

Ontologien und Ontology Engineering

Seminar „Semantic Web Technologien“

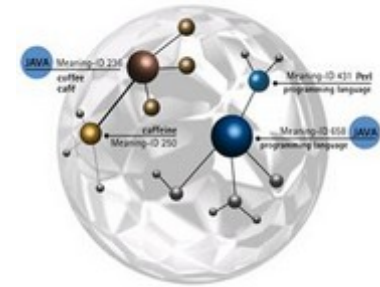
Sommersemester 2008

Prof. G. Brewka, Frank Loebe

Referent: Gregor Wiedemann

Gliederung

- **1. Ontologien**
 - 1.1 Definitionen / Anwendungen
 - 1.2 Ontologie-Formalismen
 - 1.3 Typen von Ontologien
- **2. Ontology Engineering**
 - 2.1 Begriffe und Prinzipien
 - 2.2 Methoden und Methodologien
 - 2.1 Ontology Development
 - 2.2 Ontology Re-Engineering
 - 2.3 Ontology Learning
 - 2.4 Ontology Alignment / Merging
 - 2.5 Collaborative Ontology Construction
 - 2.3 Ontology Evaluation am Beispiel OntoClean

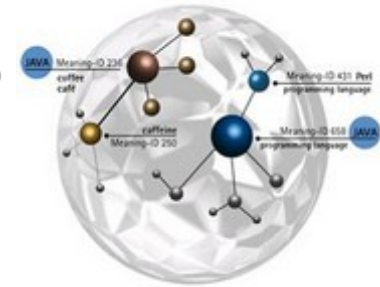


1.1 Definition / Anwendungen für Ontologien?



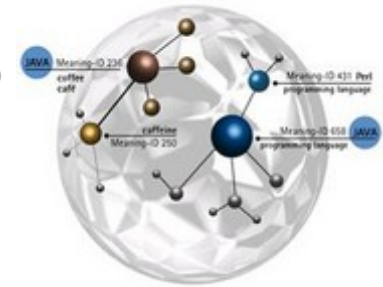
- **Ontologie != OWL, hier eher ein inhaltlich orientierter Blick auf die Begriffe / Begriffssysteme (statt wie bisher auf sprachliche Repräsentation und Schließmechanismen, etc.)**
- **weit Verbreitung von Ontologien in Bereichen**
 - Knowledge Engineering, Künstliche Intelligenz
 - Wissensmanagement
 - NLP, E-Commerce
 - Information Retrieval, Semantic Web, ...
- **Vorteile durch Ontologie-Nutzung**
 - Vereinfachte Wiederverwendbarkeit von Wissen
 - geteilte Benutzung von Domänenwissen
 - Nutzung über verschiedene Softwareplattformen hinweg
 - Abstraktion trennt Domänenstruktur/Aufgaben von Implementation
- **konkrete Anwendung: Integration verschiedener Softwaremodule miteinander (Verbindung von Applikationen durch einheitlichen semantischen Kern, Ontologie als unabhängiges Datenformat)**

1.1 Definition / Anwendungen für Ontologien?



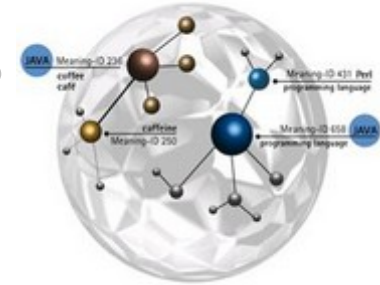
- Was ist Ontologie?
 - Herkunft aus der Philosophie: „Die Lehre des Seienden“, Unterscheidung von „essence“ und „existence“
 - Seit Anfang 90er starke Verbreitung in der Informationstechnik, hier verschiedene Definitionen
- Neches et. al 1991:
 - “An ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary”
- Gruber 1993:
 - “An ontology is an explicit specification of a conceptualization”

1.1 Definition / Anwendungen für Ontologien?



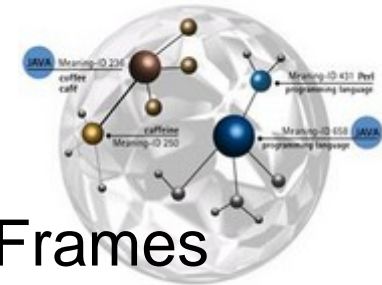
- Studer et al 1998:
 - “An ontology is a **formal, explicit** specification of a **shared conceptualization.**”
 - “Conceptualization refers to an abstract model of some phenomenon in the real world by having identified the relevant concepts of that phenomenon. Explicit means that the type of the concepts used, and the constraints of their use are explicitly defined. Formal refers to the fact that the ontology should be machine-readable. Shared reflects the notion that an ontology captures consensual knowledge, that is, it is not private of some individual, but accepted by a group.”

1.1 Definition / Anwendungen für Ontologien?



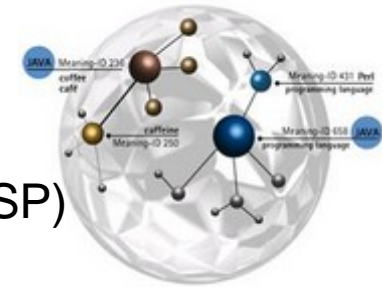
- abgeschwächte Ontologie-Definition: Taxonomien:
 - Taxonomie = hierarchische Klassifikation von Konzepten in Kategorien
- Unterscheidung von 2 Arten von Ontologien:
 - **1. Lightweight-Ontologien:** Begriffe (Konzepte), Taxonomien, Beziehungen zwischen Begriffen, Eigenschaften zu Begriffe
 - **2. Heavyweight-Ontologien:** 1. + Axiome, Constraints (Einschränkungen)
- Zusammenfassung: Ontologien versuchen konsensuales Wissen in einer allgemeinen Art und Weise abzubilden, sie können über verschiedene Software-Applikationen und Personengruppen hinweg wieder verwendet und geteilt werden

1.2 Ontologie-Formalismen



- Modellierung von Heavyweight-Ontologien mit Frames und First Order Logic (nach Gruber 1993a) 5 mögliche Komponenten:
 - **Klasse** = Repräsentation eines Konzeptes (i.d.R. in Taxonomien organisiert)
 - **Beziehungen**: $R \subset C1 \times C2 \times \dots \times Cn$ Typen von Assoziationen zwischen 2 oder mehr Konzepten (Beispiel: binäre Relation „subclass-of“ zur Modellierung von Taxonomien)
 - **Funktionen**: $F: C1 \times C2 \times \dots \times Cn-1 \rightarrow Cn$
 - **formale Axiome**: zur Modellierung von Fakten, die immer wahr sind
 - **Instanzen**: zur Repräsentation von Individuen einer Ontologie

1.2 Ontologie-Formalismen



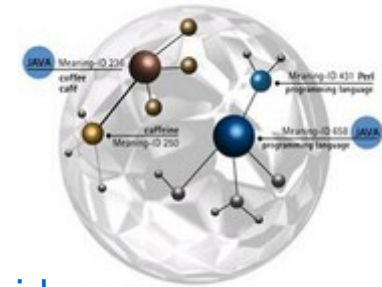
Beispiel: KIF-Notation (Knowledge Interchange Format, ähnlich LISP)

```
(define-class Travel (?travel)
  „A journey from place to place.“
:axiom-def
  (and (Superclass-Of Travel Flight)
    (Template-Facet-Value Cardinality arrivalDate Travel 1)
    (Template-Facet-Value Cardinality departureDate Travel 1))
:def
  (and (arrivalDate ?travel Date)
    (departureDate ?travel Date)
    (companyName ?travel String))

(define-relation arrivalPlace (?travel ?arrPlace)
  „A journey ends at a location“
:def
  (and (Travel ?travel)
    (Location ?arrPlace))
```

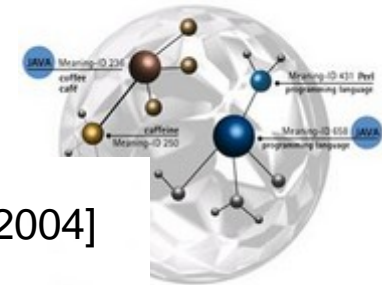
[Gomez-Perez et al 2004]

