

# Modellierungsprimitiven

Fundamentale Relationen (*is\_a*, *part\_of*,  
*instance\_of*)

Jasmin Opitz

opitz@inf.fu-berlin.de

## Überblick

Einführung und Motivation

Probleme bei der Verwendung von *is\_a*, *part\_of* und *instance\_of*

Formalisierung

Beispiel

Ausblick

Zusammenfassung

Jasmin Opitz

opitz@inf.fu-berlin.de

## Einführung

Ontologie: Beschreibung von Konzepten und Beziehungen eines Bereichs

- Katalog von Begriffen
- Relationen zwischen den Begriffen
- Axiome

Relationstypen:

- *is\_a*: Inklusionsverhältnisse zwischen Klassen
  - z.B. „Mensch *is\_a* Säugetier“
- *part\_of*: Beziehungen zwischen Teilen und Ganzen
  - z.B. „Herz *part\_of* Mensch“
- *instance\_of*: Beziehung zwischen einer Instanz und einer Klasse
  - z.B. „John *instance\_of* Mensch“

Jasmin Opitz

opitz@inf.fu-berlin.de

## Motivation (1)

Ziel:

- Fachbegriffe und Kategorien verschiedener Domänen sollen idealerweise standardisiert werden
- Bedeutung dieser Fachbegriffe semantisch erfassen
- Verfügbarkeit und Wiederverwendbarkeit elektronisch gespeicherter Informationen dauerhaft sichern

Medizin:

- Terminologiesysteme für klinische Medizin
- Ontologien in der Bioinformatik
- Integration biologischer und medizinischer Informationen
- z.B. Gene Ontology, UMLS

Jasmin Opitz

opitz@inf.fu-berlin.de

## Motivation (2)

Geoinformationssysteme/Location-based Services

- raumbezogene Daten verwalten
  - Kartographie, Fernerkundung, Umweltschutz, Stadt- und Regionalplanung, GPRS, Positionierung...
- Web Services
  - Reiseroutenplanung
  - Standortbestimmung ([www.mapquest.com](http://www.mapquest.com))
  - Wettervorhersagen ([www.wetter.com](http://www.wetter.com))
- Raum- und Zeitontologien

Konsistente Modellierung von räumlichen und zeitlichen Beziehungen notwendig.

Jasmin Opitz

opitz@inf.fu-berlin.de

## Verwendungsprobleme

- Grundprinzip: „Gleiche Termini sind genau dann zu verwenden, wenn tatsächlich von denselben Entitäten die Rede ist.“

■ wird oft vernachlässigt

- ungenau definiertes Vokabular
- Inkonsistenzen im Umgang mit *is\_a*, *part\_of* und *instance\_of* Beziehungen
  - Gleichsetzung von *is\_a*, *part\_of* und *instance\_of*
  - Beziehungstypen werden jeweils nicht fein genug unterschieden
  - In verschiedenen Systemen keine einheitliche Behandlung der Beziehungen

Jasmin Opitz

opitz@inf.fu-berlin.de

## Beispiele

- UMLS: „Pflanzenblätter *is\_a* Pflanze“ (statt *part\_of* wird *is\_a* verwendet)
- SNOMED: „teilweise\_Entfernung\_der\_Blase *is\_a* Entfernung\_der\_Blase“
- Gene Ontology: „Nukleus *part\_of* Zelle“ (nicht jede Zelle hat einen Nukleus)
- „John *is\_a* Mensch“ (statt *instance\_of* wird *is\_a* verwendet)
- „Student *is\_a* Person“ (Student ist eine Rolle einer Person)

Jasmin Opitz

opitz@inf.fu-berlin.de

## Folgen inkonsistenter Verwendung

- erschwert Verständnis der Systeme
  - schwierig zu pflegen und zu verwenden
- Mängel in der elektronischen Verarbeitung
  - Schwierigkeiten bei der Integration mit anderen Systemen
  - Modelle schwer wiederzuverwenden
- Lösung:
  - Formalisierung der Beziehungen (z.B. Description Logics)
  - Genauere Unterscheidung der einzelnen Beziehungstypen

