



Semantic Web

Vorlesung

Dr. Harald Sack

Hasso-Plattner-Institut für Softwaresystemtechnik

Universität Potsdam

Wintersemester 2008/09



<http://sw0809.blogspot.com/>

Blog zur Vorlesung: <http://sw0809.blogspot.com/>

Die nichtkommerzielle Vervielfältigung, Verbreitung und Bearbeitung dieser Folien ist zulässig
(Lizenzbestimmungen CC-BY-NC).

Semantic Web - Vorlesungsinhalt

2

1. Einführung
2. Die Sprachen des Semantic Web
3. Wissensrepräsentation
4. **Ontology Engineering**
5. Web of Trust

4. Ontology Engineering

4.1. Ontologien - eine kurze Wiederholung

4.2. Methoden des Ontologie-Entwurfs

4.2.1. Allgemeines Vorgehen

4.2.2. Methode von Uschold und King

4.2.3. Methode von Grüninger und Fox

4.2.4. METHONTOLOGY

4.2.5. On-To-Knowledge

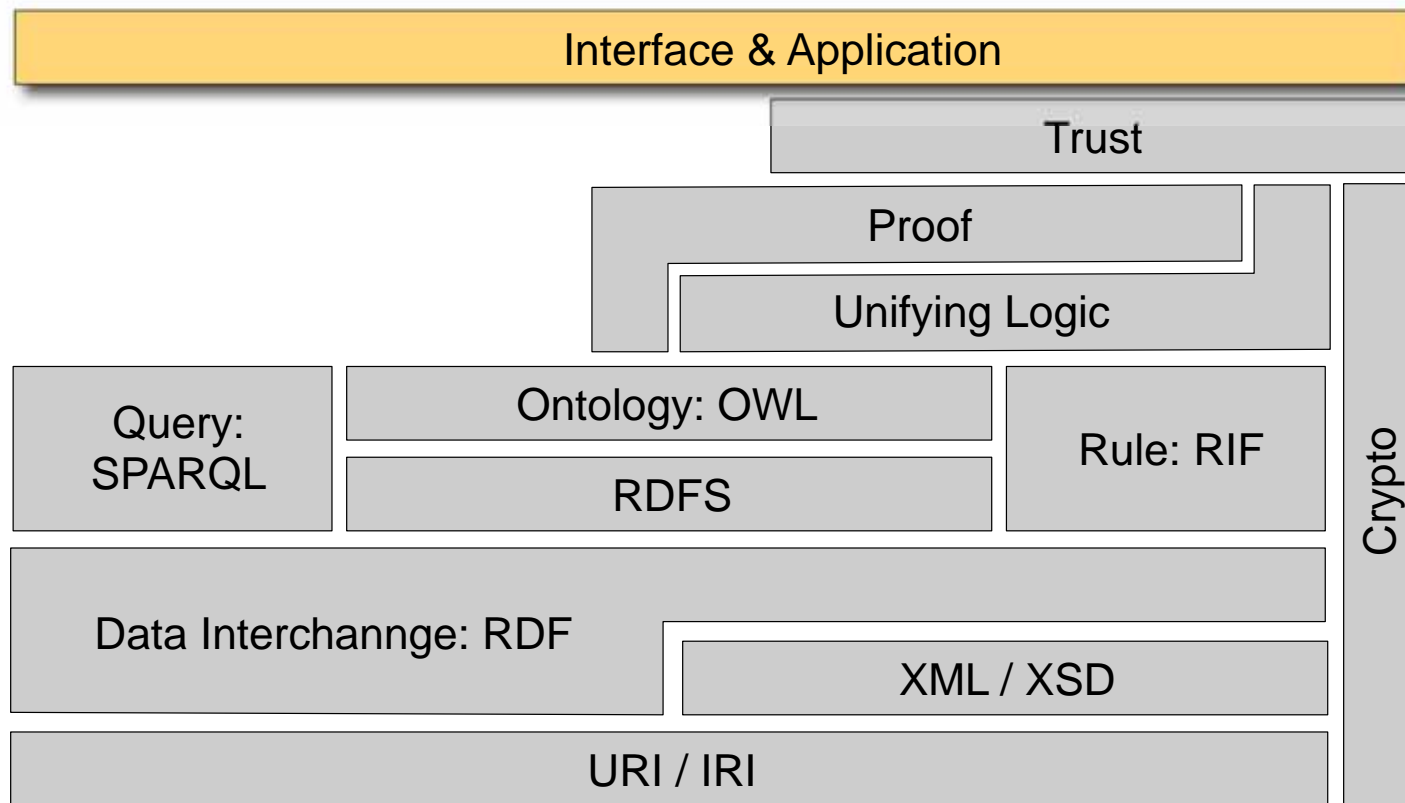
4.2.6. Ontology 101

4.3. Ontology Learning, Ontology Mapping und Ontology Merging

4. Ontology Engineering

4

Semantic Web Architektur



4. Ontology Engineering



Vorlesung Semantic Web, Dr. Harald Sack, Hasso-Plattner-Institut, Universität Potsdam

4. Ontology Engineering

6

4. Ontology Engineering

4.1. Ontologien - eine kurze Wiederholung

4.2. Methoden des Ontologie-Entwurfs

4.2.1. Allgemeines Vorgehen

4.2.2. Methode von Uschold und King

4.2.3. Methode von Grüninger und Fox

4.2.4. METHONTOLOGY

4.2.5. On-To-Knowledge

4.2.6. Ontology 101

4.3. Ontology Learning, Ontology Mapping und Ontology Merging

4. Ontology Engineering

4.1 Ontologien - Wiederholung

7

Was ist (eine) Ontologie?

- **Philosophie:** Theorie des Seins, versucht das Sein als solches zu erklären, durch Aufstellen eines Systems von universellen Kategorien und deren intrinsischer Beziehungen
- **künstliche Intelligenz (KI):** „explizite Spezifikation einer (intensionalen) Konzeptualisierung“
- **wissensbasierte Systeme:** System aus Konzepten/Vokabularien als Grundbaustein eines informationsverarbeitenden Systems
 - in Bezug auf Problemlösen unterscheidet man:
 - **Task Ontologien**, zur Beschreibung des Problemlöseprozesses und
 - **Domain Ontologien**, zur Beschreibung des Bezugsrahmens (Wissensgebiet) in dem dieser Prozess stattfindet

4. Ontology Engineering

4.1 Ontologien - Wiederholung

8

Was ist (eine) Ontologie?

An Ontology is a
formal specification ⇒ machine understandable
of a **shared** ⇒ group of people
conceptualization ⇒ about concepts
of a **domain of interest** ⇒ between general description and
individual use

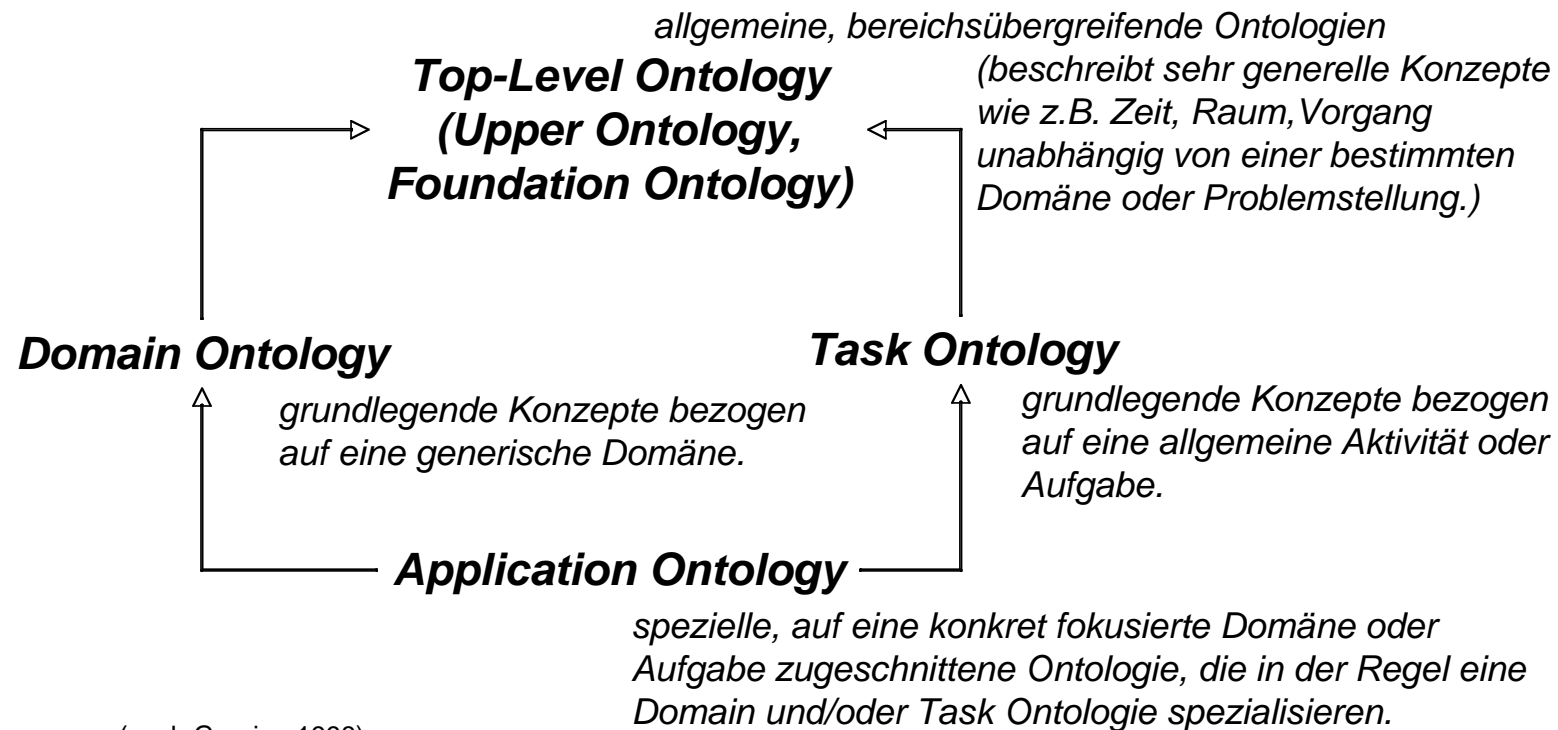
Tom Gruber, 1993

4. Ontology Engineering

4.1 Ontologien - Wiederholung

9

Was ist (eine) Ontologie?



4. Ontology Engineering

4.1 Ontologien - Wiederholung

10

Ontologien und das Semantic Web

- **Semantic Web** basiert auf der **Interoperabilität** von Metadaten
- Zwischen unterschiedlichen Metadaten besteht eine semantische Lücke (Semantic Gap), die mit Hilfe von Ontologien überbrückt werden soll
- **Problem:**
 - unterschiedliche Ontologien zur Repräsentation identischer Sachverhalte

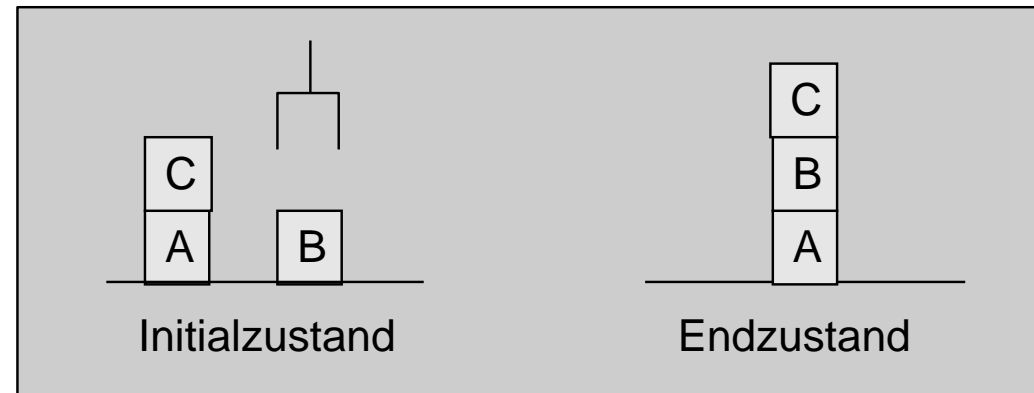
4. Ontology Engineering

4.1 Ontologien - Wiederholung

11

Semantische Lücke

- einfaches Beispiel



Modellierung 1:

<i>Objekte</i>	<i>Relationen</i>
block A	on(X,Y)
block B	clear(X)
block C	holding(X)
table T	handEmpty
hand H	

Modellierung 2:

<i>Objekte</i>	<i>Relationen</i>
block A	on(X,Y)
block B	clear(X)
block C	onTable(X)
	holding(X)
	handEmpty

