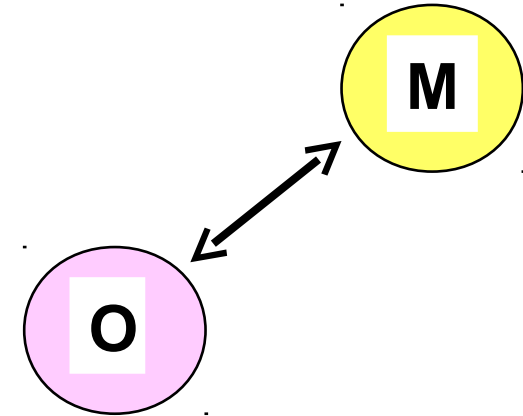


Kap. 3: Modelle und Grundlagen der Modellierung

- **Modellieren – wozu?**
- **Modelle: Definitionen**
- **Modell und Original**
- **Modelle und ihre Darstellung**
- **Modelle als Vor- und Nachbilder**
- **Ansätze und Konzepte dynamischer Modellierung**
- **Literatur**

Wozu Modellieren?

- Menschliche kreative Tätigkeit: ohne Modelle *undenkbar*
- Reale Welt steckt voller *komplexer* Zusammenhänge
- Modell erlaubt, *vereinfachende, reduzierende Sicht* auf einen Teil davon (das "*Original*") herzustellen und sich daran zu orientieren.



- Komplexe geistige Aufgaben: nur lösbar, wenn man sich nacheinander auf *ausgewählte Aspekte* konzentrieren kann.
- Mit Hilfe von Modellen kann man:
 - *Überblick* über komplexe Strukturen gewinnen,
 - Informationen und Erfahrungen sammeln,
 - Alternativen bewerten, Vergleiche anstellen, Schlüsse ziehen,
 - Entscheidungen treffen, Strategien entwickeln,
- .. - die *für die Bearbeitung des Originals von Nutzen* sind,
- .. - die sich an diesem selbst nicht (so leicht) vollziehen lassen.

Zum Modellbegriff: Definitionsversuche

Model: A semantically closed abstraction of a system (cf. [UML 99], Glossary)

Vorsicht: "semantically closed" ist nicht näher definiert, sehr vager "Semantik"begriff
Gemeint (?): eine gewisse Kohärenz des Modells.

Model: An abstraction of a physical system with a certain purpose (cf. [UML 03], Glossary)

Vorsicht: Muss es immer "*physical*" sein? Immer ein „*System*“? Ist „*Abstraktion*“ einziges Merkmal?

Modell: Idealisierte, vereinfachte, in gewisser Hinsicht ähnliche Darstellung eines Gegenstands, Systems oder sonstigen Weltausschnitts mit dem Ziel, daran bestimmte Eigenschaften des Vorbilds besser studieren zu können. (vgl. [HBB+ 94])

Achtung! "Vorbild" ist dabei *nicht zeitlich* zu verstehen. Modelle können sowohl *nach einem Vorbild* oder *als Vorbild* für ein zu erstellendes Produkt oder System gebildet werden (vgl.unten).

Modellbegriff / Merkmale

Herkunft des Modellbegriffs:

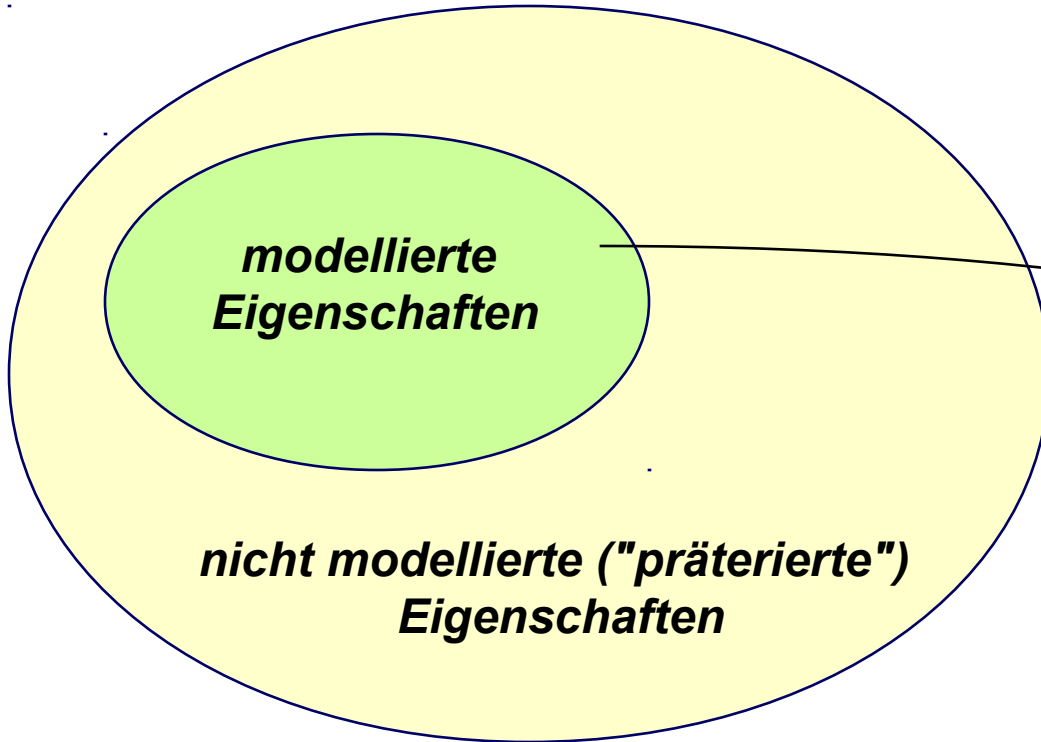
modulus (lat): Maß, Regel, Form, Muster, Vorbild

Merkmale von Modellen (n. Stachowiak, [Sta 73], vgl. auch: Ludewig [Lud 02]):

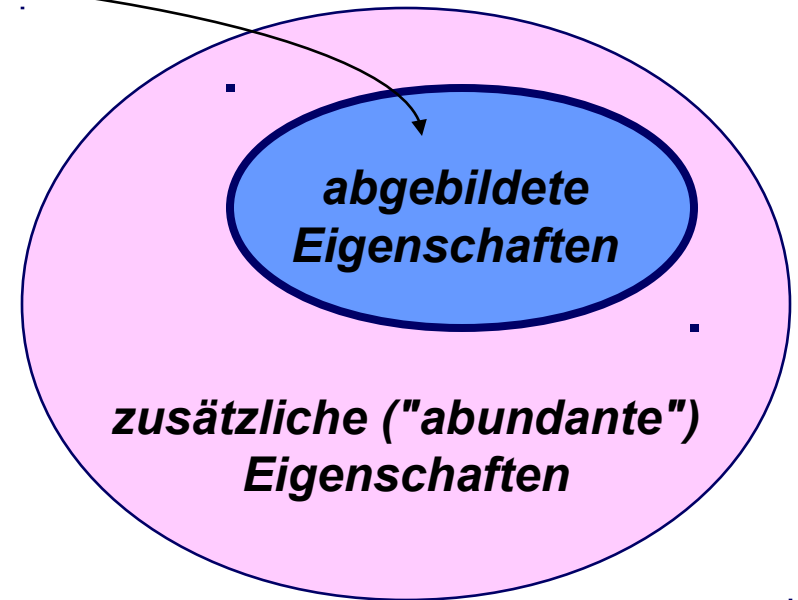
- (A) *Abbildungsmerkmal*:** Ein Modell steht immer für etwas anderes, das "Original" - ohne Original kein Modell
- (R) *Reduktionsmerkmal*:** Ein Modell weist nicht alle Eigenschaften des Originals auf, sondern nur einige - und auch die möglicherweise in veränderter, "ähnlicher" Form
- (P) *Pragmatisches Merkmal*:** Ein Modell hat den Zweck, unter bestimmten Bedingungen und bezüglich bestimmter Fragestellungen das Original zu ersetzen.

Original und Modell (n. Stachowiak)

Original:

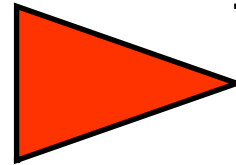


Modell:

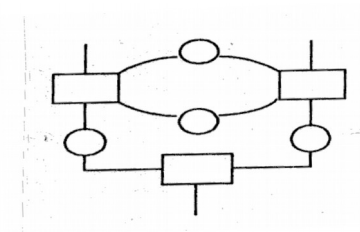


System / Modell / Original

Original (System)



Modell



Relation $O \triangleright M$

- "System" ... oder besser: "*Original*"?
(\Rightarrow Stachowiak)
- "System" setzt bestimmte Eigenschaften voraus, "*Original*" bezeichnet dagegen eine *Rolle*.

- Ein "*Modell*" repräsentiert immer ein Original
 - als Stellvertreter
 - mit seinen abgebildeten Eigenschaften

Modell und Original

.. sind vielleicht "ähnlich", aber oft in vielerlei Hinsicht sehr unterschiedlich!

