

Modeling in Business in Information Systems **Unternehmensmodellierung und -architekturen**

Unit 7: Ontology Engineering

Prof. Dr. Martin Hepp

Professur für Allgemeine BWL, insbesondere E-Business
Chair of General Management and E-Business

<http://www.heppnetz.de/teaching/dpm/>

Course Overview

- 1 Introduction
- 2 Modeling for Beginners: Conceptual Structures and Processes
- 3 Modeling Conceptual Structures
- 4 Process Modeling: Behavioral Aspects of Information Systems
- 5 ARIS – Architecture of Integrated Information Systems
- 6 Challenges in Information Systems Modeling
- 7 Reference Models and Process Libraries
- 8 Collaboration in Modeling**
- 9 Ontologies and Ontology Engineering**
- 10 Ontological Analysis of Models**
- 11 Tools
- 12 Exercises
- 13 Additional Topics

The need for a shared understanding

- People, organizations, and software systems must communicate between and among themselves
- However, due to different needs and background contexts, there can be widely varying viewpoints and assumptions regarding what is essentially the same subject matter; each may have differing, overlapping and or mis-matched concepts
- The consequent lack of a shared understanding leads to poor communication within and between people, organizations, and systems.
- It limits interoperability and re-use of data and program code.

from: Uschold/Gruninger: Ontologies: Principles, Methods and Applications, 1996

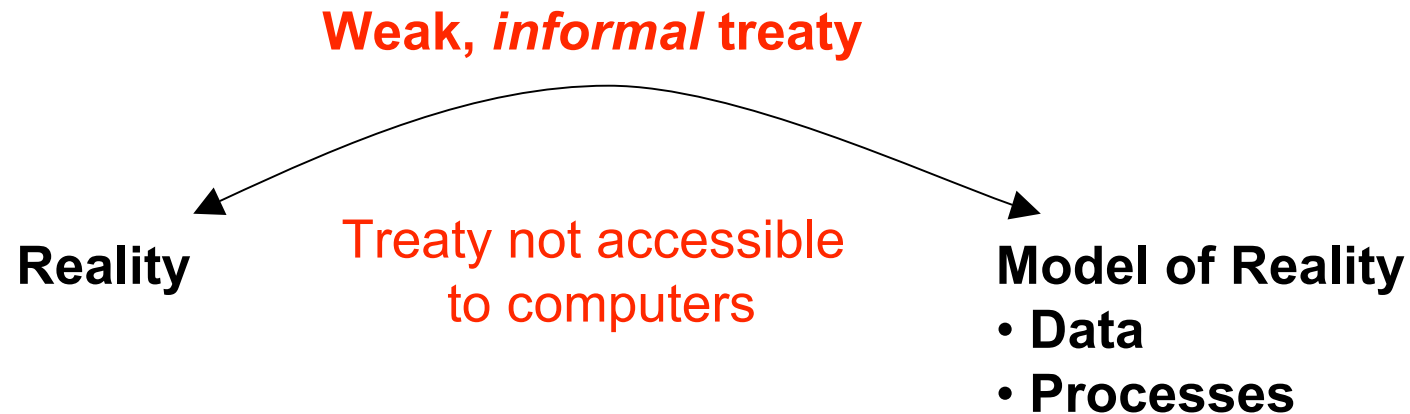
Natural language is very ambiguous...

- „I saw him standing near the **bank**.“
- „The **jaguar** was much faster than we we all had expected.“
- „The unknown species was very **tall**“.

Only the amount of contextual information (e.g. was had been said before, the speaker's identity and background,...) helps us restrict all possible interpretations of a statement to the most likely intended one.

The Root of the Problem.....

