Mapping Object Role Modeling 2 Schemes into SROIQ (D) Description Logic

Heba M. Wagih, Doaa S. ElZanfaly, and Mohamed M. Kouta

Abstract—**Ontology engineering is gaining a lot of focus due to the rapid technological development including the emergence of semantic web technology. The idea behind semantic Web is to create a web of data that can easily be shared, accessed and even transformed over a global scale independently of any application or domain. Descriptions logics achieve the goals of semantic web through providing a logical formalism for developed ontologies. SROIQ (D) represents the description logic underlying OWL2. In this paper we present the mapping rules from a rich graphical conceptual modeling technique called ORM2 to SROIQ (D).**

*Index Terms***—ORM2, OWL2, semantic web, SROIQ (D)**

I. Introduction

Semantic Web is the extension of the World Wide Web (WWW) that was invented by Tim Berners-Lee [1]. His vision was to allow data to be presented in a well defined structured form over the web. Berners-Lee defines the Semantic Web as "a web of data that can be processed directly and indirectly by machines". One of the main forces behind the idea of semantic web technology is to facilitate data sharing and integration among different sources and applications. The two main pillars of Semantic web are how to integrate and exchange data from different heterogeneous sources and which language will be used in representing the data over the web. Ontology concept is considered the most promising basis for achieving the Semantic Web goals. A common agreed definition for ontology is a "formal, explicit specification of a shared conceptualization"[2]. In other words we can say that ontology is an approach used for achieving the semantic web goals by providing knowledge for a particular domain of interest over a wider scope in an organized form. Building any ontology is based on some concepts or components regardless the ontology language used.

In 2004, the World Wide Web Consortium (W3C) designed the web ontology language (OWL) and considered it as the standard for building ontologies [3]. OWL was built on the RDF however it provides more constructs for defining properties and classes. W3C introduced three variants of OWL, with different levels of expressiveness which are the OWL DL, OWL Lite and OWL Full. Later In October 2007, W3C working group extend OWL with new features and constraints and named the new OWL version by OWL2.

Five different sublanguages where introduced in OWL2 to provide different expressiveness levels which are OWL2DL, OWL2Full, OWL2EL, OWL2QL and OWL2RL.

Descriptions logics (DL) [4, 5] present an essential step in defining, integrating, and maintaining ontologies. Developing high quality ontology is considered the most important goal of the semantic web that can be achieved by providing a logical formalism for ontologies using Descriptions logics. Semantic Web technology is heavily based on description logics. Knowledge in DL is classified into two parts which are the terminological components and the assertion components. Terminological components (T-Box) are used to represent classes and their properties while assertion component (A-Box) represents individuals and their properties. Different description logics were introduced as ALC, SHIQ, SHOIN, SHIF, SROIQ (D) and others. Since OWL2 is based on the description logic SROIQ (D) thus our study will focus on this type of description logic.

Conceptual modeling techniques represent one of the main data sources in building ontologies. Among the different modeling techniques the one we used in this paper is the ORM2. ORM2 is a powerful modeling technique characterized by its capability of representing most of the business constraints [6]. In this paper we propose an approach for mapping ORM2 to SROIQ description logic. In our approach we set a number of rules to facilitate the transformation of ORM2 schema to SROIQ (D) DL. All of the following diagrams are implemented using NORMA (Neumont ORM Architect) tool [7], which is an open source plug-in to Microsoft Visual Studio .NET. This paper is organized as follows: in Section2 we discuss some of the literature related to our topic, a brief overview for ORM2 language and SROIQ (D) DL will be discussed in section 3 and in Section 4 we introduce the mapping mechanism from ORM2 to SROIQ (D) DL. In the last section, conclusions and future work are discussed.

II. Related Work

Mapping description logics underpinning ontologies gained a lot of focus in the last few years. Many attempts were done to achieve a full mapping between different modeling techniques as ERD, UML and ORM to description logics.

