

The Jena-Based Ontology Model Inference and Retrieval Application

Luo Zhong, Mingzhu Zheng, Jingling Yuan, Jinxin Jin

Department of Computer Science and Technology, Wuhan University of Technology, Wuhan, China
Email: xiaozhutingzhou@163.com

Received March 17, 2012; revised May 24, 2012; accepted July 7, 2012

ABSTRACT

Ontology as an important representation model of semantic web has valuable application. A new ontology model on the basis of Computer Graphics (CG) knowledge is proposed, called CG ontology model. The protégé is used to build this ontology model conveniently. The Jena API is applied to store CG owl documents in MySQL, set inference rule and achieve search queries on the ontology database. Finally, the Jena-based ontology model retrieval system is developed.

Keywords: Ontology Model; Jena; Semantic Web; Inference Rule; Retrieval Application

1. Introduction

Semantic web is intelligent and gives the semantics of Internet network to connect the webpage content. IBM's technical architect Naveen Balani [1] thinks that the Semantic Web will become an effective way to represent data on the World Wide Web, or as a database that is globally linked, in a manner understandable by machines, to the content of documents on the Web. Semantic technologies use ontologies to represent meaning and provide reasoning through the relationships, rules, logic, and conditions represented in those ontologies. The open source platform protégé [2] in Stanford University generates owl documents through class hierarchy and property, and visualizes ontology model. Jena [3] is an open Java framework developed by HP laborator, and provides the programming environment for the languages like RDF, RDFS, OWL, SPARQL and so on, and includes rule-based inference engine with persistent storage. Jeremy J. Carroll [4] introduces the function and architecture of Jena.

The application range of Jena is very extensive. Wei Zhang [5] reasons with ontology model depicted with OWL using protégé and Jena API in question answering system. Sun-Young Heo [6] improves query processing performance of OWL data based on Jena. Min Wang [7] puts forward a Jena based Web page ontology extracting and processing method on the common RDF level of ontology, which includes extracting RDF model, setting operation of different RDF models, querying RDF model, modifying RDF model, and serialization of RDF model.

Ontology model has achieved the application system in the search technology of semantic web. Yi Zhang [8]

proposes ontology search engine-OntoSearch, creates an instance of class hierarchy to design food ontology model and visualizes food ontology class hierarchy with JSP. Li Ding [9] discusses how to search, index, sort semantic web documents and studies the structure of Swoogle [10] which is a crawler-based indexing and retrieval system for the semantic web documents (RDF or OWL documents).

This paper implements CG ontology model inference and query. The rest of the paper is organized as follows: Section 2 establishes CG ontology model with protégé. Section 3 stores CG ontology model in Mysql based on Jena. Section 4 designs inference rule, queries ontology database and visualizes CG ontology model using JSP and Java.

2. CG Ontology Model

Based on ontology, the model is composed of four parts: designing class hierarchy and property, creating the instances, building ontology model using protégé, forming owl documents. The following is an example of Basic graph generation algorithm on Computer Graphics to build CG ontology model.

2.1. Reference to "Computer Graphics" Course, Design CG Knowledge Class Hierarchy

The concept Line scan conversion algorithm is subclass of Basic graph generation algorithm, the concept Numerical differential algorithm is individual of Line scan conversion algorithm and so on in **Figure 1**.

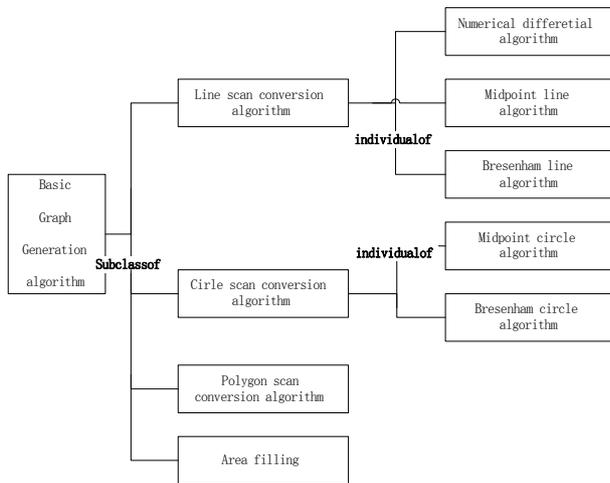


Figure 1. CG knowledge class hierarchy.

