World Academy of Science, Engineering and Technology
International Journal of Computer and Systems Engineering
Vol:3, No:5, 2009

RDFGraph: New Data Modeling Tool for Semantic Web

Daniel Siahaan, and Aditya Prapanca

International Science Index, Computer and Systems Engineering Vol:3, No:5, 2009 waset.org/Publication/11368

Abstract**—**The emerging Semantic Web has been attracted many researchers and developers. New applications have been developed on top of Semantic Web and many supporting tools introduced to improve its software development process. Metadata modeling is one of development process where supporting tools exists. The existing tools are lack of readability and easiness for a domain knowledge expert to graphically models a problem in semantic model. In this paper, a metadata modeling tool called RDFGraph is proposed. This tool is meant to solve those problems. RDFGraph is also designed to work with modern database management systems that support RDF and to improve the performance of the query execution process. The testing result shows that the rules used in RDFGraph follows the W3C standard and the graphical model produced in this tool is properly translated and correct.

Keywords**—**CASE tool, data modeling, semantic web.

I. INTRODUCTION

WORLD Wide Web has grown enormously for these last few decades. Various content have been posted on the Web, such as texts, still images, audio and video, by various individuals from various backgrounds, with various reasons and goals. World Wide Web has been a super huge content storage that ever exists.

This phenomenon has introduced a new enhancement of web that is called the Semantic Web. Semantic Web, which was first conceptualized by Tim Berners-Lee [1], is aimed on giving a context to the existing web, so that the web would no longer be consumption for only human, but also machines and appliances. Therefore, many researchers and developers have focus on developing new applications on top of the Semantic Web, such as search engines, context-aware personal guides, context-aware assistants, etc. Those applications are based on the knowledge and information which are modeled in metadata languages, such as RDF/S and OWL. These metadata languages are basically based on semantic model. As in relational model, there are a number of powerful CASE tools for helping the domain knowledge to model the domain in relational model, such as MS. Access, MS. SQL Server and Power Designer. In semantic model, there are also exist similar tools, such as Protégé [8] and RDF Editor [2].

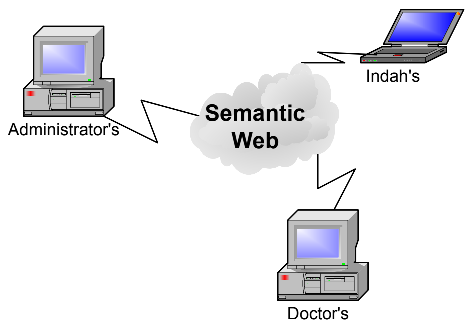
D. Siahaan is with the Institut Teknologi Sepuluh Nopember, Surabaya 60111, Indonesia (phone: +62-31-5939214; fax: +62=31-5939214; e-mail: daniel@if.its.ac.id).

Aditya Prapanca is with Surabaya State University, Surabaya 60122,Indoensia (e-mail: aditya@unesa.ac.id).

In this paper, a new CASE Tool for graphically modeling metadata in Semantic Web using new graphical notations is proposed, proposed by Siahaan [12]. This tool, which is called RDFGraph, is meant to improve the readability and easiness of the domain knowledge to graphically model a problem in semantic model.

This paper is organized as follows. The second section describes briefly about the Semantic Web. The third section shows the existing CASE Tools for modeling in semantic model. The fourth section describes the graphical notations that are used in our proposed CASE Tool. The fifth section explains in detail about the RDFGraph. The sixth section shows the testing that has been carried out and some results. The last section ends up this paper by some conclusions and proposed further works.

II. SEMANTIC WEB



(a)

Indah asked her agent to find out about the result of her regular medical checkup that was done the day before.
Indah's: Doctor, What about the result?
Doctor's: Administrator, please send me the lab. result of Indah.
Administrator's: Doctor, here is the lab. result from yesterday checkup.
Doctor's: Hemm… (*processing*). Indah, I think we need to do a blood test. How about this Monday?
Indah's: Hemm… (*checking*).. Monday is full, how about the next day?
Doctor's: Okay.
Indah's agent displays a message that tells her that she has an appointment for blood test on Tuesday.