In (2010) J. Ebert, T. Walter [8] proposed an approach to map between the conceptual modeling techniques UML (class diagrams) and OCL (object diagrams) and the description logic. The graph-based semantic description presented by the author in his research shows a high similarity between the concepts of modeling techniques and description logical models; however his mapping methodology lacks some constraints in the modeling

Manuscript received September 8, 2012; revised November 23, 2012.

Heba M. Wagih and Doaa S. ElZanfaly are with the British University in Egypt (e-mail: Heba.wageeh@ bue.edu.eg, Doaa.elzanfaly@ bue.edu.eg).

Mohamed M. Kouta is with the Arab Academy for Science and Technology (e-mail: mmkouta2004@yahoo.com).

DOI: 10.7763/IJCTE.2013.V5.684

techniques used as that of the subclassing constraints. In (2007) M. Jarrar introduced a mapping mechanism from ORM to DLRifd [9]. DLRifd [10] is an expressive description logic suitable for representing any database schema. DLRifd extends the DLR DL by adding identification constraints and functional dependencies in the T-Box component. The author summarized the ORM constructs into 29 construct and he was able to map 27 of these constructs into their corresponding description logic. Later M. Jarrar presented another contribution in mapping ORM to description logics. His new approach based on another description logic called SHOIN [11].SHOIN is the description logic underpinning OWL as recommended by World Wide Web Consortium (W3C). Although the author presented most of the mapping rules between ORM and SHOIN DL, however the SHOIN DL is still uncapable of representing some of the business constraints in ORM as that for n-ary relations and external uniqueness constraints. In (2005) another decent contribution was proposed by D. Berardi and D. Calvanese to present the mapping framework between UML class diagram and DLRifd [12]. They show that every DLRifd knowledge can be expressed as a UML class diagram preserving the completeness of reasoning. Implementing software tools capable of mapping between different conceptual modeling techniques and description logics were also put into consideration. In (2000) E. Franconi and G. Ng implemented a software tool called ICOM [13]. Their tool was designed to map the EER and UML modeling techniques to DLR and DLRifd description logics. In this paper we focus on presenting a mapping framework between ORM2 (second generation of ORM) and SRIOQ description logic. Our research provides the basis for developing a tool capable of mapping most constraints introduced in ORM2 to SROIQ.

III. Overview ORM2 and SROIQ DL

In this section, we present a brief introduction to ORM2 conceptual modeling technique and SROIQ description logic.

A. Object Role Modeling 2

Object Role Model (ORM) is a data modeling technique proposed by Terry Halpin in 1989 [14].ORM has a rich graphical notation capability in representing many business rules and semantic constraints over other modeling techniques. ORM are characterized by being attribute-free diagrams. They simplify compound facts by breaking them into elementary ones. ORM consists of two main blocks which are: objects and relationship. Entities and attributes are treated as objects that are related to each other using different types of relationships. ORM allows unary, binary, as well as n-ary facts. Also ORM supports the natural language verbalization [15] which allows the participation of end user in database development which is not provided in many other modeling techniques. ORM2 is the second generation of ORM which was later released in [6].

ORM2 added more expressivity and flexibility in representing business constraints. ORM2 substituted the English language symbols used in ORM with graphical symbols to improve the simplicity in the diagrams. It also introduced new constraints as role value constraint and objectified unaries.

B. Expressive Description Logic SROIQ (D)

Description logics (DLs) [4, 5] are the most effective knowledge representation formalism that provides a logical basis for presenting knowledge of a particular domain in an expressive well understood structure. Various description logics were proposed as DLR, DLRifd, SHOIN, SHOIQ and SROIQ. SROIQ [16] is the description logic underpinning the current web ontology language OWL2 as recommended by W3C.

SROIQ is the extension of the expressive SHOIN DL. Although SHOIN had proved its expressive power, however it lacks the ability of presenting some important business constraints as the qualified number restrictions. SROIQ presented some important rules that enrich the expressive capability of description logics with: Among these new rules are the disjoint roles where it permits the disjointness constraint between both classes and roles played by these classes. Also it introduced other important aspects as the negated role assertions and the reflexive and irreflexive roles. A full explanation for SROIQ description logic is presented in [16].