4. Ontology Engineering

4.1 Ontologien - Wiederholung

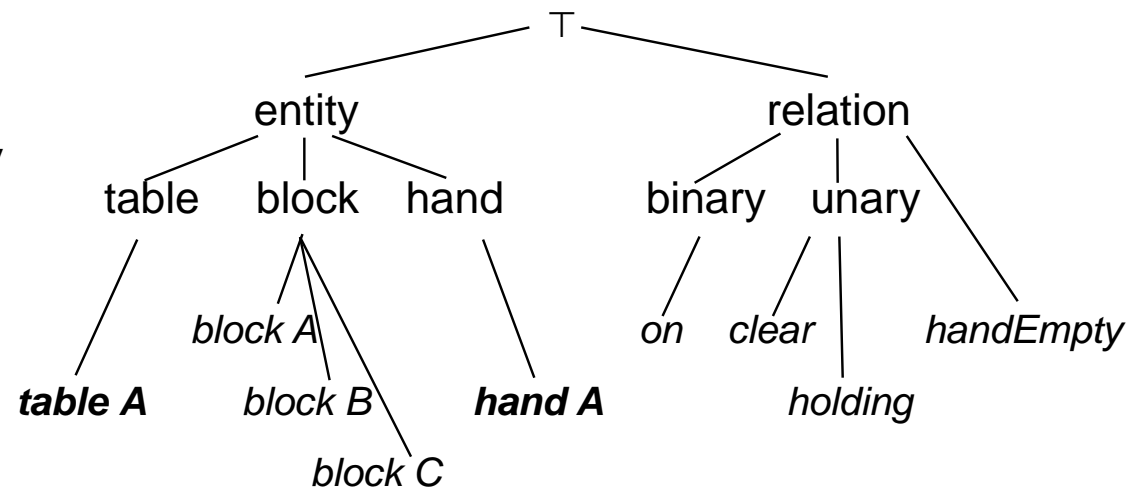
12

Modellbildung und Ontologie

- hinter den Modellen steckt eine Ontologie

Modellierung 1:

<i>Objekte</i>	<i>Relationen</i>
block A	on(X,Y)
block B	clear(X)
block C	holding(X)
table T	handEmpty
hand H	



Axiom: $on(X,Y) \wedge on(Y,Z) \rightarrow above(X,Z)$

4. Ontology Engineering

4.1 Ontologien - Wiederholung

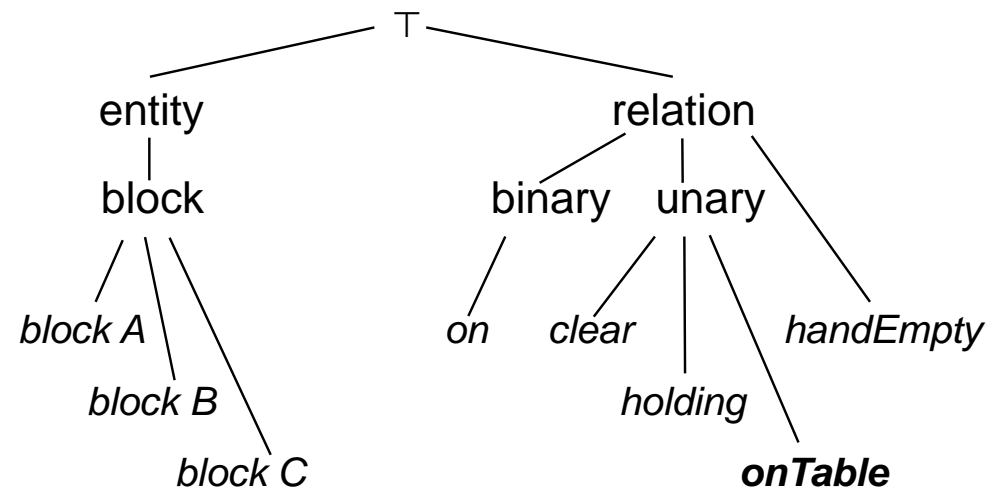
13

Modellbildung und Ontologie

- hinter den Modellen steckt eine Ontologie

Modellierung 2:

<i>Objekte</i>	<i>Relationen</i>
block A	on(X,Y)
block B	clear(X)
block C	onTable(X)
	holding(X)
	handEmpty



4. Ontology Engineering

4.1 Ontologien - Wiederholung

14

Ontology Engineering

- Ontologien sollen die Interoperabilität von Metadaten herstellen
- Daher benötigen wir
 - Methoden zum effizienten Entwurf und Design von Ontologien
(**Ontology Design**)
 - Methoden zum effizienten Vergleich von Ontologien
(**Ontology Mapping**)
 - Methoden zur effizienten Kombination verschiedener Ontologien
(**Ontology Merging**)
- Zur Unterstützung können automatisierte Verfahren herangezogen werden, zum
 - Ableiten von Ontologien aus einer Menge von Informationsressourcen
(**Ontology Learning**)
 - Bevölkern von Ontologien mit Individuen aus Informationsressourcen

4. Ontology Engineering

15

4. Ontology Engineering

4.1. Ontologien - eine kurze Wiederholung

4.2. Methoden des Ontologie-Entwurfs

4.2.1. Allgemeines Vorgehen

4.2.2. Methode von Uschold und King

4.2.3. Methode von Grüninger und Fox

4.2.4. METHONTOLOGY

4.2.5. On-To-Knowledge

4.2.6. Ontology 101

4.3. Ontology Learning, Ontology Mapping und Ontology Merging

Methoden des Ontologieentwurfs

- Eine Methodologie des Ontologie-Entwurfs beschreibt alle Aktivitäten die zur Konstruktion einer Ontologie notwendig sind
- Warum benötigt man eine formale Methodologie?
 - Entwicklung von konsistenten Ontologien
 - Effiziente Entwicklung komplexer Ontologien
 - Verteilte Entwicklung von Ontologien
- Unterscheide (nach Fernandez-Lopez et. al., 1997)
 - Ontology management activities
 - Ontology development oriented activities
 - Ontology support activities

4. Ontology Engineering

4.2 Methoden des Ontologieentwurfs

17

Ontology Management Activities

- Scheduling
 - Identifikation der durchzuführenden Aufgaben
 - Arrangement/Planung der durchzuführenden Aufgaben
 - Identifikation der benötigten Ressourcen (Zeit, Speicherplatz, etc...)
- Control
 - Garantiert korrekte Abwicklung der durchzuführenden Aufgaben
- Quality Assurance
 - Qualitätssicherung aller im dem Entwicklungsprozess anfallender Produkte (Ontologien, Software, Dokumentation)



4. Ontology Engineering

4.2 Methoden des Ontologieentwurfs

18

Ontology Development Oriented Activities

1. Pre-Development
2. Development
3. Post-Development

1. Pre-Development

- Environment study
 - Auf welchen Plattformen soll die Ontologie laufen?
 - Für welche Anwendungen ist die Ontologie bestimmt?
- Feasibility study
 - Kann die Ontologie tatsächlich erstellt werden?
 - Ist es überhaupt sinnvoll, die geplante Ontologie zu erstellen?



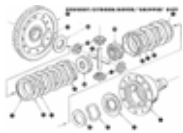
4. Ontology Engineering

4.2 Methoden des Ontologieentwurfs

19

Ontology Development Oriented Activities

2.Development



- Specification
 - Warum wird die Ontologie erstellt, was ist der beabsichtigte Nutzen und wer sind die End-Anwender?



- Conceptualization
 - Strukturiert Domain-Wissen in konzeptuellem Modell



- Formalization
 - Formalisiert konzeptuelles Modell in semi-berechenbarem Modell



- Implementation
 - Konstruktion eines berechenbaren Modells in einer Ontologiesprache

4. Ontology Engineering

4.2 Methoden des Ontologieentwurfs

20

Ontology Development Oriented Activities

3. Post-Development

- Maintenance
 - Update und Korrektur der Ontologie (falls nötig)
- Use / Reuse
 - Einsatz der Ontologie in den geplanten Anwendungen oder auch in anderen Ontologien



Ontology Support Activities

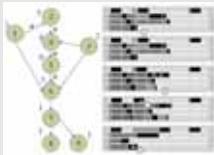
- Knowledge Acquisition
 - Wissen von Experten (semi)automatisch gewinnen (**Ontology Learning**)
- Evaluation
 - Technische Überprüfung der Ontologien in jeder Stufe der Entwicklung
- Integration
 - Wiederverwendung bereits existierender Ontologien (**Ontology Reuse**)
- Merging
 - Konstruktion einer neuen Ontologie aus bereits existierenden innerhalb einer bestimmten Domain
- Alignment
 - Abbildungsvorschriften (Mapping) zwischen den beteiligten Ontologien
- Documentation
 - Jede Stufe der Ontologie-Entwicklung wird akkurat dokumentiert
- Configuration Management
 - Verwaltet alle Versionen der Dokumentation und der entwickelten Ontologie

4. Ontology Engineering

4.2 Methoden des Ontologieentwurfs

22

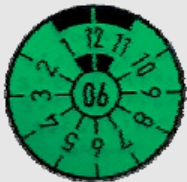
Management



scheduling



control



quality assurance

Development Oriented



environment study



feasibility study



specification



conceptualization



formalization



implementation



maintenance



use / reuse

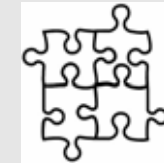
Support



knowledge acquisition



evaluation



integration



documentation



merging



configuration management



alignment

4. Ontology Engineering

23

4. Ontology Engineering

4.1. Ontologien - eine kurze Wiederholung

4.2. Methoden des Ontologie-Entwurfs

4.2.1. Allgemeines Vorgehen

4.2.2. Methode von Uschold und King

4.2.3. Methode von Grüninger und Fox

4.2.4. METHONTOLOGY

4.2.5. On-To-Knowledge

4.2.6. Ontology 101

4.3. Ontology Learning, Ontology Mapping und Ontology Merging

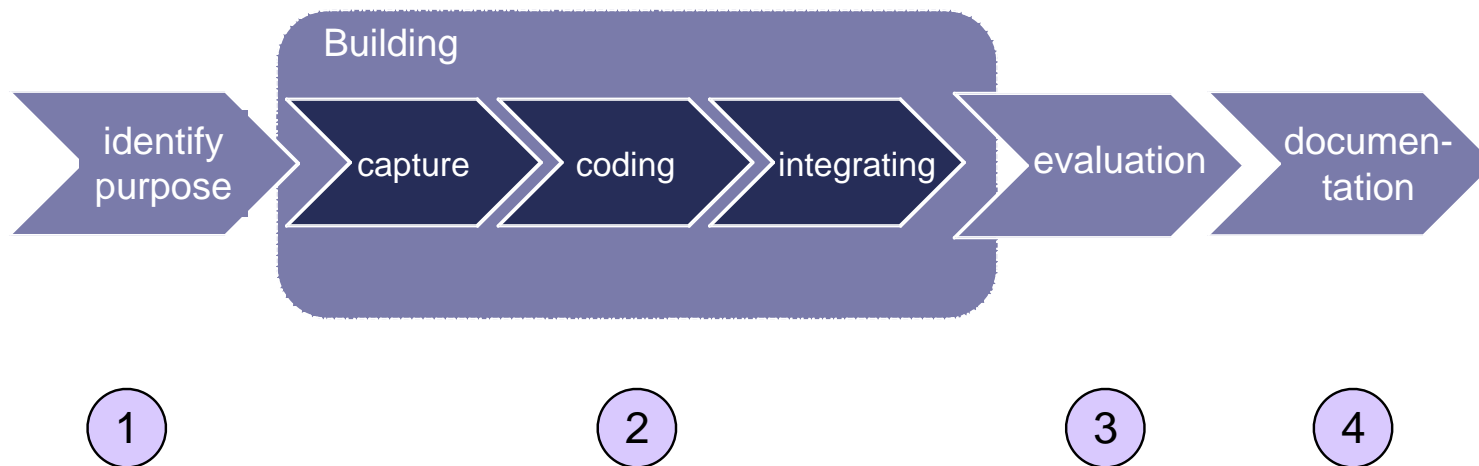
4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

24

Methode nach Uschold und King

- Prozessbasierte Entwicklung

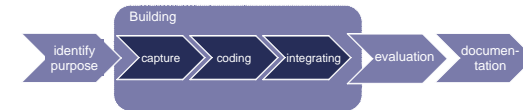


4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

25

Methode nach Uschold und King



1 Identifiziere Zweck und Anwendungsgebiet

- Warum wird die Ontologie benötigt?
- Beabsichtigte Verwendung?
 - (use / reuse / share / used as part of KB / ...)
- Identifiziere relevante Begriffe