1.2 Ontologie-Formalismen

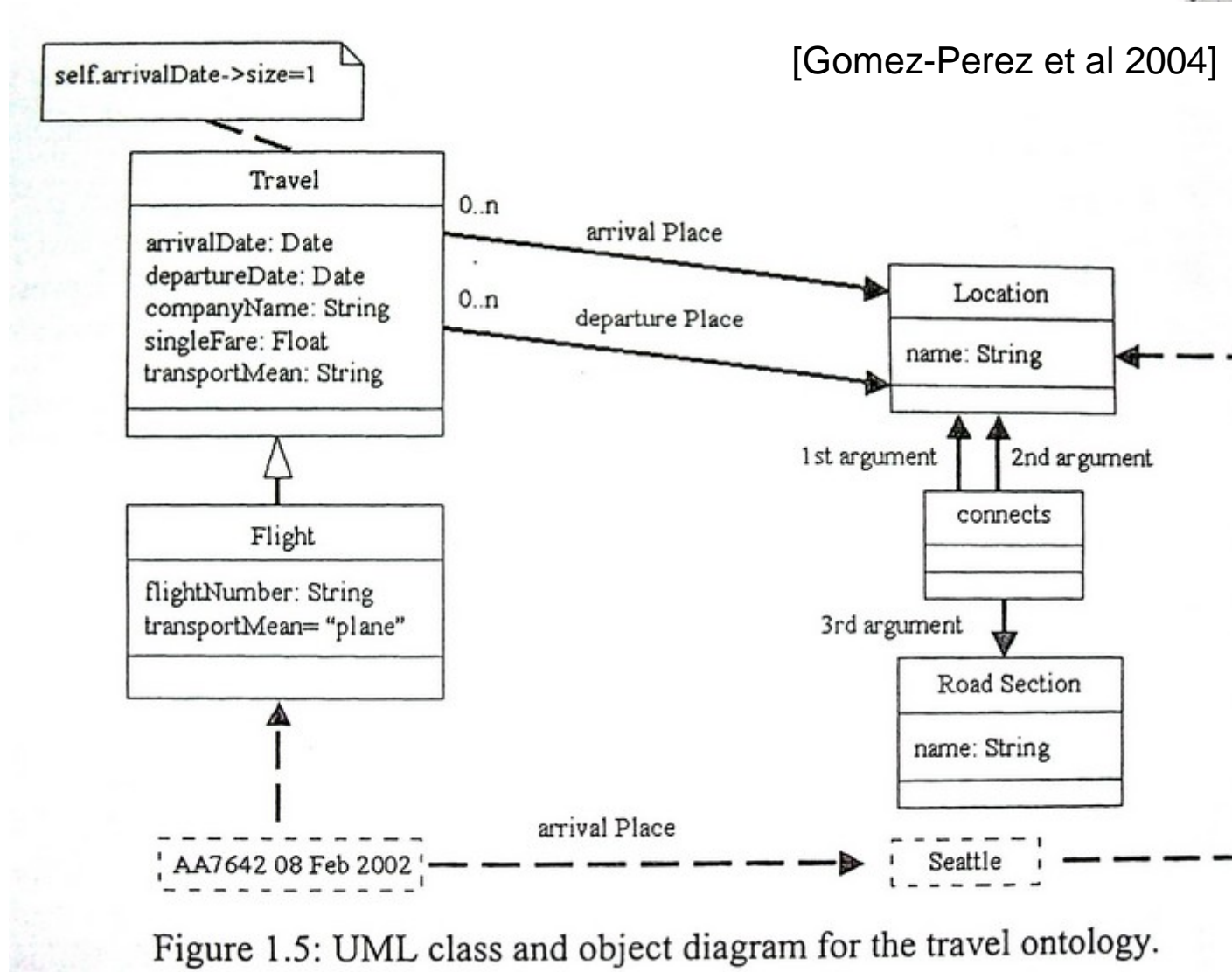


- Modellierung von Heavyweight.Ontologien Description Logic
 - Repräsentation der O. mit 3 Komponenten: Konzepte, Rollen, Individuen
 - TBox zur Definition von Konzepten und Rollen (= Beziehungen zwischen Konzepten)
 - ABox zur Definition von Individuen (Instanzen)
- Modellierung von Ontologien mit Techniken des Software Engineering
 - Idee: Nutzung von UML → gute Unterstützung durch zahlreiche, ausgereifte Software-Tools
 - jedoch nur lightweight-Ontologien mit UML möglich, für Modellierung von Kardinalitäten der Attribute Erweiterung mit OCL (Object Constraint Language) nötig
 - Klassendiagramme zur Repräsentation von Konzepten
 - Taxonomiehierarchie durch Vererbungs-/Generalisierungsmodelle im Klassendiagramm
 - UML-Objektdiagramme für die Instanzen
 - Beziehungen durch Assoziationen modellieren
 - formale Axiome: keine Möglichkeit mit UML, nur mit OCL möglich

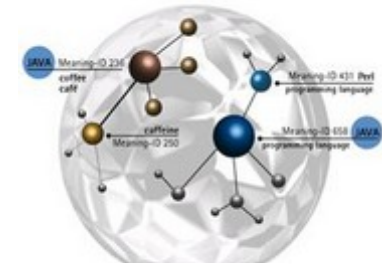
1.2 Ontologie-Formalismen



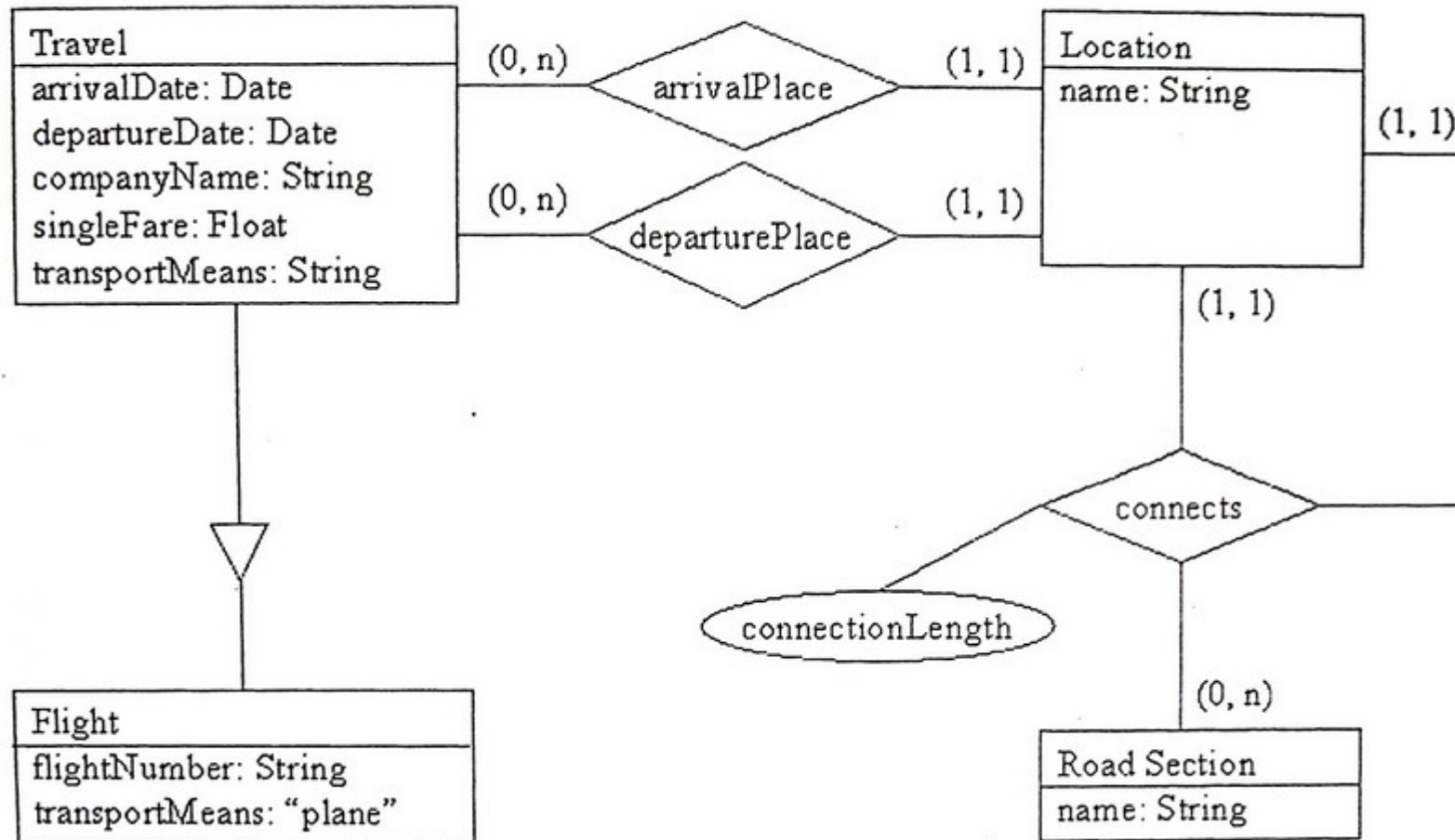
[Gomez-Perez et al 2004]



1.2 Ontologie-Formalismen



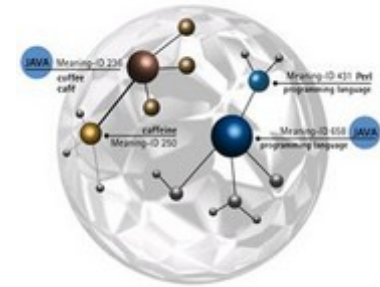
- Modellierung von Ontologien mit Datenbankmodellen



[Gomez-Perez et al 2004]

1.2 Ontologie-Formalismen

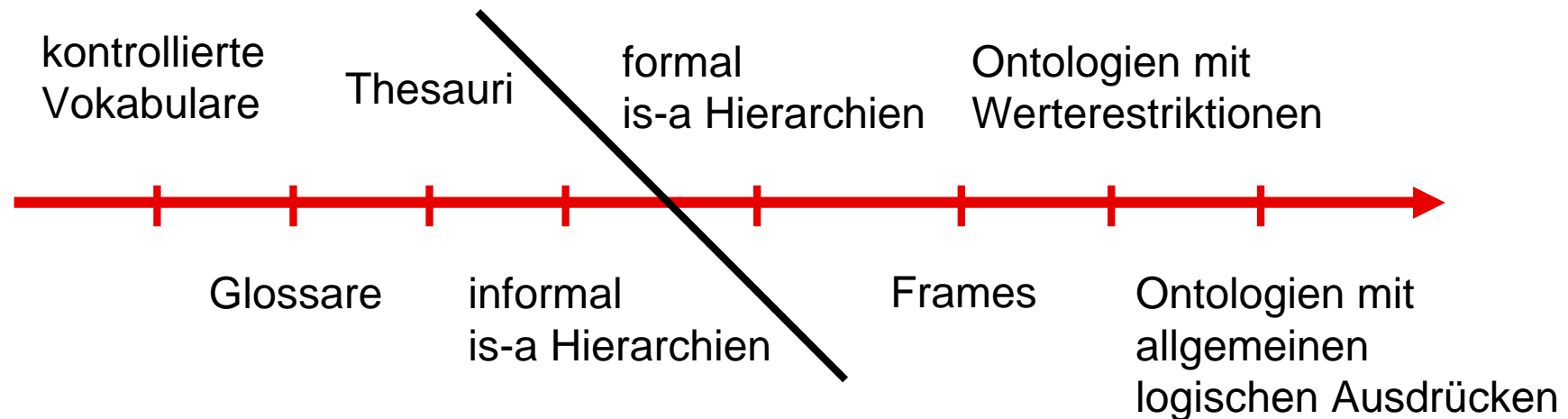
- Zusammenfassung
 - Formalismen zur Modellierung des Domänenwissens + zu dessen Implementierung benutzte Sprache schränken die Art des Wissens, welches modelliert und implementiert werden kann ein
 - AI-basierte Sprachen und ontology markup languages (RDF(S), OWL, ...) bessere Kandidaten zur Repräsentation und Implementierung von Ontologien als andere nicht-AI-basierte Ansätze wie UML oder ER/SQL