Jasmin Opitz

opitz@inf.fu-berlin.de

## Formalisierung (1)

Sicherung von Konsistenz und Stringenz:

- Formale Definitionen einführen

Verwendung des Begriffes „Entität“:

- alles, was in irgendeiner Weise existiert (Objekte, Prozesse, Funktionen, Zeiten, Orte...)
- Wichtig: Unterscheidung in Klassen und Instanzen
  - wurde bisher nicht klar genug getrennt
  - Instanzen: konkrete Individuen, örtlich und zeitlich gebunden
  - Klassen: abstrakte „Typen“, fassen Instanzen „gleicher Art“ zusammen

Jasmin Opitz

opitz@inf.fu-berlin.de

## Formalisierung (2)

Zur Definition der Begriffe -> zwei primitive Relationen *inst* und *part*

*inst*: Verhältnis zwischen Instanz und Klasse

z.B. „John ist eine Instanz von Mensch“

*part*: Verhältnis zwischen Teilen und Ganzen

z.B. „Johns Kopf ist ein Teil von Johns Körper“

Vorbemerkung:

- A,B -> Klassen
- x,y -> Instanzen

Jasmin Opitz

opitz@inf.fu-berlin.de

## Definition und Eigenschaften von *is\_a*

Alle Instanzen von A sind zugleich Instanzen von B

$$A \text{ is\_a } B := \forall x (inst(x, A) \rightarrow inst(x, B))$$

- Reflexivität
  - A *is\_a* A
- Antisymmetrie
  - A *is\_a* B & B *is\_a* A -> A=B
- Transitivität
  - A *is\_a* B & B *is\_a* C -> A *is\_a* C

Jasmin Opitz

opitz@inf.fu-berlin.de

## Differenzierung von *is\_a* Beziehungen (1)

Unterscheidung in 5 spezifische Beziehungen:

- Functional-Inclusion
  - für funktionale Beziehungen
    - „Stuhl *is\_a* Möbelstück.“
    - „Hammer *is\_a* Werkzeug.“
- State-Inclusion
  - Entitäten, die einen Zustand beschreiben
    - „Polio *is\_a* Krankheit.“
    - „Hass *is\_a* Emotion.“
- Activity-Inclusion
  - Entitäten, die eine Aktivität beschreiben
    - „Tennis *is\_a* Sport.“
    - „Mord *is\_a* Verbrechen.“

Jasmin Opitz

opitz@inf.fu-berlin.de

## Differenzierung von *is\_a* Beziehungen (2)

- Action-Inclusion
  - Entitäten, die eine Aktion beschreiben
    - „Vortragen *is\_a* Art des Sprechens.“
    - „Braten *is\_a* Art des Kochens.“
- Perceptual-Inclusion
  - Entitäten, die eine Wahrnehmung beschreiben
    - „Katze *is\_a* Säugetier.“
    - „Apfel *is\_a* Frucht.“
- Hilfsmittel für eine korrekte und konsistente Modellierung
- meistens wird trotzdem nur eine einzige *is\_a* Beziehung für alle verwendet
- Diese Unterscheidung findet z.B. Anwendung in IDEF5.

Jasmin Opitz

opitz@inf.fu-berlin.de

## Definition von *instance\_of*

$$x \text{ instance\_of } A := \text{inst}(x, A)$$

- Beispiel: „John *instance\_of* Mensch.“
- Unterschied zu *is\_a*: Es handelt sich um genau ein Individuum.

Jasmin Opitz

opitz@inf.fu-berlin.de

## Definition von *part\_of*

Zunächst definieren:

$$A \text{ part\_for } B := \forall x(\text{inst}(x, A)) \rightarrow \exists y(\text{inst}(y, B)) \& \text{part}(x, y)$$

- A's existieren ausschließlich als Teile von B's

$$B \text{ has\_part } A := \forall y(\text{inst}(y, B)) \rightarrow \exists x(\text{inst}(x, A)) \& \text{part}(x, y)$$

