An Architecture For Practical Ontology Engineering and Deployment:
the DOGMA Approach

Robert Meersman, Mustafa Jarrar
VUB STARLab
Vrije Univerisiteit Brussel, Belgium

Presented by: Dmitry Nikitov, Doctoral School 2002-2003
Contents

- What are the ontology, motivation, knowledge reusability

- The DOGMA approach
  - Ontology Base
  - Commitment Layer
  - Consistency

- Implementation and tools

- Conclusion and future work
Introduction

- What are ontologies
  - Branch of the knowledge engineering, in which semantics of certain domain is represented formally

- Ontology is more than data modeling
  - Data models should be autonomously specified within the domain

- Why ontologies: Web
  - Semantic Web: more productive use of the knowledge

- Knowledge reusability
  - Ontology stands as formal resource of knowledge
The DOGMA Approach

- Decompose Ontology into:
  - An Ontology Base (set of atomic predicates)
  - Commitment Layer (Rules)
The Ontology Base (1/2)

- A set of possible conceptualizations of the real world domain
- Knowledge components are spitted into a set of *lexons*
  - Expressed in lexical terms in a given language
  - *Lexons* are grouped into abstract contexts

```plaintext
(Organization-ContextID)
Person IsMemberOf Committee
Person Chairs Committee
Committee ChairedBy Person
Reviewer SubtypesOf Person
Author SubtypesOf Person
Reviewer Reviews Paper
Paper ReviewedBy Reviewer
Paper WrittenBy Author
Author Presents Paper
Paper Has PaperTitle
Paper Has PaperNumber
...
(Relationships-ContextID)
```
The ontology base is a set of context-specific binary fact types, called *lexons*. Notation: `<Term1, Role, Term2>`

<table>
<thead>
<tr>
<th>Context</th>
<th>Term1</th>
<th>Role</th>
<th>Term2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conference</td>
<td>Person</td>
<td>IsMemberOf</td>
<td>Committee</td>
</tr>
<tr>
<td>Conference</td>
<td>Committee</td>
<td>Includes</td>
<td>Person</td>
</tr>
<tr>
<td>Conference</td>
<td>Person</td>
<td>Chairs</td>
<td>Committee</td>
</tr>
<tr>
<td>Conference</td>
<td>Committee</td>
<td>ChairedBy</td>
<td>Person</td>
</tr>
<tr>
<td>Conference</td>
<td>Reviewer</td>
<td>SubTypesOf</td>
<td>Person</td>
</tr>
<tr>
<td>Conference</td>
<td>Person</td>
<td>Types</td>
<td>Reviewer</td>
</tr>
<tr>
<td>Conference</td>
<td>Author</td>
<td>SubTypesOf</td>
<td>Person</td>
</tr>
<tr>
<td>Conference</td>
<td>Person</td>
<td>Types</td>
<td>Author</td>
</tr>
<tr>
<td>Conference</td>
<td>Reviewer</td>
<td>Reviews</td>
<td>Paper</td>
</tr>
<tr>
<td>Conference</td>
<td>Paper</td>
<td>ReviewedBy</td>
<td>Reviewer</td>
</tr>
<tr>
<td>Conference</td>
<td>Author</td>
<td>Writes</td>
<td>Paper</td>
</tr>
</tbody>
</table>
Commitment Layer (1/2)

- The commitment layer is organized as a set of ontological commitments
  - Each ontological commitment responds to an instance of interpretation of application
    - It responds in terms of ontology base
- Ontological commitments are the sets of reusable knowledge components
  - They can interoperate since they share the same ontology base
Commitment Layer (2/2)

- A separate layer mediating between the ontology base and the application instances committing to the ontology. Organized as a set of Interpretational Views, where:
  - A set of rules in a certain syntax (consistent/closed set) that constrain a particular aspect of reality (which is conceptualized in the ontology-base).
  - Each application can create (reuse/inherit) its own interpretational view

<table>
<thead>
<tr>
<th>RuleID</th>
<th>Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Visible Lexons {1,2,3,4,5}</td>
</tr>
<tr>
<td>2</td>
<td>Each Person Chairs at most one Committee</td>
</tr>
<tr>
<td>3</td>
<td>Each Committee Includes at least one Person</td>
</tr>
<tr>
<td>4</td>
<td>Each Committee Includes at most one Person</td>
</tr>
<tr>
<td>5</td>
<td>Each Committee ChairedBy at least one Person</td>
</tr>
<tr>
<td>6</td>
<td>Each paper which is WrittenBy a Person must not beReviewedBy with this Person</td>
</tr>
</tbody>
</table>
Why: decompose into Ontology-base & Commitments Layer

- No absolute meaning; everything must be the result of *agreements* among such as designers, domain experts and users.
- All meaning (semantics) is for communication purposes, is represented independent of language but necessarily must be *entirely* rooted and described in (natural) language.
- *Identity* of concepts is a real problem; harder than agreeing about their properties.
- Domain constraints, rules and procedures are essential, but:
  - Agreement about them is very difficult.
  - Nearly always specific to a context of application.
Example: A Simple Ontology in the DOGMA Framework

The application A identifies a Paper by Paper_Number, while application B identifies by the Paper_Title.
Establishing Ontological Consistency

- Each interpretational view should be consistent (As set of rules that constrain a particular aspect of reality)

- Each set of interpretational views used by application should be consistent

- Adopting a given (well-defined) set of rule types helps analyzing the consistency and evaluating the ontology
Implementation: DOGMAModeler

- **Kernel: DOGMA Server**
  - Stores and serves the ontology base and the commitment layer
  - Consists of storage and the API
- **DOGMAModeler**: suite of ontology engineering tools
  - Ontology browser, editor, manager, mining tools
  - Supports functionality for modeling both - ontology base and commitments
Conclusion

- A database-inspired approach for engineering formal ontologies
- Knowledge are split into two groups: abstract contexts (set of lexons) and a layer of commitments
  - Lexons: binary facts
  - Commitments: rules, constraints ...
- Commitment layer mediates between ontology base and the applications
QUESTIONS?