Bsp.: Reise-Ontologie

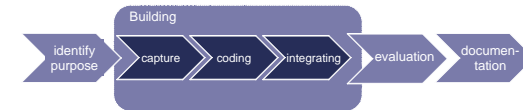
- *Aufbau eines gemeinsamen Wissensmodell über das Wissensgebiet Reisen, das in Reisebüros genutzt werden soll*
- *Ontologie könnte auch für andere Anwendungsgebiete genutzt werden, z.B. um einen Katalog für Unterkünfte oder Transportmöglichkeiten zu entwickeln*
- *relevante Begriffe z.B.: Orte, Typen von Orten, Unterkünfte, Arten von Unterkünften (Hotel / Motel / Camping / ...), Bahn, Busse, U-Bahn,...*

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

26

Methode nach Uschold und King



2 Ontologie Aufbau

- Ontology Capture
- Identifiziere Schlüsselkonzepte (Klassen) und Beziehungen (Relationen) des betreffenden Wissensgebiets und gebe diese in textueller Form an

Bsp.: Reise-Ontologie

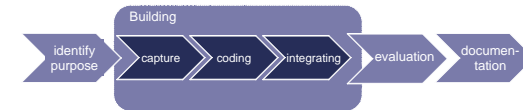
- *Transportmittel ist eine Klasse. Jeder Transport besitzt einen Startpunkt*
 - *Bus ist eine Klasse. Bus ist ein bestimmtes Transportmittel.*
 - *Stadtbus ist eine Klasse. Ein Stadtbus ist ein Bus, dessen Start- und Zielpunkt sowie dessen Zwischenstops in derselben Stadt liegen.*
- Identifikation der Ontologiekonzepte
 - Bottom-up / Top-down / Middle-Out

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

27

Methode nach Uschold und King



2 Ontologie Aufbau

- Ontology Capture
 - Bottom-up Identifikation der Ontologiekonzepte
 - erhöhter Gesamtaufwand
 - schwierig, Gemeinsamkeiten zwischen verwandten Konzepten zu finden
 - Erhöhtes Risiko von Inkonsistenzen
 - → dann Überarbeitung notwendig (noch mehr Aufwand)

Bsp.: Reise-Ontologie

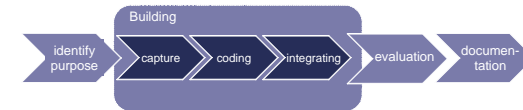
- *Transportmittel sollen in einer Bottom-up Strategie konzeptualisiert werden*

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

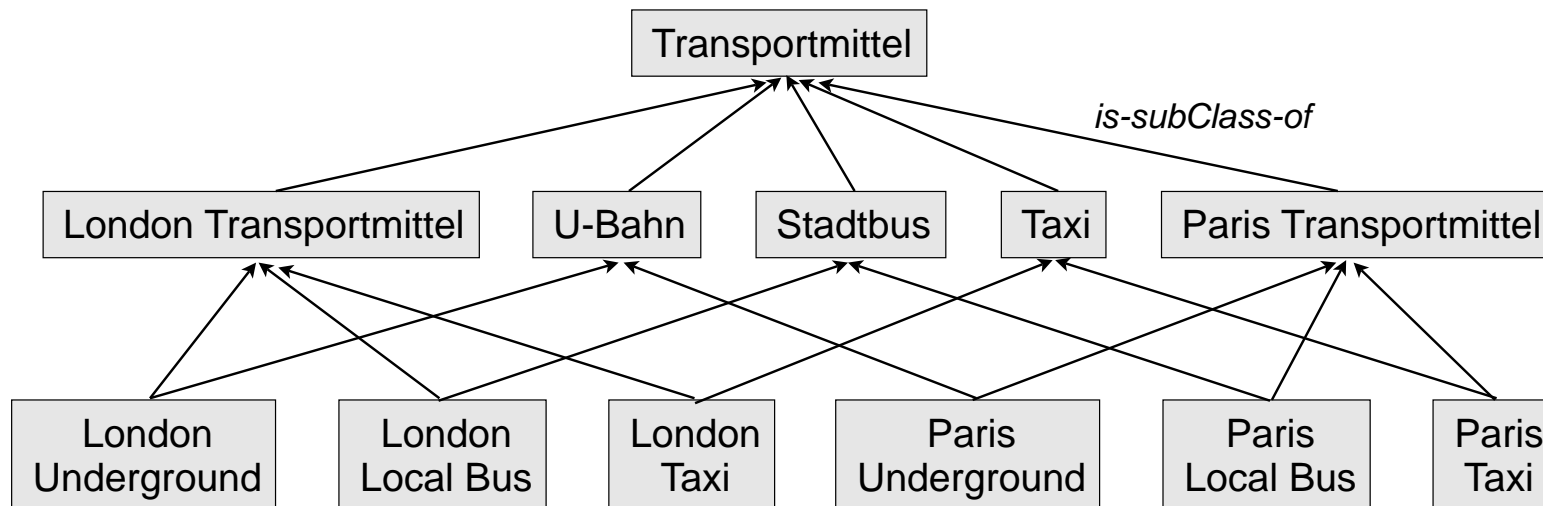
28

Methode nach Uschold und King



2 Ontologie Aufbau

- Ontology Capture
- Bottom-up Identifikation der Ontologiekonzepte

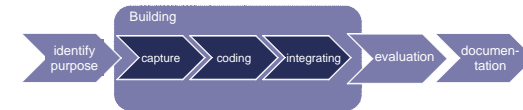


4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

29

Methode nach Uschold und King



2 Ontologie Aufbau

- Ontology Capture
 - Top-down Identifikation der Ontologiekonzepte
 - bessere Kontrolle des Detaillierungsgrades
 - möglicherweise werden aber abstrakte Kategorien gar nicht benötigt
 - geringere Stabilität des Modells
 - → dann Überarbeitung notwendig (mehr Aufwand)

Bsp.: *Reise-Ontologie*

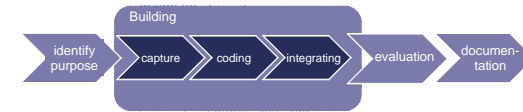
- *Transportmittel sollen in einer Top-down Strategie konzeptualisiert werden*

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

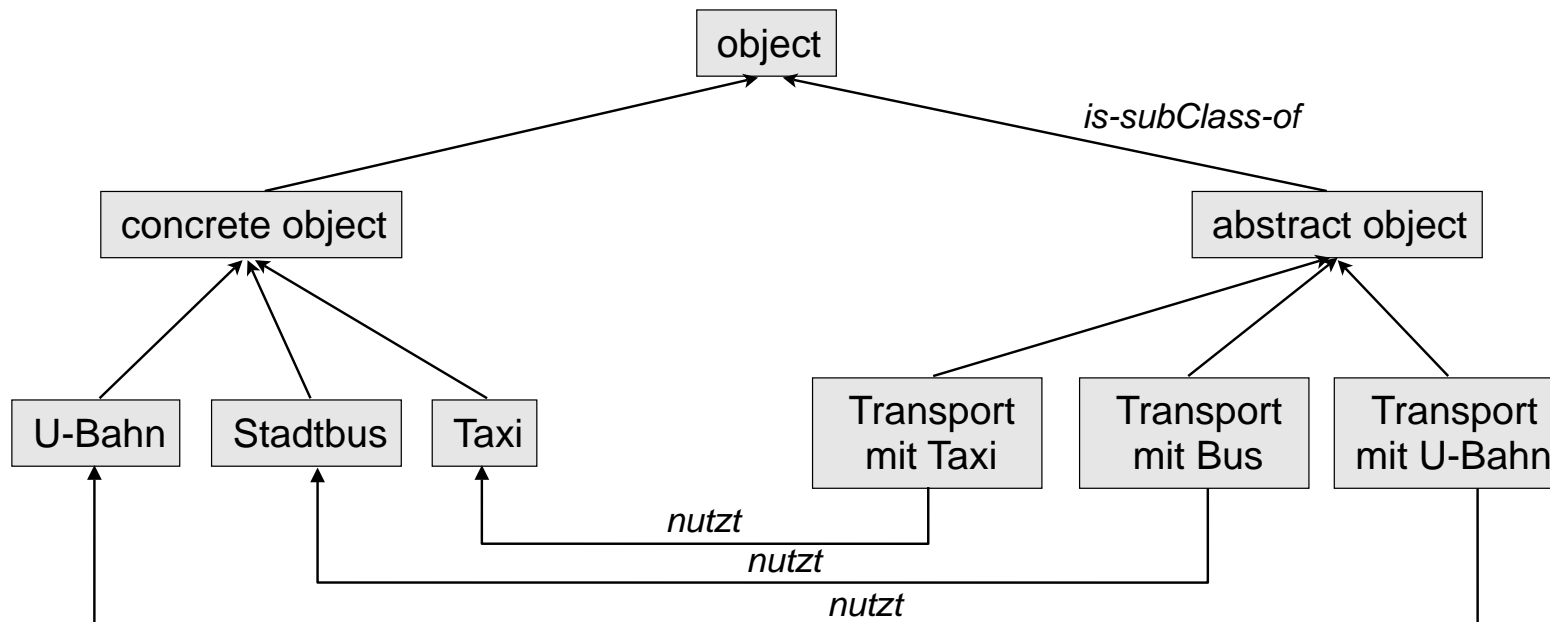
30

Methode nach Uschold und King



2 Ontologie Aufbau

- Ontology Capture
 - Top-Down Identifikation der Ontologiekonzepte

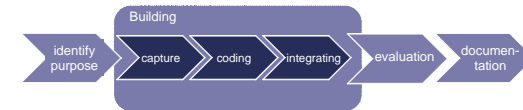


4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

31

Methode nach Uschold und King



2 Ontologie Aufbau

- Ontology Capture
 - Middle-Out Identifikation der Ontologiekonzepte
 - Starte mit Kern / Grundbegriffe, dann Spezialisierung / Generalisierung
 - ausbalanciert (bzgl. Detaillierungs-/Abstraktionsgrad)
 - stabiler als die beiden anderen Verfahren

Bsp.: Reise-Ontologie

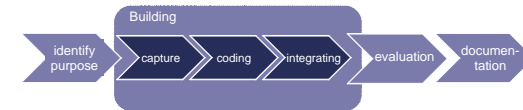
- *Transportmittel sollen in einer Middle-Out Strategie konzeptualisiert werden*

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

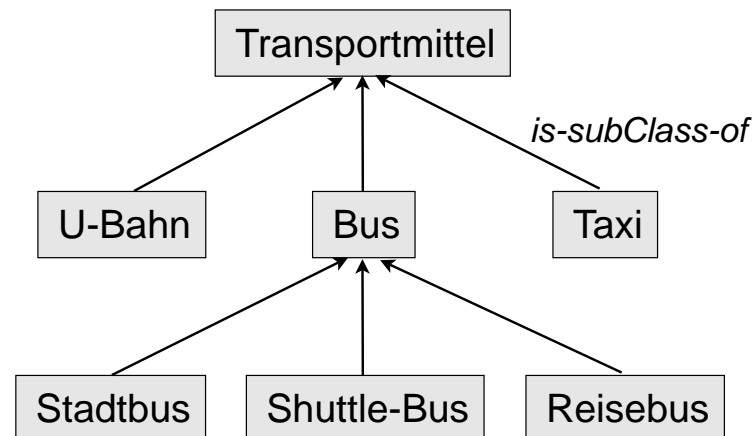
32

Methode nach Uschold und King



2 Ontologie Aufbau

- Ontology Capture
 - Middle-Out Identifikation der Ontologiekonzepte

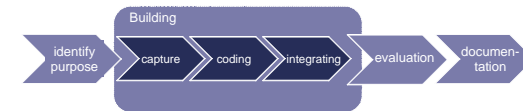


4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

33

Methode nach Uschold und King



2 Ontologie Aufbau

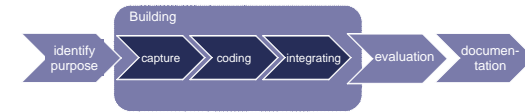
- Ontologie Aufbau
 - Coding
 - Alle an der Entwicklung beteiligten müssen sich auf eine gemeinsame Struktur der Wissensbasis geeinigt haben
- Integration of existing Ontologies
 - Entscheidung, ob und wie bereits existierende Ontologien genutzt werden sollen
 - Kann parallel zu den vorhergehenden Aktivitäten stattfinden

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.2 Uschold und King

34

Methode nach Uschold und King



3 Evaluation

- Technische Überprüfung der Ontologien und der damit assoziierten Software in jeder Stufe der Entwicklung