(b)

Fig. 1 Agents as personal assistances

All information stored in WWW are designed only for human consumption. The information within the documents is not giving any meaning for appliances in order to understand

International Science Index, Computer and Systems Engineering Vol:3, No:5, 2009 waset.org/Publication/11368

and manipulate them. On the other hand, developers want to build appliances that assist people to get their jobs done or acquire what they want by making used of the information on the web. The solution is by creating a web of semantics. In other words, the meaning of the information within documents is given to enable interoperability between appliances. And by doing this, the nature of the Web will change. Fig. 1 shows one of the Semantic Web scenarios.

Semantics enables appliances to understand the information from diverse sources, to manipulate it and to reason about it. In other words, the Semantic Web meant to transform the current WWW into a consistent logical web of data for appliances so that they can collaborate with each other and people.

A. Semantic Web ≠ Artificial Intelligence

The Semantic Web is not a realization of artificial intelligence. In artificial intelligence, the appliance is equipped with artificial intelligence to learn and understand what a person is saying or doing. On the contrary, in the Semantic Web, the appliance deals with a well-defined problem using well-defined data as the input for its well-defined operation, on which the people's part is defining them. The goal of this concept is to make the web a space for not only human-to-human communication, but also human-to-machine and machine-to-machine communication.

B. Semantic Web does not use Closed World Assumption

Closed World Assumption (CWA) is a meta-rule that states that a negation of a fact is true if the fact does not exist in the system. Database is one of the systems that use the CWA. It assumes that the database is complete. For example, consider that relation isParent(Mark,David) cannot be deduced from the database, then it can be assumed that ¬isParent(Mark, David) to be true.

On the other hand, the reasoning in the Semantic Web does not use CWA. It should be monotonic. A result from inferring set of facts P is also a result from inferring another set of facts of which P is the subset. The reasoning needs to take place in a potentially open-ended situation. Since the web is an gigantic database to search to, therefore it would take tremendous time to do inference on the whole assertions in the web to check the existence of a fact. One should assume that there is always the possibility that new information might arise from some other source, but yet did not change the validity of the previous inference result. For example, if one would like to find the ancestors of Willem Alexander, Prince of Orange, then only on the assertions that reside in certain documents would be inferred, or certain namespace, or some other parameter would be used. Therefore the reasoning should easily be controlled, so that it leads to an open-ended situation.

C. Semantic Web Languages

According to Tim Berners-Lee [1], the goal of Semantic Web is to express the real life. Therefore the language used to realize it should be sufficient, flexible and powerful to express the real world. Until now W3C has come out with one recommended language for the Semantic Web: Resource Description Framework (RDF) [11]. The RDF model provides the framework to describe resources. RDF is not a conceptually and physically centralized system, meaning that everyone does not have to agree on the same term for the same object. An object is identified by a URI, which satisfies the requirement for global consistency in resource naming. W3C is also working on the RDF Schema that provides richer modeling primitives.

There are also efforts to develop other Semantic Web languages, such as DARPA Agent Markup Language + Ontology Inference Language (DAML+OIL) [4] and Ontology Web Language (OWL). OWL is expected to support the eight design goals of the Web ontology language [10], such that it goes beyond RDF and RDF Schema basic primitives. The eight design goals are shared ontologies, ontology evolution, ontology inconsistency detection, balance of expressivity and scalability, ease of use, XML syntax and internationalization. There are also several related developments, such as XML Topic Map [13] and KIF [5], which will not be discussed in this report.

D. Ontology Services

According to Merriam-Webster's Collegiate Dictionary, ontology is a particular theory about the nature of being. Ontology in the Semantic Web can be defined as definitions of classes and their relations. As already mentioned earlier, in the Semantic Web everyone can say anything about anything anywhere. In this manner, everyone can define his own ontology. The first phenomenon is the emerging of ontology portals, where everyone can promote their ontologies. These portals can offer certain ontologies to the users according to their request and the users have to pay for the use of the ontology. And then the portals can function as brokers.

The example of the phenomenon can be seen in mp3 technology. Nowadays there are several companies have already offered this kind of service for mp3 files, for example CDDB [3], mp3.com [6] and Music Brainz [7].

III. Existing Case Tools

There is already a number of case tool for editing RDF/RDFS statements, e.g. RDF-Editor [2] and Protègè-OWL [8]. RDFEditor was designed to allow user to edit RDF document (extension .rdf) on a text base mode. Fig. 2 shows the snapshot of RDFEditor. The editor has limited functionalities, such as creating XML skeletons and removing statement.