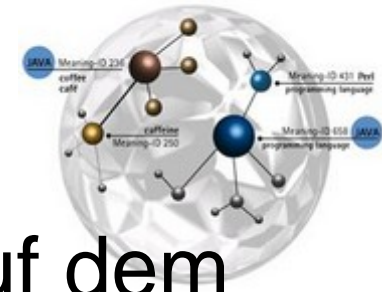
1.3 Typen von Ontologien



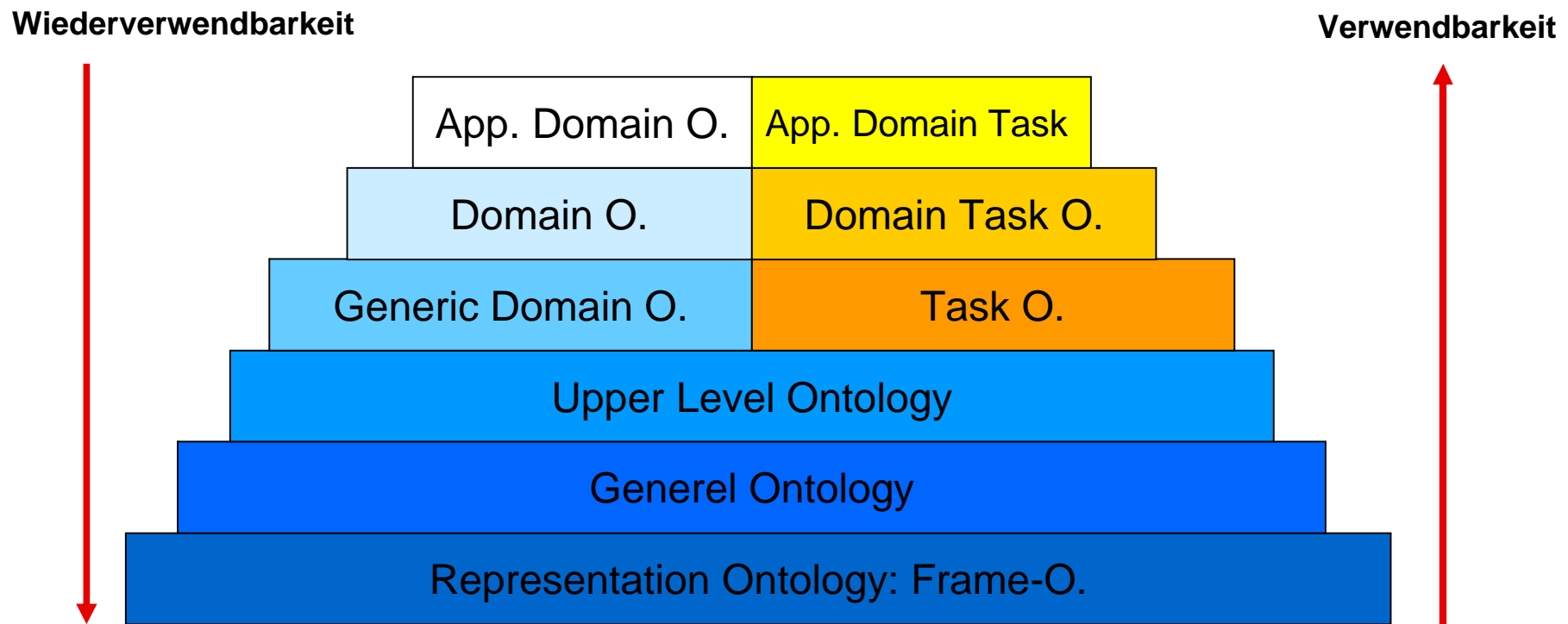
- Typen von Ontologien basierend auf der Reichhaltigkeit ihrer internen Struktur



1.3 Typen von Ontologien



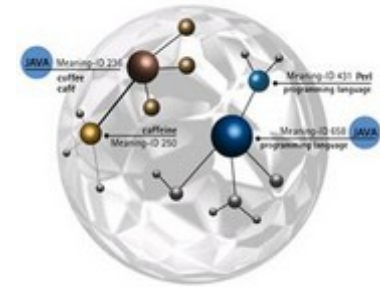
- Typen von Ontologien basierend auf dem Gegenstand ihrer Konzeptualisierung



[Gomez-Perez et al 2004]

Gliederung

- **1. Ontologien**
 - 1.1 Definitionen / Anwendungen
 - 1.2 Ontologie-Formalismen
 - 1.3 Typen von Ontologien
- **2. Ontology Engineering**
 - 2.1 Begriffe und Prinzipien
 - 2.2 Methoden und Methodologien
 - 2.1 Ontology Development
 - 2.2 Ontology Re-Engineering
 - 2.3 Ontology Learning
 - 2.4 Ontology Alignment / Merging
 - 2.5 Collaborative Ontology Construction
 - 2.3 Ontology Evaluation am Beispiel OntoClean

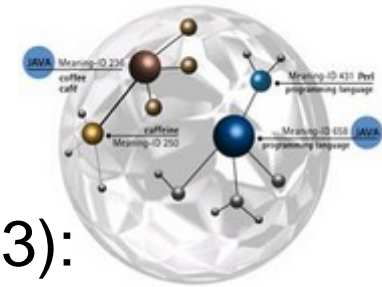


2.1 Begriffe und Prinzipien



- In Literatur / Anwendungskontext unterschiedliche Verwendung der Begriffe Methodologie, Technik, Prozess, Aktivität, etc. → Notwendigkeit einheitlicher Definitionen
- Vorschlag: Anlehnung an die IEEE (Institute of Electrical and Electronics Engineers)
 - **Methodologie:** “comprehensive, integrated series of techniques or methods creating a general systems theory of how a class of thought-intensive work ought be performed“; Verbreitung von Methodologien v.a. im Software Engineering und Knowledge Engineering
 - **Methode:** Teil von Methodologie, „set of orderly process or procedure used in the engineering of a product or performing a service“
 - **Technik:** Teil von Methodologie, “a technical and managerial procedure used to achieve a given objective”

2.1 Begriffe und Prinzipien

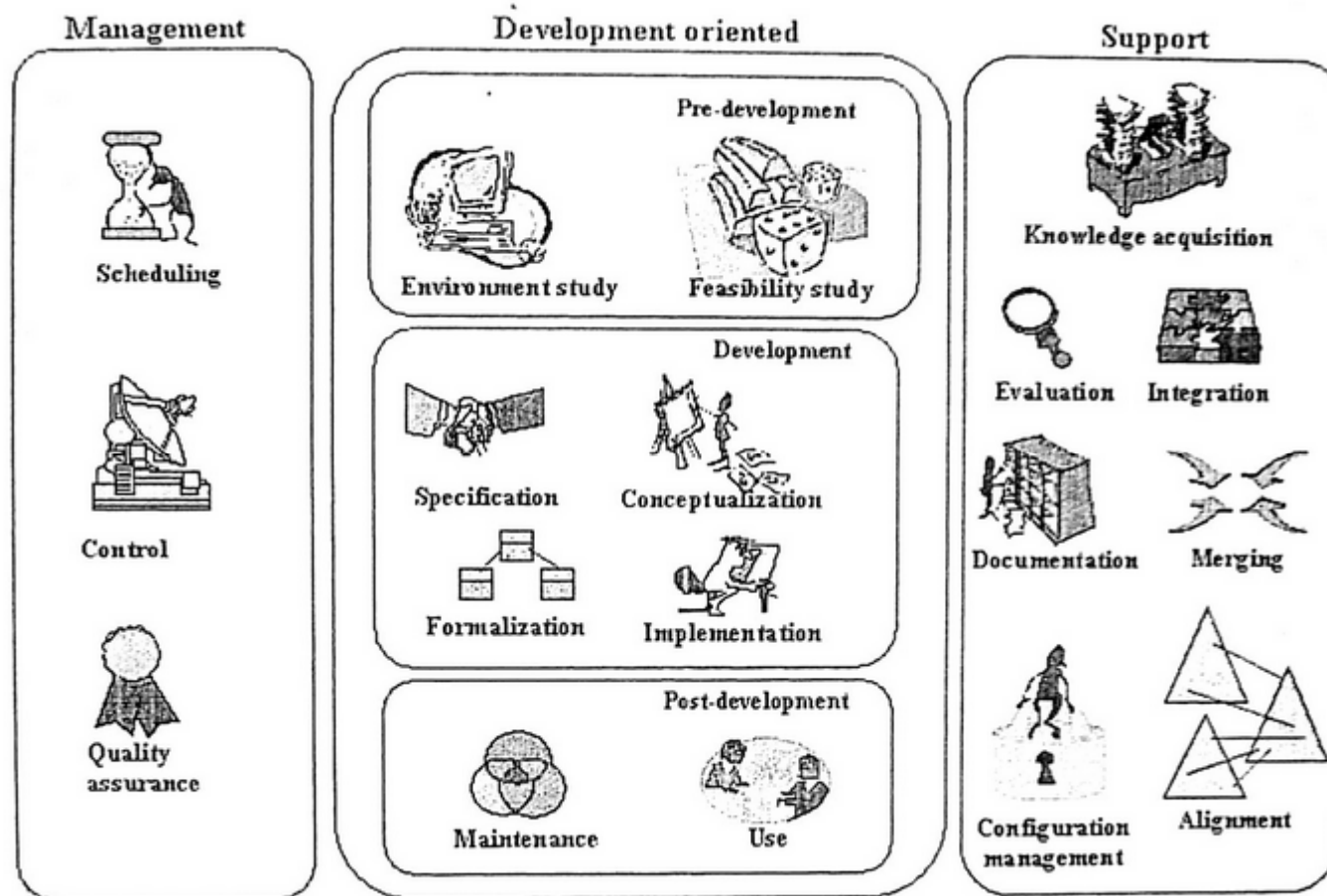


- Prinzipien des Ontologie-Designs (Gruber 1993):
 1. Klarheit
 2. Kohärenz
 3. Erweiterbarkeit
 4. minimal encoding bias
 5. minimal ontological commitment
- weitere nützliche Kriterien:
 - Berücksichtigung, ob Modell Wissen disjunkt oder vollständig repräsentiert
 - Minimierung der syntaktischen Distanz zwischen Schwester-Konzepten
 - Standardisierung von Namen / Bezeichnungen

2.1 Begriffe und Prinzipien



- Ontologieentwicklungs-Prozess: Festlegung, welche Aktivitäten in welcher Reihenfolge beim Bau von Ontologien auszuführen sind



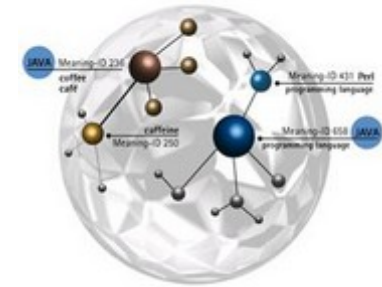
[Gomez-Perez et al 2004]

