salzburg research

How to support domain experts in creating ontologies?

Trento, 13th October 2006

Andreas Gruber

DynamOnt

Salzburg Research

- Organisation
  - Since 2000 owned by the County of Salzburg (100%)
  - Turnover approx: 4,5 Mio Euro
  - 30% basic subsidy from owner, 70% cooperative research
  - 65 staff of which are 43 researchers

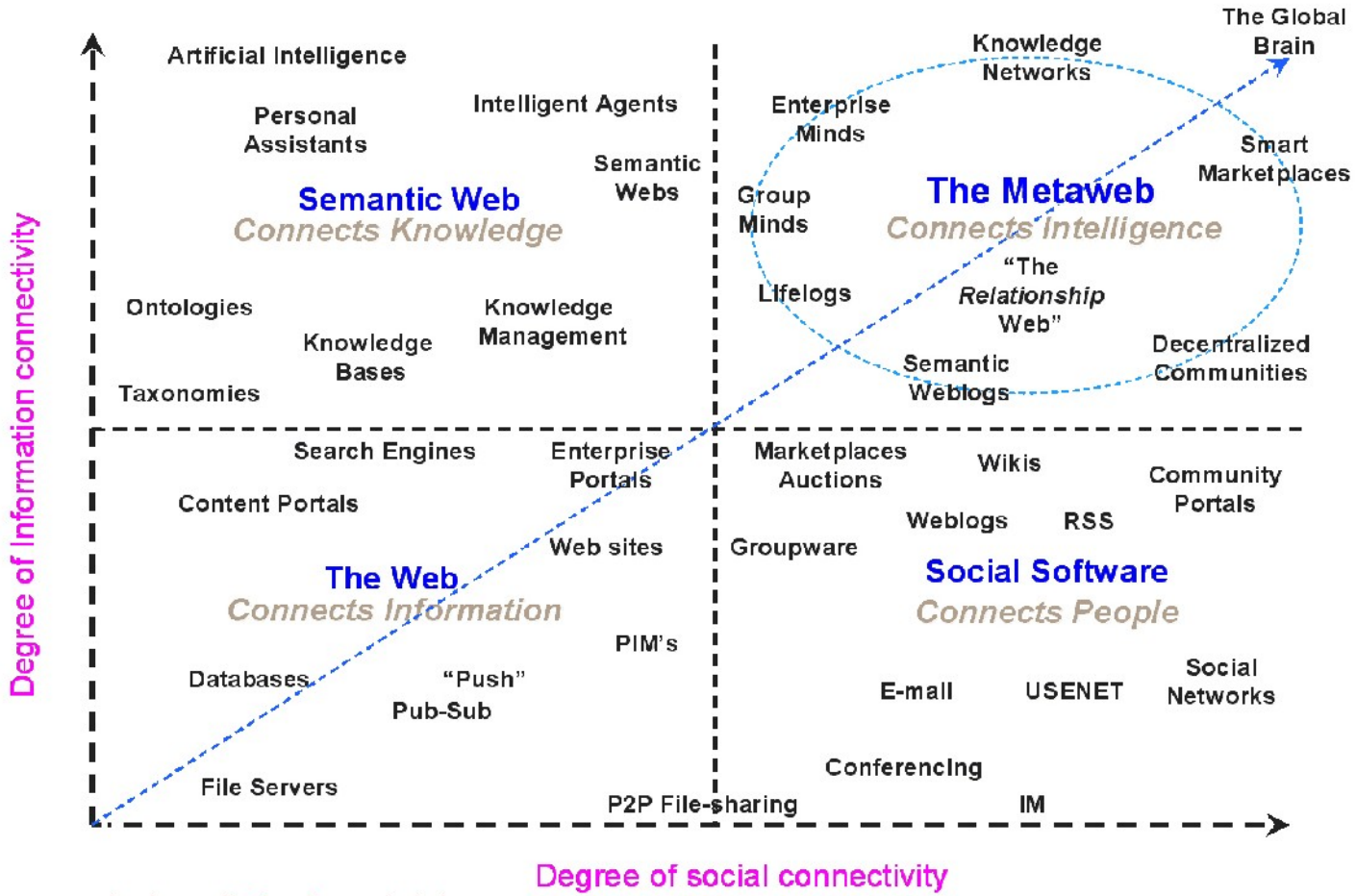
- Applied Research, Coordination and Networking, Know-how transfer within four research lines
  - ANC – Advanced Network Center (QoS, embedded systems)
  - ISR – Information society research (Studies, eCulture and eLearning)
  - MOWI – Mobility and Web-based Information Systems (location based systems, eToursim as application)
  - **KIS – Knowledge-based Information Systems**

KIS – Research Group in Media and Knowledge Technologies

- approx. 10 people, aiming at approx. 20 by 2008
  - Knowledge and content management architectures
  - Collaboration and social software for knowledge workers
  - Methodology for knowledge-based systems
- Semantic Web, Hypermedia and K-Workers' Tools
  - IkeWiki, a semantic wiki
  - RDF Gravity, an RDF/OWL visualisation tool
  - Currently in three EU Projects
    - LIVE Integrated project (Enhancing live broadcasting streams with knowledge based content)
    - QVIZ (Semantics and collaboration in archives)
    - ImportNet (Semantics based collaboration in mechatronics engineering
  - Supporting an Austrian National Competence Centre on Digital Media

A vision

Convergence of SemWeb and Social Software



My background and research activities

- Austrian Literature / History&Philosopy (Bacc.)
- Postgraduate MBA Study of Applied Knowledge Management (2004-2006)

- Project support / Requirements Engineering
  - **CULTOS** (2001-2003) Multimedia knowledge management tools for culture and arts.
  - **METOKIS** (2004-2005) Methodology and Tools Infrastructure for the Creation of Knowledge Units.

- Research and coordination
  - **DynamOnt** (2006-2007) Methodik zur Erstellung dynamischer Ontologien.

Outline of this Seminar

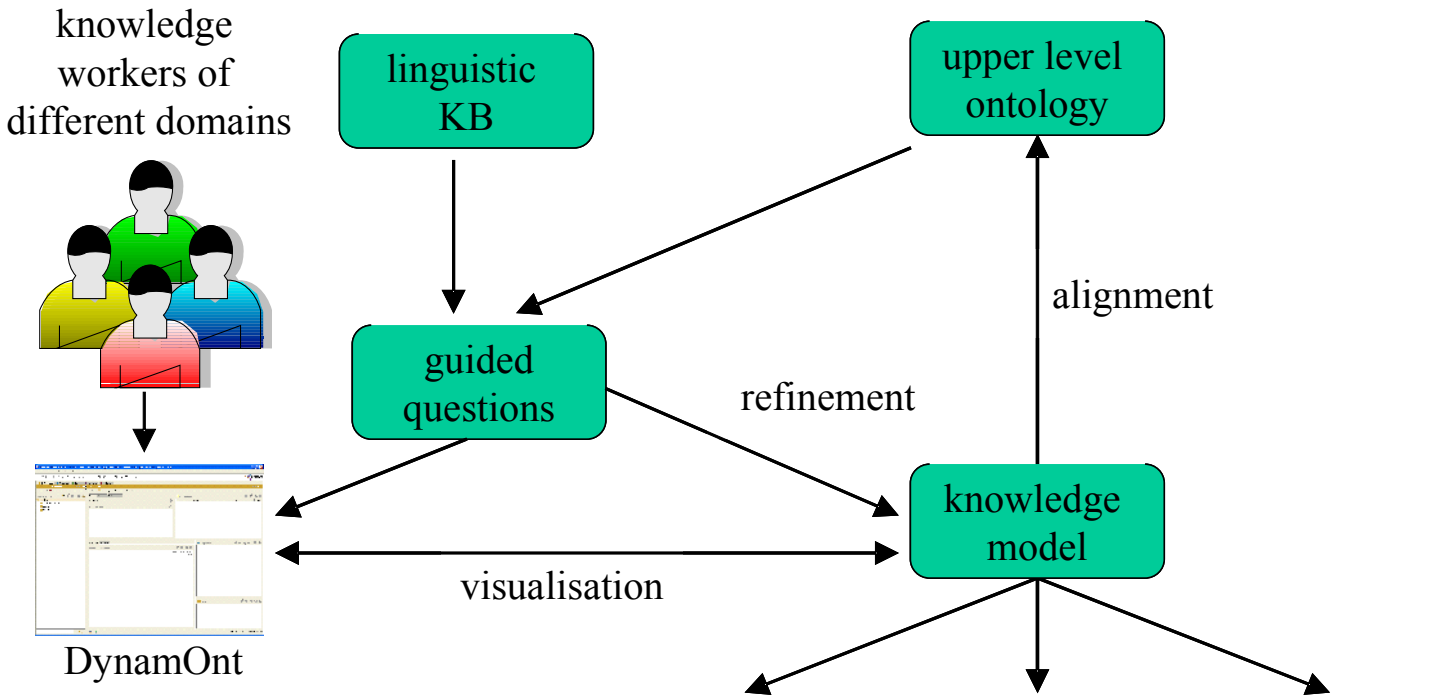
- Objectives of DynamOnt project
- Towards a methodology for ontology creation from scratch
- Purpose: 3D Matrix for complexity estimation
  - Scope
  - Expressiveness
  - Acceptance
- Creating informal knowledge models
  - Upper level ontologies as reference for ontology creation
  - Usage scenarios as basis
  - Bottom-up approach
  - Top-down approach