- jedes B enthält mindestens ein A

Definition von *part\_of*

$$A \text{ part\_of } B := A \text{ part\_for } B \& B \text{ has\_part } A$$

Jasmin Opitz

opitz@inf.fu-berlin.de

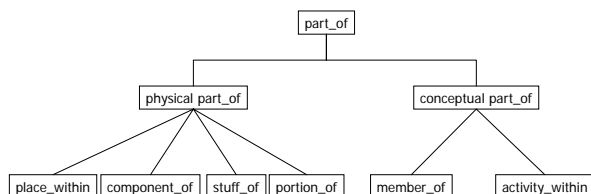
## Eigenschaften von *part\_of*

- Mutual Exclusivity
  - Wenn A *part\_of* B, dann gibt es einen weiteren Teil C *part\_of* B und es gilt A und C sind disjunkt.
- Irreflexivität
  - A *part\_of* A nicht erlaubt
- Asymmetrie
  - A *part\_of* B  $\rightarrow$  not (B *part\_of* A)
- Transitivität
  - A *part\_of* B & B *part\_of* C  $\rightarrow$  A *part\_of* C

Jasmin Opitz

opitz@inf.fu-berlin.de

## Differenzierung der *part-of* Beziehung



Jasmin Opitz

opitz@inf.fu-berlin.de

## *Part\_of* Beziehungen (1)

Physical *part\_of*:

- place\_within
  - für geographische Objekte
    - „Alpen *part\_of* Europa.“
    - „Everglades *part\_of* Florida.“
- component\_of
  - für zusammengesetzte Objekte (funktionale Komponente)
    - „Lenker *part\_of* Fahrrad.“
    - „Kapitel *part\_of* Buch.“
- stuff\_of
  - Objektmaterial
    - „Stuhl ist teilweise Holz.“
    - „Fahrrad ist teilweise Stahl.“
- portion\_of
  - ineinander eingeschlossene Objekte (inhaltliche Komponente)
    - „Kuchenstück *part\_of* Kuchen.“
    - „Sekunde *part\_of* Minute.“

Jasmin Opitz

opitz@inf.fu-berlin.de

## Part\_of Beziehungen (2)

Conceptual part\_of:

- member\_of
  - Mitglied/Teil Beziehung
    - „Baum part\_of Wald.“
- activity\_within
  - Eigenschaften/Phasen einer Aktivität
    - „Bezahlen part\_of Einkaufen.“

Jasmin Opitz

opitz@inf.fu-berlin.de

## Beispiel - Modellierungsfehler

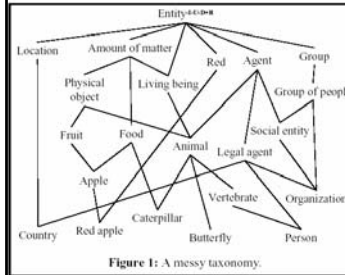


Figure 1: A messy taxonomy.

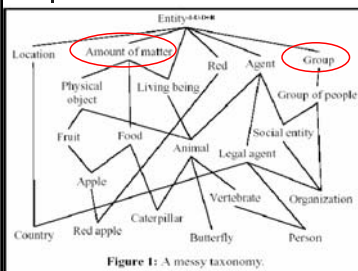
Beispiel einer unübersichtlichen und teilweise fehlerhaften Taxonomie.

Quelle:  
Nicola Guarino, Christopher Welty:  
Ontological Analysis of Taxonomic  
Relationships. International Conference  
on Conceptual Modeling. 2000.

Jasmin Opitz

opitz@inf.fu-berlin.de

## Inhomogene Modellierung

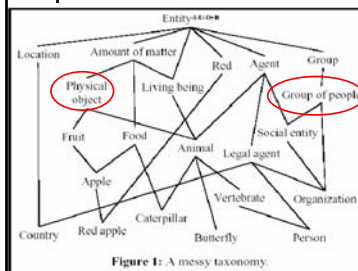


Jasmin Opitz

opitz@inf.fu-berlin.de

- allgemeine und spezielle Entitäten auf einer Ebene
  - Amount of matter / Group

## Inhomogene Modellierung

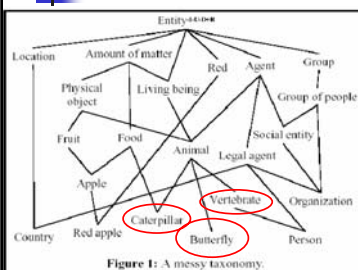


Jasmin Opitz

opitz@inf.fu-berlin.de

- allgemeine und spezielle Entitäten auf einer Ebene
  - Physical Object / Group of people