The more advance case tool for editing RDF statements is Protege-OWL. It was the extension of Protege. The current stable version of Protege-OWL is 3.4. Fig. 3 shows the snapshot of Protege. Protege allows user to do the following:

- Load and save OWL and RDF ontologies.
- Edit and visualize classes, properties and SWRL rules.
- Define logical class characteristics as OWL expressions.
- Execute reasoners such as description logic classifiers.

- Edit OWL individuals for Semantic Web markup.

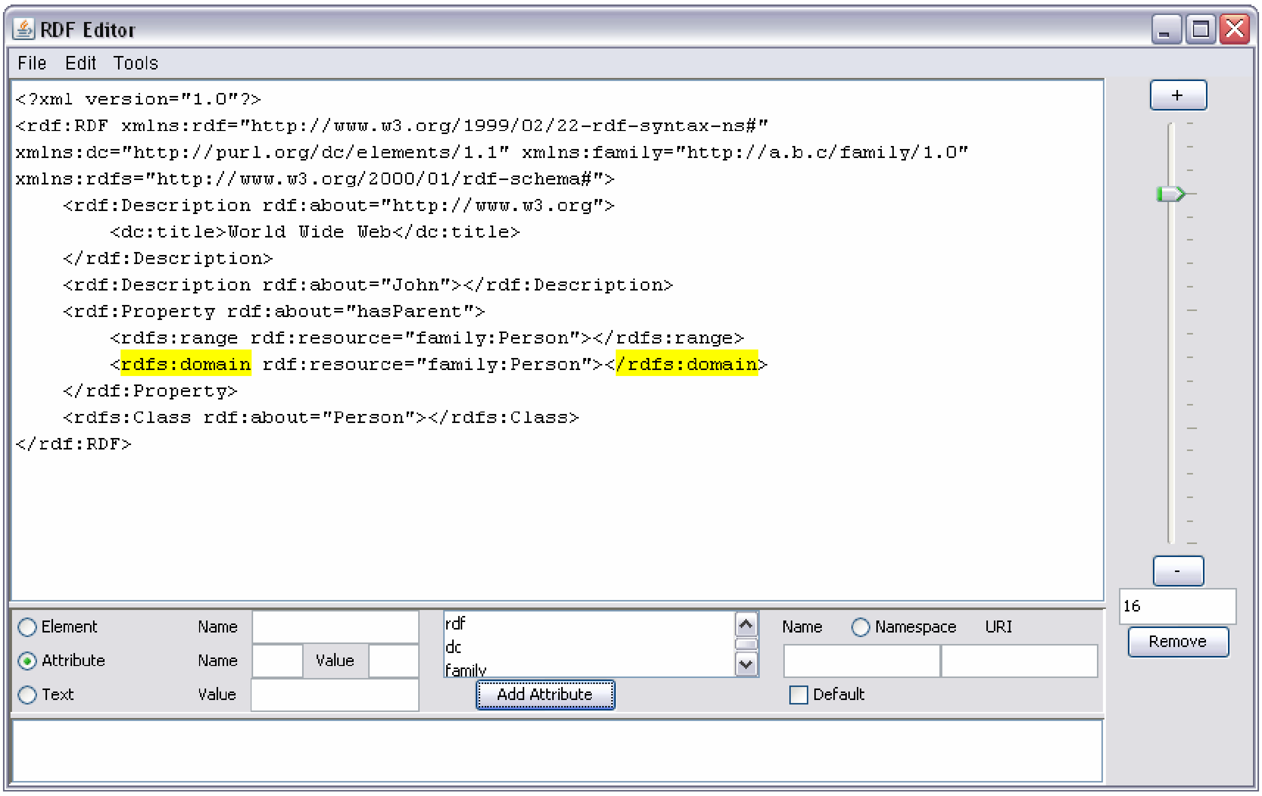


Fig. 2 Snapshot of RDF-Editor

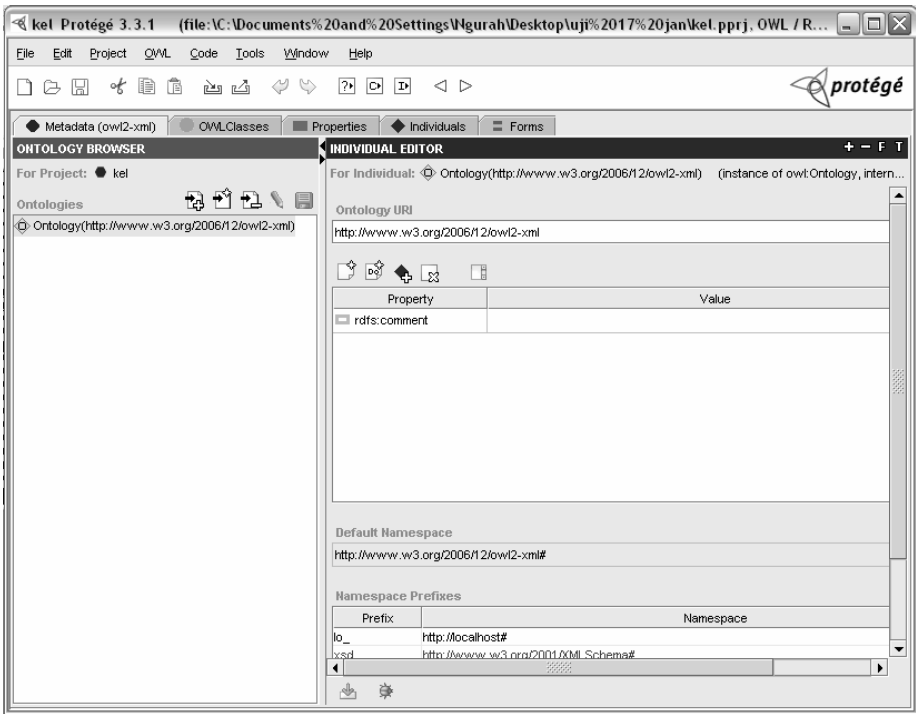


Fig. 3 Snapshot of Protégé

International Science Index, Computer and Systems Engineering Vol:3, No:5, 2009 waset.org/Publication/11368

International Science Index, Computer and Systems Engineering Vol:3, No:5, 2009 waset.org/Publication/11368

IV. Graphical Notations

Our proposed CASE Tool for RDF Modeling is based on the notations introduced from our previous work [12]. There are three base notations, as shown in Fig. 4, are used to represent a statement in RDF/S. The first notation, an ellipse, represents an instance of RDF class, property, or their inheritances. The second notation, a circle, represents a property of a resource, e.g. type, domain, range, subClassOf, subPropertyOf, etc. The property notation has a directed arrow that connects a subject of an RDF statement to an object of the RDF statement. The last notation is a rectangle, which represents a Literal data type.

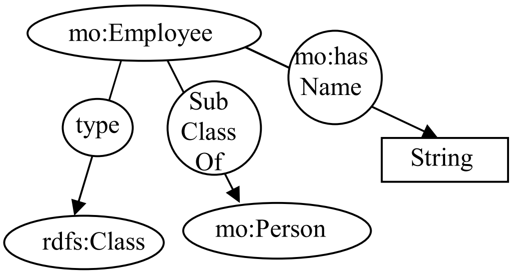


Fig. 4 Basic graphical notations [12]

V. RDFGraph

As illustrated in Fig. 4, a relationship between resources can be modeled. We have an Employee, which is a class. It is a sub class of a Person class. An Employee has a property, hasName, which data type of its value is a String. These basic notations are the main notations that are used in RDFGraph CASE Tool. In addition rules were also developed to allow syntactic verification of a model designed by a user. Given these rules, user can be ensured that he developed a syntactically correct model.

Our system was developed as a desktop application, which uses an existing database management system that support RDF/S as a storage. The system architecture of RDFGraph can be seen in Fig. 5. The architecture is built on three main modules, i.e. Design and Editing Data Model, Model Converter from OWL to graphical model, and from graphical to OWL model.

The Design and Editing Data Model component is a module that acts as an interface for a domain expert to graphically model a domain knowledge or domain problem into a metadata model. This module allows the domain expert to represent his knowledge in a graphical model using the graphical notation introduced in [12]. The output of this module is a graphical metadata file with the extension of gowl.

The first model converter module, that is, Converter from OWL to graphical notations, is a module that allows the domain expert to represent a text-based metadata model, which resides in an text file into its graphical form. The tool can understand both owl as well as rdf text file.