2.1 Begriffe und Prinzipien

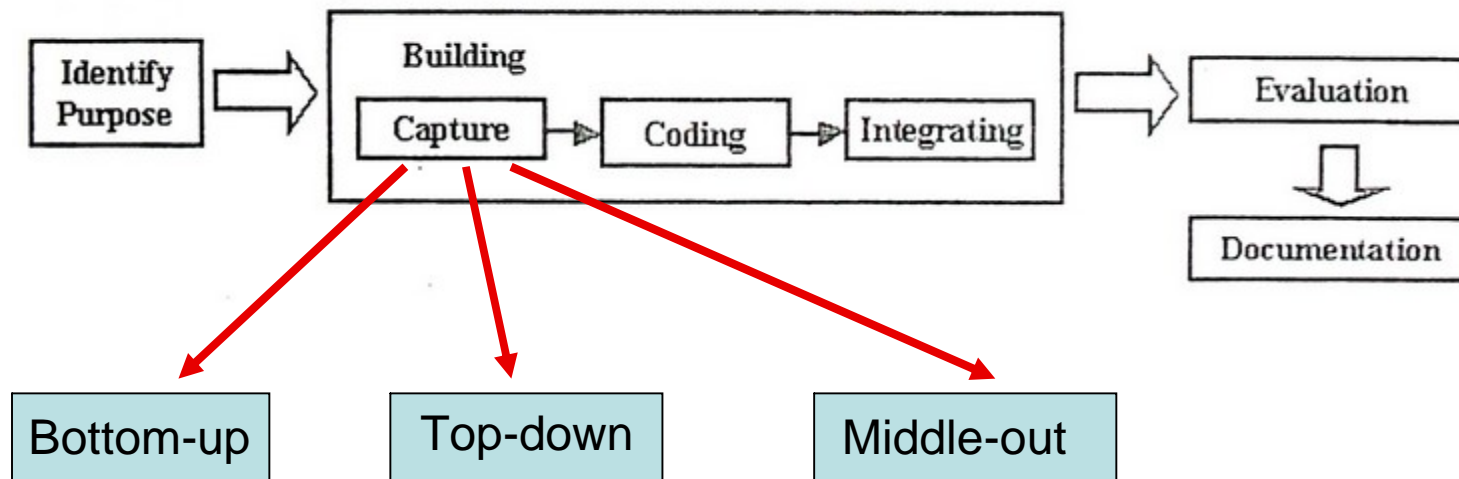


- **Ontology Methodology Evolution:** inzwischen zahlreiche Methoden und Methodologien zur Entwicklung von Ontologien entworfen:
 - a) Methoden um O. von Grund auf neu aufzubauen
 - b) Re-Engineering Methoden, bei Wiederverwendung von O.
 - c) Ontologie-Lernverfahren (semi-/automatisch)
 - d) Ontologie Alignment/Merging-Methoden
 - e) Methoden für kollaborative und /räumlich) verteilte O.-Entwicklung
 - f) Methoden zur Evaluation von O.
- **Anspruch:** vollständige Ontologie-Methodologie sollte ausführliche Beschreibungen für alle diese Bereiche zur Verfügung stellen
- praktisch existiert jedoch keine, die dies leistet

2.2 Methoden und Methodologien

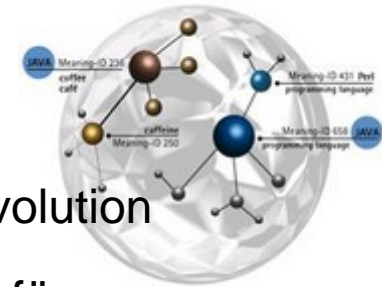


- Methode nach Uschold / King (1995):
 - eine der ersten *Methoden* zum Bau von Ontologien
 - leichtgewichtiger Ansatz: sehr einfache Abfolge von 4 Schritten

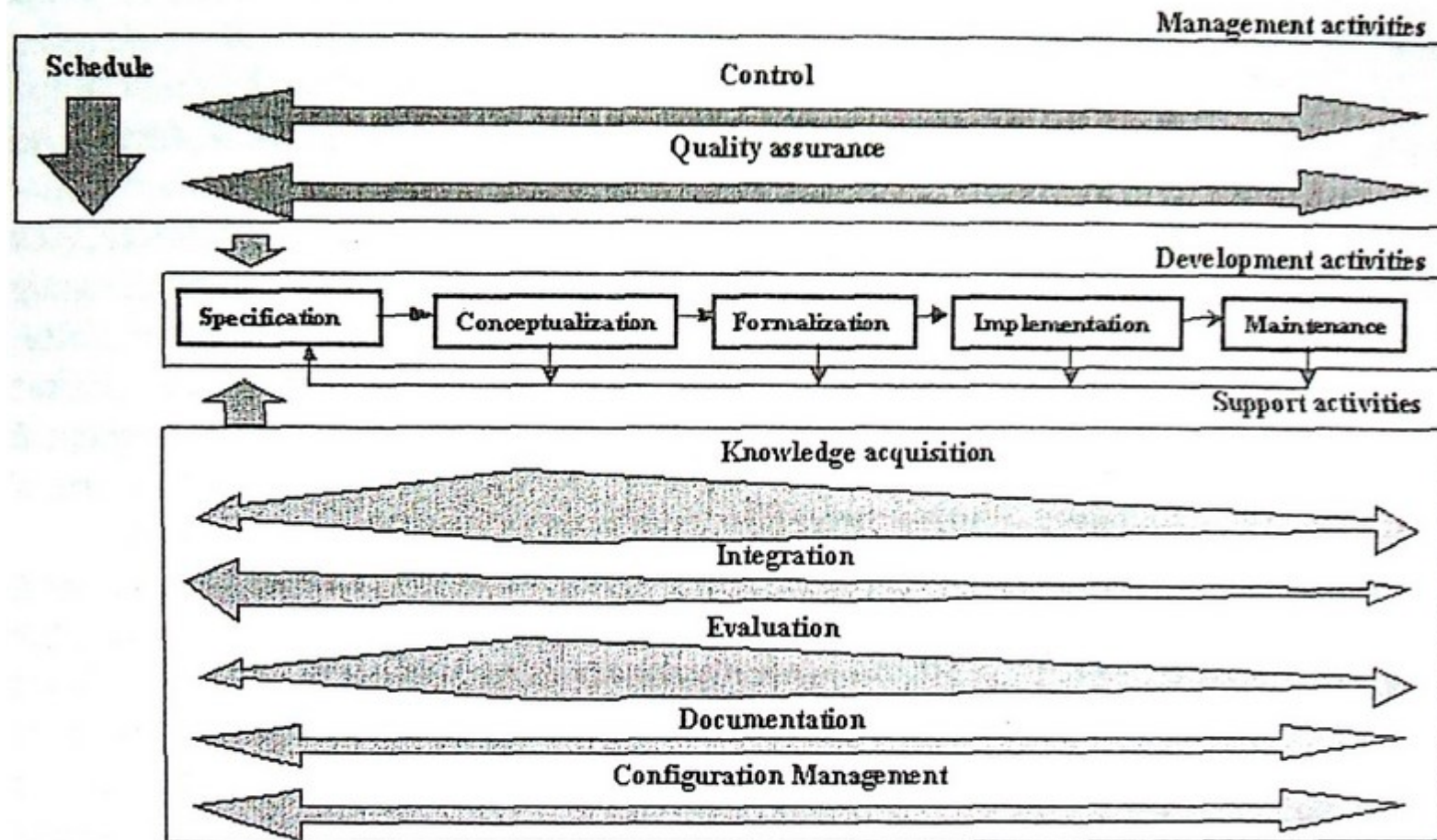


- größter Nachteil: Fehlen eines Konzeptualisierungsprozesses vor der Implementation

2.2 METHONTOLOGY

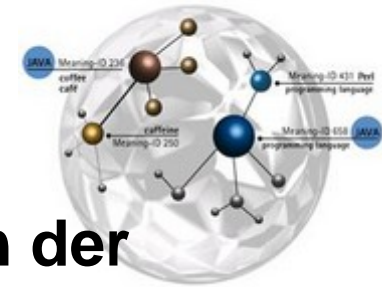


- große, umfassende Methodologie für Ontologie-Entwicklung/Evolution
- umfasst ausführliche Beschreibungen von Vorgehensmodellen für



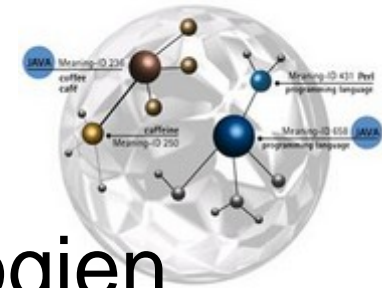
[Gomez-Perez et al 2004]

2.2 Methoden und Methodologien



- **Vergleich von Methoden und Methodologien der Ontologie-Entwicklung**
 - life cycle Modelle, die aus Prototypen heraus entwickelt werden setzen sich durch
 - eine breite Vielfalt an Ontologie-Entwicklungsstrategien existiert
 - Vielfalt an Strategien zum Taxonomie-Bau, middle-out Ansätze überwiegen
 - Keiner der Ansätze deckt alle Prozesse des Ontologie-Baus ab
 - Bisläng vollständigster Ansatz: METHONTOLOGY
 - Fokus der meisten Ansätze auf der Ontologie-Entwicklung, v.a. Implementierung
 - Meist Vernachlässigung von Management, Evolution, Evaluation der Ontologien

2.2 Methoden und Methodologien

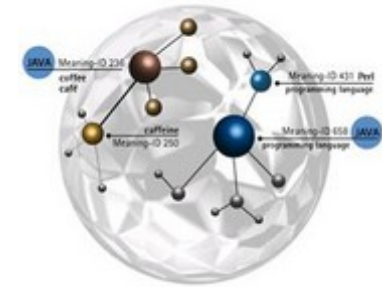


■ Vergleich von Methoden/Methodologien

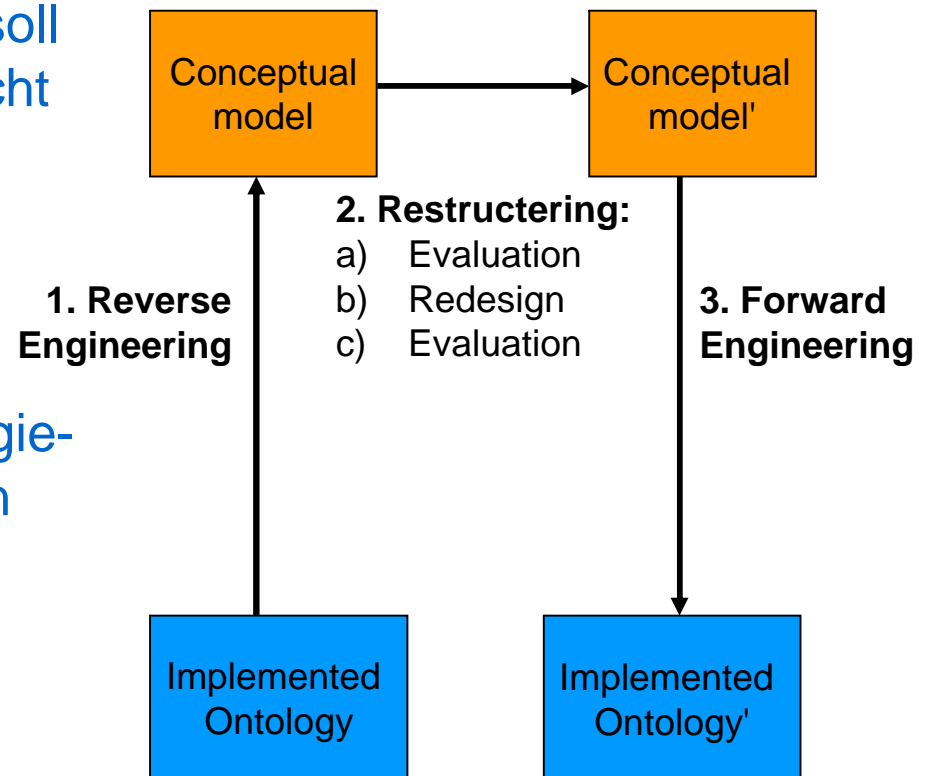
Table 3.12: Summary of the ontology development process.

