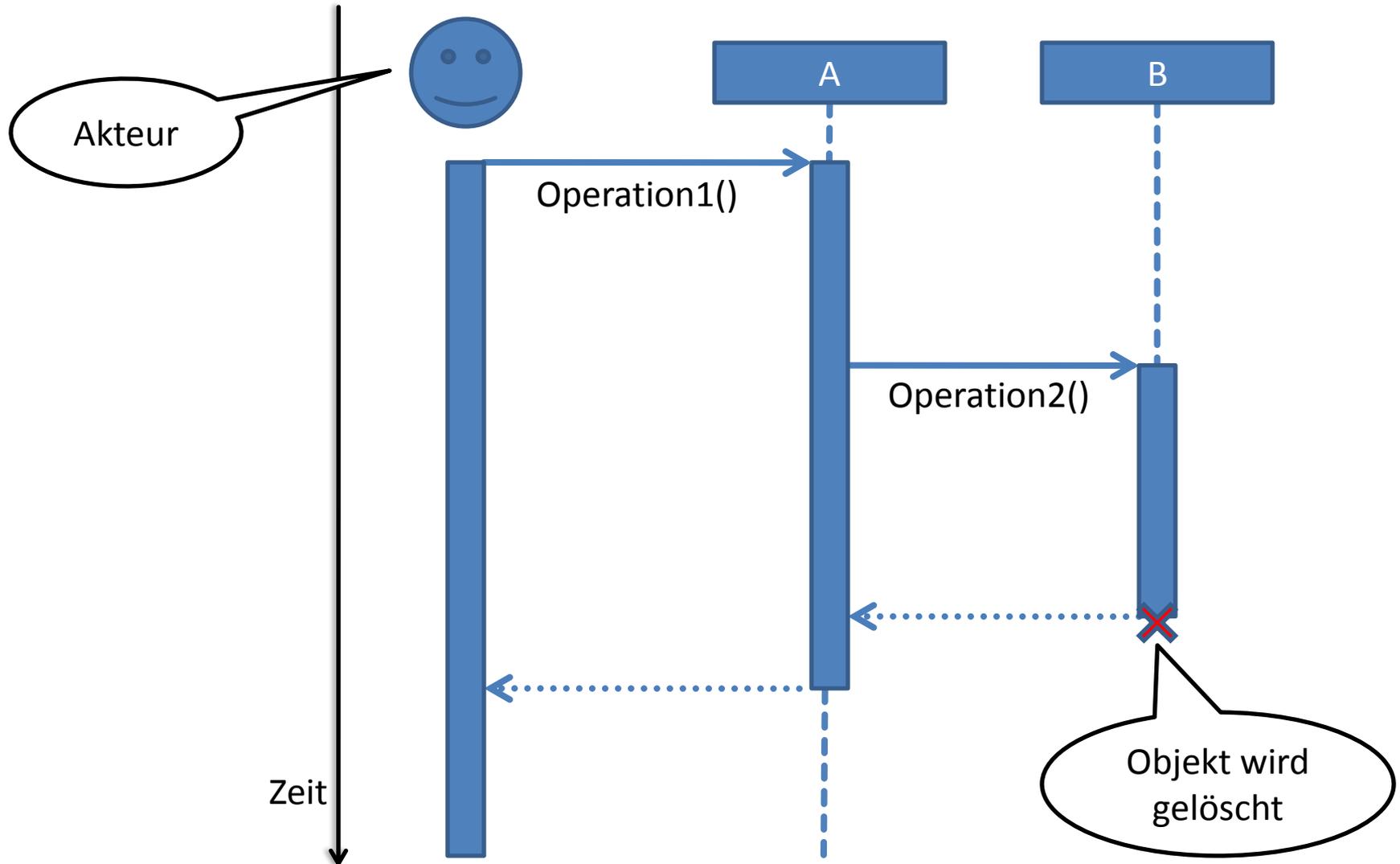
# Sequenzdiagramm

- Darstellung von Verhalten (im Gegensatz zur Struktur)
- Beschreibt die Zeitliche Abfolge von Interaktionen zwischen Objekten
- Basiskonzept der Szenario orientierten Sicht
- Reihenfolge (Zeitachse)
- Lebensdauer von Objekten