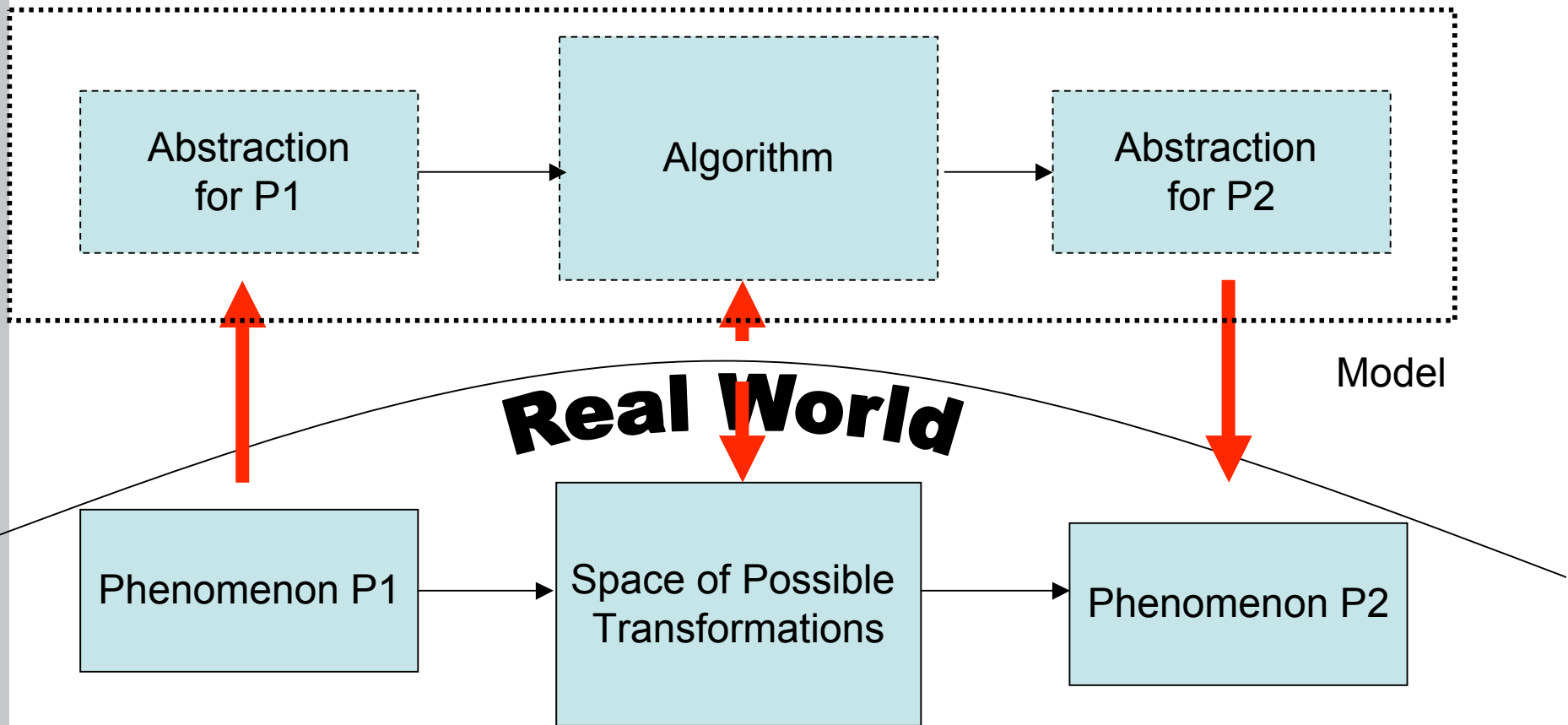
The Root of the Problem: *Weak Ties* *between Reality and Abstractions of Reality*



Symptoms:

- We can hardly validate whether a given ER model is correct
- We face difficulties making sure that the customization of SAP myERP matches the business needs of a given enterprise

Computer Systems, Models, and the Real World



What is an ontology?

- Ontology: A shared understanding of some domain of interest.
- An ontology necessarily entails or embodies some sort of world view with respect to a given domain.
- Often in the form of a set of
 - concepts (entities, attributes, processes),
 - their definitions, and
 - their inter-relationships.
- This is called a „conceptualization“.