Feature		Cyc	Uschold & King	Grüninger & Fox	KACTUS	METHONTOLOGY	SENSUS	On-To-Knowledge	
Ontology management activities	Scheduling	NP	NP	NP	NP	<i>Proposed</i>	NP	<i>Described¹²</i>	
	Control	NP	NP	NP	NP	<i>Proposed</i>	NP	<i>Described</i>	
	Quality assurance	NP	NP	NP	NP	<i>Proposed</i>	NP	<i>Described</i>	
Ontology development-oriented activities	Pre-development processes	Environment study	NP	NP	NP	NP	NP	<i>Proposed</i>	
		Feasibility study	NP	NP	NP	NP	NP	<i>Described</i>	
	Development processes	Specification	NP	<i>Proposed</i>	<i>Described in detail</i>	<i>Proposed</i>	<i>Described in detail</i>	<i>Proposed</i>	<i>Described in detail</i>
		Conceptualization	NP	NP	<i>Described in detail</i>	<i>Proposed</i>	<i>Described in detail</i>	NP	<i>Proposed</i>
		Formalization	NP	NP	<i>Described in detail</i>	<i>Described</i>	<i>Described</i>	NP	<i>Described</i>
		Implementation	<i>Proposed</i>	<i>Proposed</i>	<i>Described</i>	<i>Proposed</i>	<i>Described in detail</i>	<i>Described</i>	<i>Described</i>
	Post-development processes	Maintenance	NP	NP	NP	NP	<i>Proposed</i>	NP	<i>Proposed</i>
		Use	NP	NP	NP	NP	NP	NP	<i>Proposed</i>
Ontology support activities	Knowledge acquisition	<i>Proposed</i>	<i>Proposed</i>	<i>Proposed</i>	NP	<i>Described in detail</i>	NP	<i>Described</i>	
	Evaluation	NP	<i>Proposed</i>	<i>Described in detail</i>	NP	<i>Described in detail</i>	NP	<i>Proposed</i>	
	Integration	<i>Proposed¹³</i>	<i>Proposed</i>	<i>Proposed</i>	<i>Proposed</i>	<i>Proposed</i>	NP	<i>Proposed</i>	
	Configuration management	NP	NP	NP	NP	<i>Described</i>	NP	<i>Proposed</i>	
	Documentation	<i>Proposed</i>	<i>Proposed</i>	<i>Proposed</i>	NP	<i>Described in detail</i>	NP	<i>Described</i>	
	Merging and Alignment	NP	NP	NP	NP	NP	NP	NP	

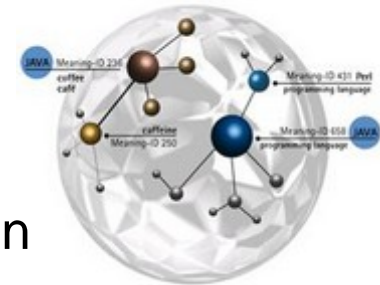
2.2 Ontology Re-Engineering



- Wozu Ontologie-Reengineering:
 - Fall 1: vorhandene Ontologie soll benutzt werden, passt aber nicht vollständig auf den neuen Anwendungsfall
 - Fall 2: Konzeptuelles Modell der Ontologie fehlt, bzw. lediglich die Ontologie-Implementierung ist vorhanden
- 3 Hauptaktivitäten:

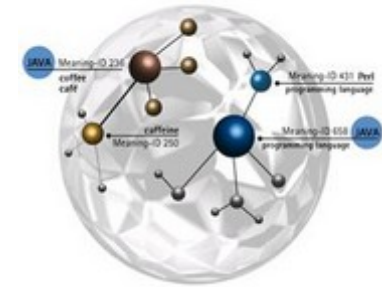


2.2 Ontology Learning

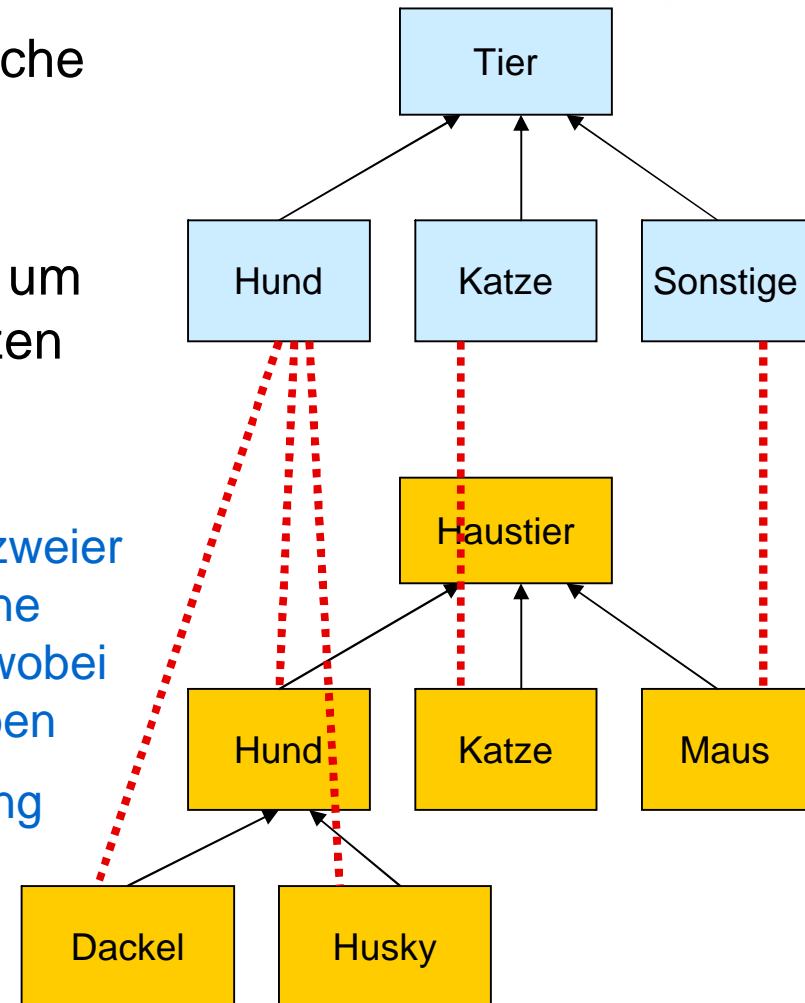


- Wissensakquisition braucht sehr viel Zeit und Ressourcen
- darum verschiedene Ansätze zur automatischen / semiautomatischen Wissensakquisition
- hierzu Nutzung von Verfahren den NLP + maschinelle Lernverfahren
- Ontologie-Lernen aus Text mittels NLP-Verfahren:
 - pattern based extraction
 - syntagmatische Relationen
 - paradigmatische Relationen
 - konzeptuelles Clustering
 - Ontology pruning
 - Concept learning auf Basis gegebener Taxonomien
- weitere Lern-Quellen für Ontologien: Instanzen, Schemata, Interoperability

2.2 Ontology Alignment / Merging



- klar: es kann mehr als eine Ontologie innerhalb einer bestimmten Domäne geben, Wissen kann auf unterschiedliche Arten modelliert werden
- darum möglicher Anwendungsfall: Zusammenführung von 2 Ontologien, um Konzepte / Instanzen aus beiden nutzen zu können:
 - **Ontology-Alignment:** Mapping / Verlinkung zwischen den Konzepten zweier Ontologien; Übersetzung in jeweils eine oder andere Richtung wird möglich, wobei die Original-Ontologien erhalten bleiben
 - **Ontology-Merging:** Zusammenführung der 2 getrennten zu einer einzigen Ontologie entlang des Mappings

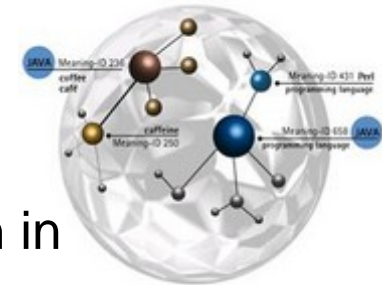


2.2 Collaborative Ont. Construction



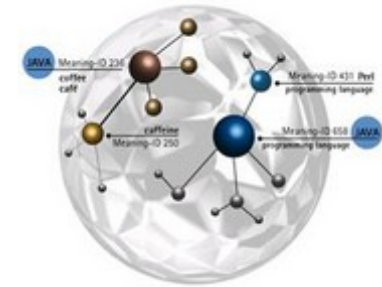
- hervorstechende Eigenschaft von Ontologien: konsensuales Wissen in einer Domäne
- hierfür notwendig: Richtlinien und Verfahren, wie konsensuales Wissen erreicht werden kann
- verschiedene Ansätze entwickelt, z.B.:
 - DILIGENT (in SWT)
 - Co4 (Protokoll für Kooperative Konstruktion von Ontologien)

2.3 Ontology Evaluation



- Evaluation vor der (Wieder-)Verwendung von Ontologien in Anwendungen oder anderen Ontologien notwendig
- gute Evaluation garantiert zwar keine Fehlerfreiheit, macht aber die Anwendung der Ontologie sicherer
- Gomez-Perez 1994b: Framework zur Ontologie-Evaluation:
 1. Teilung der Evaluation in 2 Teile :
 - a) technische,
 - b) Benutzer-Evaluation
 2. Menge an Termen zur Beschreibung von Evaluationsprozessen + deren Definition
 3. Menge an Kriterien, wie Evaluationsprozesse auszuführen sind
 4. Einbau der Evaluation in Ontologiebau-Methodologien
 5. Tools zur Ontologie-Evaluation
 6. Einbau von Evaluationsmodulen in Tools zum Bau von Ontologien