IV. Mapping Framework From Orm2 Concepts to SROIQ Description Logic

Description logics are used to provide a logical formalism for knowledge presented in any application domain specifically the semantic web applications. Integrating the relational databases supporting the Web applications into the Semantic Web requires mapping the associated schemas into equivalent description logic that underpin the ontology language used. The goal of this paper is to introduce an easy and understandable approach for mapping the ORM2 constructs to SROIQ DL syntax.

Table I presents the corresponding constructs between ORM2 and SROIQ. The set of mapping rules from ORM2 constructs to SROIQ are fully presented in the following subsections

A. Mapping Predicates from ORM2 to SROIQ

Roles played between two or more entity types are known as predicate. A predicate may be a unary type, binary, ternary or even n-ary predicate. SROIQ does not provide a mean to present the predicate in terms of Domain and Range constructs; however it can define two predicates inverse to each other as shown in Fig. 1.

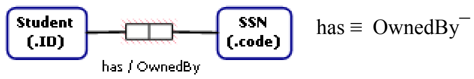


Fig. 1. Mapping inverse roles from ORM2 to SROIQ.

B. Mapping Object Constraints from ORM2 to SROIQ

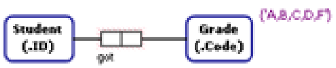
ORM2 has a number of constraints that is applied on its object types as value constraints and subtyping constraints. In the following subsections we will introduce these constraints together with their translation to SROIQ description logic.

TABLE I: CORRESPONDENCE BETWEEN SOME CONSTRUCTS OF ORM2 AND SROIQ.

ORM2 Construct	SROIQ(D) Construct
Inverse Object Property	$R \equiv R1^{-}$
Value Constraint	$C \equiv \{A, B, C\}$
Subtyping process	$C1 \sqsubseteq C$
Exclusion constraint between subclasses	$C1 \sqcap C2 \sqsubseteq \bot$
Inclusive Or constraint between subclasses	$C \equiv C1 \sqcup C2$
Exclusive Or constraint between subclasses	$C1 \sqcap C2 \sqsubseteq \bot$ $C \equiv C1 \sqcup C2$
Inclusive Or constraint between single roles	$C \equiv \exists R.C1 \sqcup \exists R1.C2$
Exclusive Or constraint between single roles	Dis($R1$, $R2$) $C \sqsubseteq \exists$ R.C1 $\sqcup \exists R1.C2$
Internal Uniqueness constraint	$C1 \sqsubseteq \leq 1\ R.C2$
Mandatory constraint	$C1 \equiv \exists R.C2$
Subset constraint between pair of roles	$R1 \sqsubseteq R2$
Equality constraint between pair of roles	$R1 \equiv R2$
Exclusion constraint between pair of roles	Dis(R, R1)
Internal Frequency constraints on a single roles	$C1 \equiv \leq n\ R.C2$
Reflexive Ring constraints	Ref(R) or $C \equiv \exists$R.Self
Irreflexive Ring constraints	Irr(R)
Symmetric Ring constraints	Sym(R)
Asymmetric Ring constraints	Asy(R)

1) Value constraint

Value constraints are value restrictions added to either the attached object type or the value type itself. Value constraints may be a list, enumeration of values, or a combination of lists and ranges. Value constraints imply stable values otherwise the schema will be continuously in changing state. Fig. 2 shows an example of mapping a value constraint added to entity type to SROIQ.



Grade $\equiv \{A, B, C, D, E\}$

Fig. 2. Mapping value constraint from ORM2 to SROIQ.