DPM - Unit 8

8

Six Effects of Ontologies

Using Philosophical Notions
as Guidance for Identifying
Stable, Reusable Conceptual Elements

Excluding Unwanted Interpretations
by Means of Formal Semantics

Unique Identifiers for
Conceptual Elements

Inferring Implicit
Facts Automatically

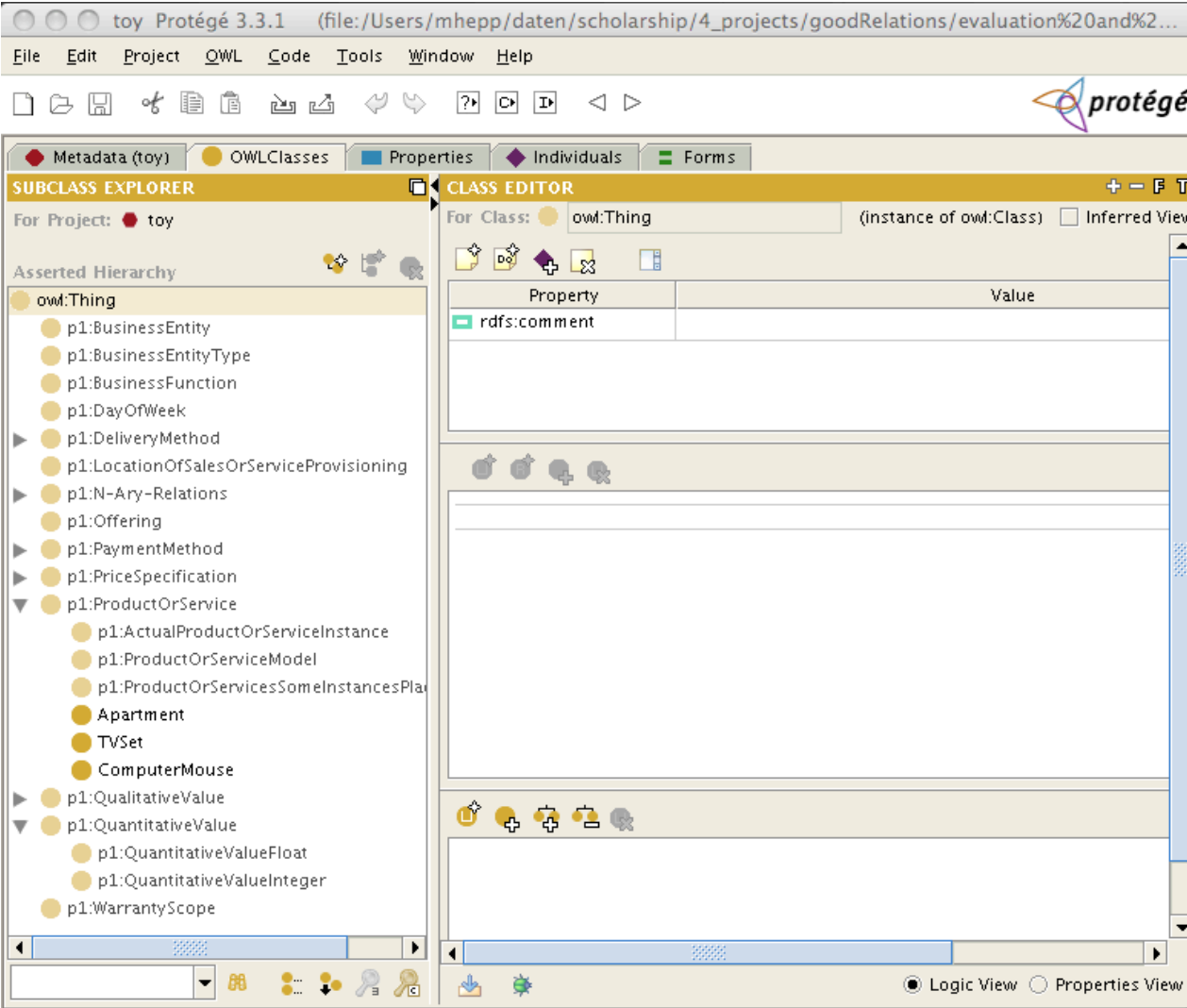
Excluding Unwanted Interpretations
by Means of Informal Semantics