4 Documentation

- Einrichten der eventuell unterschiedlichen Dokumentationsrichtlinien

4. Ontology Engineering

35

4. Ontology Engineering

4.1. Ontologien - eine kurze Wiederholung

4.2. Methoden des Ontologie-Entwurfs

4.2.1. Allgemeines Vorgehen

4.2.2. Methode von Uschold und King

4.2.3. Methode von Grüninger und Fox

4.2.4. METHONTOLOGY

4.2.5. On-To-Knowledge

4.2.6. Ontology 101

4.3. Ontology Learning, Ontology Mapping und Ontology Merging

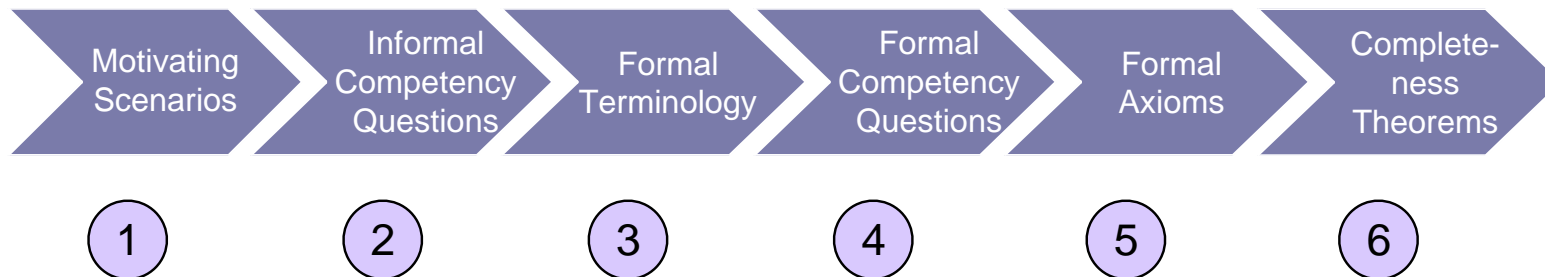
4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.3 Grüninger und Fox

36

Methode nach Grüninger und Fox

- Formaler Ansatz des Ontologie-Entwurfs
- Inspiriert durch den Entwurf wissensbasierter Systeme (Prädikatenlogik)



4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.3 Grüninger und Fox

37

Methode nach Grüninger/Fox



1 Identifikation der Hauptanwendungsszenarien

- dient der Beschreibung von Anforderungen an die Ontologie, die diese nach der Implementierung erfüllen soll
- enthält intuitive Lösungsvorschläge der im Szenario auftauchenden Probleme
- liefert erste Idee einer informell intendierten Semantik von Objekten und Beziehungen der späteren Ontologie

Bsp.: Reise-Ontologie

- *Aufbau eines gemeinsamen Wissensmodells über das Wissensgebiet "Reisen", das in Reisebüros genutzt werden soll*

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.3 Grüninger und Fox

38

Methode nach Grüninger/Fox



2 Erarbeitung informeller Kompetenzfragen

- Informelle Kompetenzfragen = Fragen in natürlicher Sprache, die mit Hilfe der zu entwickelnden Ontologie beantwortet werden können sollen
- Entspricht einer Requirement Specification gegen die die Ontologie später evaluiert werden können soll

Bsp.: Reise-Ontologie

- (a) *In Abhängigkeit der Präferenzen des Reisenden (Kulturreisen, Bergwandern, Badereisen, Kreuzfahrten, Abenteuerurlaub, ...) und anderer Einschränkungen (z.B. verfügbares Budget), welche Reiseziele wären die geeignetsten?*
- (b) *Welche Übernachtungsmöglichkeiten bieten sich für einen jungen Reisenden mit einem bescheidenen Budget ?*
- (c) ...

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.3 Grüninger und Fox

39

Methode nach Grüninger/Fox



2 Erarbeitung informeller Kompetenzfragen

- Ontologie gilt als nicht wohldefiniert, wenn nur „einfache“ Kompetenzfragen gestellt werden können
- Komplexe Kompetenzfragen können in einfachere zerlegt werden

Bsp.: Reise-Ontologie

- (a1) In Abhängigkeit der Präferenzen des Reisenden (Kulturreisen, Bergwandern, Badereisen, Kreuzfahrten, Abenteuerurlaub, ...), welche Reiseziele wären die geeignetsten?*
- (a2) In Abhängigkeit des verfügbaren Budgets, welche Reiseziele wären die geeignetsten?*

...

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.3 Grüninger und Fox

40

Methode nach Grüninger/Fox



2 Erarbeitung informeller Kompetenzfragen

- Jede Kompetenzfrage dient als Ausgangspunkt für die Definition von Aussagen, Einschränkungen (Constraints), notwendige Eingabedaten, etc...

Wie kann die gewonnene Antwort genutzt werden?

- *Daten zu den Präferenzen des Reisenden*
- *Aussagen: z.B. `jungerReisender.Alter < 30 Jahre`*
- *Einschränkungen: z.B. `Reisender hat Flugangst`*
- *Frage: Welche Reiseziele kommen in Frage für den Reisenden in Abhängigkeit der getroffenen Aussagen und Einschränkungen?*

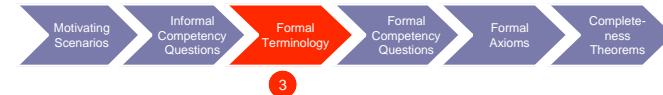
Was müssen wir wissen, um die Frage zu beantworten?

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.3 Grüninger und Fox

41

Methode nach Grüninger/Fox



- 3 Spezifikation der informellen Terminologie in Prädikatenlogik
 - Extraktion der Terminologie (Konzepte, Attribute, Relationen) aus den Kompetenzfragen und dessen formale Repräsentation

(1) Identifikation der Objekte in der zu repräsentierenden Domäne

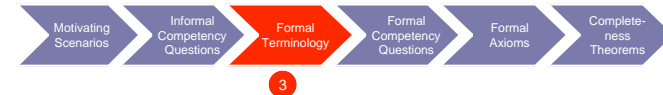
- *Bsp. Reise-Ontologie*
 - *Instanzen aus dem Wissensgebiet "Reisen"*
z.B. Rom, Madrid, London, New York, Flug DL1234, Bus 125, Zug C4, Hotel Travel Lodge in Newcastle, Hotel Hassler-Medici in Rom, Hotel Ritz in New York,

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.3 Grüninger und Fox

42

Methode nach Grüninger/Fox



3 Spezifikation der informellen Terminologie in Prädikatenlogik

(2) Identifikation der Prädikate

• *Unäre Prädikate = Konzepte, binäre Prädikate = Relationen*

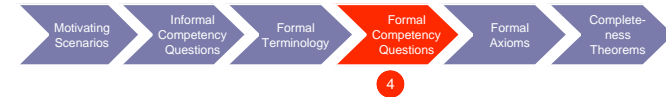
- *Bsp. Reise-Ontologie*
- *transportmittel (\$transport)*
- *bus (\$bus)*
- *zug (\$zug)*
- *reisender (\$reisender)*
- *hotel (\$hotel)*
- *reise (\$reise)*
- *ort (\$ort)*
- *budget (\$budget)*
- *reisender.name (\$reisender, \$string)*
- *hotel.name (\$hotel, \$string)*
- *alter (\$reisender, \$integer)*
- *hatReiseziel (\$reise, \$ort)*
- *hatReisestart (\$reise, \$ort)*
- *hatPräferenzen (\$reisender, \$präferenzen)*
- *möchteVerreisen (\$reisender, \$reise)*
- *hatBudget (\$reisender, \$budget)*

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.3 Grüninger und Fox

43

Methode nach Grüninger/Fox



4 Erzeugung formaler Kompetenzfragen

Bsp. Reise-Ontologie

(a1) In Abhängigkeit der Präferenzen des Reisenden

*(Kulturreisen, Bergwandern, Badereisen, Kreuzfahrten, Abenteuerurlaub, ...),
welche Reiseziele wären die geeignetsten?*

$$\exists \$x \exists \$y (ort(\$x) \wedge reise(\$y) \wedge möchteVerreisen(c, \$y) \wedge alter(c, a) \\ \wedge hatPräferenzen(c, e) \wedge verfügbaresBudget(c, b) \wedge hatReiseziel(\$y, \$x)) ?$$

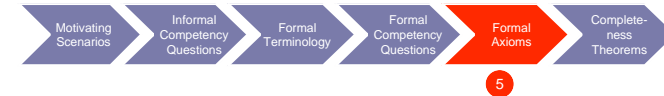
c...Reisender, b...Reisebudget, a...Alter, e...Präferenzen

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.3 Grüninger und Fox

44

Methode nach Grüninger/Fox



5 Spezifikation formaler Axiome

- Axiome spezifizieren Terme/Prädikate der Ontologie entsprechend ihrer Interpretation vermittels Prädikatenlogik

Bsp. Reise-Ontologie

Ein Reisender gilt ein Junger Reisender, wenn er noch keine 30 Jahre alt ist.

$$\forall \$x (\text{reisender}(\$x) \wedge (\forall \$y \text{ integer}(\$y) \wedge \text{alter}(x,y) \wedge (\$y < 30)) \\ \leftrightarrow \text{jungerReisender}(\$x))$$

Subklassenbeziehungen werden über Implikation ausgedrückt:

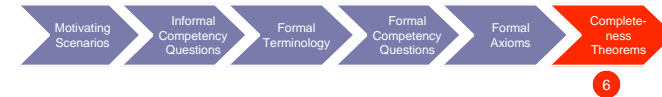
$$\forall \$x (\text{jungerReisender}(\$x) \rightarrow \text{reisender}(\$x))$$

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.3 Grüninger und Fox

45

Methode nach Grüninger/Fox



6 Spezifikation der Vollständigkeit

- Nach dem die Kompetenzfragen formalisiert wurden, muss festgelegt werden, welche Bedingungen gelten müssen, damit die Antworten auf die Kompetenzfragen als vollständig gelten
- Nicht immer notwendig

4. Ontology Engineering

46

4. Ontology Engineering

4.1. Ontologien - eine kurze Wiederholung

4.2. Methoden des Ontologie-Entwurfs

4.2.1. Allgemeines Vorgehen

4.2.2. Methode von Uschold und King

4.2.3. Methode von Grüninger und Fox

4.2.4. METHONTOLOGY

4.2.5. On-To-Knowledge

4.2.6. Ontology 101

4.3. Ontology Learning, Ontology Mapping und Ontology Merging

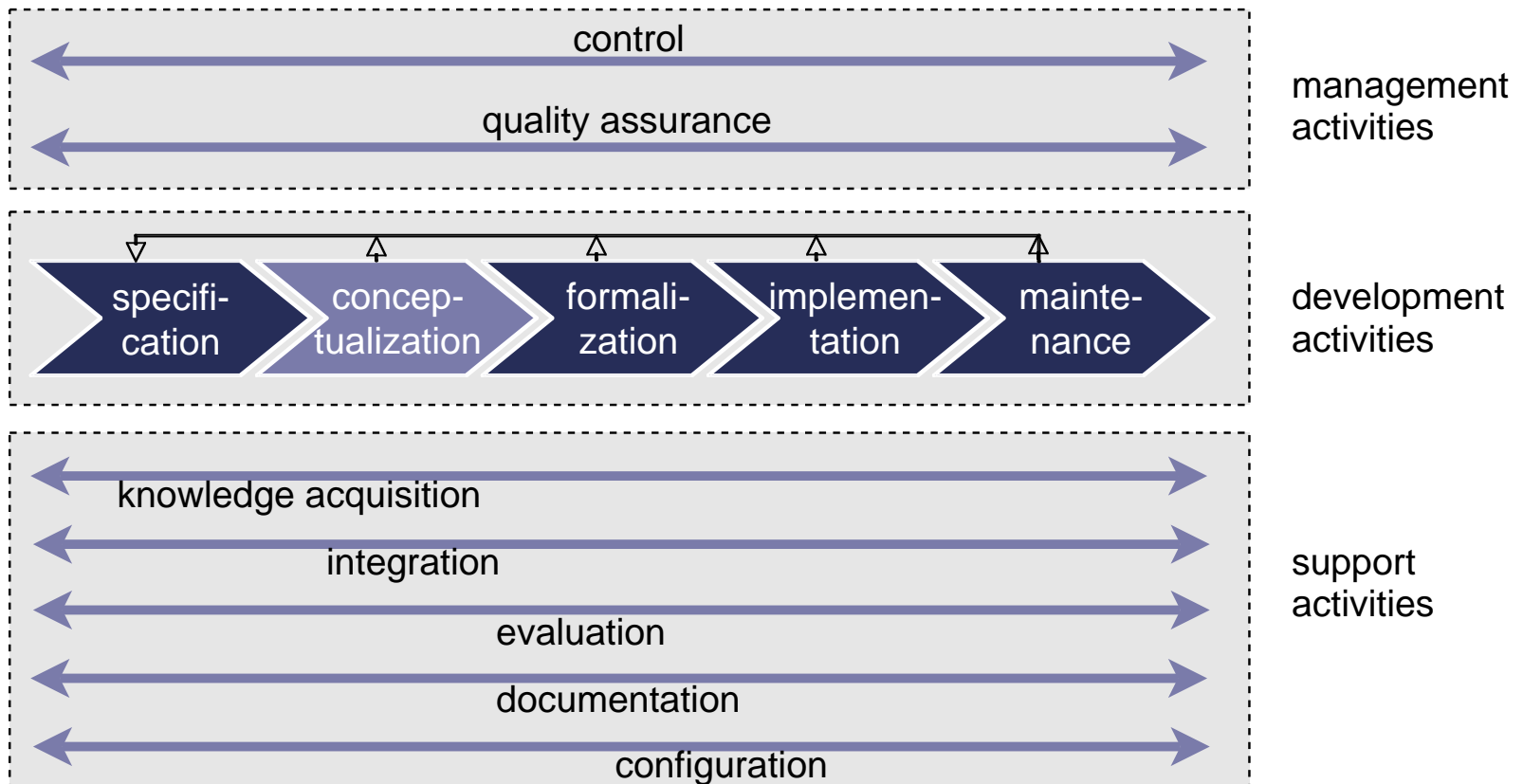
4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