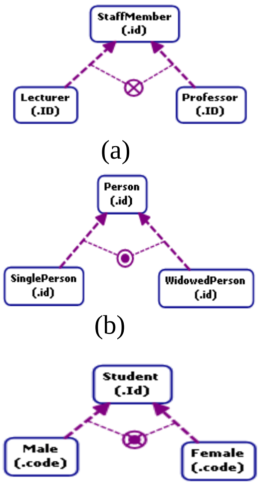
2) Subtyping

A class is considered a subtype of another class if the population in the subtype is subset of the population in the supertype. Different types of constraints are applied to Subtyping as:

- Exclusive (disjoint): superclass instance may belong to at most one of the subclasses and the subclasses are disjoint. Exclusive constraint is mapped to SROIQ as shown in Fig. (3.a).
- Inclusive-Or (total): superclass instance belongs to at least one of the subclasses and is mapped to SROIQ as shown in Fig. (3.b).
- Exclusive-Or (Partition): superclass instance must belong to at most one of the subclasses and the subclasses are disjoint and is mapped to SROIQ as shown in Fig. (3.c).

C. Mapping Role Constraints from ORM2 to SROIQ

A number of constraints are applied to role facts as uniqueness, mandatory, set comparison, frequency, exclusion and ring constraints. In the following subsections we will introduce the mapping of these constraints to their equivalent SROIQ constructs.



(a) Lecturer $\sqcap$ Professor $\sqsubseteq \bot$

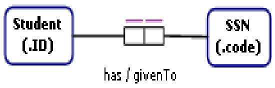
(b) Person $\equiv$ Simple Person $\sqcup$ Widowed Person

(c) Male $\sqcap$ Female $\sqsubseteq \bot$
Student $\equiv$ Male $\sqcup$ Female

Fig. 3. Mapping subtyping constraints from ORM2 to SROIQ.

1) Internal uniqueness constraint (IUC)

Internal uniqueness constraints are used to represent the one-to-one, one-to-many, and many-to-many relationships. Fig. 4 shows an example of mapping IUC to SROIQ DL.



Student $\sqsubseteq \leq 1$ has.SSN
SSN $\sqsubseteq \leq 1$ given to. Student

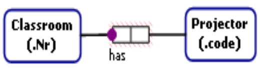
Fig. 4. Mapping 1: 1 relationship from ORM2 to SROIQ.

2) External uniqueness constraints

No construct in SROIQ syntax is capable of translating the external uniqueness constraint in ORM.

3) Mandatory constraints

Mandatory constraint implies that each instance in the population must participate in the role fact. Mandatory constraint is mapped to SROIQ using a property restriction called existential quantification which states that each instance in a class is connected through a property to another instance in another class. Fig. 5 shows the mapping of mandatory constraint to SROIQ DL.



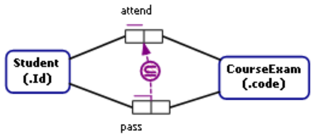
Classroom $\equiv \exists$ has. Projector

Fig. 5. Mapping mandatory constraint from ORM2 to SROIQ.

4) Set -comparison constraints

Set-comparison constraints define how the population of one role (or a number of roles) related to another role (or a number of roles). There are three types of set-comparison constraint which are subset constraint, equality and exclusion constraint.

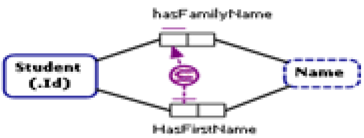
Subset_Constraint_implies that for a certain business domain, the population of one role may be a subset of the population of second role. Subset constraint is applied to single roles as well as a sequence of roles. In Fig. 6 we show an example of mapping subset constraint applied to a sequence of roles in ORM2 to SROIQ.



Pass ⊑ attend

Fig. 6. Mapping subset constraints from ORM2 to SROIQ.

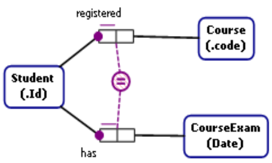
TO map the subset constraint between single roles from ORM2 to SROIQ DL, we define the first role population as a subclass of the second role population as illustrated in Fig. 7.



∃Has first name. Name ⊑∃Has family name. Name

Fig. 7. Mapping subset constraints between single roles from ORM2 to SROIQ.