Objectives of the DynamOnt project

- Methodology and software framework for dynamically creating ontologies
  - For non-IT experts in cooperative environments
  - Alignment of ontology building tools and terminology tools
  - Austrian/German language based methodology and tools

- taking into account ...
  - different formality levels of knowledge model during project livecycle
  - a range from individual, group up to sector/world acceptance
  - use of top level ontologies as reference

How we thought we would approach the topic …



Wording of DynamOnt project …

- "dynamic" ontology building means three things:
  - The ontologies can be extended and refined over time
  - Ontologies can evolve to become more formal and axiomatised
    - from glossaries, thesauri and taxonomies to ontologies, by adding formal semantics
  - Can be personalised by individuals or groups
    - from individual models to group models, to serve communities and to achieve sector acceptance

- "application profiles/models" are ontological models which
  - cover a domain of interest, but are not domain ontologies
  - act as a basis for knowledge-driven systems (various degrees of integration into the application system itself)

Main purposes of ontologies

- Uschold 1996
  - Communication between people – shared conceptualisation
  - Interoperability among systems
  - System engineering benefits
    - Re-usability of entities, attributes, processes and interrelationships
    - Knowledge Acquisition
    - Reliability or Consistency
    - Specification or requirements gathering
- Harris 2005
  - Reference Data – specify reference data for annotation
  - Data Structure – store individuals within ontology
  - Assertion and constraints – gain information about individuals

Kinds of ontologies and representations

- Kinds of ontologies
  - Foundational Ontologies (DOLCE, SUMO, OpenCyC, ..)
  - Domain Ontologies (see examples@http://www.schemaweb.info/)
  - Application Profiles/Models (Dublin Core, VCARD, LOM, FOAF...)
  - MetaOntologies (SKOS …)
  - Folksonomies (eg. del.ico.us)

- Knowledge representation languages
  - XML-Schema
  - RDF, TOPIC MAPS
  - Relational Algebra
  - RDFS, OWL, KIF
  - Rule-Languages

Successful "ontologies"

- Term Lists (Country Codes, ...)
- Thesaurus (Word-Net, DMOZ …)
- Attribute Sets (Dublin Core, LOM …)
- Data Schemas (News ML, MPEG 7...)

- Standards (Country Codes, Dublin Core, LOM, VCARD, CC/PP …)
- Application/Service Standards (iCal, FOAF …)

Towards a methodology and workbench

- Déjà vu in ontology engineering
  - Software engineering disciplines are useful to ontological engineering (Architectures, Requirements Analysis, Object oriented analysis and design, design patterns)
- Available Bits
  - Unified Methodology by Uschold 1996
  - Ontology life-cycle (e.g. METHONTOLOGY, ...)
  - Collaboration support (e.g. DILLIGENT)
- Still missing/looking forward for ...
  - Focus on building application profiles rather than domain ontos
  - Methodology according to the 3D-Matrix
  - Collaborative Tools for ontology creation
  - Support for acceptance and expressiveness stages

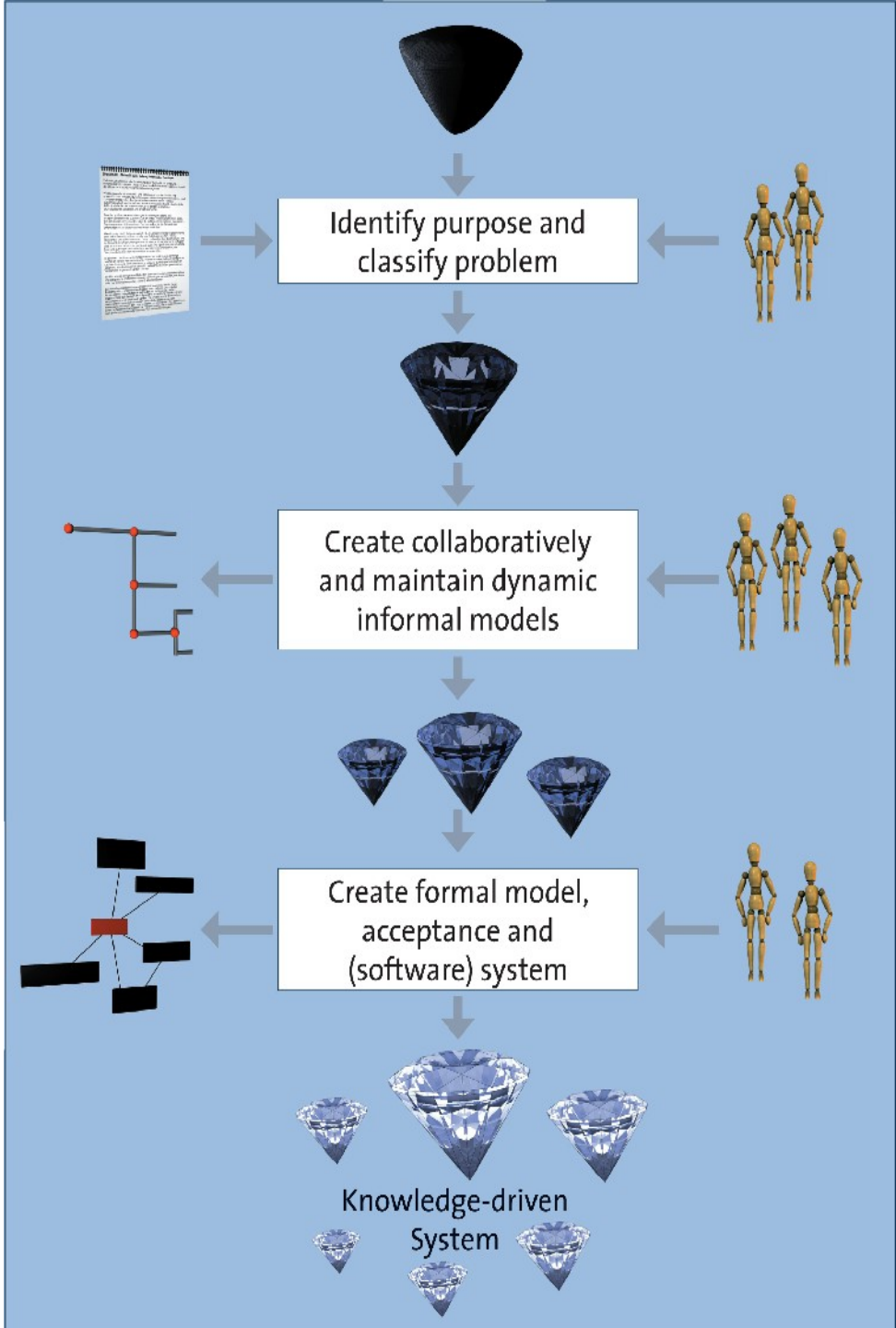
Prototypical approaches in ontology engineering