Beispiel: *Person* ► *Photo*

- Modelle betonen bestimmte Aspekte der Realität und stellen andere in den Hintergrund: ein Modell kann die Realität nie vollständig erfassen.
- Ein Modell arbeitet mit Annahmen und Vereinfachungen.
- Modelle sind häufig nicht eindeutig (in Bezug auf das modellierte Original)



Weiteres Beispiel: *Landschaft* ► *Landkarte*

- Eine Landkarte kann als Modell (= Vereinfachung) nie alle Aspekte der abgebildeten Landschaft enthalten.
- Surreales "Gegenbeispiel": *Umberto Eco: Die Karte des Reiches im Maßstab 1:1*, s. [Eco 93]

Original und Modell: Eigenschaften

- Das Original existiert möglicherweise nicht in der (konkreten, materiellen) Realität → *fiktives* Original.

Beispiel:

- Ein Modell kann selbst wieder Original für ein (weiteres) Modell sein → *Modellketten* oder - *kaskaden, Metamodelle*

Beispiel:

- Eigenschaften des Originals finden sich entweder (möglicherweise in veränderter Form) im Modell wieder - oder werden unterdrückt ("abstrahiert") → *nicht modellierte ("präterierte") Eigenschaften*

Beispiel:

- Eigenschaften des Modells sind entweder (möglicherweise in veränderter Form) aus dem Original abgeleitet - oder treten neu hinzu → *zusätzliche ("abundante") Eigenschaften*

Beispiel:

Original und Modell: Eigenschaften

- Das Original existiert möglicherweise nicht in der (konkreten, materiellen) Realität → *fiktives* Original.

Beispiel: Modell der Stadt Entenhausen

- Ein Modell kann selbst wieder Original für ein (weiteres) Modell sein → *Modellketten* oder - *kaskaden, Metamodelle*

Beispiel: UML-Metamodell (für die Kategorien von UML-Modellelementen)

- Eigenschaften des Originals finden sich entweder (möglicherweise in veränderter Form) im Modell wieder - oder werden unterdrückt ("abstrahiert") → *nicht modellierte ("präterierte") Eigenschaften*

Beispiel: Puppe – ähnlich, aber ohne "Fleisch und Blut"

- Eigenschaften des Modells sind entweder (möglicherweise in veränderter Form) aus dem Original abgeleitet - oder treten neu hinzu → *zusätzliche ("abundante") Eigenschaften*

Beispiel: Gebäudemodell aus Pappe: Klebstoff

Beispiele von Modellen

außerhalb der Informatik:

- Geographie:
- Architektur:
- Mathematik:
- Physik / Chemie:
- Biologie:
- Spielzeug:
- Mode / Unterhaltung:

...

in der Informatik:

-

-

-

- ...

Beispiele von Modellen

außerhalb der Informatik:

- Geographie: Land-, See-, Himmelskarten
- Architektur: Blaupausen, Modelle für Gebäude u. andere Bauwerke
- Mathematik: Graphen, Verbände, Algebraische Strukturen
- Physik / Chemie: Atomium, Molekularstrukturen, chem. Elemente
- Biologie: DNS-Struktur, Doppel-Helix
- Spielzeug: Eisenbahnen, Autos, Puppenhäuser, ..
- Mode / Unterhaltung: Ersatzperson
- ...

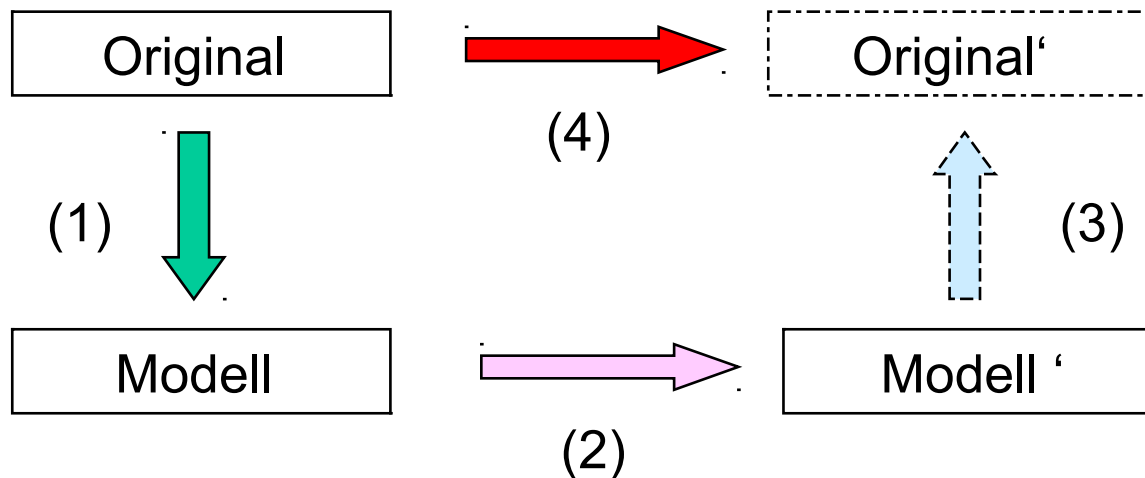
in der Informatik:

- Daten- und Prozessmodelle, Vorgehensmodelle
- Architekturmodelle
- Analyse- und Entwurfsmuster
- ...

Modellmethode

Die **Modellmethode** vollzieht **sich in vier Schritten**:

- (1) **Auswahl**: Herstellung eines dem Original entsprechenden Modells
- (2) **Bearbeitung** des Modells, um neue Informationen über Original und Modell zu gewinnen (Modellversuch)
- (3) **Schluss** auf Informationen über das Original (meist Analogieschluss)
- (4) (ggf.) **Durchführung** der Aufgabe am Original.



Zum Modellbegriff: Verwandte Begriffe

System:

Theorie:

Vergleich, Analogie, Metapher:

Paradigma:

Zeichen, Symbol:

Werkzeug:

--> vgl. auch Ludewig [Lud 02]

Zum Modellbegriff: Verwandte Begriffe

System: Schwerpunkt liegt auf verallgemeinerbaren Eigenschaften (z.B. Komponenten-Struktur, Emergenz), ist i. A. kein Vor-/Nachbild.

Theorie: stützt sich i. A. auf ein oder mehrere Modell(e), macht hypothetische Aussagen über den modellierten Bereich.

Vergleich, Analogie, Metapher: Modell ist mehr: repräsentiert ein Original und dessen (wichtige) Eigenschaften, wird analysiert und bearbeitet .

Paradigma: Denkweise, Denkschule. Kann als "Modell" (i.S.v. Beispiel, Muster) für Ableitungen dienen.

Zeichen, Symbol: hat bloße Repräsentations-Funktion; Modell soll dagegen ähnlich, analysier- und bearbeitbar sein und Ableitungen ermöglichen.

Werkzeug: Ein Modell kann (mit) als Werkzeug bei einem Projekt eingesetzt werden. Ein Werkzeug kann auf einer bestimmten Modellvorstellung aufbauen - repräsentiert aber i. A. nicht selbst ein Original.

Abgrenzung des Modellbegriffs

Modell vs. Darstellung

- Problem: Nicht-Unterscheidung zwischen (kognitiven) Modell und seiner Darstellung, z.B. werden Diagramme als "Modelle" bezeichnet.
- Mögliche Lösung: unterscheiden zwischen *kognitiven* Modell (als struktureller Vorstellung) und seiner Darstellung ("*Modell-Repräsentation*")

Modell vs. Sprache

- Problem: Nicht-Unterscheidung zwischen Beschreibungs*mittel* (= Sprache) bzw. Beschreibungs*muster* und dem, *was* damit beschrieben wird.

