

DL-Based Framework for Mapping Between Object Role Modeling 2 Schemes and OWL2 Ontologism

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. In order to integrate data into the Semantic Web, the associated database schemas should be translated into equivalent ontologism. Since Descriptions logics provide a logical formalism for developing ontologism, thus it is considered the basis for our mapping framework between OWL2 and ORM2.

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. Many ontological languages were proposed to allow data representation on the web as Cycl, LOOM, RDFS and others.

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 were introduced in OWL2 to provide different expressiveness levels which are OWL2DL, OWL2Full, OWL2EL, OWL2QL and OWL2RL. The OWL2 language has different syntaxes as RDF, OWL2XML, Manchester and RDF/XML syntax. Among these different syntaxes, the one used in this paper is the Manchester syntax [4], [5]. Manchester syntax is a user-friendly syntax for OWL2 ontologies; it is easy to write, read and edit ontologies using Manchester syntax.

Descriptions logics (DL) [6, 7] 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 [8]. In this paper we propose an approach for mapping ORM2 to Manchester syntax based on SROIQ description logic. In our approach we set a number of rules to facilitate the transformation of ORM2 schema to Manchester syntax. All of the following diagrams are implemented using NORMA (Neumont ORM Architect) tool [9], 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 modeling technique, OWL2 language and SROIQ (D) DL will be discussed in section 3 and in Section 4 we introduce the mapping mechanism from ORM2 to OWL Manchester Syntax based on SROIQ (D) DL. In the last section, conclusions and future work are discussed.

II. RELATED WORK

Extracting ontology from different modeling schemas gained a lot of focus in the last few years. Many attempts

Manuscript received August 1, 2011; revised September 28, 2011.

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

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

were done to achieve a full schema matching between different modeling techniques as ERD, EERD, UML and ORM to ontological language. Some of these attempts were based on the concept of description logics.

In (2010) J. Ebert, T. Walter [10] proposed an approach to map between two different conceptual modeling techniques which are 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 techniques used as that of the subclassing constraints.

In (2009) I. Myroshnichenko and M. C. Murphy introduced another approach for mapping ER Schemas to OWL Ontologies [11]. Their approach was so close to the work done by Upadhyaya and Fahad [11] in 2008; however their focus was based on the ERD schema without mentioning the new features

In (2007) M. Jarrar introduced a mapping mechanism from ORM to DLRifd [9]. DLRifd [12] 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 authors 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 [13]. 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 incapable of representing some of the business constraints in ORM as that for n-ary relations and external uniqueness constraints.

Also in (2007) D. B. Bach introduced an approach to Import and Export OWL ontologies into and from DOGMA [14] using description logics, his approach was based on the first generation of ORM disregarding many of the new features which was previously introduced in ORM2[6] as duplicated object types, independent object types and role value constraints. Also the author in his research did not cover some of the important aspects in translating ORM to OWL as Ring constraints and objectification

In (2005) another decent contribution was proposed by D. Berardi and D. Calvanese to present the mapping framework between UML class diagram and DLRifd [15]. 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 [16]. Their tool was designed to map the EER and UML modeling techniques to DLR and DLRifd description logics.

In (2004) R. Colomb [17] proposed an approach to map between different data models as UML and ER; also he introduced a set of rules to map these models to OWL

ontologies. His approach focused on the syntax conversion between data models disregarding the semantics. In (2005) Upadhyaya and Kumar [18] proposed another approach for mapping extended ER models to OWL ontologies.

In this paper we focus on presenting a mapping framework between ORM2 (second generation of ORM) and OWL2 web ontology language based on SROIQ description logic. Our research provides the basis for developing a tool capable of mapping most constraints introduced in ORM2 to OWL2.

III. OVERVIEW FOR ORM2 , OWL2 AND SROIQ DL

In this section, we present a brief introduction to ORM2 conceptual modeling technique, OWL2 language and SROIQ description logic.