47

METHONTOLOGY *Fernandez-Lopez/Gomez-Perez, 1999*

- Lebenszyklus-Modell basierend auf entwickelten Prototypen



Vorlesung Semantic Web, Dr. Harald Sack, Hasso-Plattner-Institut, Universität Potsdam

4. Ontology Engineering

4.2 Ontologientwurf / 4.2.4 METHONTOLOGY

48

METHONTOLOGY

- Einzelaufgaben der Konzeptualisierung

Task 1: Aufbau eines Term-Glossars

Task 2: Aufbau einer Konzept-Taxonomie

Task 3: Aufbau von ad-hoc binären Relationen

Task 4: Aufbau eines Konzept-Wörterbuchs

Task 5: detaillierte Beschreibung von ad-hoc binären Relationen /
Instanz-Attributen / Klassen-Attributen / Konstanten

Task 6: formale Beschreibung der Axiome und Regeln

Task 7: detaillierte Beschreibung der Instanzen

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

49

METHONTOLOGY

- Task 1: Aufbau eines Term-Glossars

Name	Synonyme	Akronyme	Beschreibung	Typ
Delta Flight		DL Flight	Flug, der von Delta Airlines durchgeführt wird	Konzept
Geschäftsreise	business trip		Spezielles Angebot für Geschäftsleute bestehend aus Flug und einm Hotel mit gehobenen Standards	Konzept
5-Sterne Hotel			Hotel mit sehr hohen Standards	Konzept
Luxusreise			Luxuriöse und teure Reise	Konzept
Reise	travell		Eine Reise von einem Ort zu einem anderen	Konzept
Reiseziel	destination		Ort, an dem die Reise endet	Relation
Ankunftszeit	arrival date		Zeitpunkt der Ankunft	Instanz Attribut

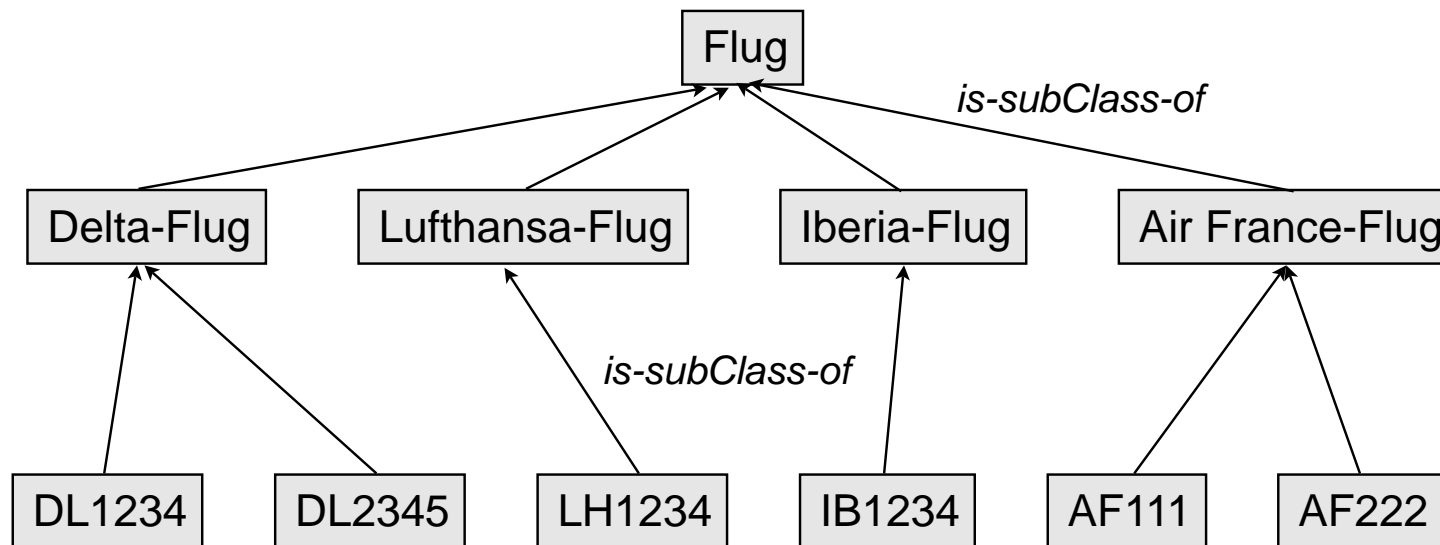
4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

50

METHONTOLOGY

- Task 2: Aufbau einer Konzept-Taxonomie



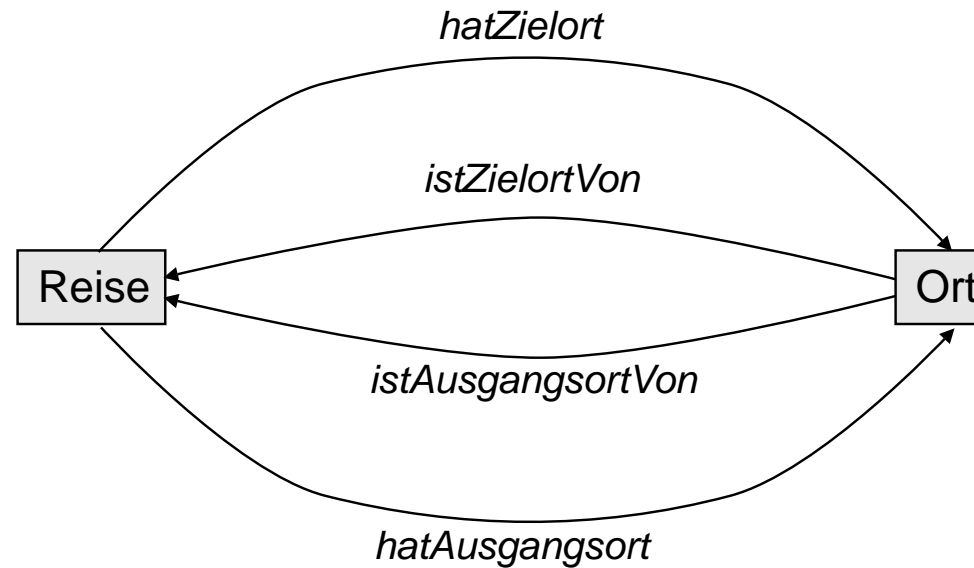
4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

51

METHONTOLOGY

- Task 3: Aufbau von ad-hoc binary relationships



4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

52

METHONTOLOGY

- Task 4: Aufbau eines Konzept-Wörterbuches

Konzeptname	Klassenattribut	Instanzattribut	Relationen
DL2345			derselbe Flug wie
Delta Flug	Firmenname		
5-Sterne Hotel	Anzahl Sterne		
Unterkunft		Preis für Standard-Zimmer	ist gelegen in
Reise		Ankunftszeit Firmenname Abfahrtszeit Kosten	

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

53

METHONTOLOGY

- Task 5.1: Detaillierte Beschreibung der binären Relationen

Relation	Domain	Kardinalität	Range	Eigenschaft	Inverses
derselbe Flug wie	Flug	N	Flug	symmetrisch transitiv	-
ist gelegen in	Unterkunft	1	Ort	-	-
Zielort	Reise	1	Ort	-	ist Zielort von
Ausgangs- ort	Reist	1	Ort	-	ist Ausgangsort von

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY



54

METHONTOLOGY

- Task 5.2: Detaillierte Beschreibung von Instanzattributen

Attribut-name	in Konzept	Typ	Einheit	Genauigkeit	Wertebereich	Kardinalität
Ankunftszeit	Reise	Datum	-	-	-	(0/1)
Abfahrtszeit	Reise	Datum	-	-	-	(0/1)
Kosten Hinweg	Reise	Float	currency	0,01	-	(0/1)
Kosten Rückweg	Reise	Float	currency	0,01	-	(0/1)
Firmenname	Reise	String	-	-	-	(0/N)

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

55

METHONTOLOGY

- Task 5.3: Detaillierte Beschreibung von Klassenattributen

Attribut-name	in Konzept	Typ	Einheit	Genauigkeit	Werte	Kardinalität
Anzahl Sterne	5-Sterne Hotel	Integer	Sterne	1	5	(1/1)
Anzahl Sterne	4-Sterne Hotel	Integer	Sterne	1	4	(1/1)
Anzahl Sterne	3-Sterne Hotel	Integer	Sterne	1	3	(1/1)
Anzahl Sterne	2-Sterne Hotel	Integer	Sterne	1	2	(1/1)
Firmenname	Lufthansa Flug	String	-	-	-	(1/1)

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

56

METHONTOLOGY

- Task 5.4: Detaillierte Beschreibung der Konstanten

Name	Typ	Wert	Einheit
max. Anzahl von Reisenden in einem Flugzeug	Integer	200	Personen

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

57

METHONTOLOGY

- Task 6.1: formale Beschreibung der Axiome

Axiom Name	Züge innerhalb Europas
Beschreibung	jeder Zug, der von einem Ort in Europa startet muss an einem anderen Ort in Europa enden
Ausdruck	$\forall(x,y,z) (\text{Zugreise}(x) \wedge \text{AbfahrtOrt}(x,y) \wedge \text{AnkunftOrt}(x,z) \wedge \text{OrtInEuropa}(y) \rightarrow \text{OrtInEuropa}(z))$
Konzepte	Zugreise, OrtInEuropa
beteiligte Attribute	-
binäre Relationen	AbfahrtOrt, AnkunftOrt
Variablen	x,y,z

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

58

METHONTOLOGY

- Task 6.1: formale Beschreibung der Regeln

Regel Name	Lufthansa Flug Regel
Beschreibung	Jeder Flug, der in Deutschland startet wird von Lufthansa durchgeführt
Ausdruck	$\text{OrtInDeutschland}(y) \wedge \text{Flug}(x) \wedge \text{abflugort}(x,y) \rightarrow \text{Firmenname}(x, \text{"Lufthansa"})$
Konzepte	OrtInDeutschland, Fluf
beteiligte Attribute	Firmenname
binäre Relationen	abflugort
Variablen	x,y

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.4 METHONTOLOGY

59

METHONTOLOGY

- Task 7: detaillierte Beschreibung der Instanzen

Instanzname	Konzeptname	Attribute	Werte
LH1234_Dez152008	LH1234	Firmenname	Lufthansa
		Abflugzeit	15/12/2008
		Ankunftszeit	15/12/2008
		Kosten Hinweg	349
LH2345_Dez152008	LH2345	Firmenname	Lufthansa
		Abflugzeit	15/12/2008
		Ankunftszeit	15/12/2008
		Kosten Hinweg	789