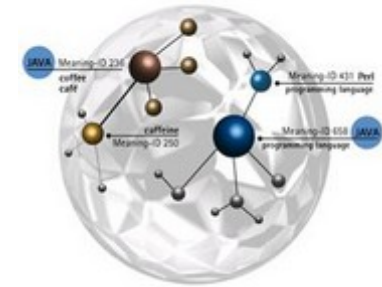
2.3 Ontology Evaluation



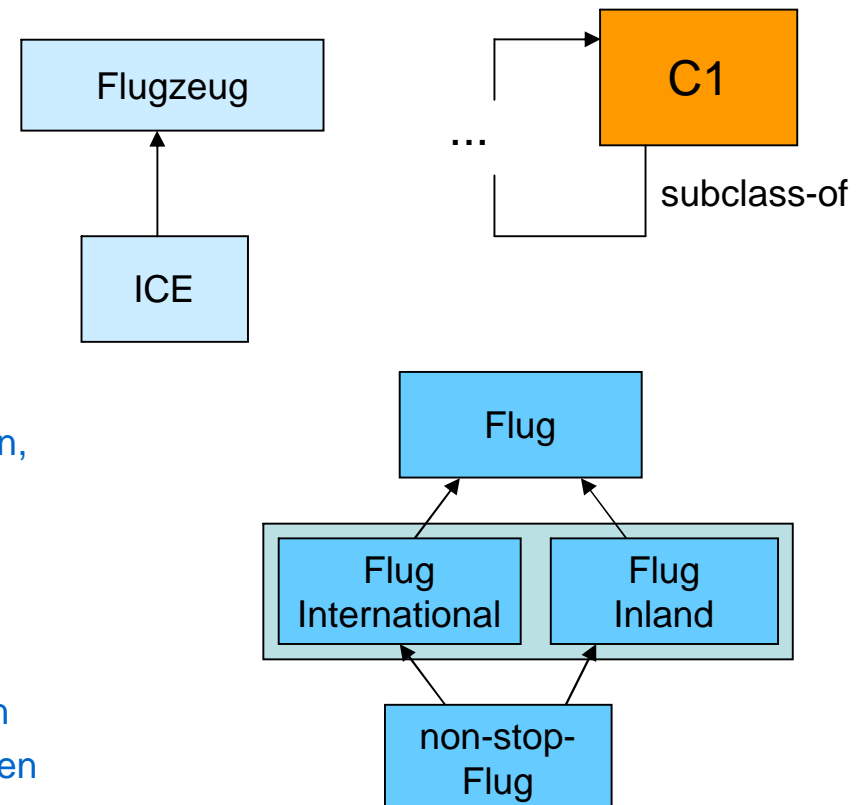
Begriffe der Ontologie-Evaluation:

- a) **Verifikation** (Entsprechung Spezifikation)
- b) **Validation** (Entsprechung der realen Welt)
- c) **Assessment** (Benutzerurteilssicht)
- d) **Konsistenz** (Widerspruchsfreiheit)
- e) **Vollständigkeit** (completeness)
- f) **Prägnanz** (conciseness, Redundanzfreiheit)

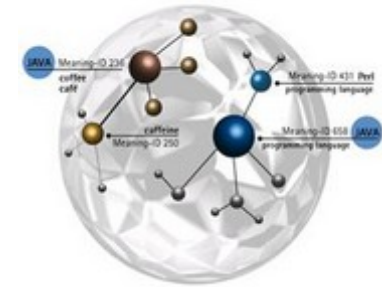
2.3 Ontology Evaluation



- **Taxonomie-Evaluation:** wesentlicher Teil der Ontologie-Evaluation
- verschiedene Fehler innerhalb von Taxonomien möglich:
- **Inconsistency:**
 - Circularity Errors
 - Semantic Inconsistency
 - Partition Errors
- **Incompleteness**
 - Incomplete Concept Classifikation
 - Partition Errors: disjoint knowledge omission, exhaustive knowledge omission
- **Redundanz:**
 - redundante Subklassen-Beziehungen
 - redundante Instance-Of-Beziehungen
 - identische formale Definitionen von Klassen
 - identische formale Definitionen von Instanzen



2.3 Ontology Evaluation mit OntoClean



- OntoClean: von der Ontology Group des CNR (Consiglio Nazionale della Ricerche) entwickelte Methode
- Ziel: Entfernung / Berichtigung von falschen Subklassenbeziehungen in Taxonomien anhand philosophischer Vorstellungen zu Rigidität, Identität und Einheit (+ evtl. Abhängigkeit)
- Zuordnung dieser Meta-Eigenschaften zu den Konzepten
- Anwendung bestimmter Vererbbarkeitsregeln bezüglich dieser Meta-Eigenschaften zum Finden und Berichtigen von Fehlern innerhalb der Ontologie-Taxonomie

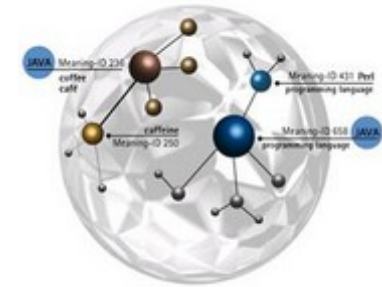
Metaeigenschaften beschreiben eine Klasse K

- **+K**: Alle Instanzen von K besitzen die Eigenschaft
- **-K**: nicht alle Instanzen von K besitzen die Eigenschaft
- **~K**: Keine der Instanzen von K besitzt die Eigenschaft

Betrachtung folgender Metaeigenschaften:

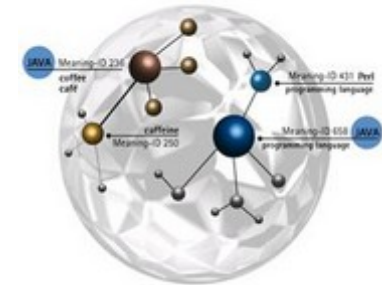
- Rigidität (+R, -R, ~R)
- Identität (+I, -I, +O, -O)
- Einheit (+U, -U, ~U)
- Abhängigkeit (+D, -D)

2.3 Ontology Evaluation mit OntoClean



- **Rigidität (+R, -R, ~R):** $\phi(x) \Rightarrow (x)$
 - Eigenschaft ist rigide (essentiell), wenn beschriebene Entität die Eigenschaft in jeder möglichen Welt ausgeprägt hat
- **Identität (+I, -I, +O, -O):** $\phi(x) \wedge \phi(y) \Rightarrow (p(x,y) \Leftrightarrow x = y)$
 - Identitätskriterien ermöglichen, verschiedene Instanzen einer Klasse voneinander zu unterscheiden
 - Klassen, die Identitätskriterien bereit stellen (also nicht von einer Oberklasse erben) werden mit (+O) getaggt, sonst (+I)
- **Einheit (+U, -U, ~U):** $\forall y(PART(y,x,t) \Rightarrow \forall z(PART(z,x,t) \Leftrightarrow \omega(z,y,t)))$
 - Eine Klasse besitzt das Einheitskriterium, wenn alle Teile einer Instanz der Klasse über eine Relation miteinander verbunden sind, diese Relation jedoch ausschließt, dass sie die einzelnen Teile darüber hinaus mit anderen Instanzen verbunden sind. ω ist also eine unifizierende Relation für x , falls alle Teile von x über ω verbunden sind.
- **Abhängigkeit (+D, -D):** $\forall x(\phi(x,t) \Rightarrow \exists y(\psi(y,t) \wedge \neg PART(y,x,t)))$
 - Abhängigkeit von externer Klasse, Bsp. +D(VATER, KIND)

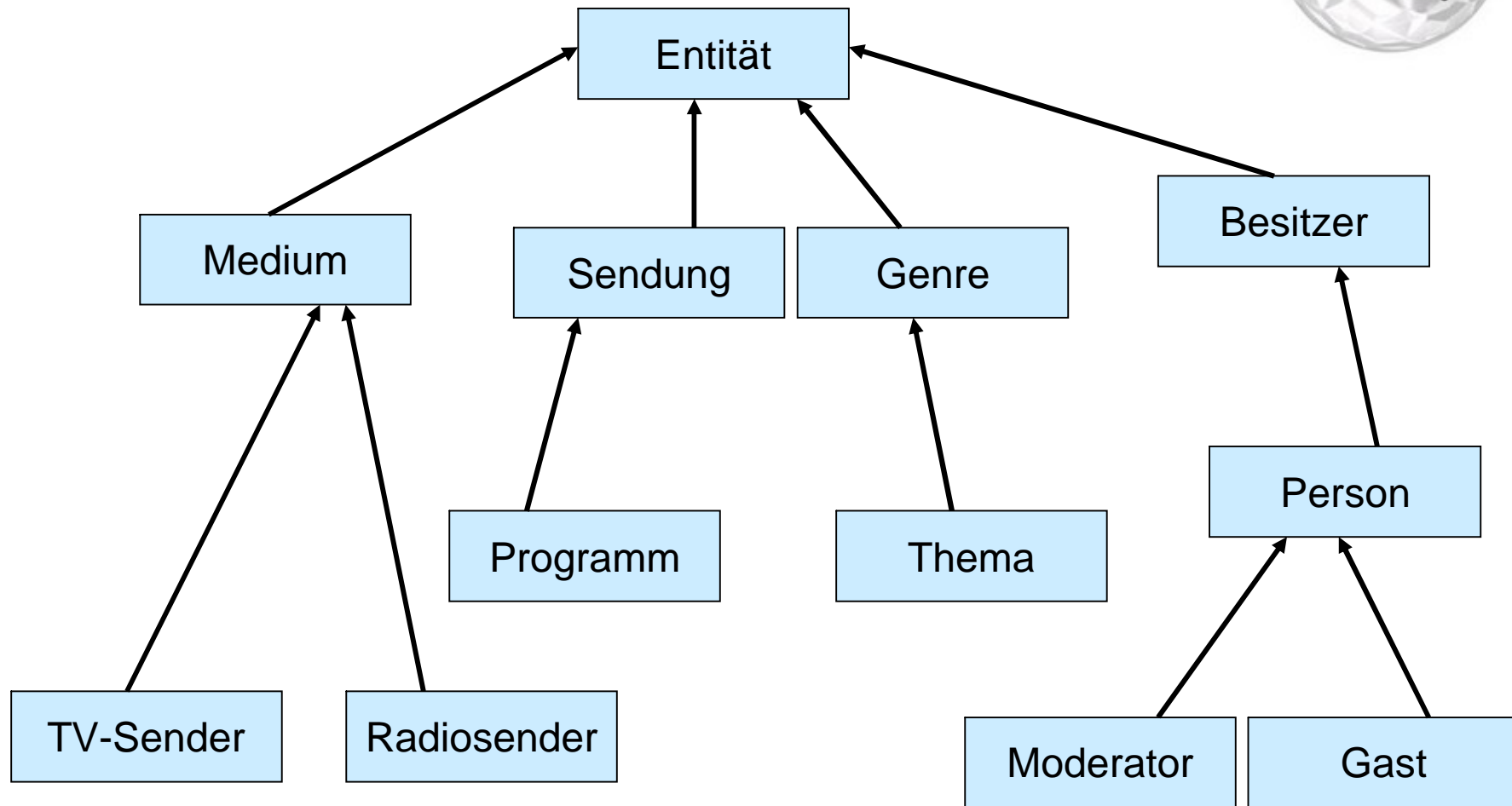
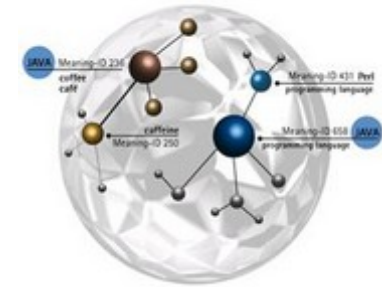
2.3 Ontology Evaluation mit OntoClean



