Ideas for Collaborative Ontology Development on the upcoming Web 3.0 Era

Matthias Loskyll
(matthias@xantippe.cs.uni-sb.de)
Dominikus Heckmann

Saarland University
German Research Center for Artificial Intelligence (DFKI)
Outline

- Context and Motivation
- UbisEditor
  - Ontology Visualization
  - Editing Functionalities
  - Aspects of Collaborative Editing
  - Personalized Ontology Views
- Conclusions and Future Work
Web 3.0

Definition by Wahlster and Dengel:
Web 3.0 = Web 2.0 + Semantic Web

Semantic Web
(Protégé Editor)

Web 1.0
(e.g. Homepages)

Web 2.0
(e.g. Folksonomies)

Web 3.0
(UbisEditor 3.0)
UbisWorld

- Combining user modeling and ubiquitous computing
- UbisWorld = GUMO + UbisOntology + External Ontologies
Motivation

- Online ontology editor for UbisWorld needed
- Shall be used by ontology experts and by user community
- Ontologies grow in size and complexity
- Difficult to manage by one person or small group
- Early approaches (e.g. WebOnto, Ontolingua) had performance problems
- Basic idea comparable to WebProtégé
Ontology Visualization

- Ajax-based ontology tree visualization
- Data requested from server only when user opens tree node
- Visualization of very large knowledge sets possible
- Additional structure by using limitation nodes
- Properties of objects shown in the tree
Editing Process

- Right-click opens context menu
- Changing parents of nodes using drag&drop
- Multilingual editing of labels
- Editing of properties using similar grid
- User has to push button to commit changes to server
- Database backend altered according to performed changes
Rights Management and Quality Control

- Rights Management:
  - Role-based (e.g. standard user, ontology developer)
  - Context menu adaptive to rights of user (e.g. delete not allowed)
  - Visibility of tree branches

- Quality Control:
  - Store ID of user with performed changes
  - Five-star rating system
  - Needed extension: rating of raters
Collaborative Ontology Editing

- **Transaction management:**
  - Locking of tree branches when user starts editing process

- **Change management:**
  - Display change history of an object and of whole ontology
  - Annotation and discussion of changes
  - Possibility to revert changes

- **Conflict resolution:**
  - Role-based strategy
  - Community-based (voting)

- **Search functionality**
Personalized Ontology Views

- User-created personalized ontology view
- Check boxes to select needed objects
- Appropriate OWL-file provided for download

```xml
<owl:Class rdf:ID="N..00100006.Furniture">
    <rdfs:label> Furniture </rdfs:label>
    <ubis:category> 10 </ubis:category>
    <ubis:creator> USER..matthias </ubis:creator>
    <rdfs:subClassOf rdf:resource="#N..00100004.Thing"/>
</owl:Class>
```
Conclusions

- Ontology visualization using Ajax-based trees
- Editing using context menu and grids
- Role-based rights management
- Locking mechanism
- Search functionality
- Personalized ontology views
Future Work

- UbisEditor as a stand-alone application
- Import functionality
- Change management
- Annotation and discussion facilities
- Conflict resolution
- Evaluation of visualization and editing techniques
Thank you.

Please register at
www.ubisworld.org

Thank you for your attention.