4. Ontology Engineering

50

4. Ontology Engineering

4.1. Ontologien - eine kurze Wiederholung

4.2. Methoden des Ontologie-Entwurfs

4.2.1. Allgemeines Vorgehen

4.2.2. Methode von Uschold und King

4.2.3. Methode von Grüninger und Fox

4.2.4. METHONTOLOGY

4.2.5. On-To-Knowledge

4.2.6. Ontology 101

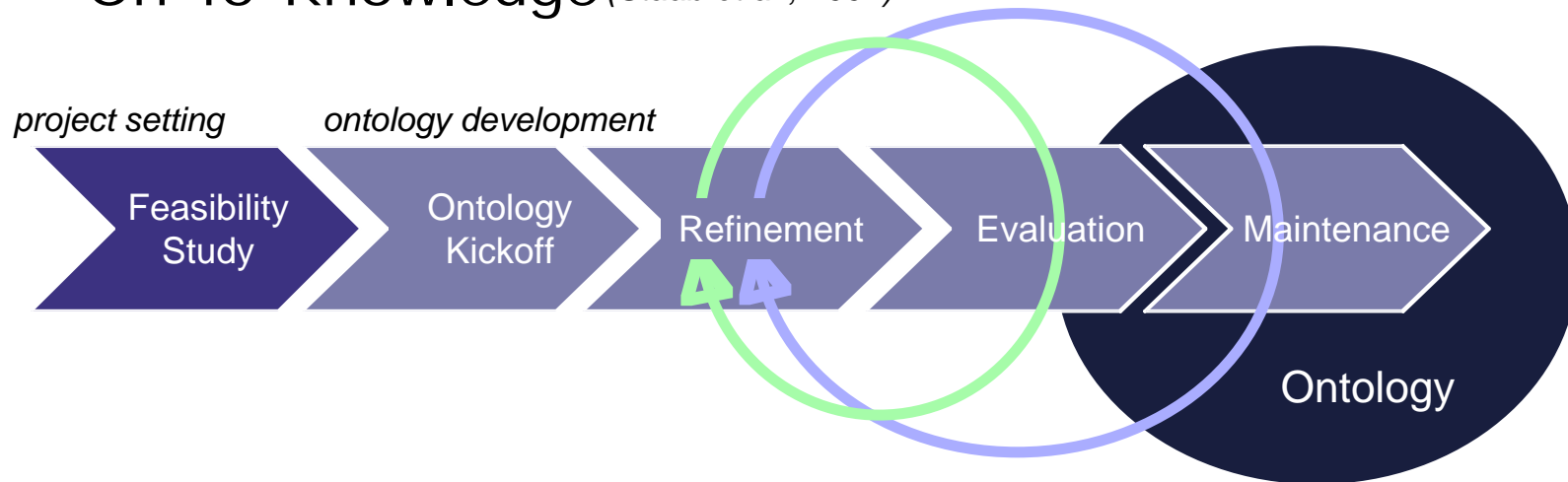
4.3. Ontology Learning, Ontology Mapping und Ontology Merging

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.5 On-To-Knowledge

51

On-To-Knowledge (Staab et al., 2001)



- identify **problem and opportunity areas**
- select most promising **focus area and target solution**

- **requirement specification**
- analyze input sources
- develop **baseline taxonomy**

- concept elicitation with domain experts
- develop baseline taxonomy
- conceptualize and formalize
- add relations and axioms

- identify **problem and opportunity areas**
- select most promising **focus area and target solution**

- manage organizational maintenance process

4. Ontology Engineering

52

4. Ontology Engineering

4.1. Ontologien - eine kurze Wiederholung

4.2. Methoden des Ontologie-Entwurfs

4.2.1. Allgemeines Vorgehen

4.2.2. Methode von Uschold und King

4.2.3. Methode von Grüninger und Fox

4.2.4. METHONTOLOGY

4.2.5. On-To-Knowledge

4.2.6. Ontology 101

4.3. Ontology Learning, Ontology Mapping und Ontology Merging

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.6 Ontology 101

53

Ontology Development 101 *(Noy, McGuinness, 2000)*

- Am Beispiel einer Ontologie für Weinbau und Nahrungsmittel



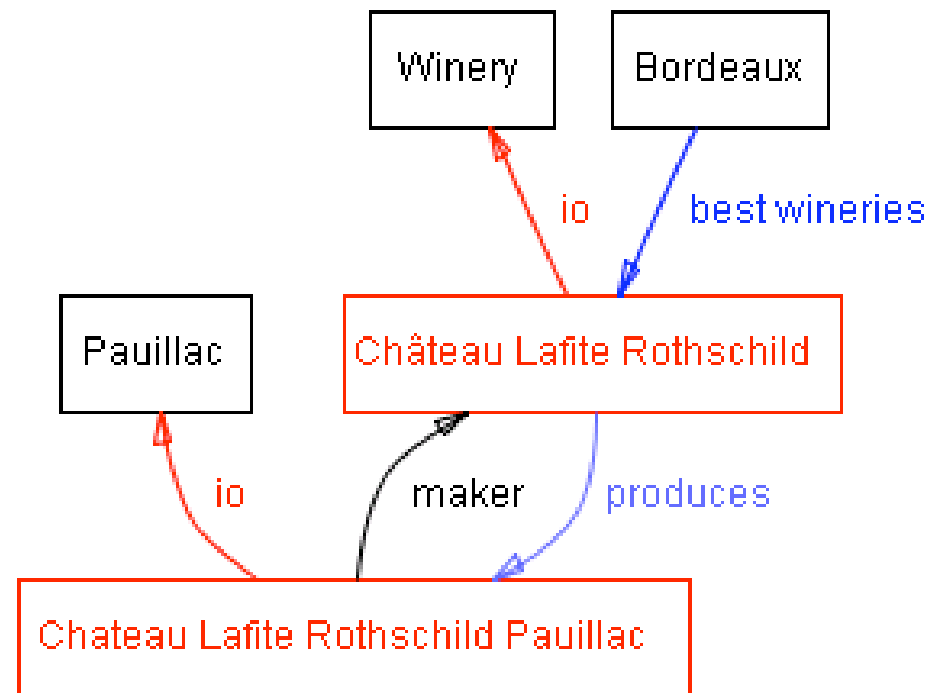
4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.6 Ontology 101

54

Ontology Development 101 *(Noy, McGuinness, 2000)*

- Am Beispiel einer Ontologie für Weinbau und Nahrungsmittel



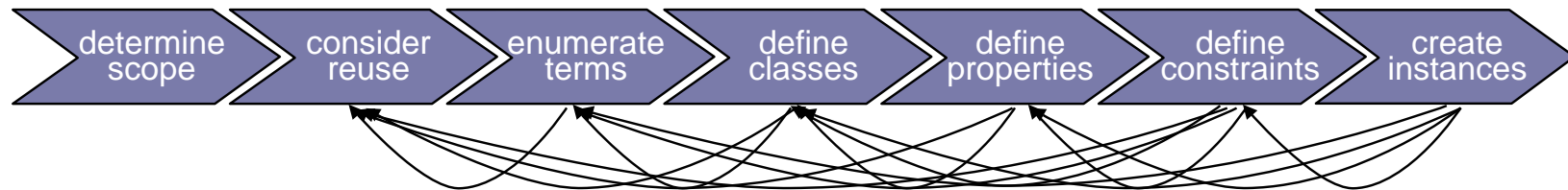
4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.6 Ontology 101

55

Ontology Development 101 *(Noy, McGuinness, 2000)*

- Am Beispiel einer Ontologie für Weinbau und Nahrungsmittel



- in der Praxis iterativer Prozess, der sich beständig wiederholt und die Ontologie verbessert
- es gibt immer unterschiedliche Vorgehensweisen bei der Modellierung einer Ontologie
- in der Praxis entscheidet immer die angestrebte Anwendung über das Vorgehen bei der Modellierung

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.6 Ontology 101

56

Ontology Development 101 *(Noy, McGuinness, 2000)*

- Beurteile Fachgebiet und Fokus



- Welches Fachgebiet soll die Ontologie abdecken?
- Wozu soll die Ontologie genutzt werden?
- Welche Arten von Fragen sollen die in der Ontologie repräsentierten Informationen beantworten können?
- Wer wird die Ontologie nutzen und pflegen?
- Formulierung von Kompetenzfragen (vgl. Grüninger/Fox)

Im Ontologie Lifecycle können sich diese Fragen
auch verändern...

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.6 Ontology 101

57

Ontology Development 101 *(Noy, McGuinness, 2000)*

- Wiederverwendbarkeit berücksichtigen



- Warum soll man Wiederverwendbarkeit berücksichtigen?
 - um Aufwand zu sparen
 - um mit Werkzeugen von anderen Ontologien verwendet werden auch auf die eigene zugreifen zu können
 - um Ontologien wiederzuverwenden, wenn sie durch erfolgreichen Einsatz validiert wurden

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.6 Ontology 101

58

Ontology Development 101 *(Noy, McGuinness, 2000)*

- Terminologie entwickeln



- Von welchen Begriffen soll die Rede sein?
- Welche Eigenschaften haben diese Begriffe?
- Was wollen wir über diese Begriffe aussagen?

Example: Wine Ontology

*wine, grape, winery, location,
wine color, wine body, wine flavor, sugar content
white wine, red wine, Bordeaux wine
food, seafood, fish, meat, vegetables, cheese,...*

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.6 Ontology 101

59

Ontology Development 101 *(Noy, McGuinness, 2000)*

- Klassen und Klassenhierarchien entwickeln



- Klassen sind Konzepte in der betreffenden Domain
 - Klasse der Weine
 - Klasse der Weinbaubetriebe
 - Klasse der Rotweine
- Klassen sind Sammlungen von Objekten mit gleichartigen Eigenschaften
- Wähle top-down / bottom-up / middle-out Ansatz zur Modellierung der Klassenhierarchien

4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.6 Ontology 101

Ontology Development 101 *(Noy, McGuinness, 2000)*

- Definiere Properties



- Properties in einer Klassendefinition beschreiben Attribute von Instanzen
 - jeder Wein hat eine Farbe, Restzuckergehalt, Produzent, etc,...



Property	Domain	Cardinality
hasColor	single WineColor	(cardinality 1)
hasWineDescriptor	multiple WineDescriptor	
madeFromGrape	multiple WineGrape	(minCardinality 1)
hasBody	single WineBody	(cardinality 1)
hasFlavor	single WineFlavor	(cardinality 1)
hasMaker	allValuesFrom Winery	(cardinality 1)
hasSugar	single WineSugar	(cardinality 1)
locatedIn	multiple Region	(someValuesFrom Region)
food:madeFromFruit	multiple food:SweetFruit, food:NonSweetFruit	

4. Ontology Engineering

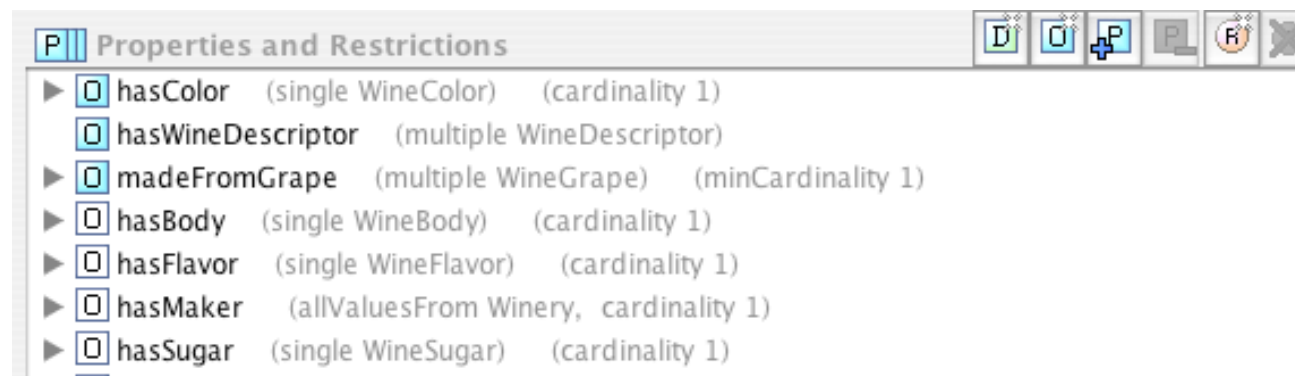
4.2 Ontologieentwurf / 4.2.6 Ontology 101

Ontology Development 101 *(Noy, McGuinness, 2000)*

- Definiere Beschränkungen auf Properties



- Property constraints (Restriktionen) beschreiben bzw. beschränken die Menge der möglichen Property-Werte
 - Der Name eines Weines ist ein String
 - Weinproduzent ist eine Instanz von Winzer



4. Ontology Engineering

4.2 Ontologieentwurf / 4.2.6 Ontology 101

72

Ontology Development 101 *(Noy, McGuinness, 2000)*

- Definition von Klasseninstanzen



- Erzeuge die Instanzen der Klassen
- Jede Klasse wird zum direkten Typen für ihre Instanzen
- Jede Superklasse eines direkten Typs ist Typ der Instanz
- Zuweisung von Property-Werten für Instanzen entsprechend Constrainits
 - „das Glas spanischen Rotweins, das xy gestern abend zum Abendessen getrunken hat“