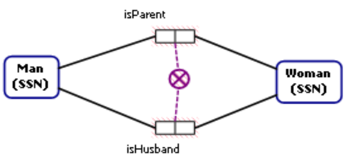
Equality Constraint implies that for a certain business domain, the population of one role must be equal to the population of the second role. Equality constraint can be substituted by two subset constraints in opposite directions. Mandatory constraint must be applied to the roles of the relation before using the equality constraints. If one role is mandatory and the other is optional, then equality constraints can not be applied. Fig. 8 shows an example of mapping equality constraint between a pair of roles to SROIQ.



has ≡ registered

Fig. 8. Mapping equality constraints from ORM2 to SROIQ.

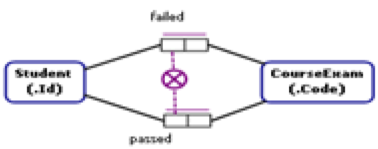
Exclusion Constraint between single roles implies that an instance can participate in one role only. Fig. 9 shows the mapping of exclusion constraint between a pair of roles is mapped to SROIQ.



Dis (is Husband, is Parent)

Fig. 9. Mapping exclusion constraint from ORM2 to SROIQ.

Mapping the exclusion constraint between single roles requires getting the complement of the intersection between these single roles as shown in Fig. 10.

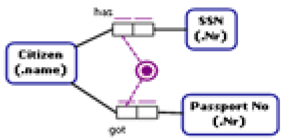


Student ⊑ ¬(∃ failed. Course Exam ∏ ∃Passed. Course Exam)

Fig. 10. Mapping exclusion constraint between single roles from ORM2 to SROIQ.

5) Inclusive or constraint

Inclusive or Constraint (also known as disjunctive mandatory constraint) implies that an object instance must at least participate in one (possibly all) of the associated roles. Inclusive or Constraint is mapped to SROIQ as shown in Fig. 11.

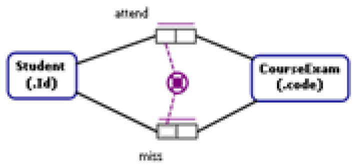


Citizen ≡ ∃ has. SSN ⊔∃ got. Passport no

Fig. 11. Mapping inclusive-or constraint from ORM2 to SROIQ.

6) Exclusive-or constraint

Exclusive or Constraint implies that an object instance must at least participate in one of the attached roles and that these fact type roles are disjoint. Exclusive or Constraint is a combined Inclusive-Or and Exclusion constraints. Fig. 12 shows the mapping of Exclusive or Constraint between fact type roles to SROIQ.



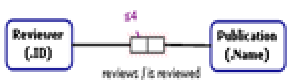
Dis (miss, attend)
Student ⊑ ∃attend. Course Exam ⊔∃miss. Course Exam

Fig. 12. Mapping exclusive or constraint between single roles fromORM2 to SROIQ.

7) Frequency constraints

Frequency constraint added to any role specifies the number of occurrences of this role by its object type. Frequency constraint is considered a generalized form of uniqueness constraints. Frequency constraints are either internal or external.

Internal Frequency Constraint is added to a single role or multiple roles in the same predicate. It is translated to SROIQ using the (≥, ≤) constructs as shown in Fig. 13.



Reviewer ≡ ≤ 4 reviews.Publication

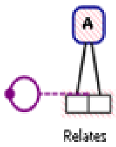
Fig. 13. Mapping internal frequency constraint over single role from ORM2 to SROIQ

External Frequency Constraint is applied to roles from two or more different predicates. No SROIQ constructs are available to map this constraint.

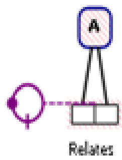
8) Ring constraints

Ring constraint is defined in [17] as “When two roles in a predicate are played by the same object type, the path from the object type through the role pair and back to the object type forms a ring”. A number of constraints can be applied to this ring. These constraints are:

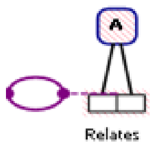
- Reflexive: relation on a set where an element can relate to itself. Fig. 14.(a) shows an example of mapping Reflexive constraint to SROIQ.
- Irreflexive: relation on a set where no element is related to itself. Fig. 14.(b) shows an example of mapping Irreflexive constraint to SROIQ.
- Symmetric: if and only if for each relationship instance, the inverse relationship also holds. Fig. 14.(c) shows an example of mapping Symmetric constraint to SROIQ.
- Asymmetric: if a relationship holds then its inverse cannot hold. Fig. 14.(d) shows an example of mapping Asymmetric constraint to SROIQ.
- Antisymmetric: if a relationship holds between non-identical objects then its inverse cannot hold
- Acyclic: no cycles of any length are allowed.
- Intransitive: if a first object bears the relationship to a second, and the second bears the relationship to a third, then the first cannot bear the relationship to the third.



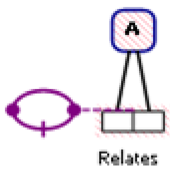
Ref (Relates) or
A≡ ∃Relates.Self
(a)



Irr (Relates)
(b)



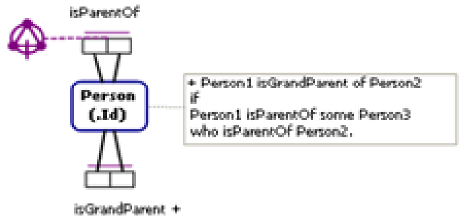
Sym (Relates)
(c)



Asy (Relates)
(d)

Fig. 14. Ring constraints in SROIQ.

No available SROIQ constructs can directly map the acyclic and intransitive ring constraint; however both of these constraints are used together to represent the property chain feature in SROIQ. Property chain feature is mainly useful in some domains as in representing family relations. Property chain is used to restrict the population of certain relation to a chain of fixed number of properties as in isGrandParent/ isParentOf relation. In isGrandParent / isParentOf relation a property chain is used to represent the population of isGrandParent as all individuals that are linked by a chain of exactly two isParentOf properties. For class person, we assume we have three instances P_1, P_2 and P_3. So if (P_1) isParentOf (P_2) and (P_2) isParentOf (P3), then P_1 isGrandParent P3 (forward chain is allowed and is presented using the intransitive ring constraint). The acyclic ring characteristic is also used to restrict that no backward chain will occur from P_3 to P_1. Derivation rule will be then used to define the relation between isParentOf and isGrandParent properties where isGrandParent property will be derived from isParentOf property. Fig. 15 shows an example of mapping acyclic and intransitive ring constraint from ORM2 to SROIQ DL.



Is parent o is parent ⊑ is grandparent

Fig. 15. Mapping subproperty chain construct from ORM2 to SROIQ.

D. Mapping Objectification Constrains from ORM2 to SROIQ

Objectification is the process of transforming a relationship between objects into a new object thus it can be defined as a nested object type. Objectification process usually requires at least two roles with either single (1: *n*), spanning (m: n) or (1:1) uniqueness constraint as shown in Fig. 16. No Avaialbe SROIQ constructs to map this constraint.

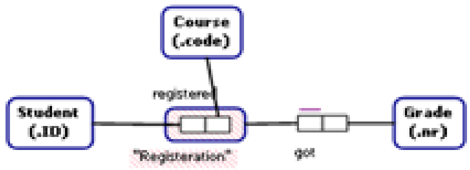


Fig. 16. Objectification in ORM2

V. Conclusions and Future Work

Ontology is Description logics present knowledge for any application domain in a logical formalism specifically for the semantic web applications. Among different DL languages, the one we used in our research is the SROIQ. SROIQ provides the solution for mapping most of the ORM2 constraints as ring constraints and disjoint role properties.

SROIQ still lacks the capability of mapping primary keys and n-ary relations where $n > 2$. Several attempts were presented to narrow the gap between ORM2 and DL languages mainly the DLRifd and SHOIN. In this paper we present a new approach to map between ORM2 to SROIQ description logic. Our research is considered the basis for implementing DL-based reasoning tools that supports knowledge acquisition through ORM2 modeling technique. Our future work is mainly concerned in implementing a tool capable of mapping between ORM2 and SROIQ constructs.