- Prototypical approaches [Uschold96] in ontology engineering
  - Approach 1: Start Ontology Editor and define terms and axioms
  - Approach 2: Scope the ontology and begin formal encoding
  - Approach 3: Produce intermediate documents/results (informal ontology)
  - Approach 4: Proceed by converting informal competency questions to formal ones and then specify axioms and definitions of ontology
- We follow the Uschold's' approaches 3 and 4
  - Domain experts create and maintain knowledge model (“informal model”)
  - Knowledge- and Software Engineers support in building the application profile (“formal model”) with the help of patterns from top level ontologies

Simple overview of process model version 0.5

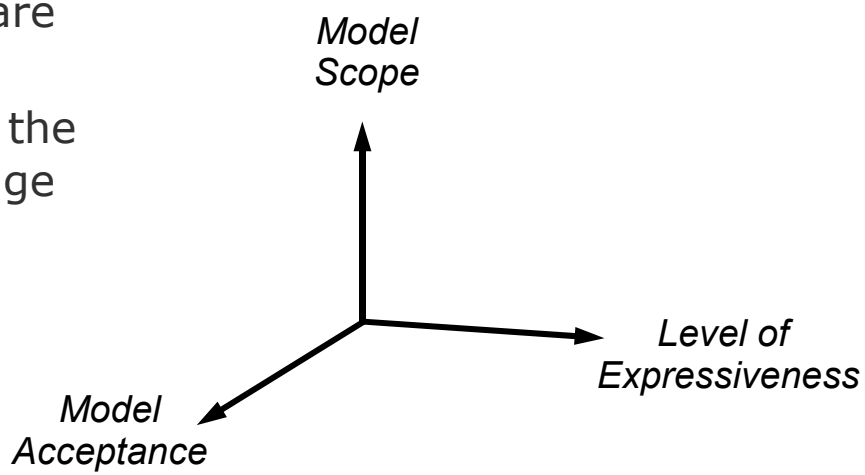
- Identify purpose
- Create informal knowledge model
- Design formal models, acceptance and (software) ssystem

... not biased towards specific models of SE (sequential, waterfall or spiral)



3D Matrix overview

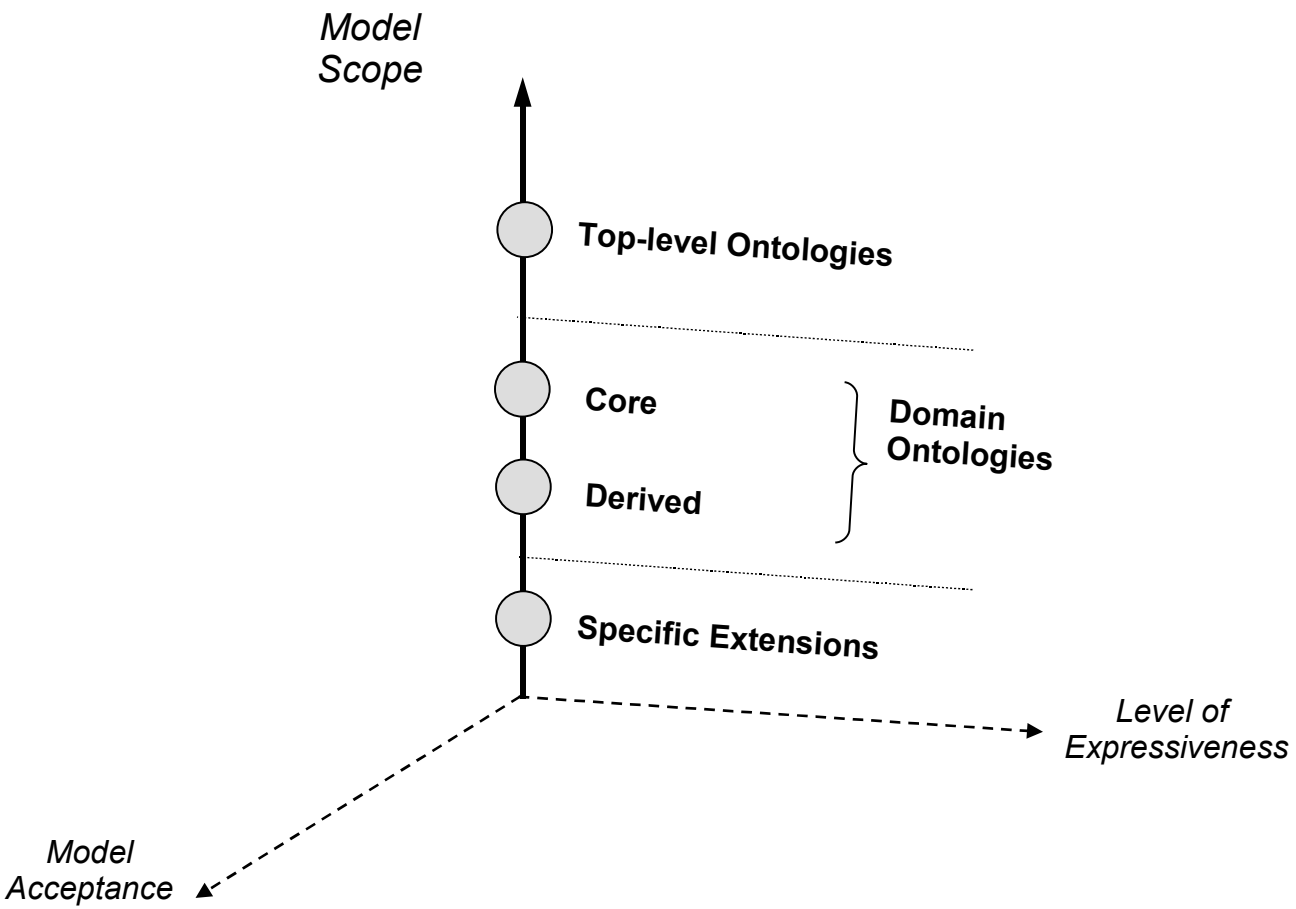
- Scope
  - Which parts of semantics are modelled by the ontology?
  - What is the perspective of the ontology onto the knowledge of the users?