4. Ontology Engineering

73

4. Ontology Engineering

4.1. Ontologien - eine kurze Wiederholung

4.2. Methoden des Ontologie-Entwurfs

4.2.1. Allgemeines Vorgehen

4.2.2. Methode von Uschold und King

4.2.3. Methode von Grüninger und Fox

4.2.4. METHONTOLOGY

4.2.5. On-To-Knowledge

4.2.6. Ontology 101

4.3. Ontology Learning, Ontology Mapping und Ontology Merging

4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

74

- Ontologie-Entwurf ist sehr aufwändig bzgl. Zeit und Ressourcen
kann das Verfahren (teil-)automatisiert werden?
- Ontologien können (automatisch) „gelernt“ werden
- **Ontology Learning** definiert Menge von Methoden und Techniken
 - zum grundlegenden Aufbau einer neuen Ontologie
 - zur Erweiterung oder Anpassung einer bereits existierenden Ontologie
 - in einer (teil-)automatisierten Weise aus unterschiedlichen Ressourcen
- auch Ontology-Generation, Ontology-Mining, Ontology-Extraction,...
- Automatisierung benötigt Hilfestellung
 - Natural language processing (NP)
 - Machine learning techniques (ML)

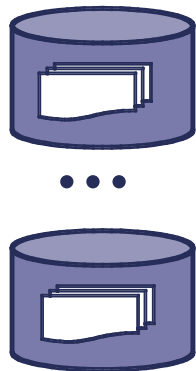
4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

75

Prinzipielles Vorgehen

Dokumentenkorpus



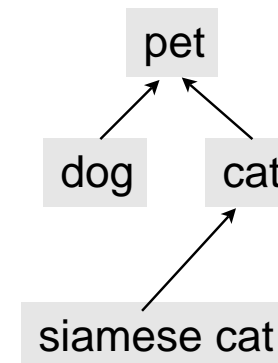
(1) Term-
extraktion

Terme

<dog> *<dogs>*
<cat>
<siamese cat>

(2) Konzep-
tualisierung

Ontologie



(3) Evaluation und
Nachbesserung

4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

76

Ontology Learning Tasks

- welche Aufgaben im Ontologie-Entwurf können (teil-)automatisiert werden?

- Ontology Learning Tasks
 - Ontology creation
 - Ontology schema extraction
 - Extraction of ontology instances
 - Ontology integration and navigation
 - Ontology update
 - Ontology enrichment

4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

77

Ontology Learning Tasks

- **Ontology creation**

- Entwurf von Grund auf (from the scratch) durch einen Experten
- Maschinelles Lernen (machine learning, ML) unterstützt den Experten beim Entwurf durch
 - Vorschlag von geeigneten Relationen zwischen den Konzepten
 - Überprüfung der Integrität/Konsistenz der entworfenen Ontologie

- **Ontology schema extraction**

- Extraktion von Schemata aus Web-Dokumenten / Texten
- ML benutzt Eingabedaten und Meta-Ontologien, um fertige Domain-Ontologien (ggf. mit Hilfe des Experten) zu erzeugen

4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

78

Ontology Learning Tasks

- **Extraction of ontology instances**
 - Extraktion von Ontologie Instanzen aus Web-Dokumenten / Texten, um vorgegebene Ontologie-Schemata mit Instanzen zu füllen
 - nutzt Techniken aus den Bereichen Information Retrieval und Data Mining
- **Ontology integration and navigation**
 - Umbau (Rekonstruktion) von existierenden Wissensbasen und Navigation in vorhandenen Wissensbasen,
 - z.B. Übersetzung einer Wissensbasis aus FOL nach OWL

4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

79

Ontology Learning Tasks

- **Ontology update**

- Erweiterung, Umbau und Veränderung von bereits bestehenden Ontologien, z.B. zur Anpassung an veränderte Domain
- betrifft Teilbereiche von Ontologien, die speziell so angelegt wurden, dass diese verändert werden können

- **Ontology enrichment**

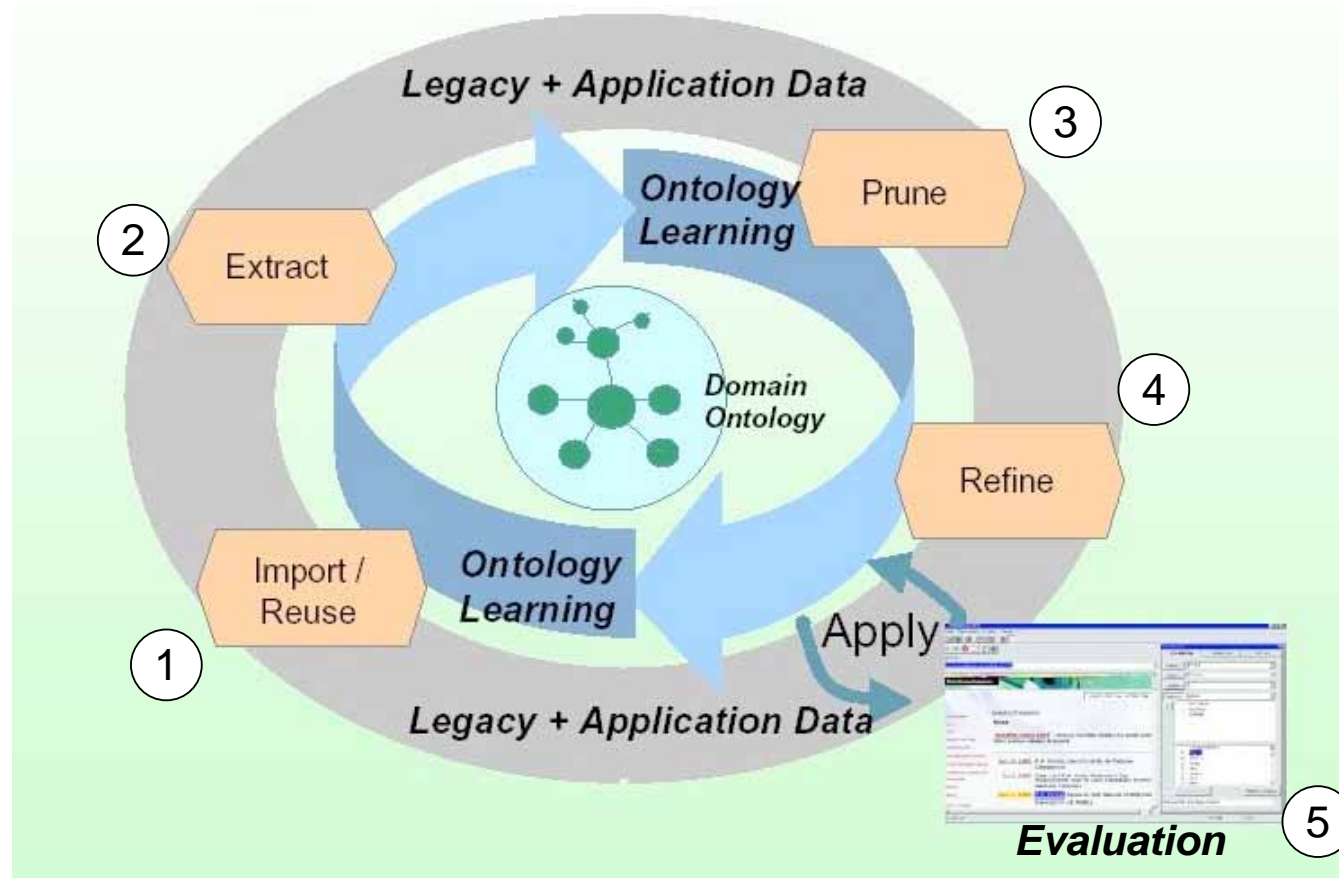
- (auch Ontology tuning) betrifft automatische Aktualisierung kleinerer Teilbereiche bereits existierender Ontologien
- verändert nicht wichtige (bedeutende) Konzepte und Relationen, sondern präzisiert diese
- anders als bei Ontology update sind Teilbereiche von Ontologien betroffen, die an sich nicht verändert werden können sollten

4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

30

Ontology Learning Process (Maedche, Staab, 2001)



4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

31

Ontology Mapping

- um sich verstehen zu können, müssen zwei Parteien dieselbe formale Spezifikation einer gemeinsamen Konzeptualisierung verwenden
-sich aber auf dieselbe Ontologie zu einigen ist nicht immer einfach (unterschiedlicher Anwendungszweck, unterschiedliche Sichtweisen, unterschiedliche Meinungen...)
- Parteien mit unterschiedlichen Ontologien (zur einer identischen Domain) können sich nicht verstehen



- Ontologien müssen aufeinander abgebildet werden
(= Ontology Mapping / Ontology Alignment / Ontology Mediation)

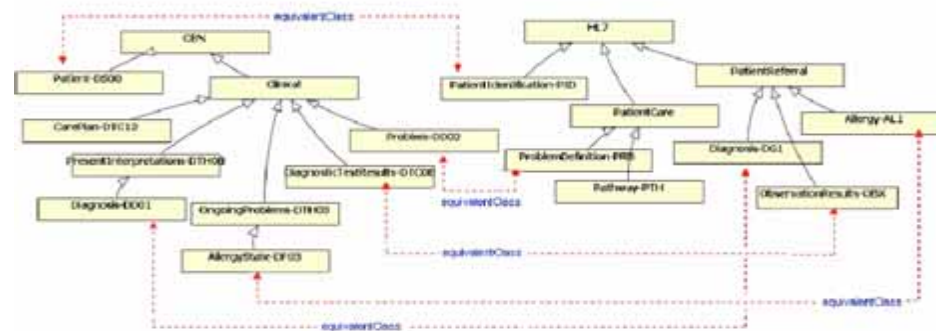
4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

32

Ontology Mapping

- ist ein Prozess, in dem zwei Ontologien semantisch auf der konzeptionellen Ebene miteinander in Beziehung gesetzt werden.
- Dabei werden Instanzen der Start-Ontologie O_S entsprechend der ermittelten semantischen Beziehungen in Instanzen der Ziel-Ontologie O_T mit Hilfe einer Abbildung $M: O_S \rightarrow O_T$ transformiert.
- Die Abbildung M kann dabei
 - injektiv (nicht umkehrbar) oder auch
 - bijektiv sein

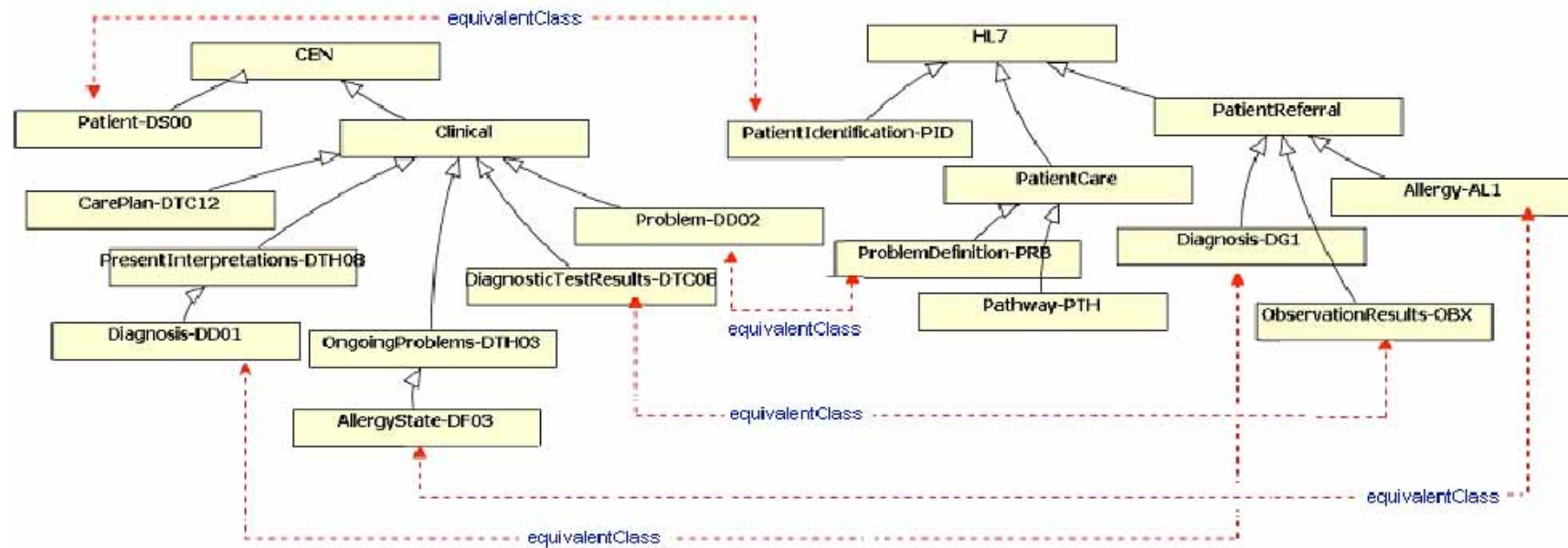


4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

33

Ontology Mapping



4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

34

Ontology Mapping

- Ontology Mapping ist kein “neues” Problem...
- dieselben Probleme treten bei der Datenintegration z.B. in föderierten Datenbanken auf
- Föderierte Datenbanken verwalten lokale Schemata für jede einzelne beteiligte Datenbank
- Datenintegration (Schema Matching) erfolgt entweder über
 - bilaterale Abbildungen oder über
 - globale Schemata zu denen jeweils eine Abbildung zu jedem lokalen Schema existiert (Abbildung erfolgt über view)