# Sequenzdiagramm: Syntax (1)



# Sequenzdiagramm: Syntax (2)

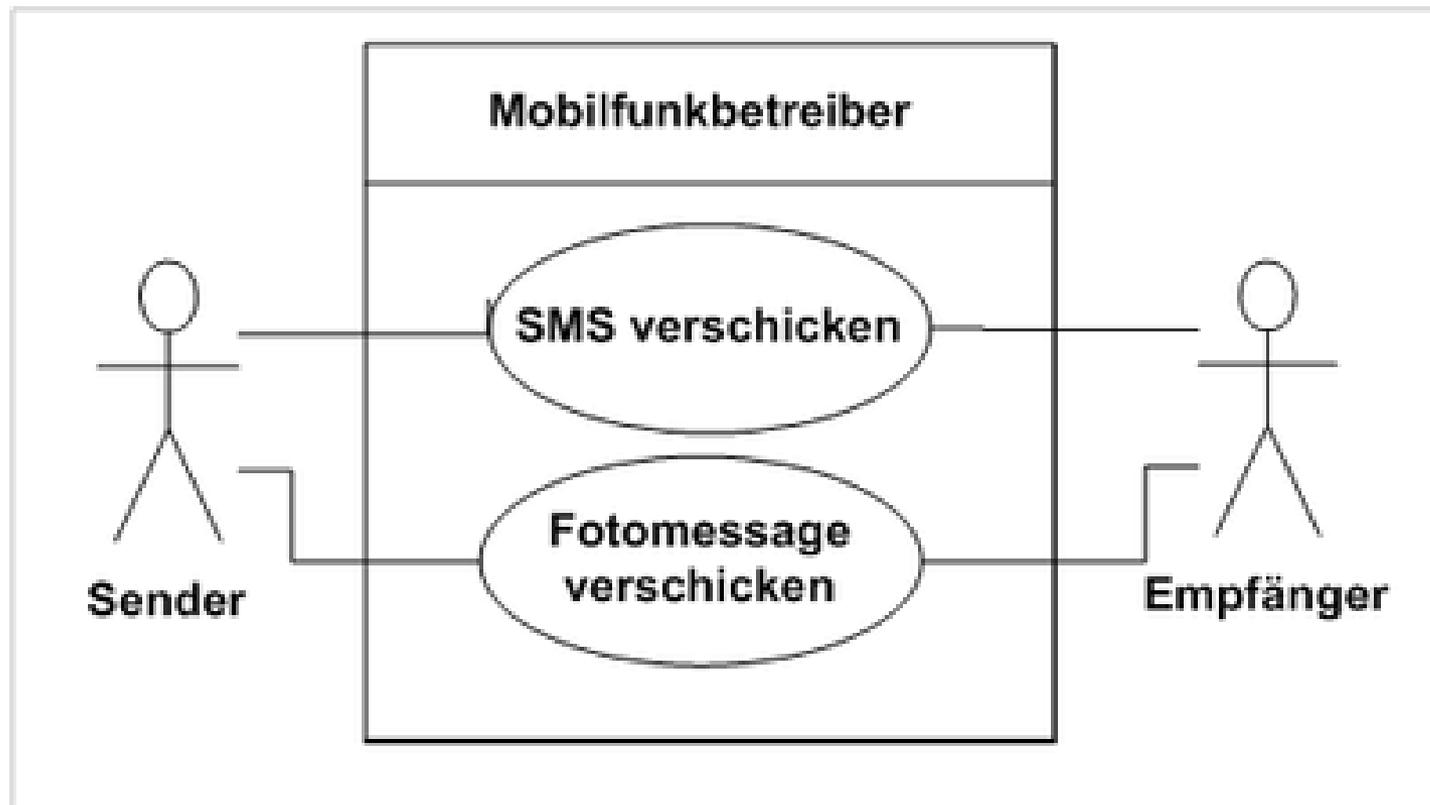


# Use Cases (1)

- Stellt Außenverhalten eines Systems dar
  - nur nach außen hin sichtbares Verhalten
- Beschreibt Aktionen, welche es Akteuren ermöglicht, ein fachlich motiviertes Ziel zu erreichen