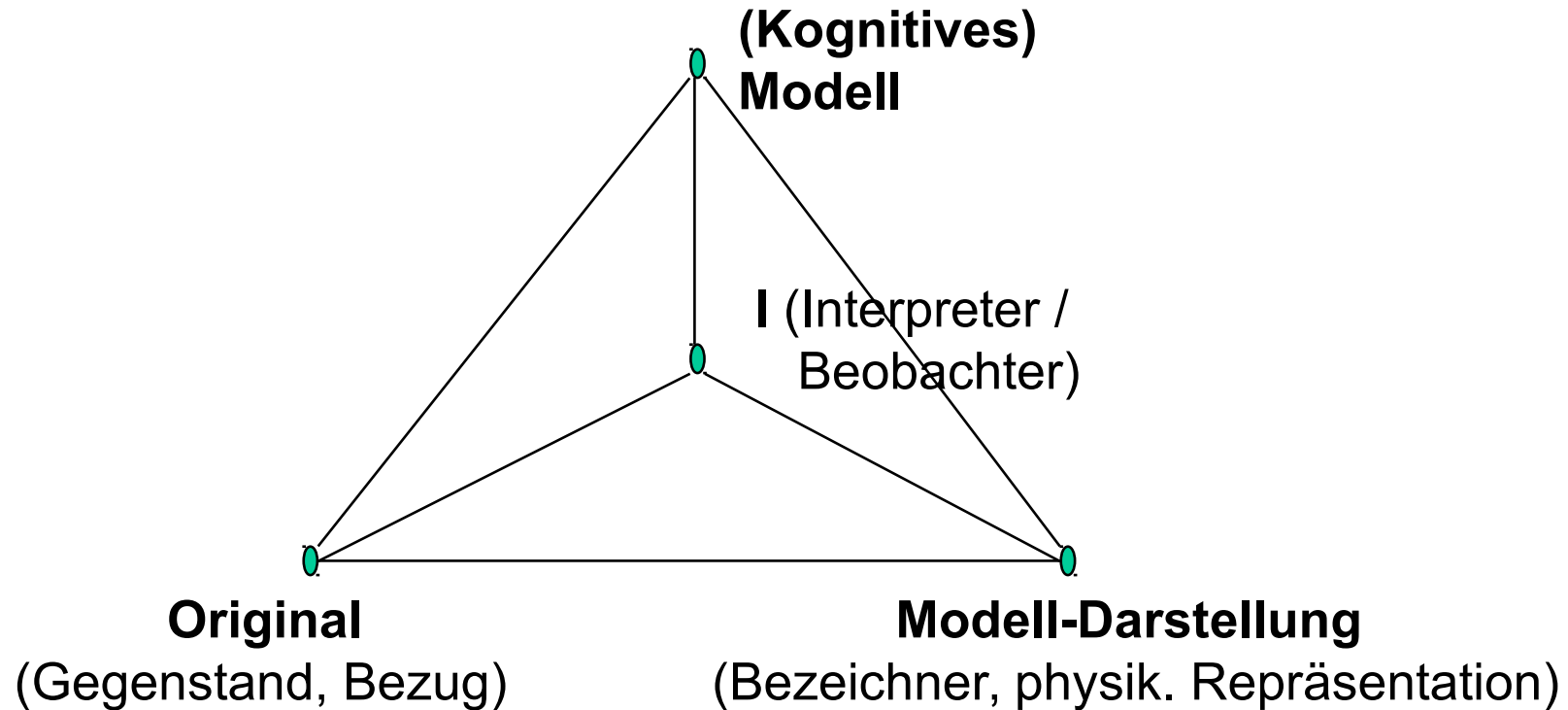
Beispiel: *Entity/Relationship (E/R-) "Modell"* – ist eine Diagrammtechnik (Sprache), mit der man *Datenmodelle* (Artefakte) herstellen kann.

Abweichendes Verständnis in der *mathematischen Logik*:

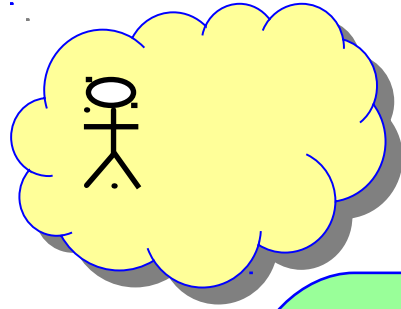
Modell = Interpretation eines Axiomensystems, bei der alle Axiome dieses Systems wahre Aussagen darstellen.

n. Meyers Neues Lexikon (1993), zit. nach [K-K 14]

Original - Modell - Repräsentation



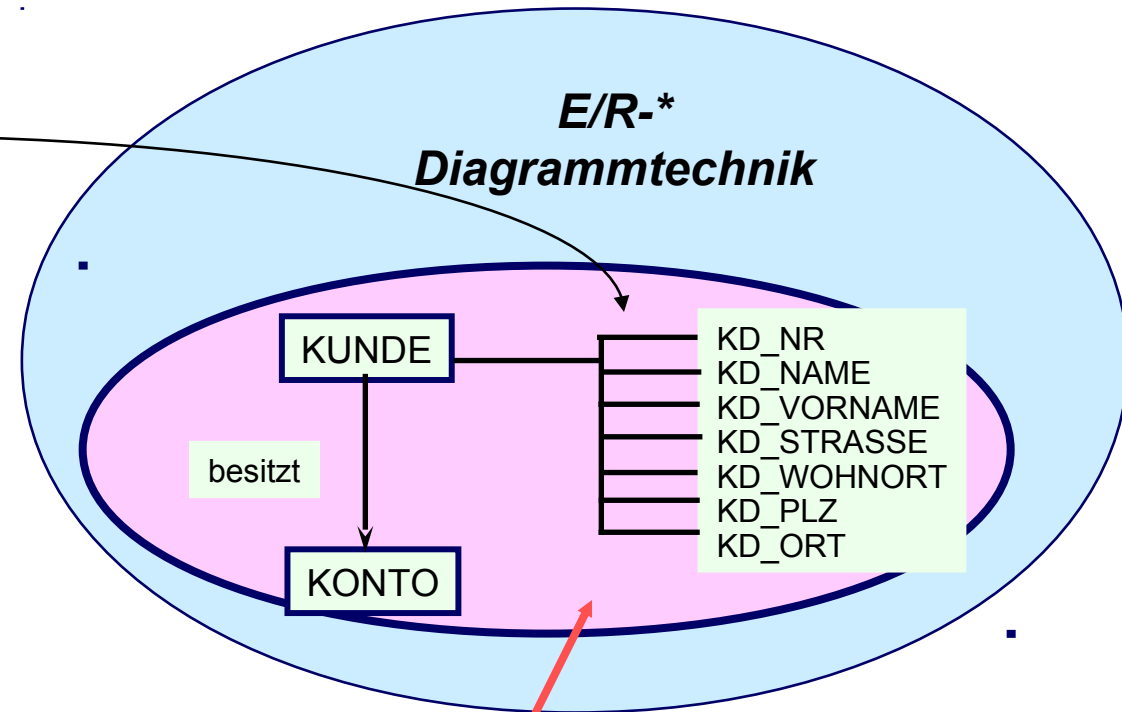
Modelle und ihre Darstellung



- **Entität Kunde, charakterisiert durch**
- **Attribute Kunden-Nr., Name, Vorname, Adresse, ...**
- **Entität Konto, charakterisiert durch**
- ...

(Kognitives) Modell
("im Kopf des Betrachters")

(Modellierungs-) Sprache



Modell (-Repräsentation)

Modellierungssprache UML

Unified Modelling Language (UML):

- entwickelt aus verschiedenen Ansätzen mit (vorwiegend) graphischen Elementen, seit 1997, Standard der OMG (*Object Management Group*)
- gestattet Darstellung mittels verschiedener **Sichten**, u.a. ...

(a) für die *statische* Systemstruktur:

- *Klassen-/Objekt-Diagramme (Class/Object diagram)*

(b) für das *dynamische* Verhalten:

- *Anwendungsfall-Diagramme (Use Case diagram)*
- *Interaktions- / Sequenzdiagramme (Interaction / Sequence diagram)*
- *Zustandsdiagramme (State diagram)*
- *Aktivitäts-Diagramme (Activity diagram)*

(c) für die *System-Umsetzung*:

- *Komponenten-/ Verteilungs-Diagramme (Component /Deployment diagram)*

Beispiel eines UML-Klassendiagramms

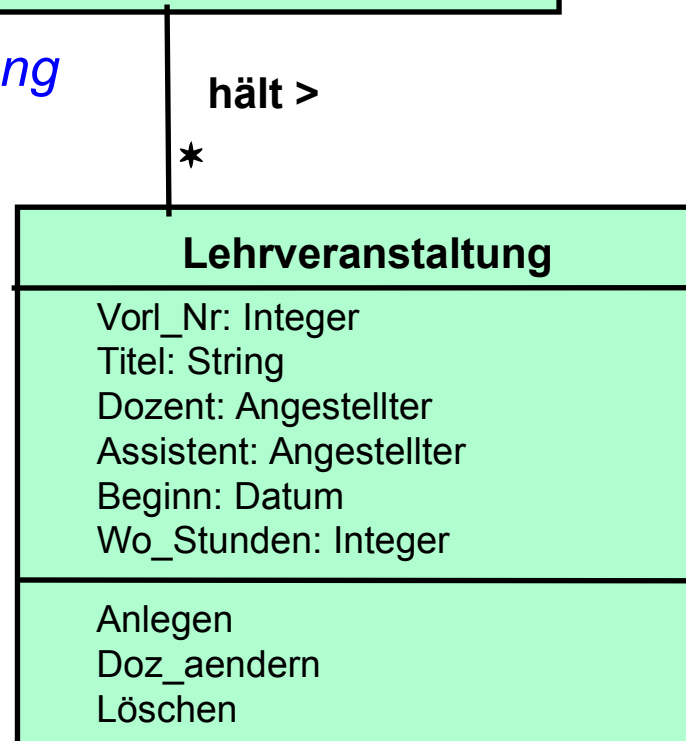
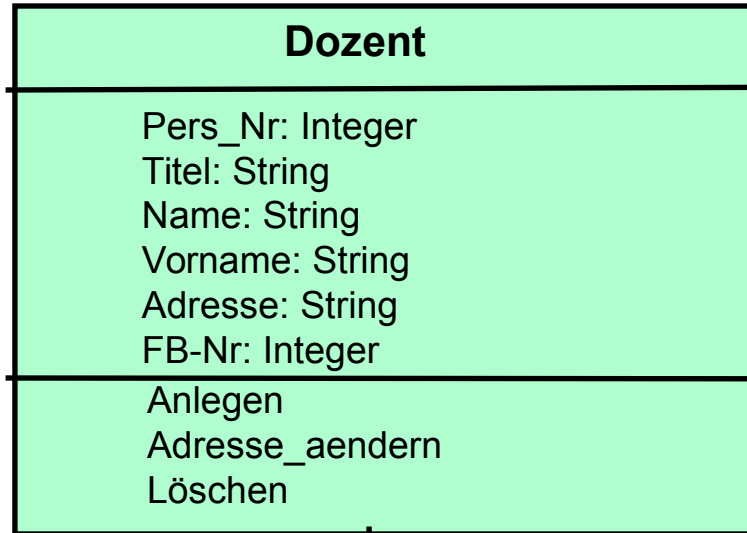
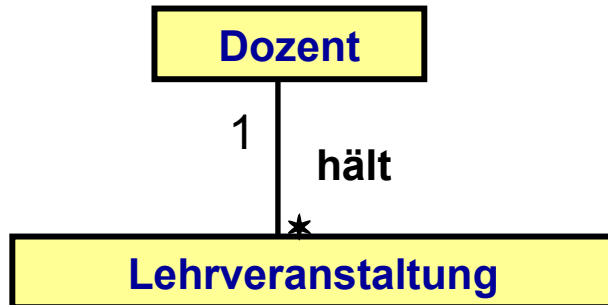
Klassen-Bezeichner