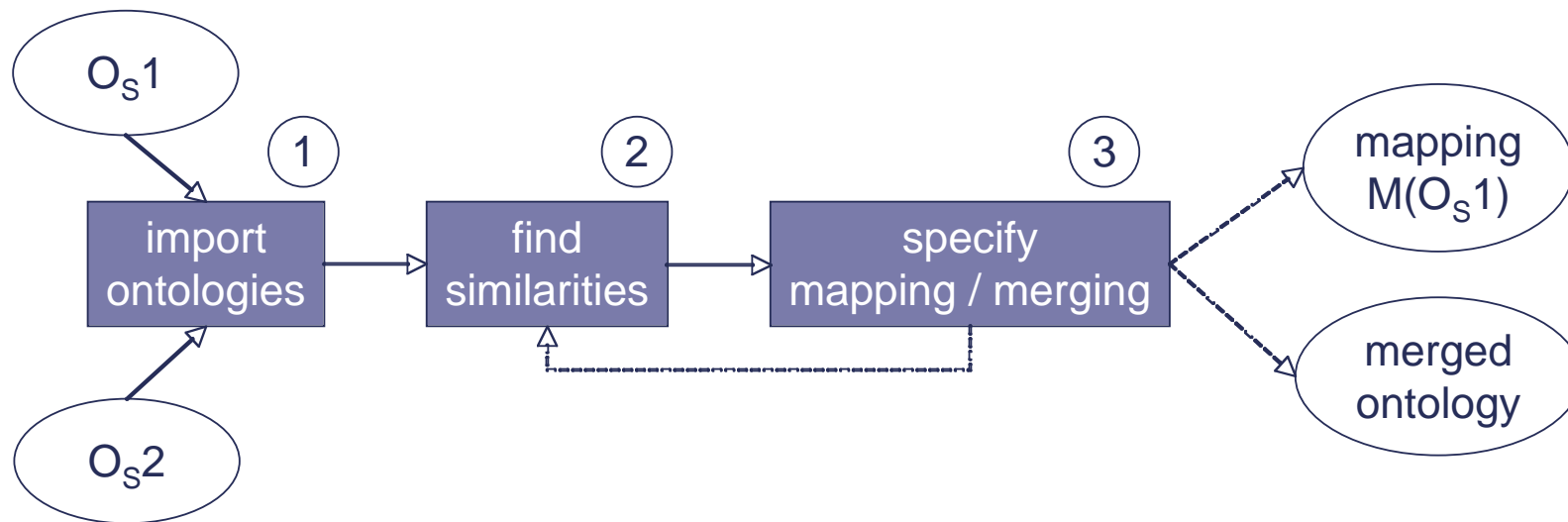
4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

35

Ontology Mapping Process

- prinzipieller Ablauf



4. Ontology Engineering

4.3 Ontology Learning, Mapping, Merging

36

Ontology Merging

- ist ein Prozess, in dem aus zwei oder mehr Start-Ontologien eine **neue** Ontologie erzeugt werden soll.
- die neue Ontologie **vereinheitlicht** und **ersetzt** die originalen Start-Ontologien.

- **Union Approach**

Die neue Ontologie ist die **Vereinigung** aller Entitäten der Start-Ontologien, wobei Konflikte aus unterschiedlichen Repräsentation identischer Konzepte der Start-Ontologien aufgelöst wurden.

- **Intersection Approach** (extensional)

Die neue Ontologie besteht nur aus den Teilen der Start-Ontologien, die überlappen.

4. Ontology Engineering

37

4. Ontology Engineering

4.1. Ontologien - eine kurze Wiederholung

4.2. Methoden des Ontologie-Entwurfs

4.2.1. Allgemeines Vorgehen

4.2.2. Methode von Uschold und King

4.2.3. Methode von Grüninger und Fox

4.2.4. METHONTOLOGY

4.2.5. On-To-Knowledge

4.2.6. Ontology 101

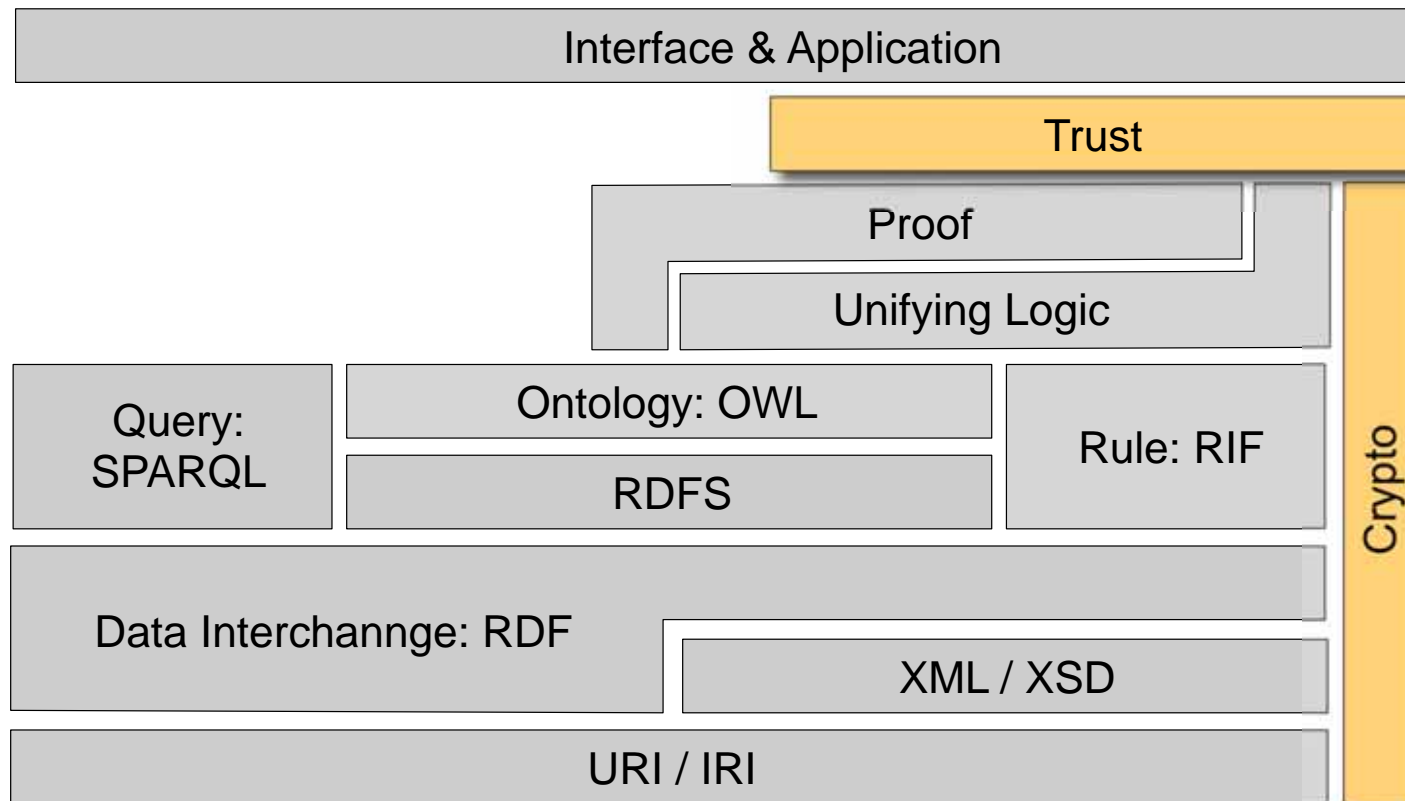
4.3. Ontology Learning, Ontology Mapping und Ontology Merging

3. Wissensrepräsentationen

3.8 Regeln mit SWRL / RIF

38

Semantic Web Architektur

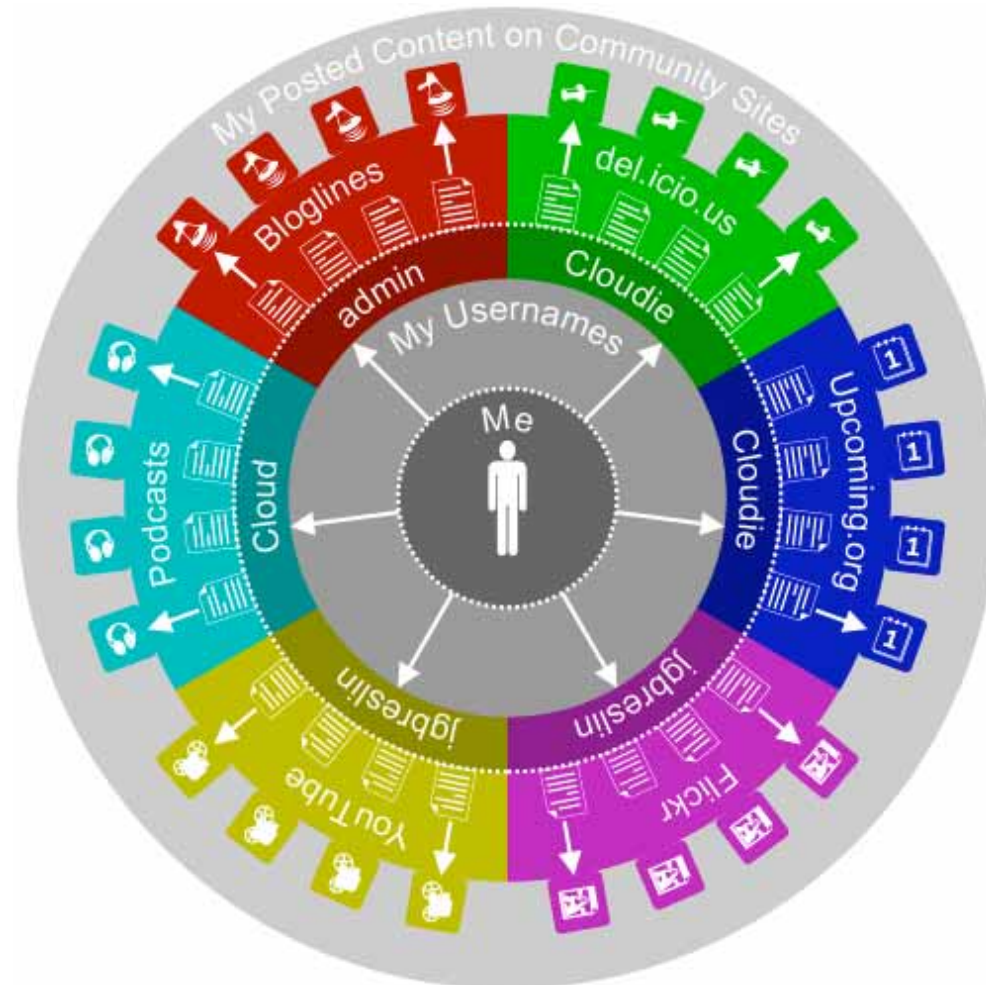


3. Wissensrepräsentationen

3.8 Regeln mit SWRL und RIF

39

Nächste Vorlesung: Web of Trust



Semantic Web - Vorlesungsinhalt

90

1. Einführung
2. Die Sprachen des Semantic Web
3. Wissensrepräsentation
4. Ontology Engineering
5. Web of Trust

3. Wissensrepräsentationen

3.8 Regeln mit SWRL / RIF

91

Literatur



- » A. Gomez-Perez et al.
Ontological Engineering,
Springer, 2004.

3. Wissensrepräsentationen

3.8 Regeln mit SWRL / RIF

92

Literatur



- Blog
<http://sw0809.blogspot.com/>
- Materialien-Webseite
http://www.hpi.uni-potsdam.de/meinel/teaching/semantic_web_ws08090.html



- bibsonomy - Bookmarks
<http://www.bibsonomy.org/user/lysander07/sw0809-12>