## Inhomogene Modellierung

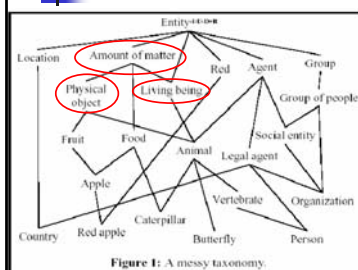


Jasmin Opitz

opitz@inf.fu-berlin.de

- allgemeine und spezielle Entitäten auf einer Ebene
  - Butterfly / Vertebrate

## Modellierung von Unterklassen



Jasmin Opitz

opitz@inf.fu-berlin.de

- bedenkliche Klassifizierung
  - Physical Object is\_an Amount of matter
  - Living being is\_an Amount of matter

## Modellierung von Unterklassen

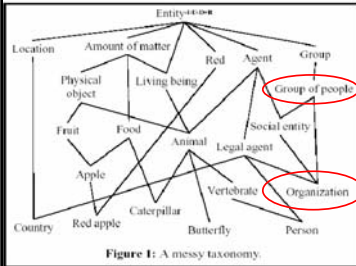


Figure 1: A messy taxonomy.

Jasmin Opitz

opitz@inf.fu-berlin.de

- bedenkliche Klassifizierung
  - Organization is\_a Group of people

## Modellierung von Unterklassen

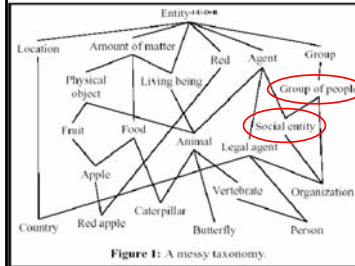


Figure 1: A messy taxonomy.

Jasmin Opitz

opitz@inf.fu-berlin.de

- bedenkliche Klassifizierung
  - Social entity is\_a Group of people

## Modellierung von Unterklassen

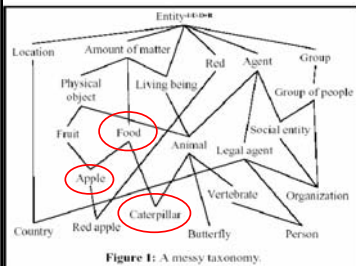


Figure 1: A messy taxonomy.

Jasmin Opitz

opitz@inf.fu-berlin.de

- bedenkliche Klassifizierung
  - Apple is\_a Food
  - Caterpillar is\_a Food

## Vermischung verschiedener Klassifikationsarten

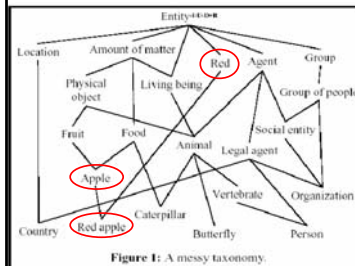


Figure 1: A messy taxonomy.

Jasmin Opitz

opitz@inf.fu-berlin.de

- Vermischung verschiedener Beziehungsarten im selben Unterbaum
  - Red apple is\_an Apple
  - Red apple has\_colour Red

## Vermischung verschiedener Klassifikationsarten

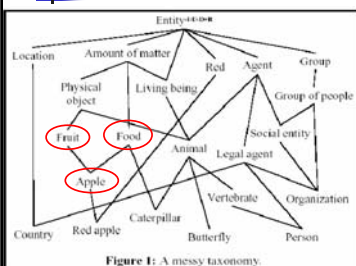


Figure 1: A messy taxonomy.