The second model converter module, that is, Converter from graphical notations into OWL file, is a module that allows the domain expert to store its graphical metadata model into an owl file. The stored file has an extension of owl. Aside from owl extension, this tool also allow the user to store in different

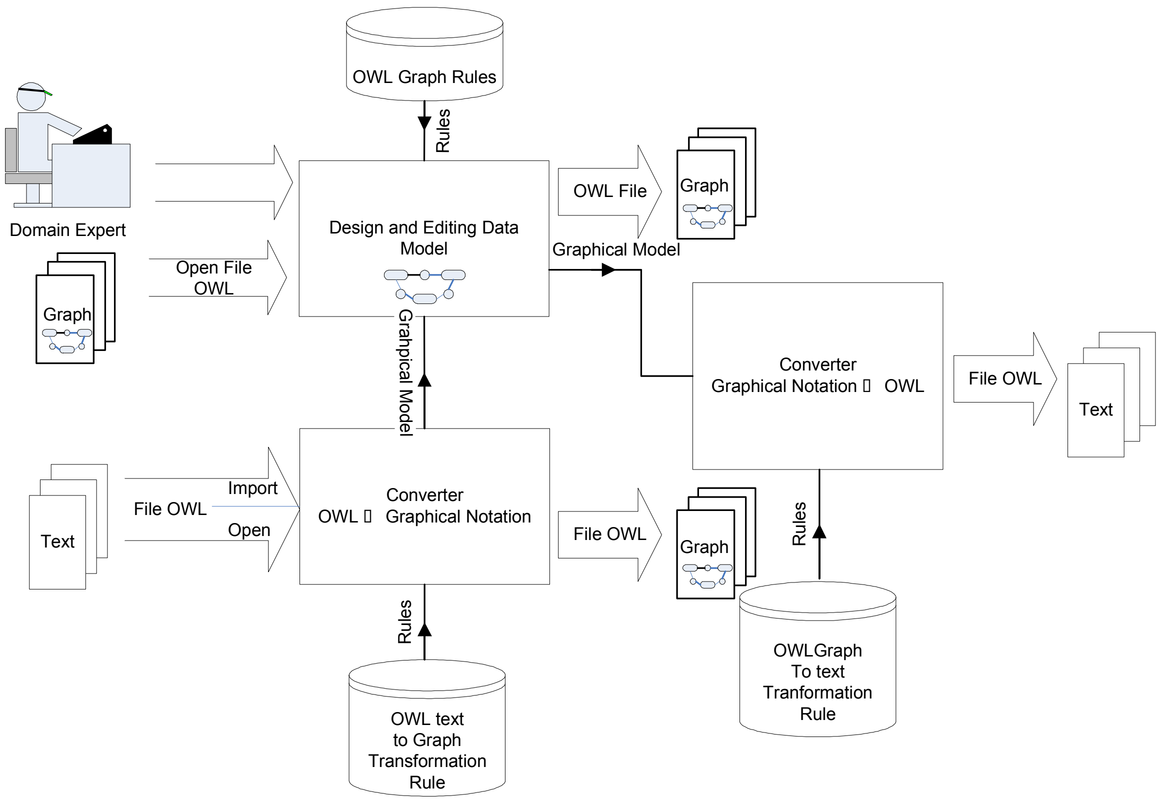


Fig. 5 System Architecture

extension, such as rdf, n-triple, and oracle rdf query format.

There are three basic scenarios, which represent the main functionalities of RDFGraph. In the first scenario, a domain expert can design a new model immediately on the design window using the graphical notations provided. Later, the domain expert can later save his design in a text-based RDF, with an RDF extension file (.rdf) along with its graphical extension (.grdf). In the second scenario, a domain expert can work on an existing model by opening a text based rdf file. Immediately the tool displays a graphical view of the rdf file. Later then, the domain expert can edit and save his updated model. In the last scenario, the domain expert can import a schema located in a local file or from a remote site.