Evaluation / Überarbeitung der Taxonomie in 5 Schritten:

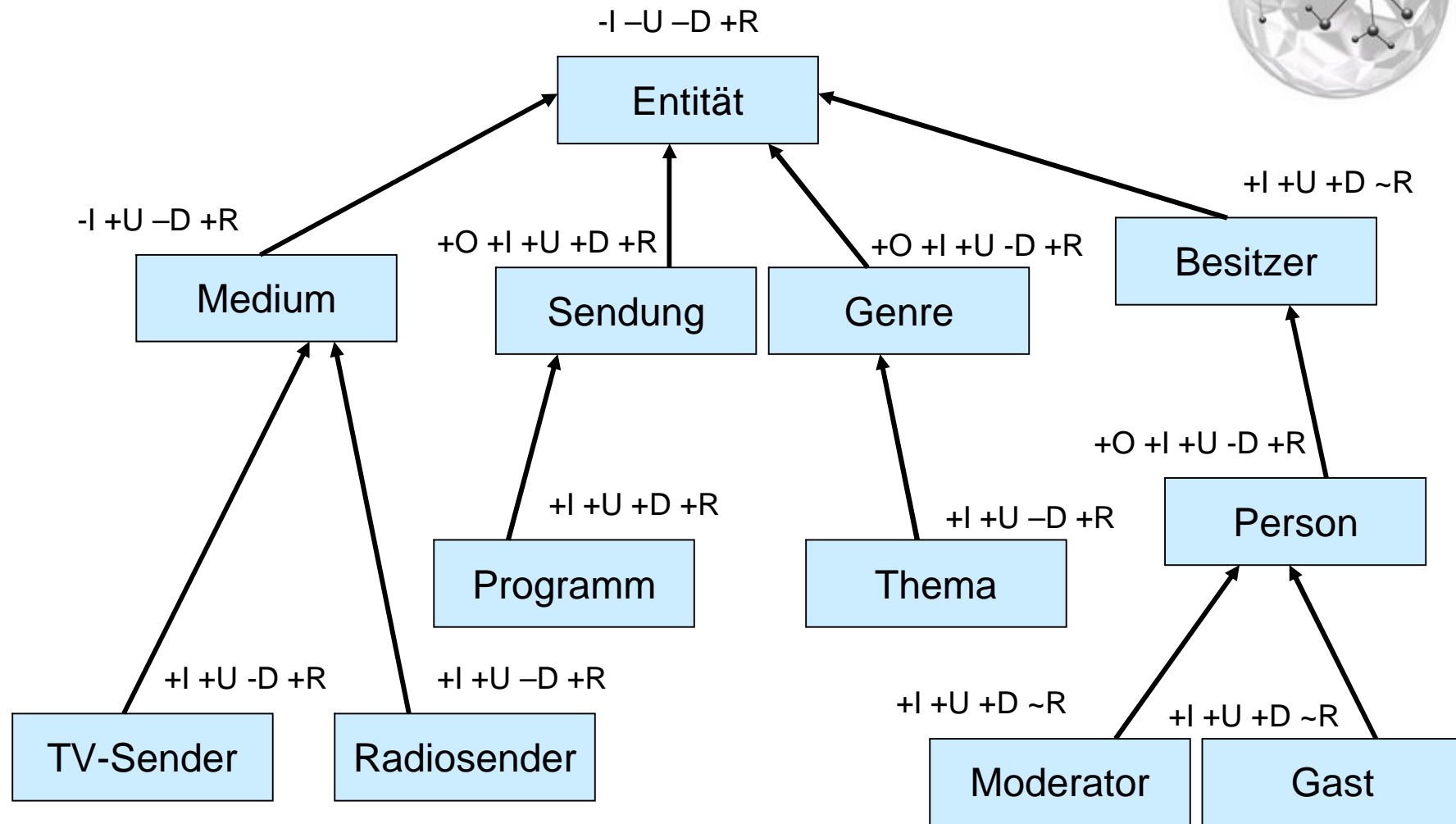
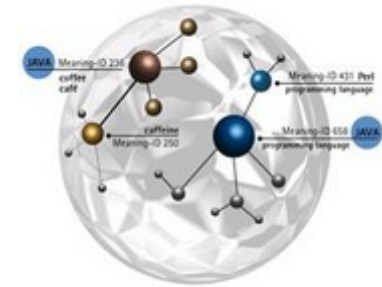
1. Zuordnung von Metaeigenschaften zu Termen
2. Backbone-Taxonomie aus den rigiden Konzepten bilden
3. Evaluation und Umformung der Taxonomie anhand von Regeln die Metaeigenschaften betreffend
 - Regel 1: Zwei Konzepte mit inkompatiblen Identitätskriterien (+I) können keine gemeinsamen Individuen haben, sind notwendigerweise disjunkt
 - Regel 2: Zwei Konzepte mit inkompatiblen Einheitskriterien (+U) können keine gemeinsamen Individuen haben, sind notwendigerweise disjunkt
 - Regel 3: gilt $\sim U$ kann für Subklassen nicht +U gelten
 - Regel 4: gilt $\sim R$ für die Superklasse, darf nicht $R+$ für Subklassen gelten
4. Nicht-Rigide Eigenschaften wieder einbeziehen → Regelanwendung
5. Vervollständigung der Taxonomie mit zusätzlichen Konzepten und Beziehungen

2.3 Ontology Evaluation mit OntoClean

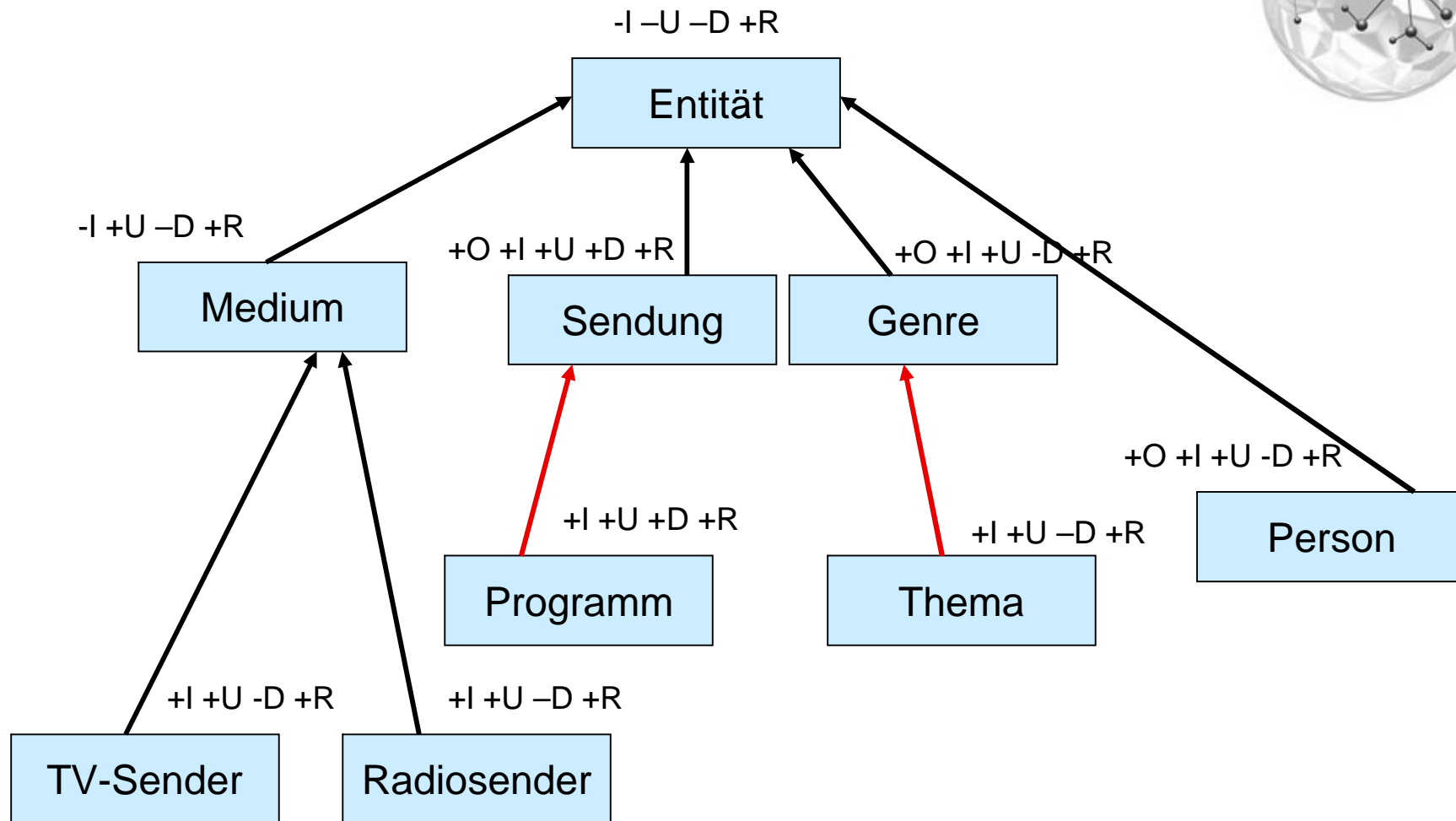
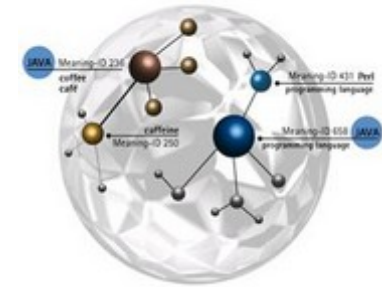


[Herb 2006]