Attribut-Bezeichner: Typ

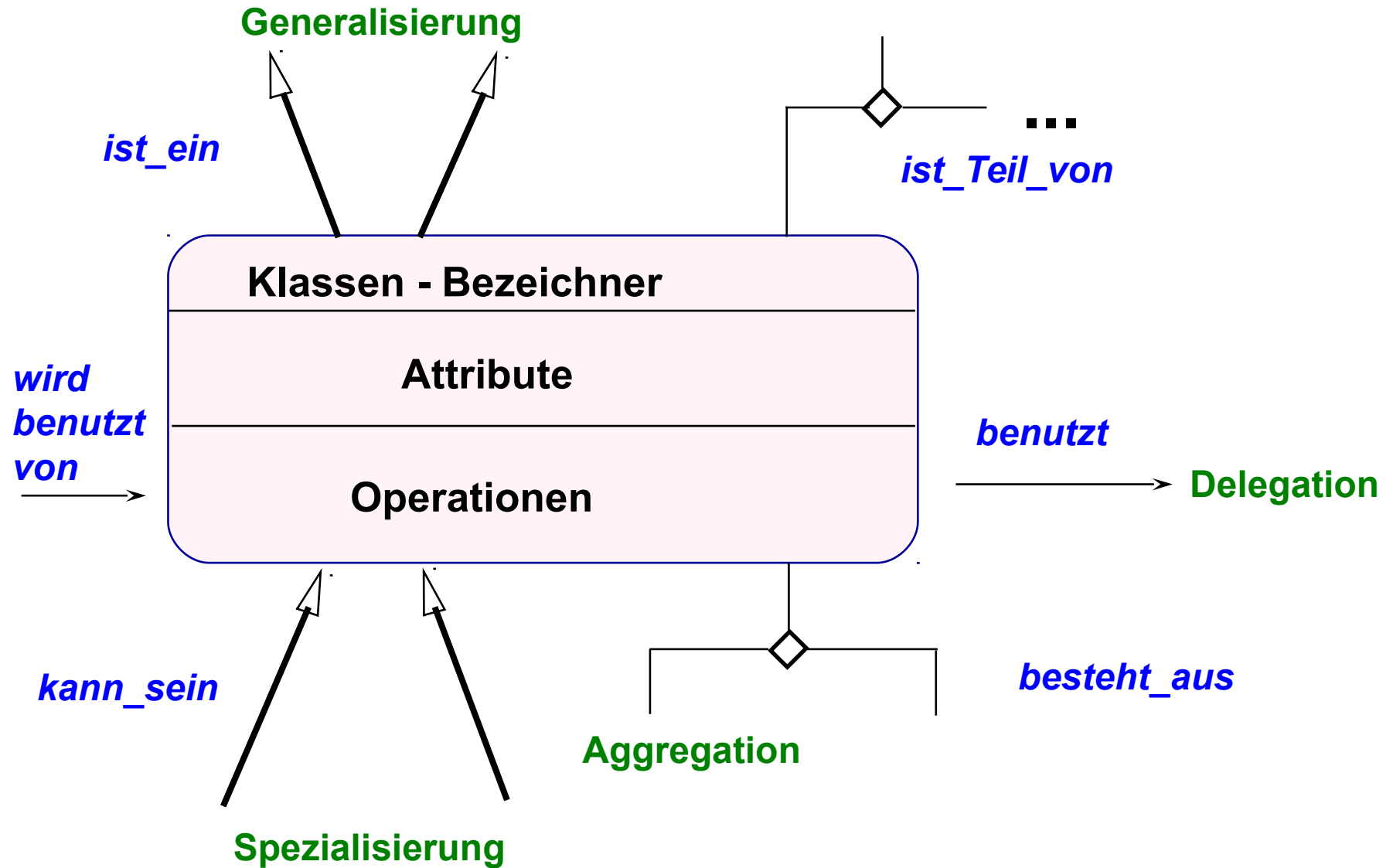
Operations-Bezeichner

Assoziation (mit Bezeichner, Richtung und Kardinalität)

Kurzform:



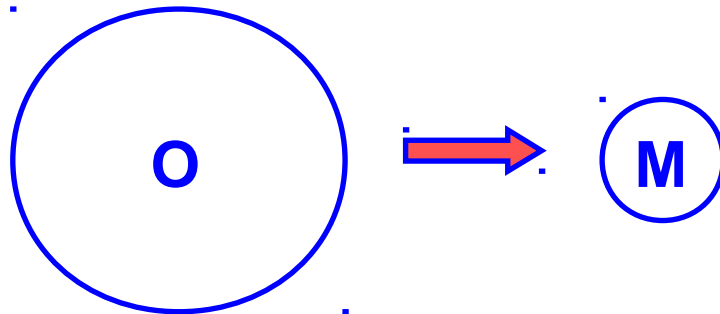
Klassen und Klassenbeziehungen (Zusammenfassung)



Nähere Charakterisierung der Original-Modell-Relation

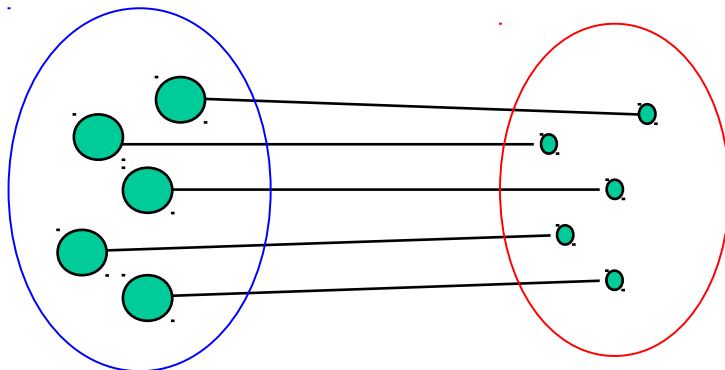
Vom Original wird "*abstrahiert*". Was bedeutet das genau?