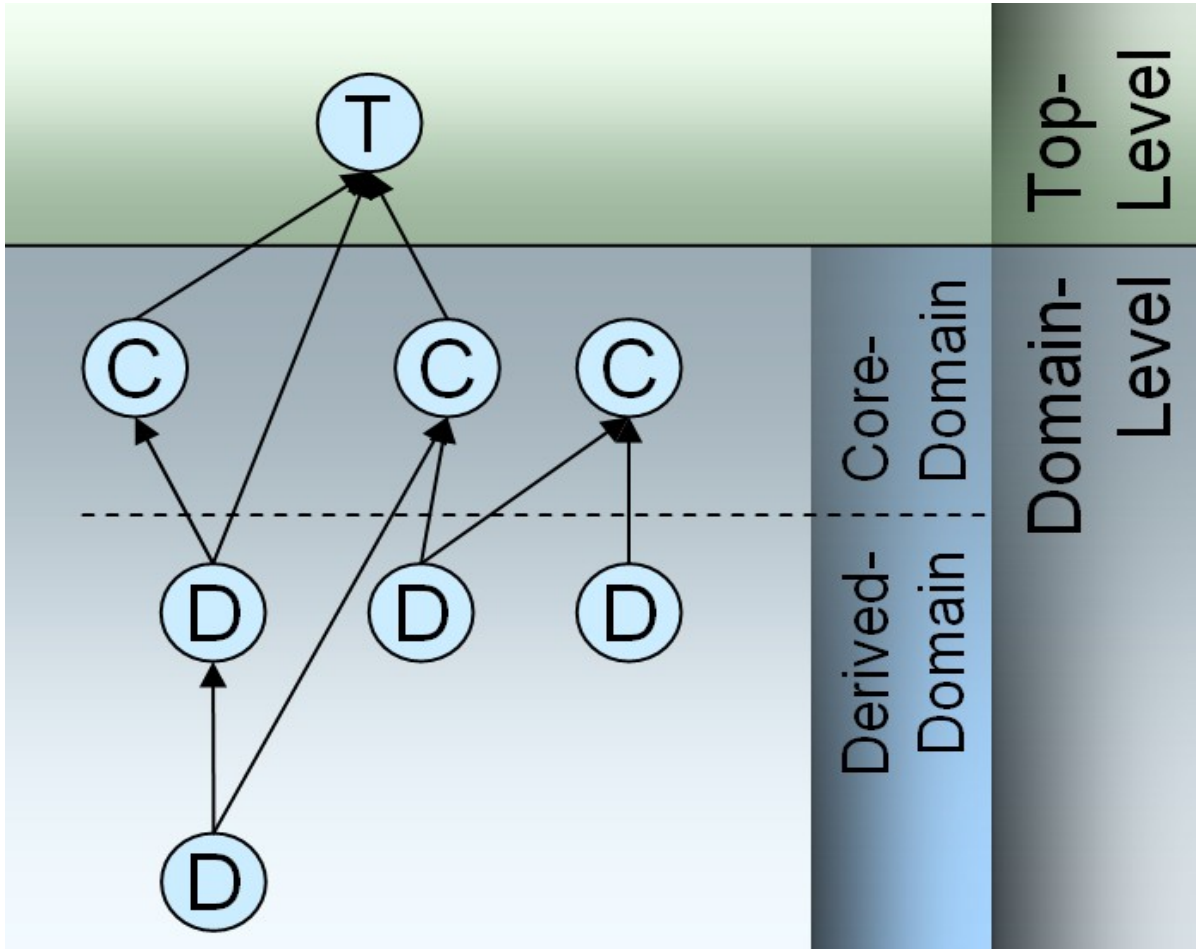
- Expressiveness
  - What kind of semantics is used to describe knowledge?
  - What semantics are needed to fulfil requirements?

- Acceptance
  - Which user communities will be using the ontology?
  - Which communities accept the ontology?