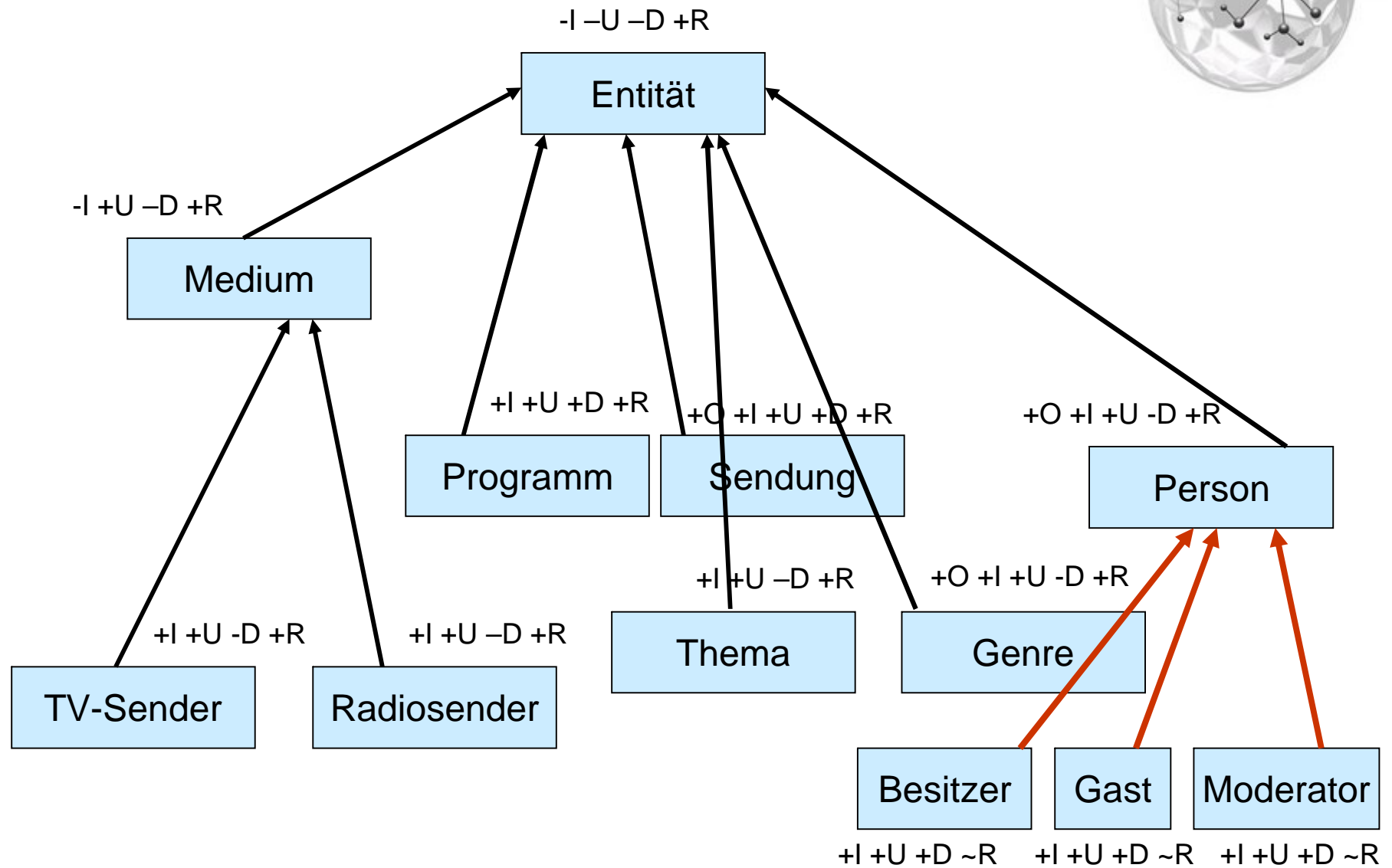
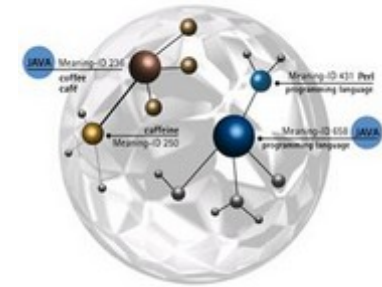
2.3 Ontology Evaluation mit OntoClean



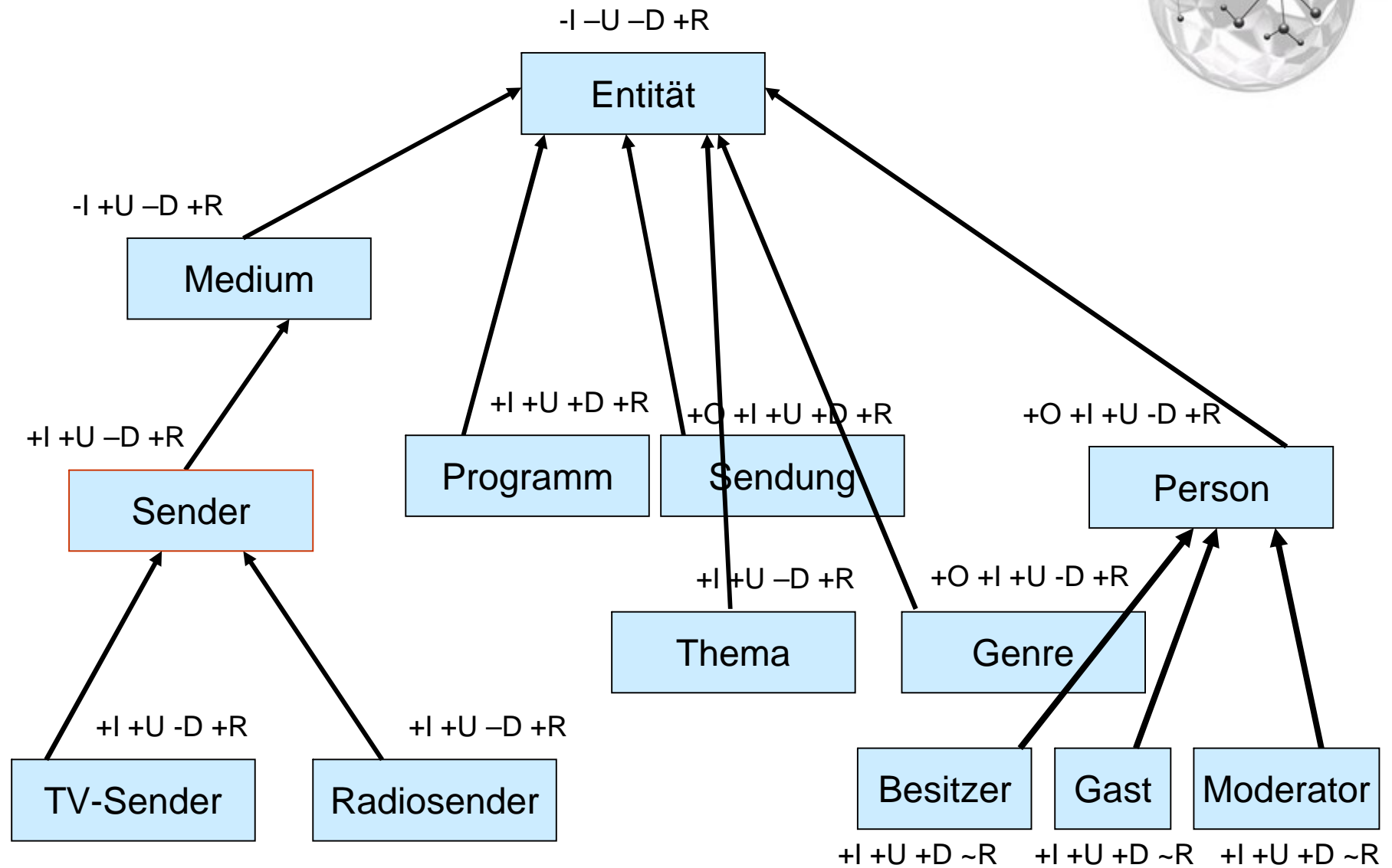
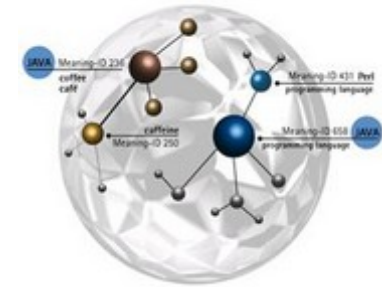
2.3 Ontology Evaluation mit OntoClean



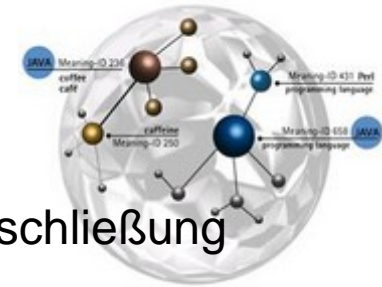
2.3 Ontology Evaluation mit OntoClean



2.3 Ontology Evaluation mit OntoClean

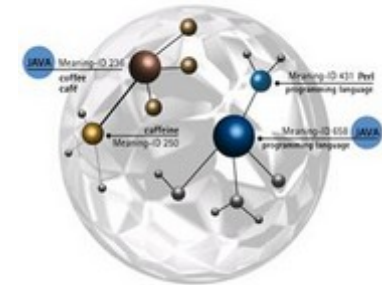


3. Zusammenfassung



- Ontologien unterstützen die Organisation, Verarbeitung und Erschließung von Wissen im Rahmen der EDV
- Ontologien lassen sich mit verschiedenen Formalismen repräsentieren (Logische, Datenbankbasierte, softwaretechnische Ansätze)
- Ontologien lassen sich nach der Reichhaltigkeit ihrer internen Struktur oder nach dem Gegenstand ihrer Konzeptualisierung kategorisieren
- es existieren zahlreiche Methoden und Methodologien zur Entwicklung von Ontologien mit unterschiedlichen Stärken und Schwächen
- diese beschreiben Vorgehensmodelle für die folgenden Vorgänge:
 - Methoden um Ontologien von Grund auf neu aufzubauen
 - Re-Engineering Methoden, bei Wiederverwendung von Ontologien
 - Ontologien -Lernverfahren (semi-/automatisch)
 - Ontologie Alignment/Merging
 - Methoden für kollaborative und räumlich verteilte Ontologie-Entwicklung
 - Methoden zur Evaluation von Ontologien

Literatur



- Davies et al. (2006) : Semantic Web Technologies, Wiley : Chichester.
- Gomez-Perez, Asuncion / Corcho, Oscar / Fernandez-Lopez, Mariano (2004): Ontological Engineering : with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web. First Edition (Advanced Information and Knowledge Processing), Springer.
- Guarino, Nicola (1998): Formal Ontology and Information Systems, in: Guarino, Nicola (Hrsg.), Formal Ontology in Information Systems. Proceedings of FOIS'98, Trento, Italy, 6-8 June 1998. Amsterdam, IOS Press, pp. 3-15.
- Guarino, Nicola / Giaretta, Pierdaniele (1995): Ontologies and Knowledge Bases: Towards a Terminological Clarification, in: Mars, N. J. I. (Hrsg.): Towards Very Large Knowledge Bases: Knowledge Building & Knowledge Sharing, IOS Press, S. 25-32.
- Herb, Mario (2006): Ontology Engineering mit OntoClean, in: www.ipd.uni-karlsruhe.de/~oosem/S2D2/material/1-Herb.pdf, 04.06.2007.