Spotting Logical
Inconsistencies

From: Martin Hepp: Ontologies: State of the Art, Business Potential, and Grand Challenges,
in: Martin Hepp, Pieter De Leenheer, Aldo de Moor, York Sure. (Eds.):
Ontology Management: Semantic Web, Semantic Web Services, and Business Applications,
ISBN 978-0-387-69899-1, Springer, 2007, pp. 3-22.

- Ontology in Computer Science: The **approximation of a specification** of a **shared** understanding of some domain of interest.
- Often in the form of a set of
 - conceptual entities (concepts/classes/categories, attributes, relations, behavioral aspects),
 - their definitions in the form of
 - modalities for human usage (text, pictures,...)
 - logical axioms that help reduce unwanted interpretations and infer implicit facts
 - their inter-relationships.
- An ontology necessarily entails or embodies some sort of world view with respect to a given domain.

Example



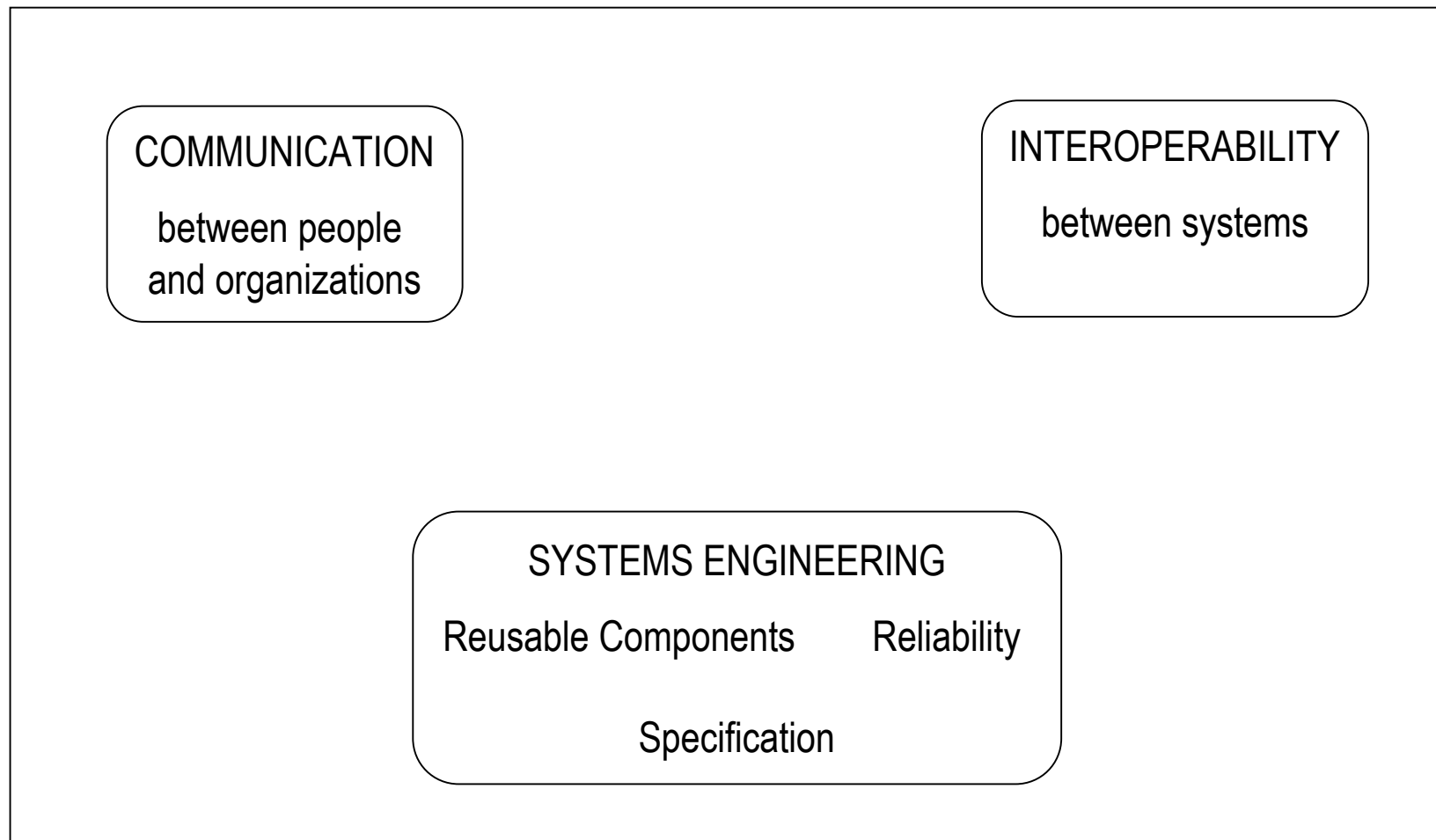
The screenshot shows the Protégé 3.3.1 interface. The title bar indicates the project is 'toy' and the file path is '(file:/Users/mhepp/daten/scholarship/4_projects/goodRelations/evaluation%20and%20...'. The menu bar includes File, Edit, Project, OWL, Code, Tools, Window, and Help. The toolbar contains various icons for file operations and navigation. The main interface is divided into several panes:

- SUBCLASS EXPLORER:** Shows the asserted hierarchy for the project 'toy'. The root class is 'owl:Thing', which has several subclasses listed, including 'p1:BusinessEntity', 'p1:BusinessEntityType', 'p1:BusinessFunction', 'p1:DayOfWeek', 'p1:DeliveryMethod', 'p1:LocationOfSalesOrServiceProvisioning', 'p1:N-Ary-Relations', 'p1:Offering', 'p1:PaymentMethod', 'p1:PriceSpecification', 'p1:ProductOrService', 'p1:ActualProductOrServiceInstance', 'p1:ProductOrServiceModel', 'p1:ProductOrServicesSomeInstancesPlac', 'Apartment', 'TVSet', 'ComputerMouse', 'p1:QualitativeValue', 'p1:QuantitativeValue', 'p1:QuantitativeValueFloat', 'p1:QuantitativeValueInteger', and 'p1:WarrantyScope'.
- CLASS EDITOR:** Shows the editor for the class 'owl:Thing'. It includes a table for properties and values:

Property	Value
rdfs:comment	

The interface also includes a status bar at the bottom with icons for Logic View and Properties View.

Main categories of uses for ontologies



Ontological Commitment

- The actual essence of an ontology to which one commits when using this ontology.
- Traditionally, only the formal ontological commitment
 - The set of logical statements made by the ontology that constrains the interpretation of conceptual elements
- Practically, also the informal ontological commitment

Examples

- Ontological part of ER models: e.g. cardinality constraints
- Modeling elements in formal ontology languages
 - symmetry
 - transitivity
 - reflexivity
 - disjunction of classesetc.