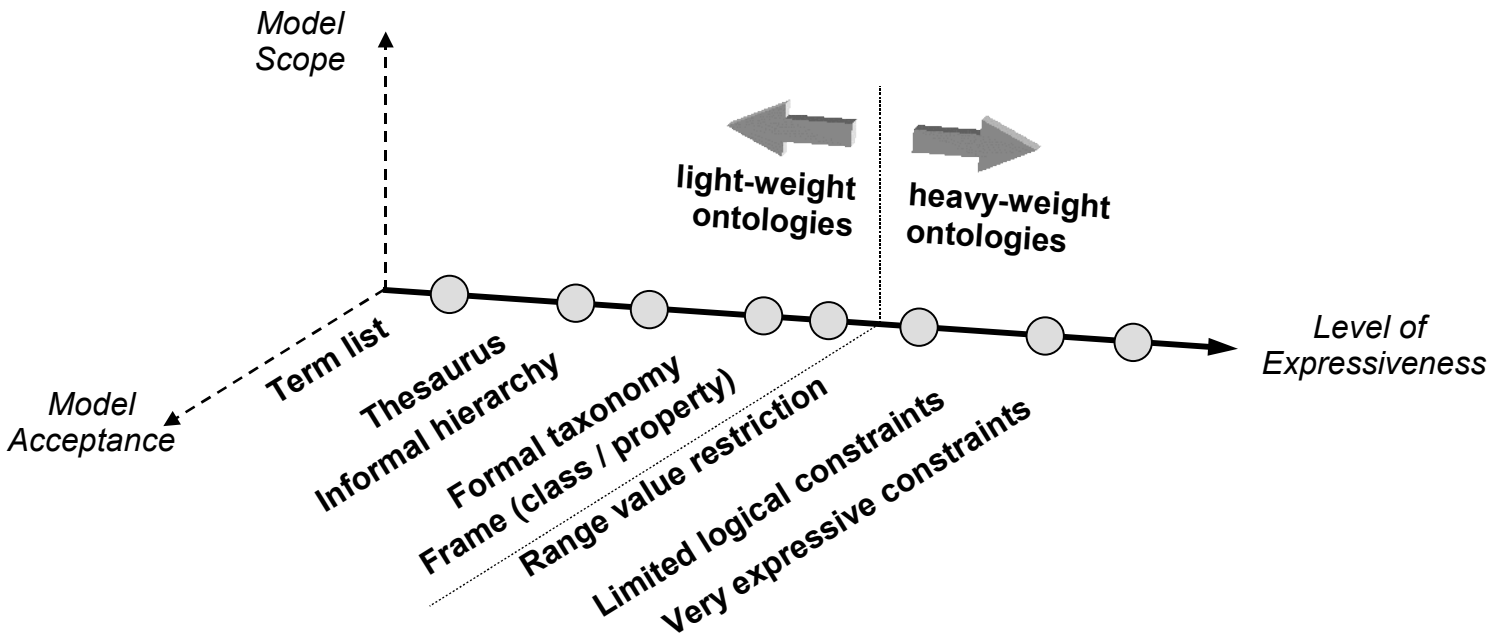
Model Scope



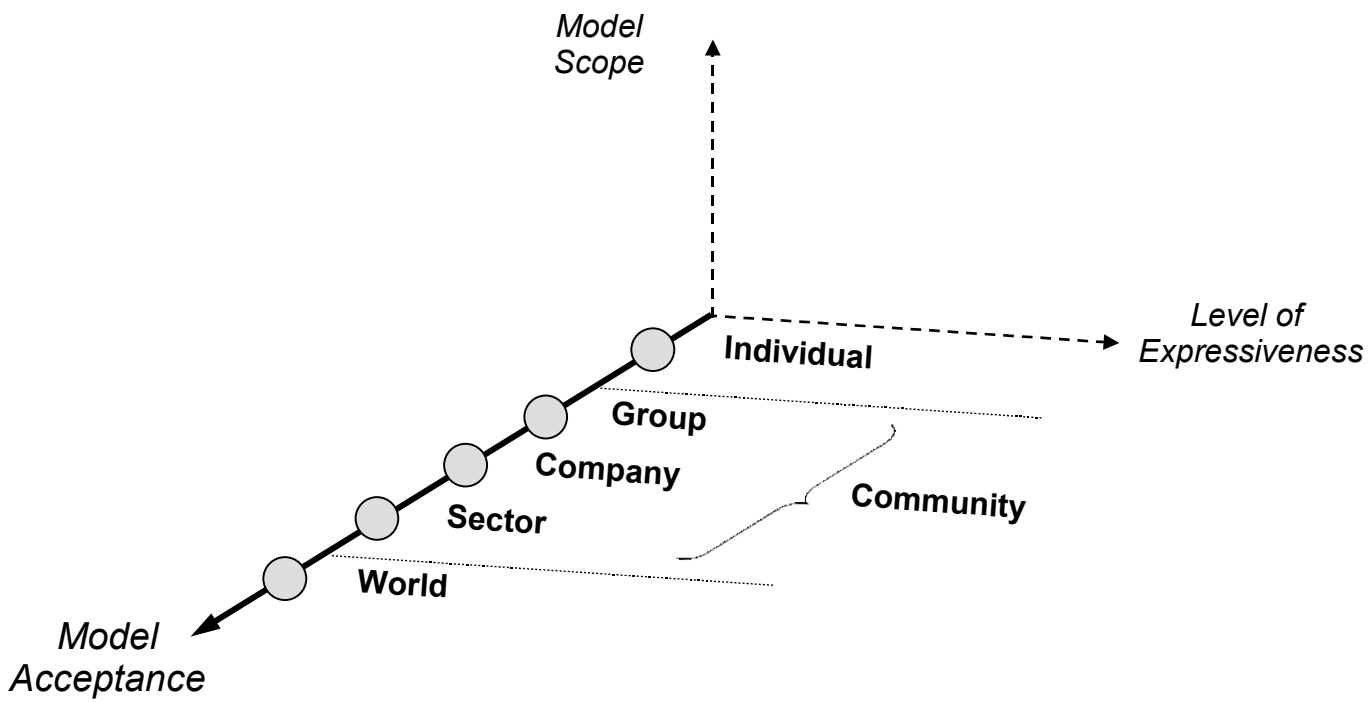
Model Scope



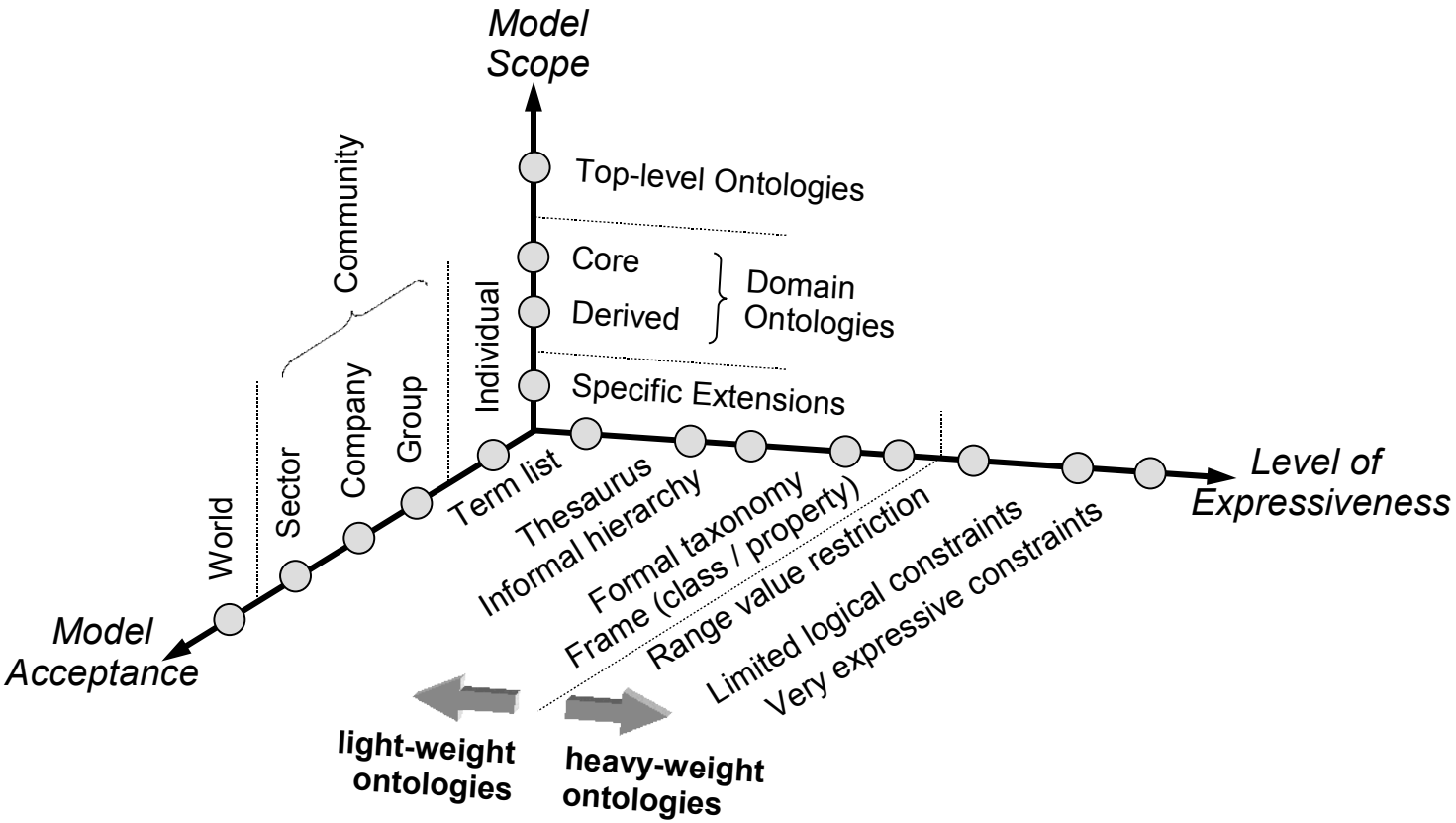
Expressiveness / Formality



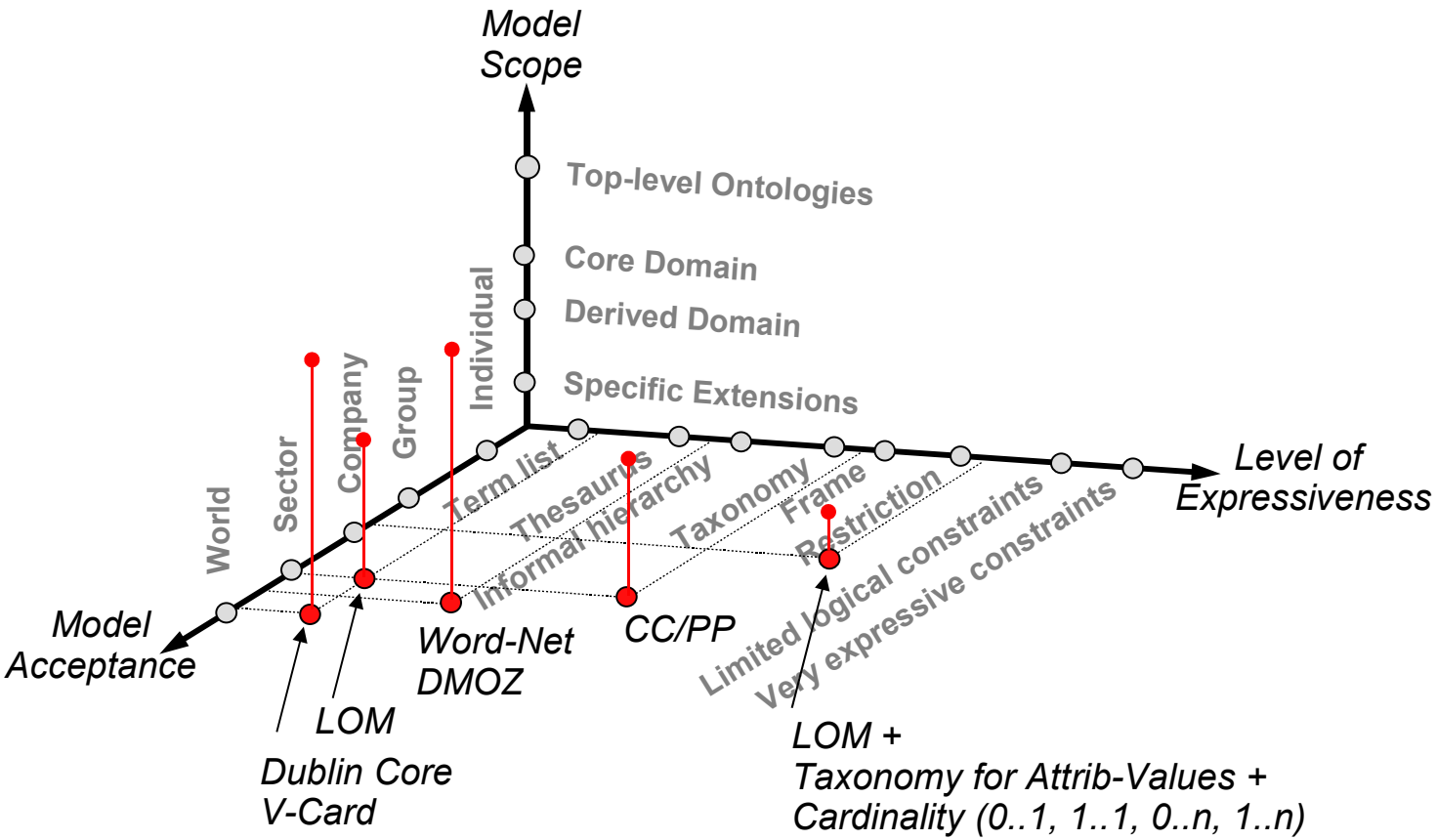
Model Acceptance



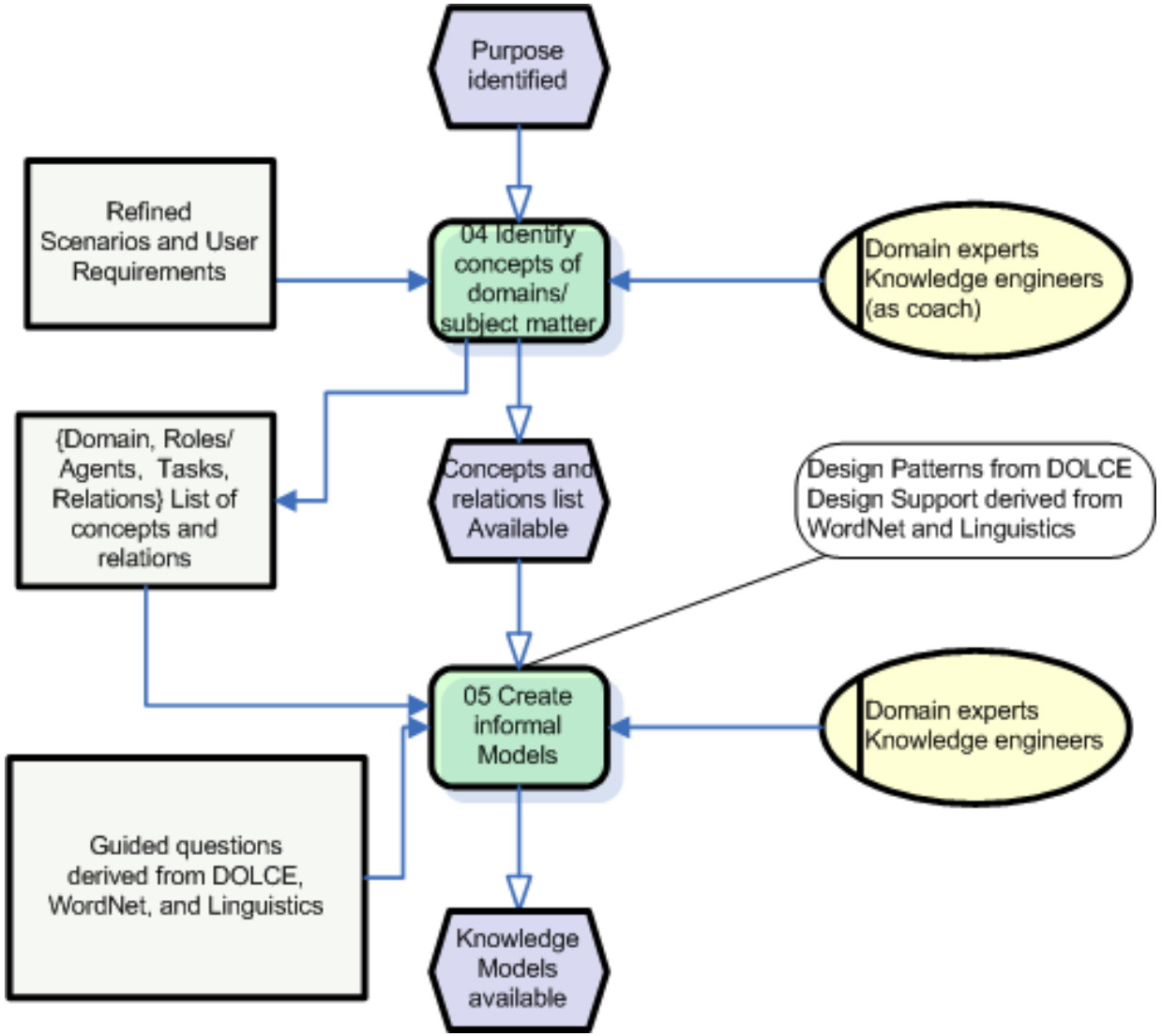
3D Matrix



Estimating the complexity/costs of modelling