- Kann/soll in UML das Verhalten beliebiger Modellelemente beschreiben
  - Klasse, Komponente, ...

# Use Cases (2)

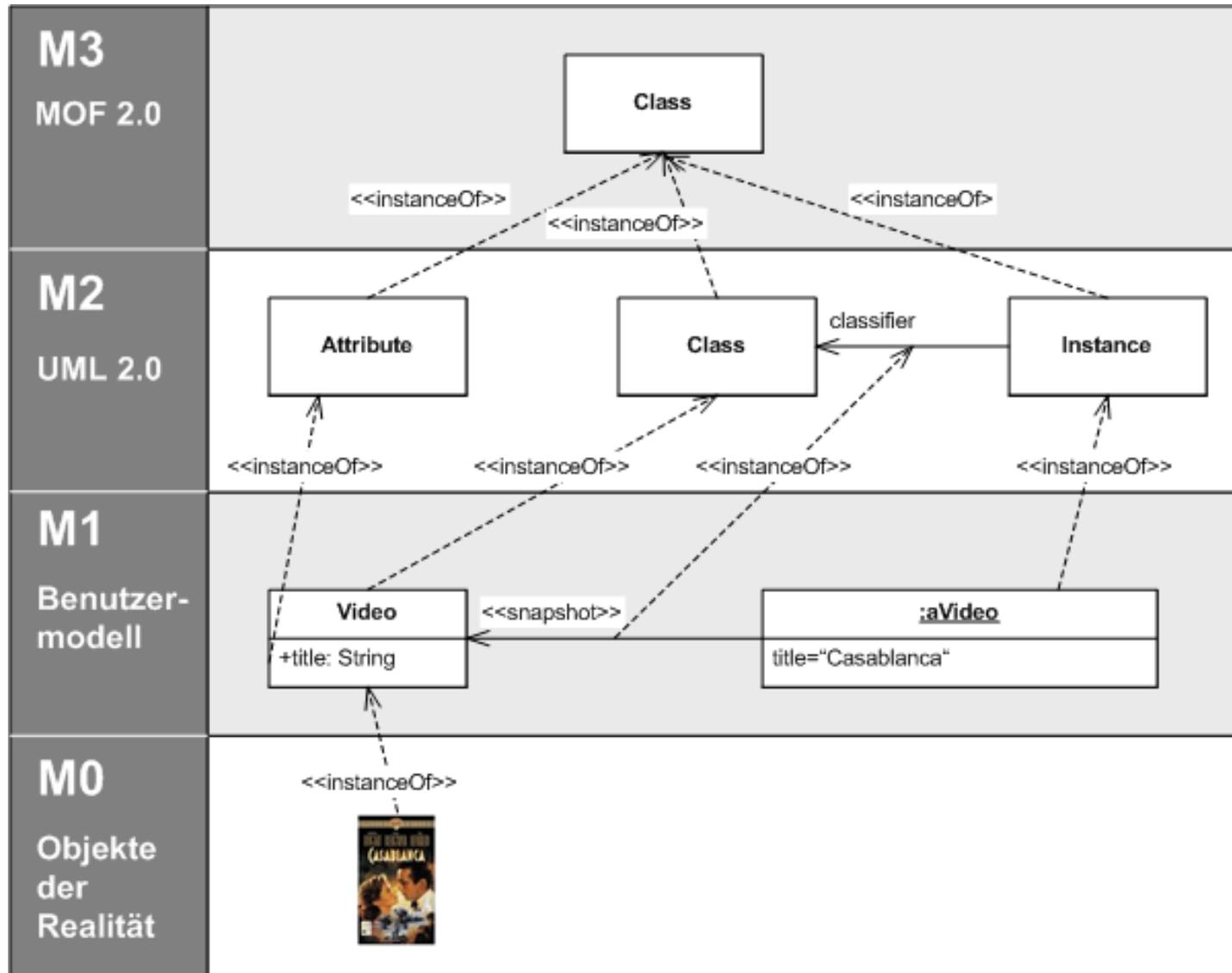
- Akteure stoßen Verhalten an und/oder sind vom Resultat abhängig