Jasmin Opitz

opitz@inf.fu-berlin.de

- Vermischung verschiedener Beziehungsarten im selben Unterbaum
  - Apple is\_a Fruit
    - Perceptual Inclusion
  - Apple is\_a Food
    - Functional Inclusion

## Ausblick (1)

Modellierungsfehler in Taxonomien:

- schwierig per Hand zu berichtigen, wenn Taxonomien groß und unübersichtlich werden
  - Ansätze zur automatisierten Verarbeitung (nächster Vortrag)

Konsistente Ontologien durch Formalisierung:

- Formalisierung ist schwierig zu verstehen
  - hängt von der Anwendung ab, wie detailliert man modelliert

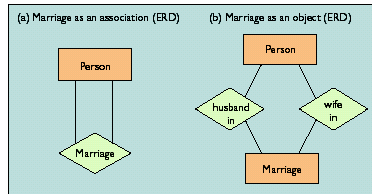
Die formale Betrachtung der Beziehungstypen löst nicht alle Modellierungsprobleme ...

Jasmin Opitz

opitz@inf.fu-berlin.de

## Ausblick (2) – Modellierung von zusammengesetzten Komponenten

Idee: Komponenten sollten immer als Entitäten / Klassen modelliert werden und nie als Assoziationen.



Quelle:  
Representing Composites in Conceptual Modeling. Communications of the ACM. July 2004/Vol. 47. No.7

Jasmin Opitz

opitz@inf.fu-berlin.de

## Modellierung von zusammengesetzten Komponenten (1)

Probleme bei der Modellierung von Komponenten als Assoziation:

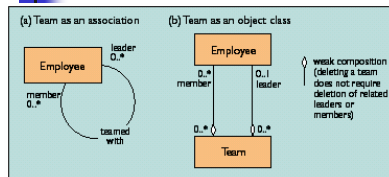
- „emergent properties“: entstehen, wenn mehrere Komponenten zu einem Kompositum zusammengesetzt werden
  - z.B. Team / Anzahl der Mitglieder
- „mutual properties“: wechselseitige Eigenschaften zwischen den Komponenten
  - z.B. Eine Person ist Stiefvater/Stiefmutter in einer Ehe.

Emergent und mutual properties können nur korrekt dargestellt werden, wenn die Teile als Entitäten dargestellt werden.

Jasmin Opitz

opitz@inf.fu-berlin.de

## Modellierung von zusammengesetzten Komponenten (2)



Quelle:  
Representing Composites in Conceptual Modeling. Communications of the ACM. July 2004/Vol. 47. No.7

Bemerkung: Bestimmte Situationen können nur modelliert werden, wenn die Teile als Entitäten dargestellt werden.

- z.B. Kann ein Team keine Mitglieder haben?

Jasmin Opitz

opitz@inf.fu-berlin.de

## Zusammenfassung

- Einführung in die grundlegenden Beziehungstypen (*is\_a*, *part\_of*, *instance\_of*) in konzeptueller Modellierung
- Anwendung in verschiedenen Domänen (Medizin, GIS)
- Beispiele und Folgen inkonsistenter Verwendung der Beziehungen in bestehenden Systemen
- Formalisierung und Differenzierung der Beziehungstypen
- Ausblick
  - Ansätze zur automatisierten Korrektur von fehlerhaften Systemen
  - Komplexität der Modellierung geht über Formalisierung hinaus

Jasmin Opitz

opitz@inf.fu-berlin.de

## Interessante Links

- UMLS (Unified Medical Language System):
  - <http://www.nlm.nih.gov/research/umls/>
  - gesehen am 29.05.2005
- Gene Ontology:
  - <http://www.geneontology.org/>
  - gesehen am 29.05.2005
- SNOMED (Systematized Nomenclature of Medicine):
  - <http://www.snomed.org/>
  - gesehen am 29.05.2005

Jasmin Opitz

opitz@inf.fu-berlin.de

## Literatur

- Barry Smith, Cornelius Rosse: The Role of Foundational Relations in the Alignment of Biomedical Ontologies. Proceedings of MedInfo 2004, San Francisco.
- Nicola Guarino, Christopher Welty: Ontological Analysis of Taxonomic Relationships. International Conference on Conceptual Modeling, October, 2000.
- Graeme Shanks, Elizabeth Tansley, Ron Weber: Representing Composites in Conceptual Modeling. Communications of the ACM. July 2004/Vol.47.No.7.
- Information Integration for Concurrent Engineering: IDEF5 Method Report. The IDEF5 Relationship Library.

Jasmin Opitz

opitz@inf.fu-berlin.de