- *Weglassen*: Modell enthält *weniger Details* als Original
- *Verkleinern*: Modell ist "*kleiner*" als Original
- *Übertragen*: Modell befindet sich in einem *anderen Bereich* als das Original
- Wesentlicher Bestandteil der Abstraktion: *Projektion*



Arten von Projektion

- Projektion kann in *zweierlei Hinsicht* erfolgen:



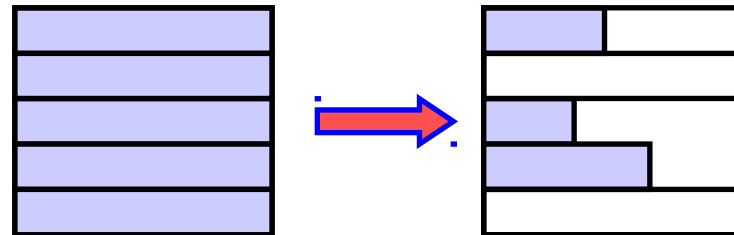
Original-
Bereich **O**

Modell-
Bereich **M**

(1) Verkürzung der Eigenschaften

⇒ *Merkmals*-Projektion

(*feature projection*) $O \triangleright_i M$



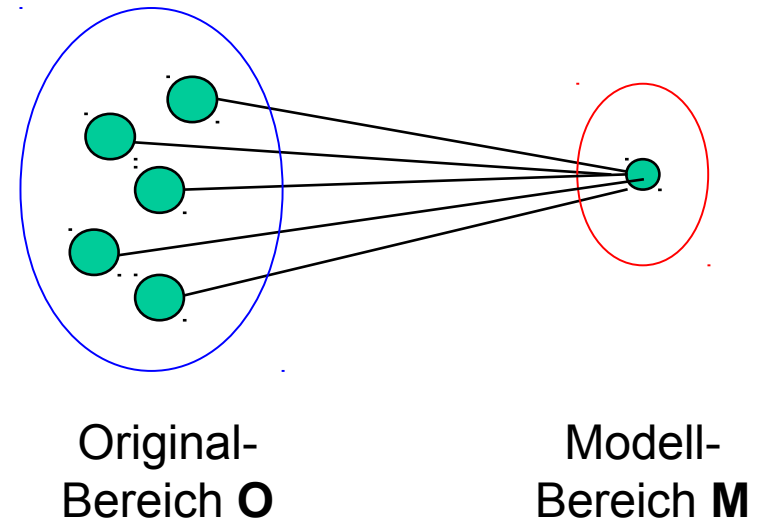
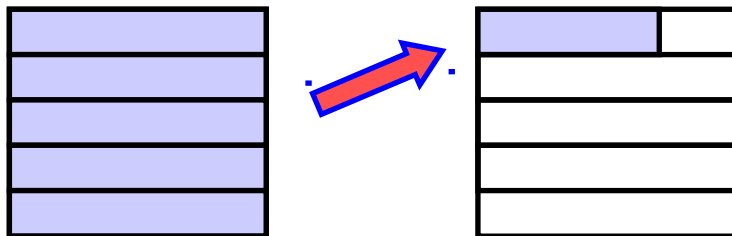
Beispiel: Städte in Landkarte

Stellvertreter-Projektion

(2) Repräsentation einer Menge von Elementen durch einen Platzhalter

⇒ *Stellvertreter-Projektion*

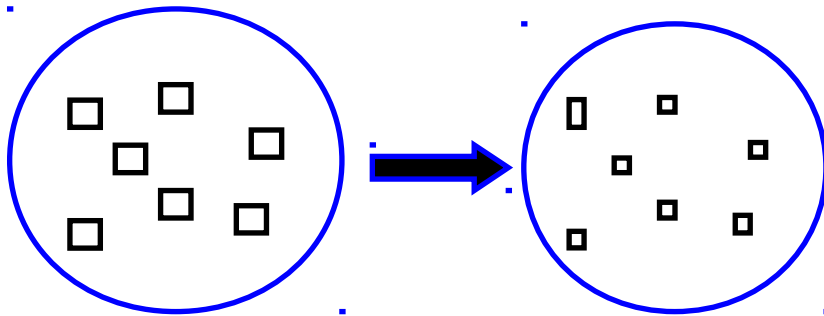
(placeholder projection) $\mathbf{O} \triangleright_t \mathbf{M}$



Beispiel: Balkendiagramme, 1 Balken steht (mit seiner Länge) für eine Menge von Gegenständen

Token- vs. Typ-Modelle

- *Zweierlei* Projektion führt zu *zweierlei Arten* von Modellen:



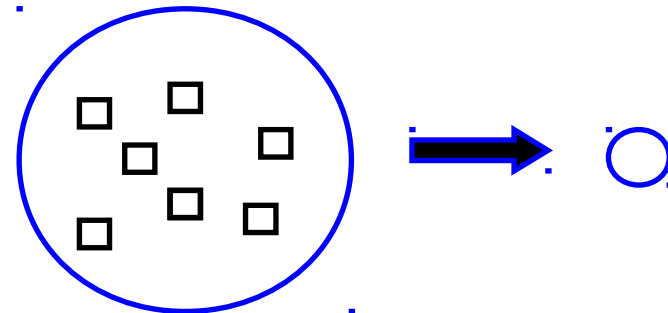
(1) **Token-Modelle** $\mathbf{O} \triangleright_i \mathbf{M}$

- abgeleitet von *feature projection*
- Anzahl der *Exemplare* bleibt *gleich*
- Anzahl und Details der Attribute werden reduziert

(2) **Typ-Modelle** $\mathbf{O} \triangleright_t \mathbf{M}$

- abgeleitet von *placeholder projection*
- Anzahl der *Exemplare* wird auf 1 Stellvertreter *reduziert*

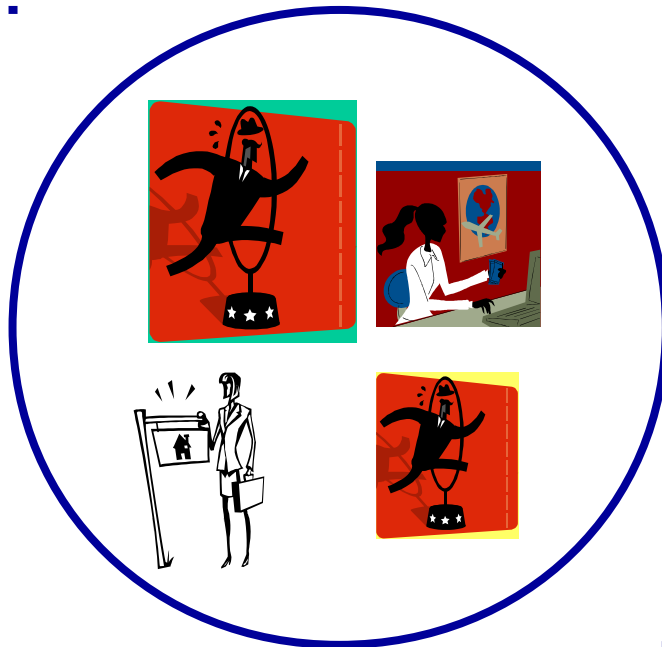
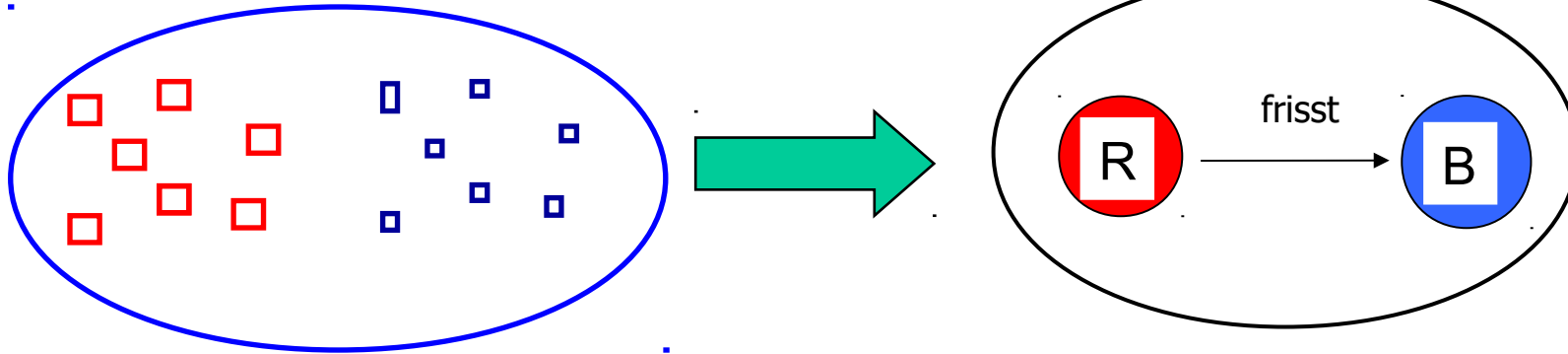
Alternative Notation: $\mathbf{O} \blacktriangleright \mathbf{M}$



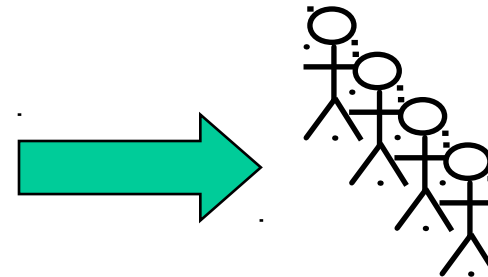
Beispiele ?