2.2. CG Property Relations

Property relations show the links between classes, including two types: object property and datatype property. Object properties are the relationship between two instances of classes, and datatype properties is the datatype of class instance, including domain and rang.

Object properties in CG knowledge classes are described as follows: 1) Reference extended relationship means point of knowledge associated with another point, including symmetry. For example, the concept bresenham line algorithm and Bresenham circle algorithm; 2) Precursor relationship means point A is the prior knowledge with point B, including transitive; 3) Follow-up relationship means point B is the follow-up knowledge with point A, including transitive, and relationship between precursor and follow-up is inverse for each other; 4) Overall and part relationship, such as Basic graph generation algorithm and Line scan conversion algorithm.

CG datatype properties include the property name, concept, principle, text description of algorithm and code description of algorithm. The domain is the concept Basic graph generation algorithm and the range is string. In **Figure 2**, CG resource class hierarchy and CG experiment class hierarchy are created to link with CG course, and set the corresponding property relations. At last, CG owl document has been completed by protégé, and CG ontology model is formed.

3. Based on Jena CG Ontology Model Stored in Database

Jena is an open source Java framework. Its core function includes RDF API, RDQL query language, reasoning subsystem, persistent storage memory, ontology subsystem, and provides the appropriate interface.

Jena provides relational database interface into RDF data, Model, Resource, Query can be used to access and

maintain RDF data. CG ontology model is mainly stored in Mysql. Jena stores ontology model using two types of seven tables. The first type is asserted statements and the second is reified statements in **Tables 1** and **2**.

Jena supports memory model and database model with com.hp.hpl.jena.db package, Database Persistence is following as steps:

- 1) Load the database JDBC driver Class. For Name (strDriver);
- 2) Create the database connection IDBConnection conn = new DBConnection (strURL, strUser, strPassWord, strDB);
- 3) Create a ModelMaker for database ModelMaker maker = ModelFactory.createModelRDBMaker (conn);
- 4) Create a default model “MyOntology” for ontology;
- 5) Perform data conversion, stored ontology data in the database defModel.read (in, null).

4. CG Ontology Model Inference and Retrieval Application

4.1. CG Ontology Model Retrieval System by Jena and JSP

The CG ontology model retrieval system is achieved using C/S mode, implements complex reasoning query process through Jena API, and is composed by user search interface, retrieval process, Jena analyze and ontology model database. In the client, user submits requests to the retrieval page. Server accepts the requests, search the results in CG ontology database through Jena inference. The architecture is shown in **Figure 3**.

4.2. CG Ontology Model Inference and Query

Next, an example of using the retrieval system is given. In the login page, user should submit class name and property, such as class name is <http://www.owl-ontologies.com/CGeg.owl#Numerical_differetial_algorithm>, and property is <http://www.owl-ontologies.com/CGeg.owl#follow_up>, click OK, **Figure 4** is shown.

In **Figure 5**, Search result show Numerical_differetial_algorithm follow-up relationship, but there are Mid_circle_algorithm and Bresenham_circle_algorithm in the direct follow-up relationship. Why is the search result different from direct relationship in **Table 3**? The answer is the inference rule designed. The rule and query is as follows:

```

String rule =
"[rule:(?x <http://www.owl-ontologies.com/CGeg.owl#follow_up> ?y)"
+ "(?y <http://www.owl-ontologies.com/CGeg.owl#follow_up> ?z)" +
"->(?x <http://www.owl-ontologies.com/CGeg.owl#follow_up> ?z)"]";
  