References

[1] T. B. Lee and M. Fischetti, "Weaving the web: The past, present and future of the World Wide Web by its inventor," *Britain: Orion Business*, 1999.

[2] T. Gruber, "A translation approach to portable ontology specifications," in *Knowledge Acquisition*. vol. 5, pp. 199-220, 1993.

[3] S. Bechjofer, F. V. Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. P. Schneider, and L. A. Stein, "OWL web ontology language reference W3C recommendation 10," *W3C Working Group*, 2004.

[4] F. Baader, D. Calvanese, D. M. Guinness, D. Nardi, and P. F. P. Schneider, *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2003.

[5] F. Baader and U. Sattler, "An overview of tableau algorithms for description logics," *Studia Logica*, vol. 69, pp. 5-40, 2001.

[6] T. Halpin, "ORM 2," *On the Move to Meaningful Internet Systems* 2005: OTM 2005 Workshops, eds.

[7] R. Meersman, Z. Tari, P. Herrero *et al.*, *Springer LNCS* 3762, pp. 676-687, 2005.

[8] NORMA: Neumont ORM Architect for Visual Studio. [Online]. Available: http://sourceforge.net/projects/orm.

[9] E. Jürgen and W. Tobias, "Interoperability services for models and ontologies," in *Databases and Information Systems VI. IOS Press*. *Bd*. vol. 224, 2010.

[10] M. Jarrar, "Towards automated reasoning on ORM schemes. -mapping ORM into the DLR idf description logic," in *Proc. 26th International Conference on Conceptual Modeling*, Auckland, New Zealand, 2007, pp. 181-197.

[11] C. M. Keet, "Prospects for and Issues with Mapping the object-role modeling language into DLRifd," in *Proc. 20th International Workshop on Description Logics*, Bressanone, Italy, vol. 250, pp. 331-338, 2007.

[12] M. Jarrar, "Mapping ORM into the SHOIN/OWL description logic- towards a methodological and expressive graphical notation for ontology engine," in *Proc. on the Move Federated Conferences and Workshops*, Portugal, 2007, vol. 4805, pp.729-741.

[13] D. Berardi, D. Calvanese, and G. De Giacomo. "Reasoning on UML class diagrams," *Artificial Intelligence*, 2005.

[14] F. Franconi and G. Ng. "The ICOM tool for intelligent conceptual modeling," *7th Workshop on Knowledge Representation Meets Databases (KRDB'00)*, 2000.

[15] T. A. Halpin, "Object-role modeling (ORM/NIAM)," *Handbook on Architectures of Information Systems,* Springer, Heidelberg, pp. 81-103, 2006.

[16] M. Jackson, "Object role modelling and conceptual database design," *Proc. Visual Developer's Academy*, Oxford, 1996.

[17] I. Horrocks, O. Kutz, and U. Sattler, "The even more irresistible SROIQ," in *Proc. 10th International Conference on Principles of Knowledge Representation and Reasoning*, Lake District, UK, 2006, pp.57-67.

[18] T. Halpin, *Information Modeling and Relational Databases: From Conceptual Analysis to Logical Design*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001.

Heba M. Wagih received the B.S. degree in Informatics and Computer Sciences from Ain Shams University in 2003. She received the M.S. degree in Information Systems from the Arab Academy for Science, Technology and Maritime Transport in 2012. All of her researches are concerned with the semantic web technologies and conceptual modeling techniques.

Doaa S. Elzanfaly received the B.S. degree in Management and Information Systems from .Sadat Academy for Management Sciences. She received the M.S. and Ph.D. degrees in Information Systems from Helwan University and Connecticut University in 2000 and 2004, respectively. She is a lecturer in Faculty of Informatics & Computer Science, British University in Egypt (BUE), Information Systems Department.

Mohamed Kouta received the B.S. degree in Electrical Engineering from Military Technical College in 1972. He received the M.S. and Ph.D. degrees in computer science from Jons Hopkins University and Clarkson University in 1982 and 1985, respectively. He is the Dean for college of management and technology, Arab academy for science and technology (AAST) (Cairo Branch).