Token- vs. Typ-Modelle: Beispiele

Bsp. Räuber-Beute-Modell



Bsp. Agentenmodell



Weitere: [DB-/ UML-Klassenmodelle](#)

Unterscheidungen von Modellen

- nach **Zweck**:

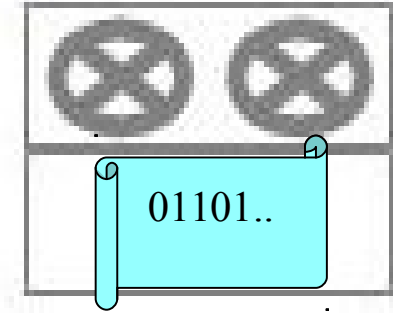
- . *Deskriptive* (Abbild-) / *präskriptive* (Vorbild-) / *transiente* Modelle
- . *Experimentelle*, *explorative* Modelle
- . *Idealisierende* / *Vorhersage-/ Erklärungs-Modelle*
- . *Kausale* Modelle
- . *Prognostische* (--> Prototypen, Simulationen)
- . *Konstitutive* Modelle
- . *Dokumentations-* Modelle,
- . *Lehr-* und *Spielmodelle*

Deskriptive vs. präskriptive Modelle

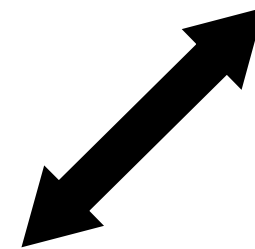
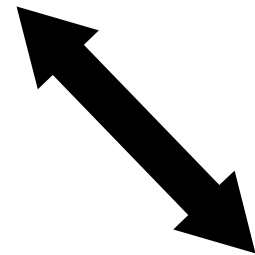
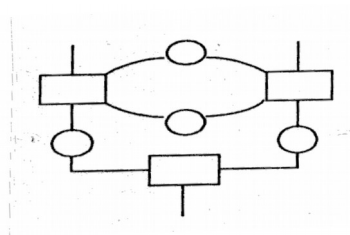


Anwendung
(Original 1)

Software-
System
(Original 2)



Modell



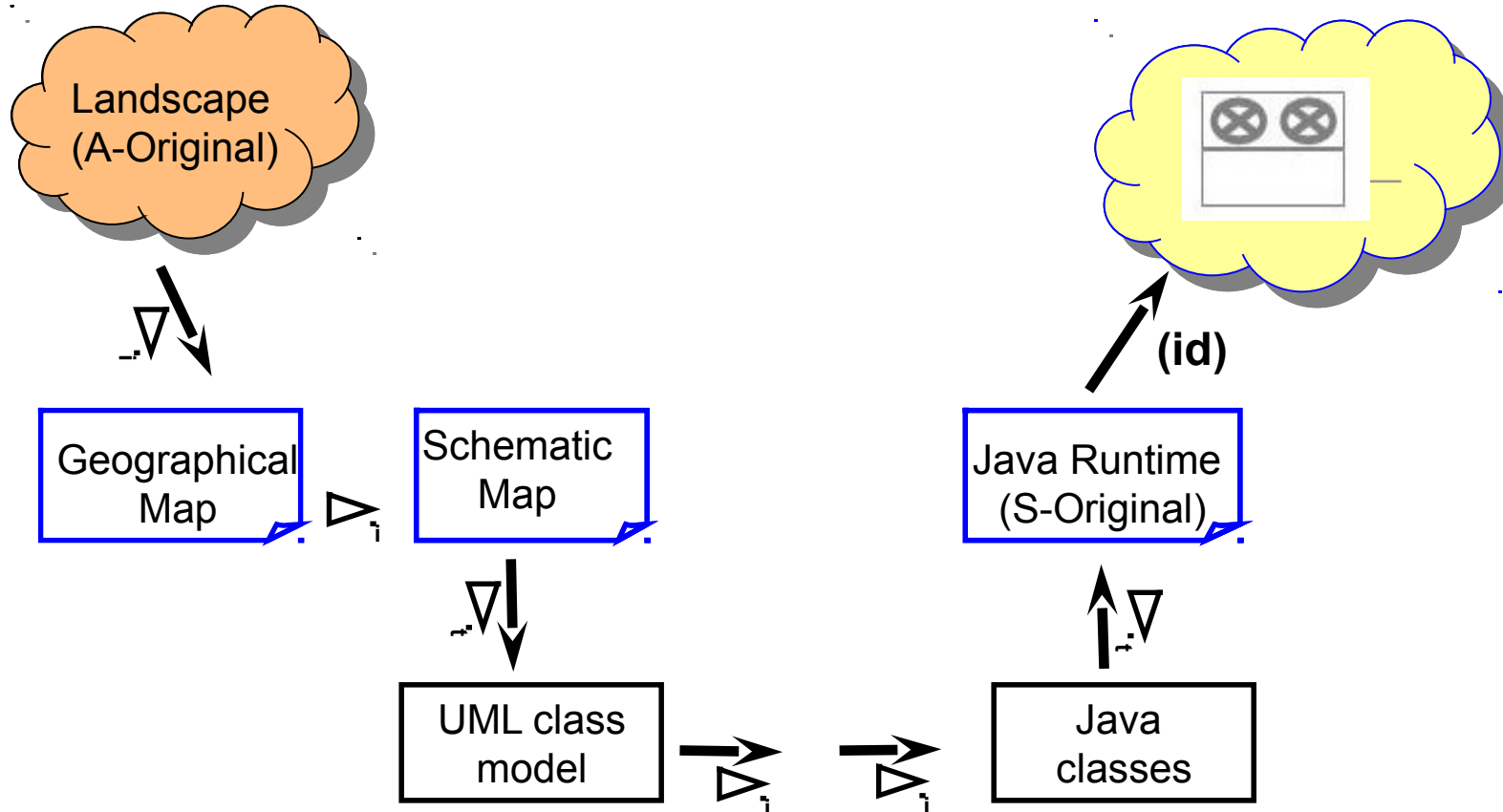
Janus schaut in *beide* Richtungen!

Modell als Nachbild:
A ► M



Modell als Vorbild:
M ◀ S

(Token- und Typ-)Modelle als Vor-/Nachbilder: Beispiel

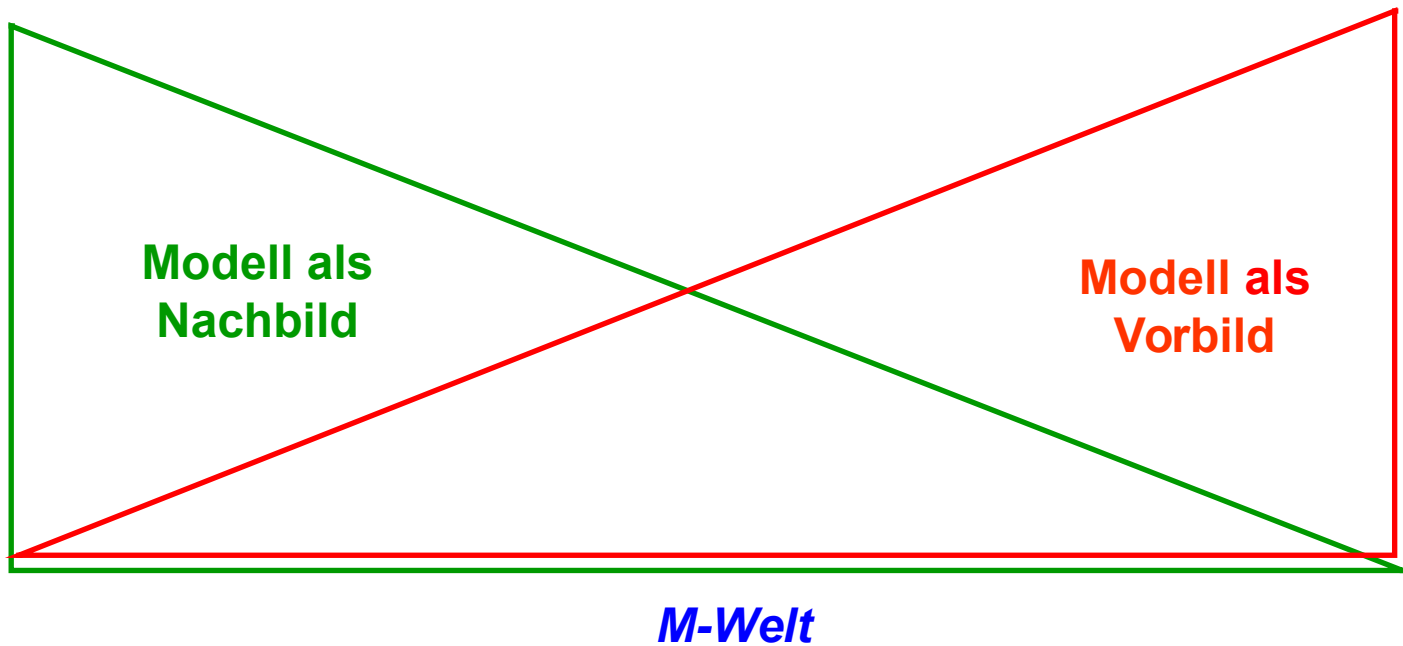
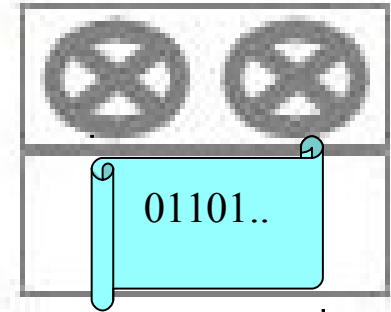