```

The rule means the condition y follow up x and z follow

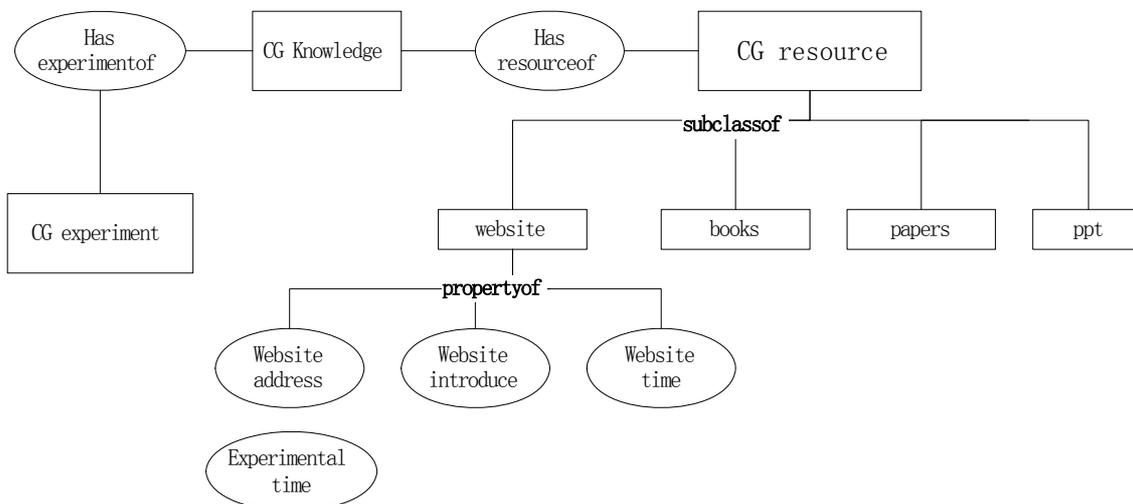


Figure 2. CG resource class relation.

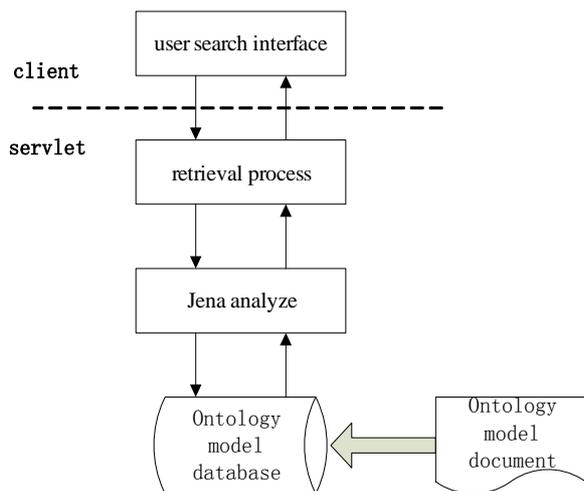


Figure 3. CG ontology model retrieval structure.

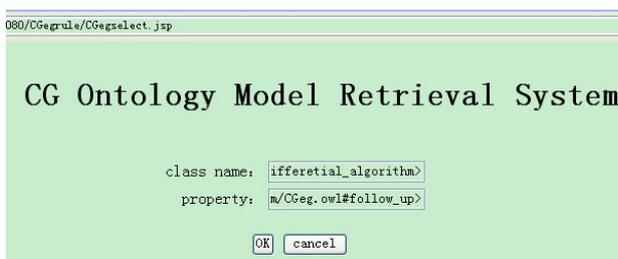


Figure 4. Ontology model login page.

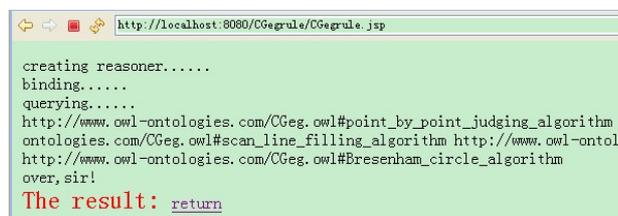


Figure 5. CG ontology model inference page.