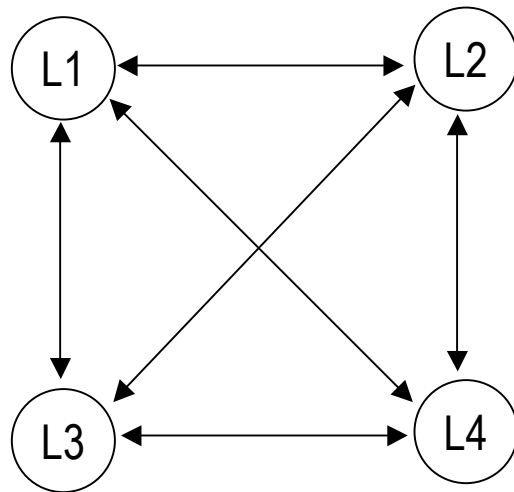
Varying degree of formality

- **highly informal:** expressed loosely in natural language
- **semi-informal:** expressed in a restricted and structured form of natural language
- **semi-formal:** expressed in an artificial, formally defined language
- **rigorously formal:** meticulously defined terms with formal semantics, theorems and proofs of such properties as soundness and completeness

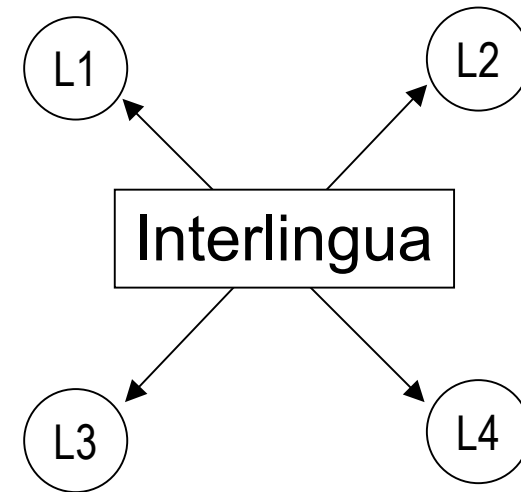
Higher degree of formality: Costs and benefits

- The more formal an ontology is, the more does it exclude unwanted interpretations.
- Also, it increases the amount of inferences that can be drawn.
- However, it is more costly in terms of labor and engineering delay to create a more formal conceptualization.
- Also, achieving consensus is more difficult and time-consuming for a higher degree of formality.
- A higher degree of formality also imposes high entry barriers on the ones who are to model the respective domains and thus excludes potential contributors.

Ontologies as a hub for meaning



$2*n$ more translators
for each additional node



2 more translators
for each additional node

from: Uschold/Gruninger: Ontologies: Principles, Methods and Applications, 1996

The contractual dimension of ontologies in business settings

- Ontologies allow to make explicit what is agreed **and what is not agreed**.
- This avoids misunderstandings about how narrow or broad a given term is meant in a business relationship.

What is Ontology Engineering?

- The art and science of building, maintaining, and evaluating consensual representations of a domain of interest.