Vor- und Nachbild-Anteil der Modelle

A-Welt



S-Welt



Modelle – statisch vs. dynamisch

Verschiedene Modelltypen helfen, *unterschiedliche Aspekte* hervorzuheben:

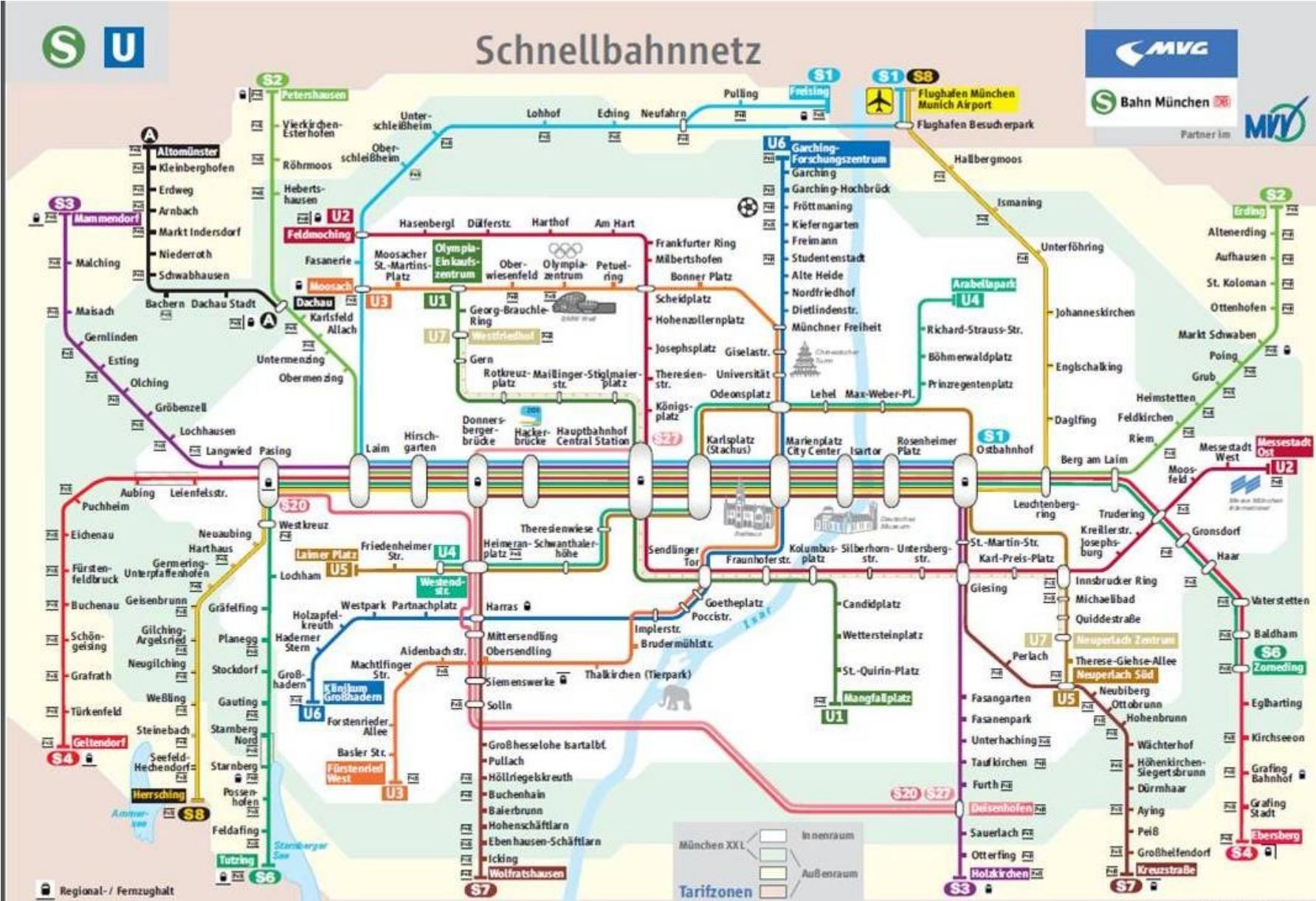
- **Statische Modelle**

- Gegenstands- / Objektmodelle
- Strukturmodelle
- Entitätsmodelle
- Klassenmodelle

- **Dynamische Modelle**

- Vorgehensmodelle
- Aktions- / Aktivitätsmodelle
- Prozessmodelle
- Zustandsmodelle

Modell: Schnellbahnnetz München



Eigenschaften des Beispielmmodells (MVV)

statisch:

- *Gegenstands- / Strukturmodell* (*Token-Modell*) für Netz mit Linien, Knoten und statischen (d.h. räumlichen, nicht zeitlichen) Verknüpfungen.
- *Dynamik* des Systems ist nicht abgebildet - keine einzelnen Züge, Fahrzeiten, Korrespondenzen (Anschlüsse), etc.

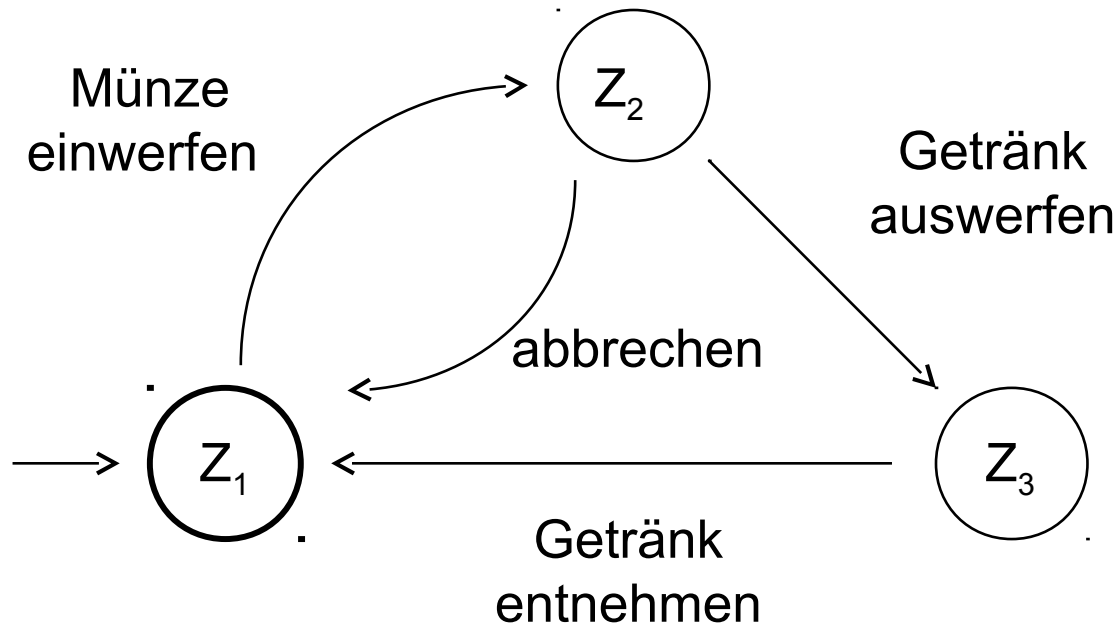
Modellauswahl:

- *präteriert*: dynamische Eigenschaften, Züge, Zugläufe, Anschlüsse, Wagenumlauf, Standplätze etc.
- *abundant*: (falsche) geographische Ausrichtungen (z.B. „Knicks“ in U3 und U6 n. Süden)

Modellzweck:

- Für die *Zielgruppe* (= MVV-Benutzer) Linienverläufe, Knoten (Umsteigepunkte) und Verknüpfungen visuell darstellen und damit Orientierung im System schaffen

Modell: Endlicher (deterministischer) Automat



Eigenschaften des Beispielmotells (Getränke-Automat)

dynamisch:

- *Zustandsmodell* (Typ-)Modell für (sehr) einfache Getränkeautomaten.
- *Dynamik* des Systems steckt in Zuständen, Zustandsübergängen etc.

Modellauswahl:

- *präteriert*: physikalische, materielle Eigenschaften: Gehäuse, Zubehör, Stromversorgung, Münzen, Getränke, ...
- *abundant*: graphische Symbole (Kreise, Pfeile, ..)