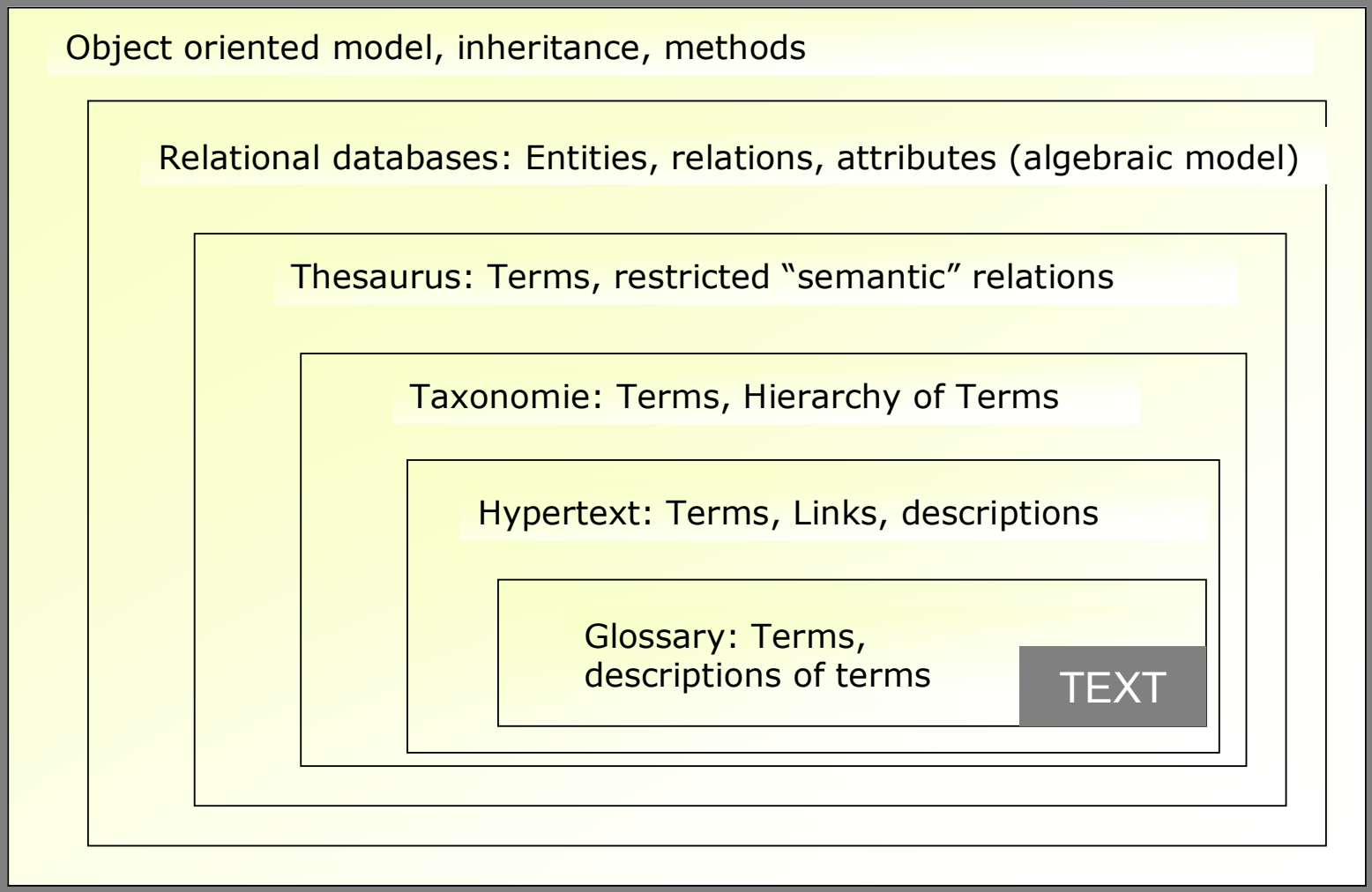
Creating informal models



Two phases of the methodolgoy

Human readable vs. formal semantics: an approach to (symbolic) knowledge representation

Formal ontology: adding a layer of logic



How to link domain expert knowledge from Texts (in natural language) to DolceLite+ ?