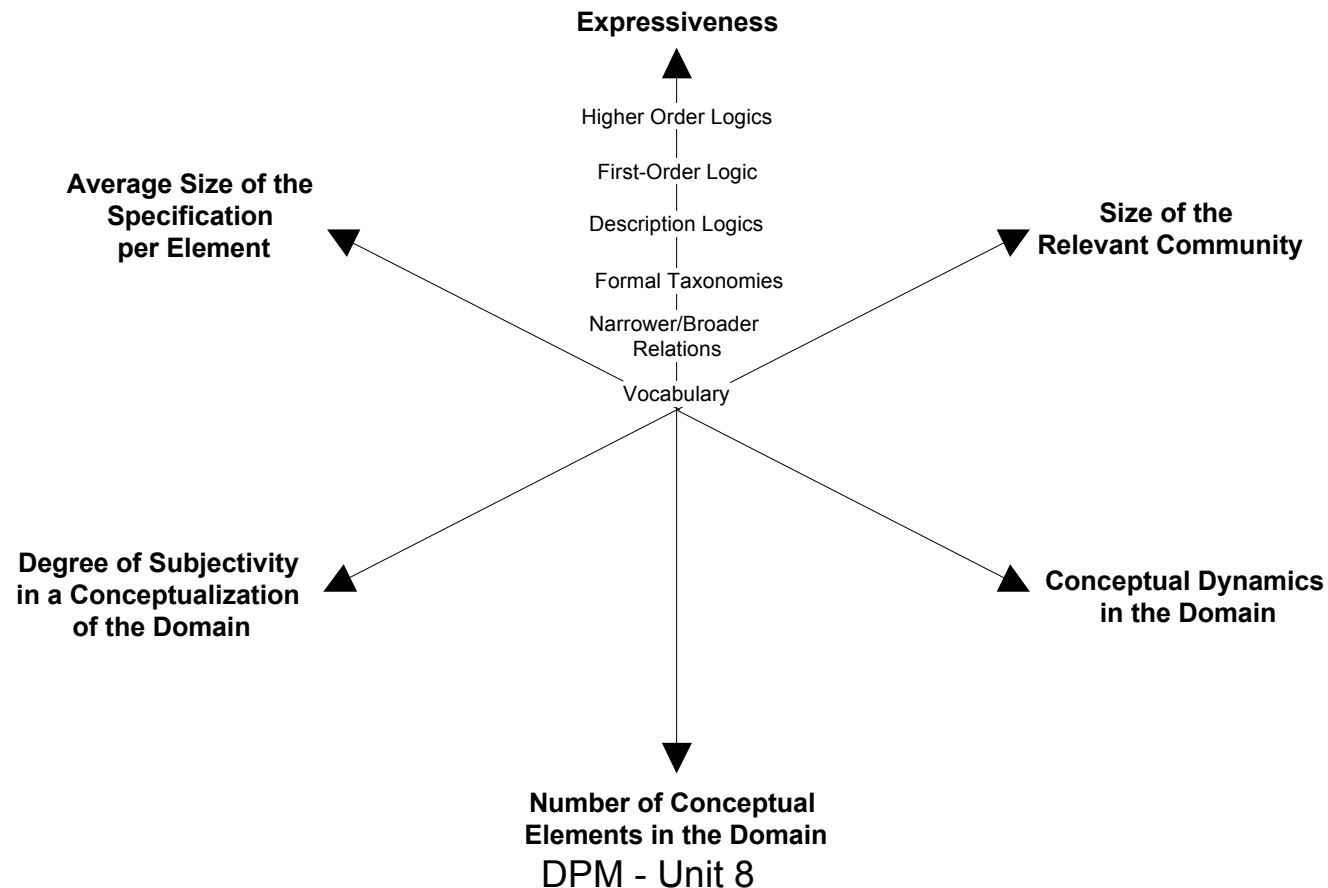
Skeletal methodology for building ontologies

- Identify Purpose and Scope
- Building the Ontology
 - ontology capture
 - ontology coding
 - integrating existing ontologies
- Evaluation
- Documentation

Initial guidelines for designing ontologies

- **Clarity**: Communicate effectively the intended meaning to humans who use the ontology
 - Natural language definitions
 - Give examples
 - Motivate distinctions
 - Formal axioms, if possible
- **Coherence**: The ontology should be internally consistent
 - Axioms
 - Definitions
- **Extensibility**: Crafted so that one can extend and specialize the ontology monotonically
- **Minimal ontological commitment**
 - Make as few claims as possible about the world being modeled
 - Too many (usage restricted) vs. too few (compatible with unintended models)

Six Characteristic Variables of Ontology Projects



Step 1: Purpose and Scope

- Be clear about why the ontology is being built and what its intended usages are
- E.g.
 - interoperability between systems
 - systems engineering
 - communication between people and organizations
- Recommended: Written document

Motivating Scenarios

- Define one or more motivating scenarios
- Should justify the need for a new ontology
- Should clarify the context of usage of the new ontology

from: Uschold/Gruninger: Ontologies: Principles, Methods and Applications, 1996

Competency Questions

- A set of queries which place demands on the underlying ontology.
- Ontology must be able to represent the questions using its terminology and the answers based on the axioms
- Ideally, in a staged manner, where consequent questions require the input from the preceding ones.
- A rationale for each competency question should be given.