Table 1. Statement table.

Name	Function	Category
Jena_GiTj_Stmt	Store ontology data	Asserted statement table
Jena_GiTj_Reif	Deal with ontology data	Reified statement table

Table 2. System table.

Name	Function	Category
Jena_Sys_Stmt	Store system metadata	System statement table
Jena_Long_Lit	Store a long string constant in the statement table	Long literals table
Jena_Long_URI	Store URI in the statement table	Long resources table
Jena_Prefix	Store URI prefix, only one	Prefixes table
Jena_Graph	Store user name and unique identifier	Graph table
Jena_Mutex	No contents, if table exists, the database will be locked in a certain period	Lock table

Table 3. The query result.

x
CG:point_by_point_judging_algorithm
CG:scan_line_filling_algorithm
CG:Mid_circle_algorithm
CG:Bresenham_circle_algorithm

up y deduce the result z follow up x.

String queryString =
 "PREFIX
 CG:<http://www.owl-ontologies.com/CG.owl#>"
 + "SELECT ?x " + "WHERE"
 + "{" + classname + property + "?x";
 The query result show in Table 3.

1) Create reasoner

```
Reasoner reasoner = new GenericRuleReasoner(Rule.  
parseRules(rule));
```

2) Reasoner and ontology model binding

```
InfModel inf = ModelFactory.createInfModel(reasoner,  
instances);
```

3) Construct the query

```
Query query = QueryFactory.create(queryString).
```

5. Summary and Future Work

Semantic web research is very hot now. Ontology is one of key technologies in Semantic web. This paper designs CG ontology model with protégé, stores the model in database, sets inference rule, queries ontology database, and finally realizes CG ontology model retrieval system. But this is only for a single ontology model, the scope of the application is small. According to the existing ontology search engine, such as Ontosearch [8] and swoogle [9], we will research multiple ontology model to apply to semantic search engine, extract web page under the concept of ontology model, search the semantic web documents to return the best results, in order to intelligent search technologies.

REFERENCES

- [1] N. Balani, "The Future of the Web Is Semantic," 2005. <http://www.ibm.com/developerworks/cn/java/wa-semweb>
- [2] "Welcome to Protégé." <http://protege.stanford.edu/>
- [3] "Jena—A Semantic Web Framework for Java." <http://jena.sourceforge.net/>
- [4] J. J. Carroll and D. Reynolds, "Jena: Implementing the Semantic Web Recommendations," *Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters*, New York, 2004, pp. 74-83.
- [5] W. Zhang and L. G. Duan, "Reasoning and Realization Based on Ontology Model and Jena," *IEEE Fifth International Conference on Theories and Applications (BIC-TA)*, Taiyuan, 23-26 September 2010, pp. 1057-1060.
- [6] S.-Y. Heo, "A Study on the Improvement of Query Processing Performance of OWL Data Based on Jena2," *International Conference on Convergence and Hybrid Information Technology*, Cheonan, 28-30 August 2008, pp. 678-681.
- [7] M. Wang and J. P. Ding, "The Research on the Jena-Based Web Page Ontology Extracting and Processing," *First International Conference on Semantics, Knowledge and Grid*, Shanghai, 27-29 November 2005, p. 105.
- [8] Y. Zhang, "OntoSearch: An Ontology Search Engine," *SGAI Conference*, 2004, pp. 58-69.
- [9] L. Ding and T. W. Finin, "Search on the Semantic Web," *IEEE Computer*, Vol. 38, No. 10, 2005, pp. 62-69. [doi:10.1109/MC.2005.350](https://doi.org/10.1109/MC.2005.350)
- [10] L. Ding, T. Finin, A. Joshi, R. Pan and Y. Peng, "Swoogle: A Search and Metadata Engine for the Semantic Web," *Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management*, Washington DC, 9 November 2004, pp. 652-659.