I) Brute force: Aligning directly to DolceLite+

- Will be done with domain experts and the support of a "Dolce-aware" knowledge engineer
- evaluated within DynamOnt

II) Use of WordNet and the WordNet mapping to DolceLite+

- This option will be tested in DynamOnt by using the OntoWordNet to suggest a link to DolceLite+

III) Use DolceLite+ Design Patterns

a) Bottom up approach as well as

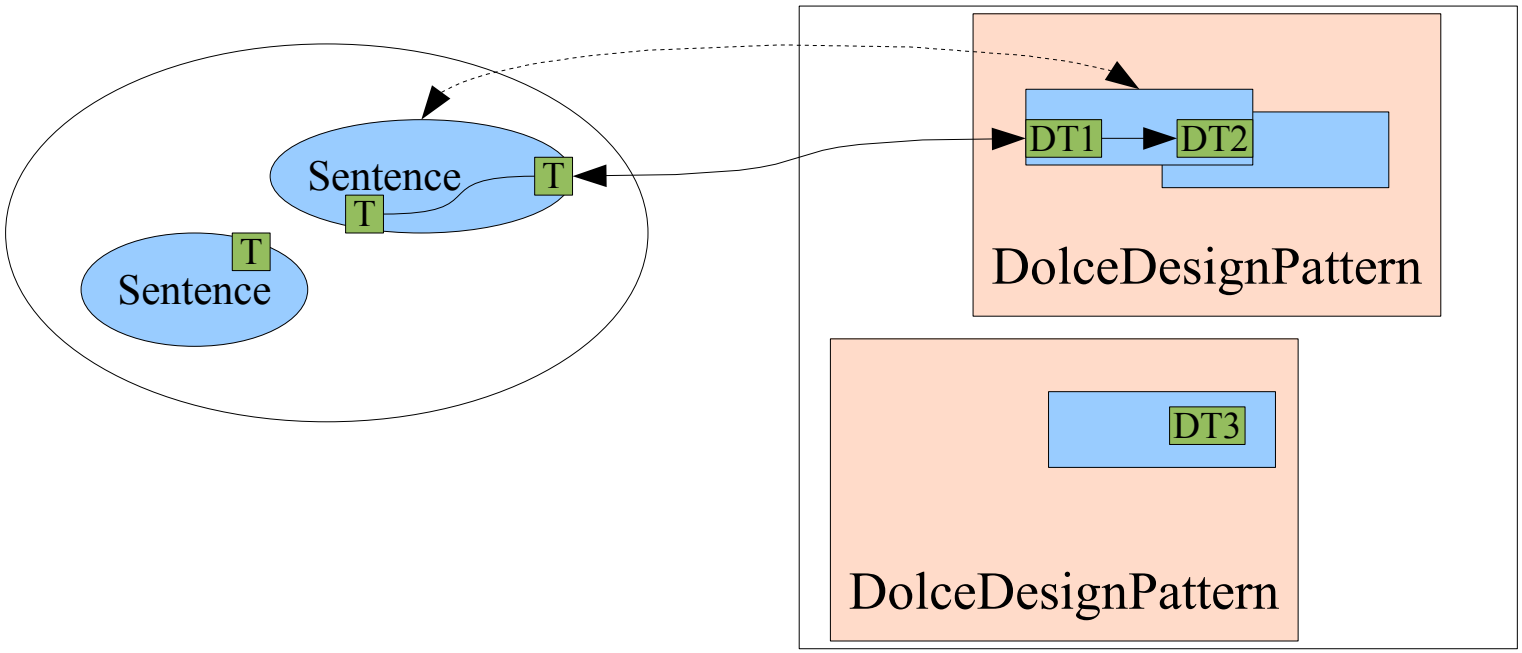
b) Top down approach

Expected pros and cons ..

- Methods I+II
  - centred on single terms (--> concepts classes)
  - context will be lost during the work ...
  - does not scale, has no methodological support
  - WordNet mapping might not work in specific domains
- Method III
  - centred on associations and features of terms
  - based on the notion of “text”, “statements” and “terms”
  - the tracking of terms within statements will provide some context and support collaborative development
  - usage of design patterns might facilitate the linking to upper level ontologies (DolceLite+ in our case)
  - Missing Link: Guided questions ...
  - Still unclear: Usage of FrameNet

What we intend to do …

NL text (usage scenario) ⟺ DolceLite+



IIIa) The bottom-up approach

- Start with approved usage scenarios (assume mutual understanding of domain experts) as input
- Experts then write simple NL "statements"
  - collaboration support during this elicitation process
  - no restrictions from KR languages
  - statements can be used as 'informal competency questions'
- Experts use available (Dolce)DesignPatternStatements as templates and try to subsume their statements under the patterns
- Experts "extract" terms and properties from statements and create (freely) their model – the statements remain as contextual information

IIIa) Bottom up-approach: Example

1. Natural language statements from experts

   - "ePortfolio is a tool for self-directed learning"
   - "A professor is member of an university"
   - "Peter coordinates the AST'06 within I-KNOW'06"