RDFGraph is designed with several functionalities. These functionalities are grouped as follows:

- Modeling functionalities, which allow the domain expert to create and update a graphical metadata model. It includes the following commands:
  - Adding a resource
  - Deleting a resource
  - Editing a resource
  - Save a model
  - Open a model
- Validation, which allows the domain expert to validate a model against the standard RDF and OWL rules defined by W3C. It includes the following features:
  - Validate the model against the RDF rules
  - Suggest correction
- User friendliness, which helps the domain expert to modeling a domain knowledge using a graphical notations. It includes the following features:
  - Zoom in or Zoom out
  - Friendly name
  - Color configuration
- Advance modeling functionalities, which is meant to allow integration with other system, i.e. database management system, search engines, editors, etc. These functionalities will allow other potential application of this tool in software engineering field in the future.It includes the following functionalities:
  - Importing a schema from local or remote site
  - Saving in a graphical format
  - Export the model into N3 format
  - Export the model into a set of queries to be stored in Oracle Spatial Database
  - Compatible with OWL

Since the feature import schema is supported in this case tool, it can be assured that this case tool is also compatible with OWL, as the ontology language. Fig. 6 shows a snapshot of how a problem is modeled in RDFGraph. In the model it can be seen that each resource name is started with namespace. A resource can also have a friendly name, which shorten the original name, i.e. sC stands for rdfs:subClassOf.

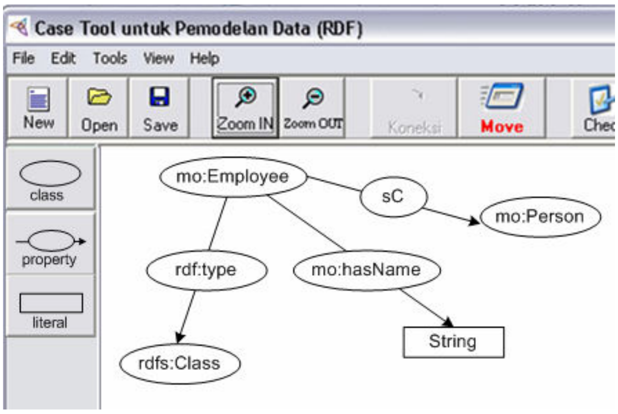


Fig. 6 A sub class relationship is modeled in RDFGraph

VI. Testing and Result

Several tests on the implementation of RDFGraph have been conducted. The goals of the tests are to proof that the rules used in RDFGraph follows the standard set by W3C[3] and to ensure that the model really translated into accepted design as it is in other relational modeling tools, such as Access and Oracle.

For the first goal, a text base RDF recommended by W3C [9] was selected. RDFGraph opens each file and then resaves as a local text base RDF file. Both files (the original and the produced RDF files) were compared for statement similarity. The result shows that both versions of RDF files are similar.

For the second goal, two test case problems were selected, as shown in Fig. 7 and 8. The model was generated using RDFGraph and export it as a set of query for Oracle RDF Spatial database. The model is also generated in a relational database management system, i.e. Microsoft Access. Both models were also compared by running a set of queries on both models. Table I shows the queries used to test the models. Our experimental result shows that the two models, semantic model generated by RDFGraph and relational model, can produce the same results for each query.

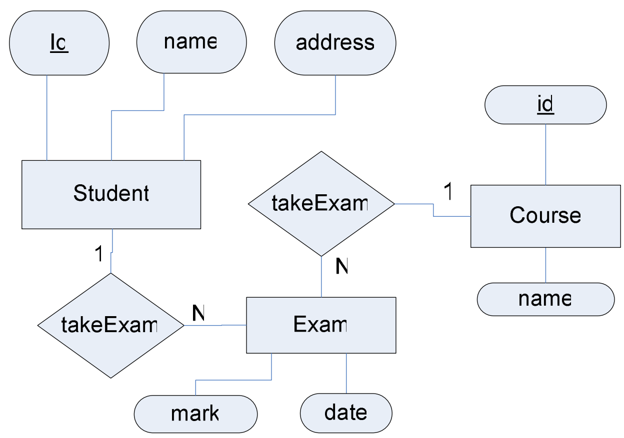


Fig. 7 Student taking examination relationship

International Science Index, Computer and Systems Engineering Vol:3, No:5, 2009 waset.org/Publication/11368

International Science Index, Computer and Systems Engineering Vol:3, No:5, 2009 waset.org/Publication/11368