# UML Erweiterbarkeit

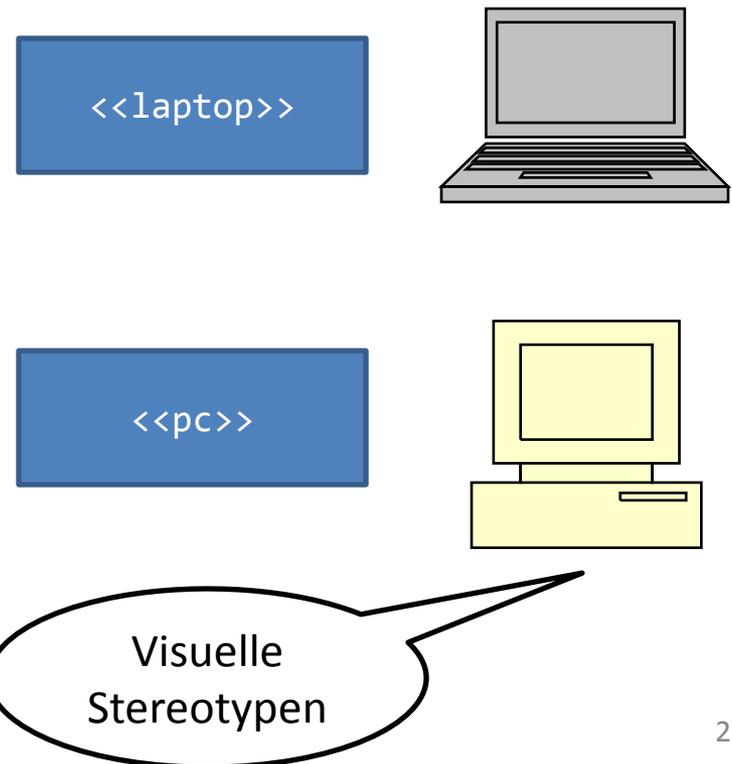
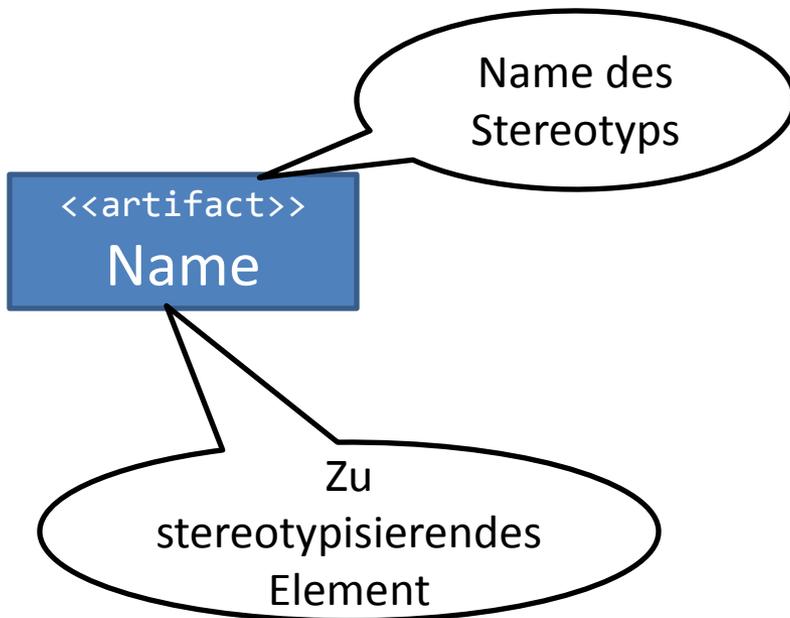
- UML erweitern bedeutet: Erweiterung des Metamodells
  - Neue Klassen
  - Neue Attribute
- Bestehendes kann nicht entfernt werden
- Z.B. verwendet für nichtfunktionale Anforderungen
- Drei Möglichkeiten der Erweiterung...