2. Agreement levels by tagging (based on T. Gruber, 2006)

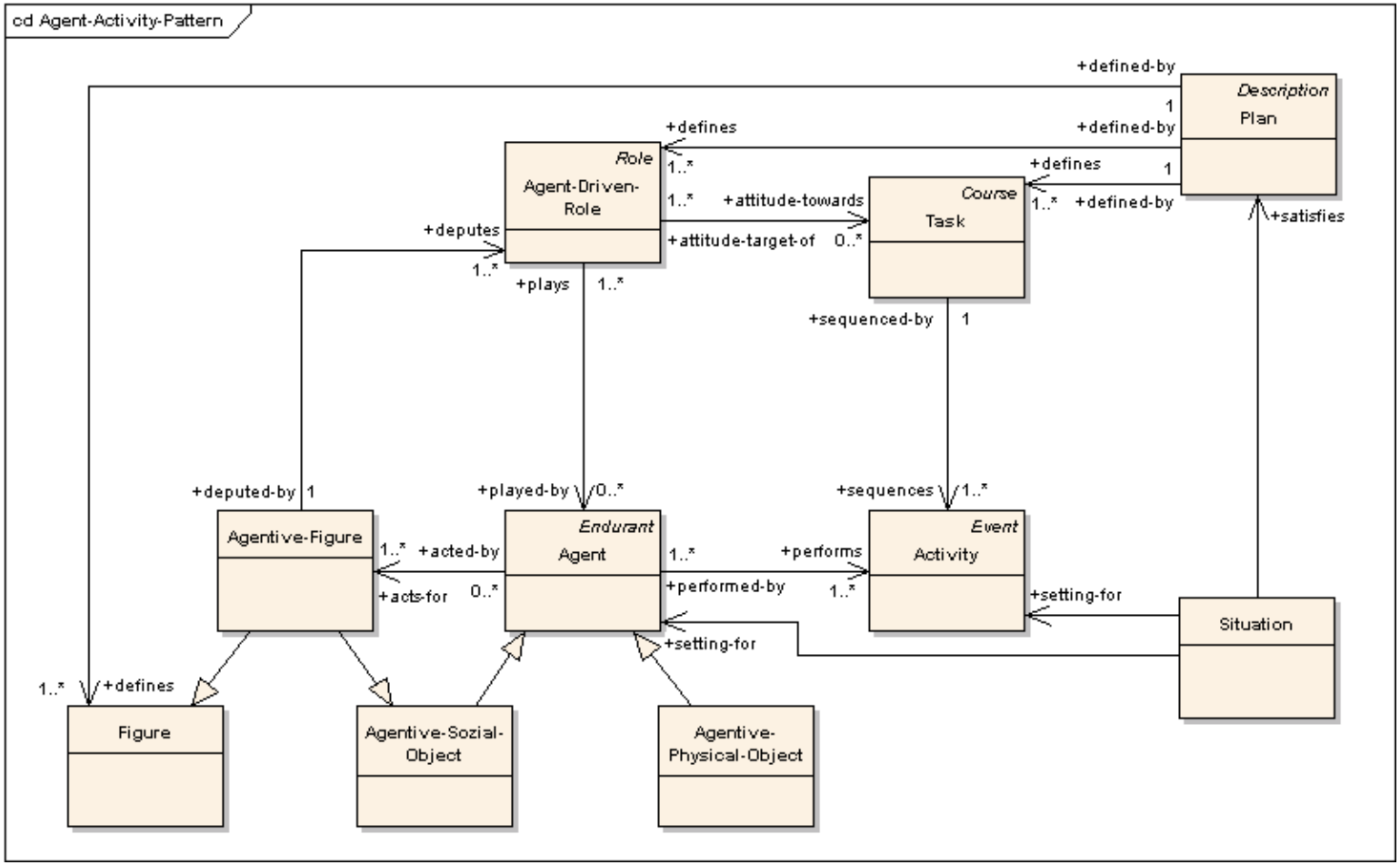
   - Tagging (statement1ver3, true, expert1, date-time, dyn-prj)
   - Tagging (statement1ver3, true, expert2, date-time, dyn-prj)
   - Tagging (statement1ver1, false, expert3, date-time, dyn-prj)

IIIa) Bottom up-approach: Example cont.

3. Use DOLCE LitePlus Design patterns as templates for NL Statements

- AGENT performs ACTIVITIES
  - Peter coordinates the AST'06 within I-KNOW'06
- AGENT acts for ORGANISATION (agentive-figure)
  - Peter coordinates the AST'06 within I-KNOW'06
- ROLE (a-d-r) deputed-by ORGANISATION (a-f)
  - A professor is member of an university
- AGENT plays ROLE
  - ePortfolio is a tool for self-directed learning

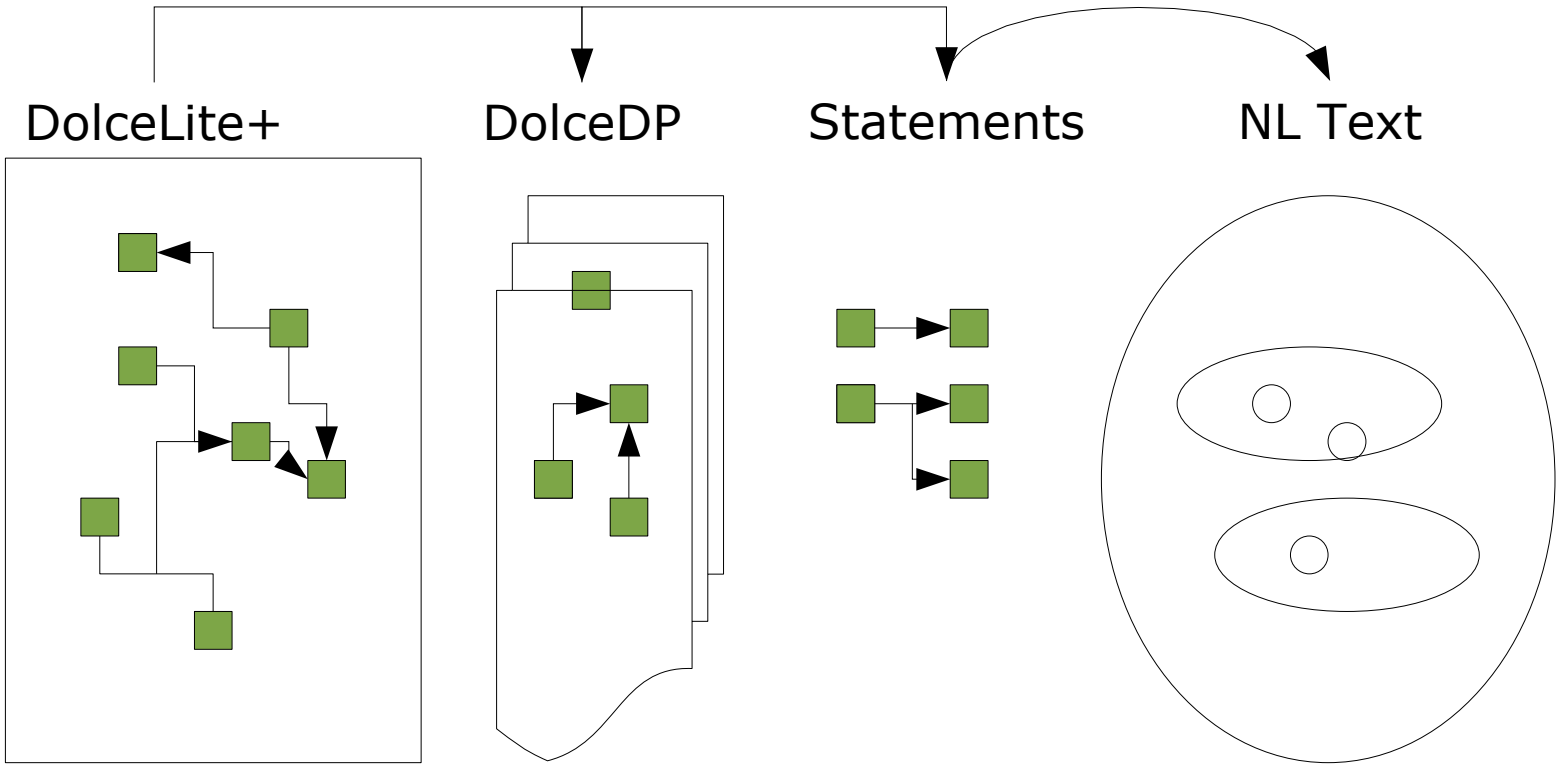
DOLCE Lite+ Agent-Activity pattern



IIIa) Expected pros and cons …

+ Experts can describe their knowledge in free form as statements.
- ExpertStatements might not match with the structure of PatternStatements
- 1-n mapping requires specific guiding to get the correct PatternStatement
+-ExpertStatements might use terminology that are not represented

IIIb) Top down approach



IIIa) Top down approach pros and cons

+ Transitions down to DesignStatements is formalized
+ Experts are guided to use 'well-formed' statements templates in order to create their model
+ Complexity of supporting experts is lower than in bottom-up approach
- Experts are restricted to constructs provided by DolceLite+

FAQ

- Is this methodology domain independent? **YES, but** it might prove, that it works better for some domains.
- Are the process model and its methods validated? **NO, not yet.** It will be evaluated within the DynamOnt project by experts from the eLearning domain.
- Will the methodology provide methods for text mining, ontology learning, ontology merging & mapping, ontology re-engineering. **NO.**
- Does the methodology provide methods for ontology creation and ontology building from scratch. **YES.**