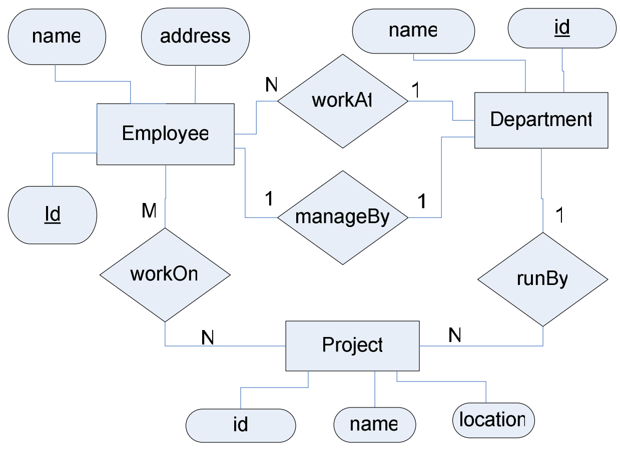


Fig. 8 Employee, Department and Project relations

TABLE I
LIST OF QUERIES FOR EACH DATA MODEL

Data Model	Queries
Student – Course relationship	Find all students who took exam on math course Find student's marks of all courses Find average mark on math course Find all students who took today's exam
Employee-Department-Project relationships	Find employee who works at Department of Finance Find names of managers of each department. Find employee who works on project e-learning Find project done by David. Find employee who works on projects run by Department of IT Find manager whose department run projects done by David

The result of the query executions shows that data models implemented in semantic model (exported in Oracle) and relational model (in MS. Access) are similar. These similarities suggest that the implementation the tool is sufficient as a system to manage semantic database

VII. CONCLUSION

A case tool that can help domain expert to model a problem in a RDF/S and also ontology language such as OWL has been developed. The case tool is not only allowing domain expert to graphically model a problem in a semantic model, but also to create new knowledge on top on existing one (schema) and to store them in Oracle RDF Spatial Database.

There are further works that should be carried out to improve the functionalities of this tool, i.e. how to allow the tool to work directly with RDF/S DBMS, how to check the integrity of the imported schema and query directly on the instances of the schema.

ACKNOWLEDGMENT

The author would like to thank Prof. Toshihiro KITA from Institute for e-Learning Development of Kumamoto University, JICA, through PREDICT-ITS and Kumamoto University for the opportunity and support of our participation in CICIT 2009.

REFERENCES

[1] Berners-Lee, Tim. (2008, November 5). What the Semantic Web Can Represent. [Online]. Available: http://www.w3.org/DesignIssues/RDFnot.html
[2] Bernstein, Orr. (2009, January 2). RDF Editor. CS Department of Rensselaer Polytechnic Institute. Available: http://www.cs.rpi.edu/~puninj/rdfeditor/
[3] CDDB. (2009, January 19). Available: http://www.cddb.com
[4] DAML+OIL. (2008, July 6). http://www.daml.org/2001/03/daml+oil-index.html.
[5] Knowledge Interchange Format. Knowledge System Laboratory. (2008, August 3). Stanford University. Available: http://www-ksl.stanford.edu/knowledge-sharing/kif/
[6] MP3. (2009, January 3). Available: http://www.mp3.com
[7] Music Brainz. (2009, January 3). Available: http://www.musicbrainz.org
[8] Protégé. (2008, December 23). Standford University. Available: http://protege.stanford.edu.
[9] RDF Test-Case. (2008, August 2). W3C. Available: http://www.w3.org/TR/rdf-testcases/
[10] Requirements for a Web Ontology Language. (2008, July 6). W3C WebOnt Working Group. Available: http://www.w3.org/TR/webont-req.
[11] Resource Description Framework. (2008, July 6). W3C. Available: http://www.w3.org/RDF
[12] D. Siahaan. 2006. "Graphical Notations For Semantic Web Language," in *2006 Proc. on ICTS Conf.*, pp.75-80.
[13] XML Topic Maps 1.0. (2008, August 2). Available: http://www.topicmaps.org/xtm

Daniel Siahaan was born in Palembang, Indonesia in 1974.Daniel Siahaan earned his master of science in 2002 in the field of mediamatics from Technische Universiteit Delft, The Netherlands. Daniel Siahaan received his Professional Doctorate in Engineering in 2004 in the field of software engineering from Technische Univesriteit Eindhoven, The Netherlands.

He was working for two years at Philips Research Laboratories, Eindhoven, The Netherlands during his master and doctorate study. He is now working as a lecturer in the field of software engineering at Informatics Department, Institut Teknologi Sepuluh Nopember. His current interest is in the field of semantic web, especially its implementations in e-learning.