A Protégé 4 Backend for Native OWL Persistence

June 25, 2009

Jörg Henss
Joachim Kleb
Stephan Grimm

FZI Research Center for Information Technology
at the University of Karlsruhe
Germany
Motivation

- Why do we need a Persistence Backend for Protégé 4?
  - Storage
  - Maintenance
  - Collaborative Work

- Why do we need a *new* Persistence Backend for Protégé 4?
  - Native support for OWL
  - It was missing ;(-)
By **nativeness** we understand:
- Mapping OWL language constructs one-to-one to storage layer

### Triple Structure
- RDF-Store
- CLOS model

### Axiomatic view
- Restrictions, cardinalities
- OWL acts on **objects** not on **nodes**
  - E.g. blank nodes are only recognizable via URI in RDF
- An object model for OWL is required
Schema Representation

- **OWL as Objects**
  - Concepts, Individuals, etc.

- **OWL-API as Object Model for OWL**
  - Java based API for OWL
  - Maintained by University of Manchester
  - OWL 2 ready
  - Protégé 4 is based upon

- **Use of Object-Relational mapping for persistence**
  - Stores object information in database
  - Restriction on necessary parts for Ontology Persistence
    - E.g. minimisation of redundancy
## Possible Strategies
- One Java Class One Table
- One Inheritance Tree One Table
- One Inheritance Path One Table
- Mixed forms

## Our Strategy
- Mixed form
  - One class one table
  - One inheritance tree one table
- Results in 56 tables
Relational Model
Complex Classes

common superclass
Complex Classes

Conjunction/Disjunction
Complex Classes
Complex Classes

Enumerations
Complex Classes

Restrictions

OWLRestriction

- OWLQuantifiedRestriction
  - filler
    - OWLDataAllRestriction
    - OWLDataSomeRestriction
    - OWLObjectAllRestriction
    - OWLObjectSomeRestriction

- OWLValueRestriction
  - value
    - OWLDataValueRestriction
    - OWLObjectValueRestriction

- OWLCardinalityRestriction
  - cardinality
    - OWLDataExactCardinalityRestriction
    - OWLDataMaxCardinalityRestriction
    - OWLDataMinCardinalityRestriction
    - OWLObjectExactCardinalityRestriction
    - OWLObjectMaxCardinalityRestriction
    - OWLObjectMinCardinalityRestriction

- OWLObjectSelfRestriction

OWLDescription

- OWLClass

- OWLNaryBooleanDescription
  - url
    - OWLObjectIntersectionOf
    - OWLObjectUnionOf
    - OWLObjectComplementOf

- OWLObjectOneOf
  - values
ABox

Instance Affiliation
ABox

Assertions
Comparison to other systems

- Most systems focus on optimisation techniques for reasoning. In contrast we focus on direct manipulation.

- No in-Memory parsing necessary.

- Highly similar to other systems on schema level, *e.g.* SOR.

- Direct manipulation
  - Complete ontology is editable on database level.

- Instance information persistence is similar to triple stores

- Ensures all functionalities of OWL-API
Database Persisted Ontology
OWL-API

Creation & Loading
OWL-API

Persisting
Architecture

Diagram showing the relationship between the Configuration File, Mapping, OWLDBOntology Factory, HibernateCfgUtil, Hibernate Config., Interceptor, and EventListener with the DBMS.
Benefits

- Minimisation of necessary joins compared to triple stores
  - Better retrieval
- Management facilities of RDBMS
  - Query optimisation
  - Transactions
  - Caching, etc.
- OWL 2 compatible
  - Mapping approach also usable for another API
- Modularisation via owl:import
  - Several ontologies possible
- Seamless integration into the OWL-API
  - Non-invasive
Seamless Protégé Integration

- Open Dialog
  - Similar to Protégé 3x
  - Additional dialect
Protégé Integration

Create ontology wizard

Physical Location/Database Location

Please specify the database access configuration of the database where your ontology is located (JDBC driver class, Table name, JDBC URL, Username and Password)

Database Configuration

driver
com.mysql.jdbc.Driver

dialect
org.hibernate.dialect.MySQL5Dialect

url
jdbc:mysql://localhost/owldb

username
root

password

Go Back Finish Cancel
Conclusion

- No change in interaction regarding the in-Memory implementation of Protégé (as well as in interaction with the OWL-API)
- No changes on the OWL-API object implementations (non-invasive)
- Project files desirable
- Still Prototype
- Download address

- Part of the German Theseus Research Project
Questions?