Step 2: Building the ontology

- Step 2a: Ontology capture
 - Identify key concepts and relationships
 - produce clear text definitions for these
 - identification of terms that refer to these
 - Consensus
- Step 2b: Ontology coding
 - Explicit representation of the conceptualization in some formal language

Ontology Capture: Scoping

- Brainstorming
 - Produce all potentially relevant terms and phrases
 - People involved must have substantial domain expertise
- Grouping: Structure terms loosely into work areas
 - Provisionally categorize them for inclusion or exclusion (-> purpose and scope)
 - Keep notes of these decisions
 - Group similar terms and potential synonyms together

Ontology Capture: Produce Definitions

- Determine suitable **meta-ontology**
 - especially: use words and modeling primitives in a consistent manner (e.g. role, entity, relationship, instance,...)
- **Work areas:** Start with those the have the most semantic overlap with others
 - must be right in the first place, otherwise lot of redundant re-working
- **Terms:** Produce definitions in a middle-out fashion rather than top-down or bottom up.
 - see next slide

Produce Definitions: The Middle-out Approach

- Define the most basic (i.e. important) terms first in each work area before moving to more abstract or more specific terms
- Example:
 - Dog – fundamental
 - Spaniel – specific
 - Mammal – abstract
- Guideline: Basic is what is the same for most people

Advantages of the Middle-out Approach

- A bottom-up approach results in a high degree of detail
 - increases overall effort
 - makes it difficult to spot commonality between related concepts
 - increases risk of inconsistencies and re-work
- Top-down allow better control of degree of detail
 - risk of arbitrary high-level categories
 - risk of limited stability
- Middle-out strikes a balance in terms of the level of detail
- The higher level categories naturally arise and are thus more likely to be stable.

Reaching Agreement: Some suggestions

- Produce a natural language text definitions
- Ensure consistency with terms already in use
 - use existing thesauri and dictionaries
 - avoid introducing new terms in the definitions
- Indicate relationships with other commonly used terms
 - synonyms, variants, broader / narrower terms, such referring to different dimensions
- Give examples

Step 3: Integrating Existing Ontologies

- Check overlap with existing ontologies
- Establish formal links
 - Produce mappings to existing concept definitions
 - Import and extend existing ontologies
- Avoid re-inventing the wheel!

Step 4: Ontology Evaluation

- Validate whether the ontology produced is consistent and meets the requirements specification
 - Intended scope and usages
 - Competency questions

from: Uschold/Gruninger: Ontologies: Principles, Methods and Applications, 1996

Step 5: Documentation

- Give natural text definitions for each element of the ontology,
- Ideal: retrievable resources at the very same URI, if URIs are used as ontology element identifiers
- Use multi-media elements, if feasible

Most Popular Modeling Primitives

- Class/Concept
- Is-A Relationships
 - InstanceOf / `rdf:type`
 - subclassOf / `rdfs:subclassOf`
- Generic Relationship/Property
 - Object Property
 - Datatype Property
- Individual/Instance

Modeling Examples

- Peter loves Frieda, a brown-eyed Spaniel.
- Peter loves Frieda, his friend Paul's Spaniel.
- Peter wrote down a poem by Shakespeare.

Exercises

- All: Build a small ontology for the domain of a university library.
- Teams:
 - EM2008 Ontology
 - Recipe Ontology
 - Application Software Ontology
 - Computer Equipment Ontology
 - Alpine Leisure Activities Ontology
 - Documents and Media Ontology

References

- Uschold & Gruninger: *Ontologies: Principles, Methods and Applications*, 1996

Thank you.

Prof. Dr. Martin Hepp

*Professur für Allgemeine BWL, insbesondere E-Business
Chair of General Management and E-Business*

<http://www.heppnetz.de/teaching/dpm/>