Modellzweck:

- Für die *Zielgruppe* (= Programmierer, Simulations-Ersteller) Abläufe, ihre Zusammenhänge und Kausalitäten darstellen, um richtige (=beabsichtigte) Funktionsweise zu verifizieren

Modell vs. Ontologie

- **Ontologie** (von griech: $\tau\omicron$ $\omicron\nu$) = Lehre vom *Seienden*.
- In der Informatik (anfänglich in der KI, J. McCarthy, um 1970) bezeichnet O. die *formale Beschreibung eines Wissens- (bzw. Gegenstands)bereichs* zur gemeinsamen Nutzung von (vorwiegend automatisierten) Anwendungen
- T. Gruber: **Ontology** = "*a formal explicit specification of a shared conceptualisation*" [Gru 93].
- O. ist also ein „*Modell im Großen*“, das vielfältigen Nutzungen dient und automatisiert be-/verarbeitet werden kann.
- Eine Ontologie beschreibt einen Gegenstandsbereich mit Hilfe
 - einer standardisierten *Terminologie* (Taxonomie),
 - *Beziehungen* (zwischen den Begriffen der Taxonomie),
 - *Ableitungsregeln* (zum Verbinden der Begriffe)
- Pro Wissensbereich gibt es (mindestens!) eine Ontologie
→ viele "**Ontologien**" (im Plural!) sind notwendig

- [Bos 04] H. Bossel: Systeme, Dynamik, Simulation: Modellbildung, Analyse und Simulation komplexer Systeme. Books on Demand, Norderstedt/Germany, 2004
- [B-S 04] M. Broy, R. Steinbrüggen: Modellbildung in der Informatik. Springer 2004
- [Cha 01] D. Chandler: Semiotics: The Basics. Routledge, London/New York 2001, überarb. Ausgabe 2006,
- [Che 76] P.P. Chen: The entity/relationship model - Toward a unified view of data. *ACM Transact. on DB Systems Vol. 1, no. 1*, pp 9-36 (1976)
- [DIN 82] DIN 44300: Informationsverarbeitung - Begriffe. Beuth-Verlag, Berlin 1982
- [Dör 84] D. Dörner: Modellbildung und Simulation, in: E. Roth (Hg.): Sozialwissenschaftliche Methoden. Oldenbourg-Verlag 1984, S. 337–350
- [Eco 93] U. Eco: Platon im Striptease-Lokal: Parodien und Travestien, dtv 1993
- [Gru 93] T. Gruber: A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2), pp. 199-220 (1993)
- [HBB+94] Hesse, W., Barkow, G., v. Braun, H., Kittlaus, H.B., Scheschonk, G.: Terminologie der Softwaretechnik - Ein Begriffssystem für die Analyse und Modellierung von Anwendungssystemen, *Informatik-Spektrum* 17.1, S. 39-47 u. 17.2, S. 96-105 (1994)
- [Hes 02] W. Hesse: Das aktuelle Schlagwort: Ontologie(n). in: *Informatik-Spektrum* 25.6, S. 477-480 (2002)
- [Hes 06] W. Hesse: Modelle - Janusköpfe der Software-Entwicklung - oder: Mit Janus von der A-zur S-Klasse. *Proc. Modellierung 2006*, pp. 99-113. Springer LNI P-82 (2006)

Literatur (Forts.)

- [Hes 08] W. Hesse: Engineers discovering the "real world" – From Model-driven to Ontology-based Software Engineering (Invited Talk). Proc. 7th Int. Conf. on Inf. Systems UNISCON 2008; Springer LNBIP 5, pp. 136-147 (2008)
- [Hes 14] W. Hesse: Ontologie und Weltbezug. Informatik-Spektrum 37.4, pp. 298-307 (2014)
- [H-E 14] W. Hesse, H. Engesser: Ontologie. Informatik-Spektrum 37.4, pp. 281-282 (2014)
- [H-M 08] W. Hesse, H.C. Mayr: Modellierung in der Softwaretechnik: eine Bestandsaufnahme Informatik-Spektrum 31.5, pp. 377-393 (2008)
- [K-K 14] U. Kastens, H. Kleine Büning: Modellierung – Grundlagen und formale Methoden. 3. Aufl. Hanser 2005
- [Lud 02] J. Ludewig: Modelle im Software Engineering - eine Einführung und Kritik. In: M. Glinz et. al (Hrsg.): Proc. Modellierung 2002. LNI P-12 Koellen-Verlag 2003
- [Pre 86] A. Prestel: Einführung in die Mathematische Logik und Modelltheorie. Vieweg, Braunschweig 1986.
- [Scn 97] H.J. Schneider (Hrsg.): Lexikon Informatik und Datenverarbeitung, Version 4.0, Oldenbourg 1997
- [Sta 73] H. Stachowiak: Allgemeine Modelltheorie. Springer, Wien 1973
- [Tab 06] P. Tabeing: Softwaresysteme und ihre Modellierung
- [Tro 90] K. Troitzsch: Modellbildung und Simulation in den Sozialwissenschaften. Westdeutscher Verlag 1990.
- [UML 06] *OMG Unified Modelling Language Specification Version 2.0*, 2006. www.uml.org/#UML2.0

Einschub: Semiotik

σημείον (griech.: *sēmeĩon*): Zeichen, Signal

Semiotik: (seit ca. 1890, begründet durch *Charles S. Peirce*)

- Lehre von den Zeichen, ihren Bedeutungen und Wirkungen.
- direkte Verbindung zu: Erkenntnistheorie (Philosophie), Wissenschaftstheorie, Sprachwissenschaften

Wesentliche semiotische Kategorien:

- **Syntax** (*Form*),
- **Semantik** (*Bedeutung*),
- **Pragmatik** (*Wirkung*)

Traditionelle *Zeichen-/Bedeutungs*-Beziehung (*F. de Saussure, 1857-1913*)



Zeichen und ihre Bedeutung



Signifiant,
Zeichengeber,
Representamen, sign token

Signifié,
bezeichnetes
Ding, *sign object*

- Beispiel:

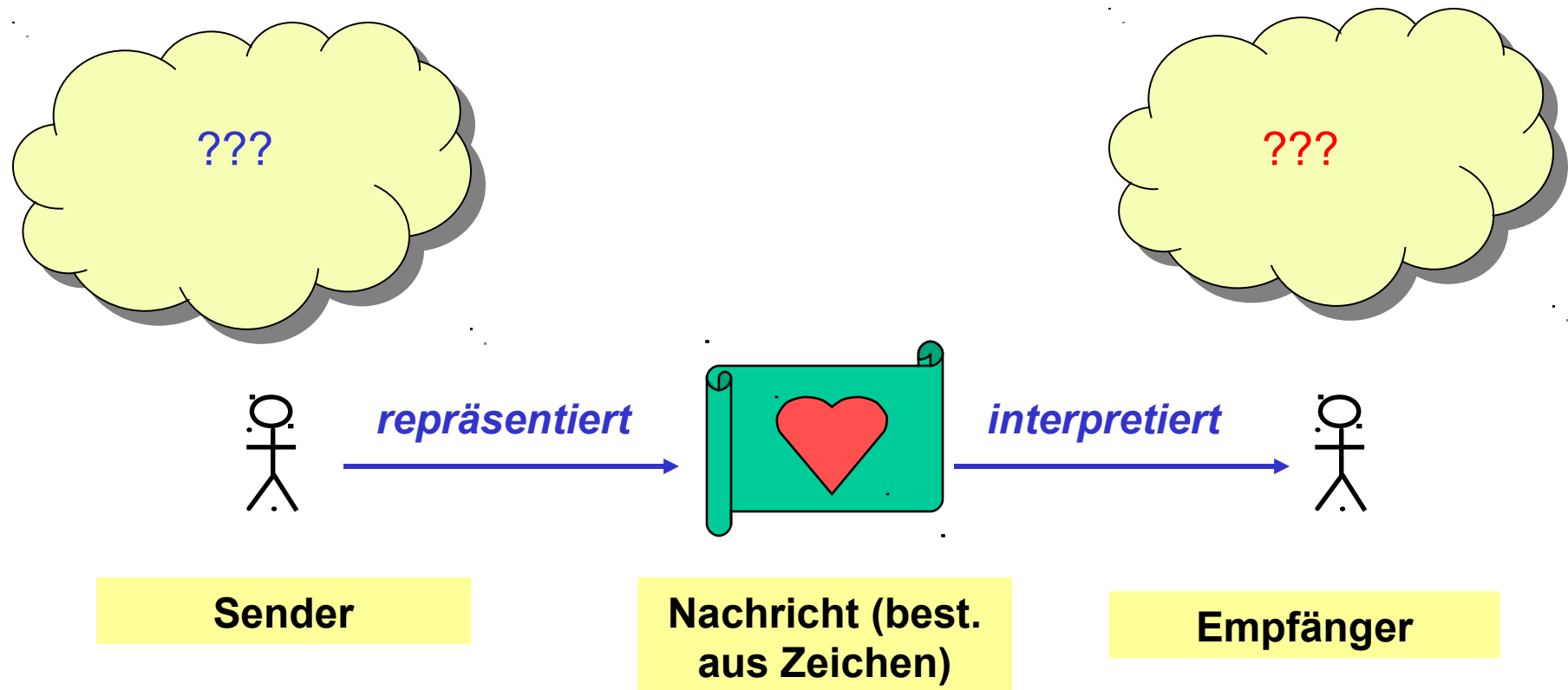


"Aktor in einem
Anwendungsfall" (*use case*)

Die Beziehung *steht_für* ist offenkundig kontext- und interpretations-abhängig.

Zeichen dienen der Kommunikation ...

... zwischen Sender (Zeichengeber) und Empfänger (Zeichennehmer):



Zeichen dienen der Kommunikation ...

... zwischen Sender (Zeichengeber) und Empfänger (Zeichennehmer):