# Das UML-Metamodell



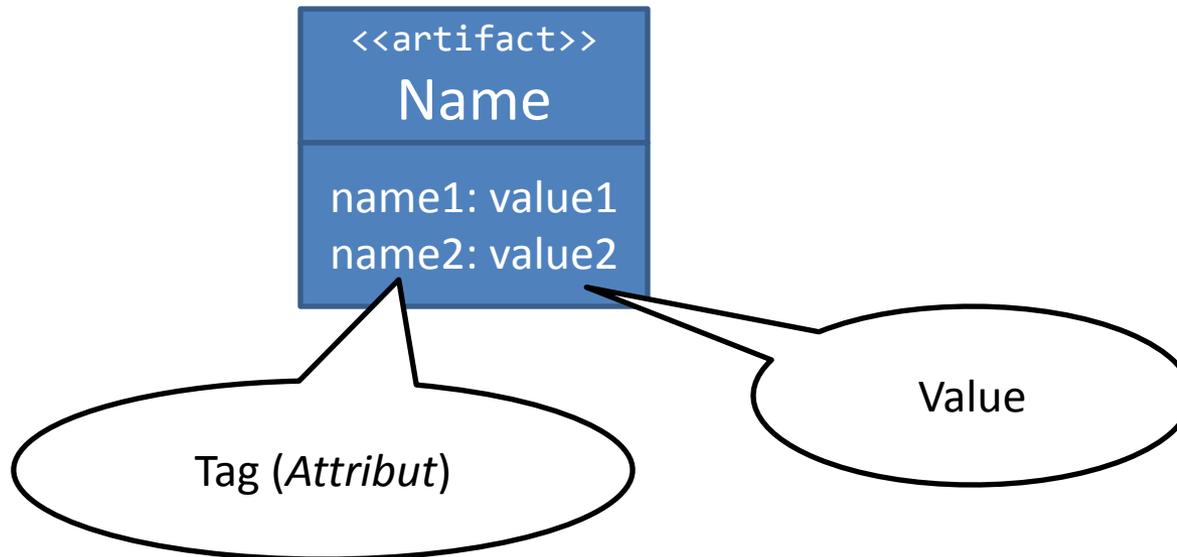
# UML erweitern (1)

- Stereotyp
  - Kennzeichnung spezifischer Elemente eines Modells („Etikett“)



# UML erweitern (2)

- Tagged Values



# UML erweitern (3)

- Profile
  - Zusammenfassung von Stereotypen und Tagged Values zu Profilen

**FRAGEN?**

# Quellen

- Vorlesung „Software Engineering“, Sommersemester 2009, Prof. Bothe.
- Störrle, Harald: UML 2 für Studenten. Pearson Studium, 2005.
- Wikipedia: Anwendungsfall. URL: <http://de.wikipedia.org/wiki/Anwendungsfall>, abgerufen am: 11.11.2009
- Wikipedia: Unified Modeling Language. URL: [http://de.wikipedia.org/wiki/Unified Modeling Language](http://de.wikipedia.org/wiki/Unified_Modeling_Language), abgerufen am: 11.11.2009