TABLE I: CORRESPONDENCE BETWEEN SOME CONSTRUCTS OF ORM2,SROIQ (D) AND OWL2

ORM2	SROIQ(D)	OWL2
Inverse Object Property	$R \equiv R^{-1}$	InverseOf
Value Constraint	$C \equiv \{A, B, C\}$	EquivalentTo
Subtyping process	$C1 \sqsubseteq C$	SubClassOf
Exclusion constraint between subclasses	$C1 \sqcap C2 \sqsubseteq \perp$	DisjointClasses
Inclusive Or constraint between subclasses	$C \equiv C1 \sqcup C2$	Union construct "OR"
Exclusive Or constraint between subclasses	$C1 \sqcap C2 \sqsubseteq \perp$ $C \equiv C1 \sqcup C2$	DisjointUnionOf
Inclusive Or constraint between single roles	$C \equiv \exists R.C1 \sqcup \exists R1.C2$	Union construct "OR"
Exclusive Or constraint between single roles	$Dis(R1, R2)$ $C \sqsubseteq \exists R.C1 \sqcup \exists R1.C2$	DisjointProperties SubClassOf Union construct "OR"
Internal Uniqueness constraint	$C1 \sqsubseteq \leq 1 R.C2$	Functional characteristic or cardinality restrictions
Mandatory constraint	$C1 \equiv \exists R.C2$	Some
Subset constraint between pair of roles	$R1 \sqsubseteq R2$	SubPropertyOf
Equality constraint between pair of roles	$R1 \equiv R2$	EquivalentTo
Exclusion constraint between pair of roles	$Dis(R, R1)$	DisjointWith
Internal Frequency constraints on a single roles	$C1 \equiv \leq n R.C2$	Cardinality restrictions
Reflexive Ring constraints	Ref(R) or $C \equiv \exists R.Self$	Reflexive, characteristic
Irreflexive Ring constraints	Irr(R)	Irreflexive characteristic
Symmetric Ring constraints	Sym(R)	Symmetric characteristic
Asymmetric Ring constraints	Asy(R)	Asymmetric characteristic

A. Object Role Modeling 2

Object Role Model (ORM) is a data modeling technique proposed by Terry Halpin in 1989 [19]. 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 [20] 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 [8].

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. OWL 2 Web Ontology Language

Ontology is a promising technology that supports the goals of the semantic web. It allows data exchange over a wide range independently from any application domain. Among the different ontology languages that were proposed, we choose the OWL as the standard for our work. OWL is a semantic markup language for publishing and sharing ontologies on Web.

Two versions of OWL language were introduced; OWL1 was released in 2004 [3] and was considered the standard for building ontologies as recommended by the World Wide Web Consortium (W3C). OWL1 presents information in the form of classes and properties. Properties are either DataProperty or ObjectProperty. More constructs were introduced in OWL1 as that for representing the relation between classes (DisjointWith) and the cardinality restrictions. Also OWL1 presents some property characteristics as Transitive and Symmetric Properties. Later a second version of OWL was introduced inheriting OWL1 constructs and adding new features to it. Among the new features that were added are DisjointUnion, DisjointClasses, Reflexive, Irreflexive, and Asymmetric Object Properties, Property Chain Inclusion and the keys.

C. Expressive Description Logic SROIQ (D)

Description logics (DLs) [6, 7] 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, DLR_{ifd}, 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 [21].

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 ontologies which are achieved through using the description logic underpinning the ontology language used. The goal of this paper is to introduce an easy and understandable approach for mapping the ORM2 constructs to OWL Manchester syntax. Table 1 presents the corresponding constructs between ORM2, OWL2 and SROIQ. The set of mapping rules between ORM2 constructs, OWL2 and SROIQ are fully presented in the following subsections..

A. Mapping Predicates from ORM2 to OWL

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. Roles are mapped to OWL2 using the constructs ObjectProperty, InverseOf, Domain and Range. SROIQ DL 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).

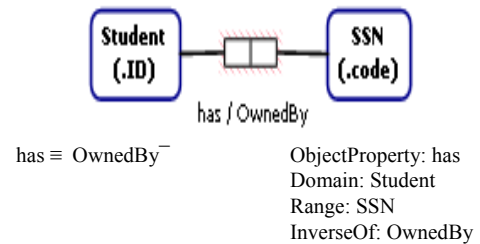


Fig. 1. Mapping predicates from ORM2 to OWL2 based on SROIQ

B. Mapping Object Constraints from ORM2 to OWL

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 and OWL2 web ontology language.

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 both SROIQ and OWL2.

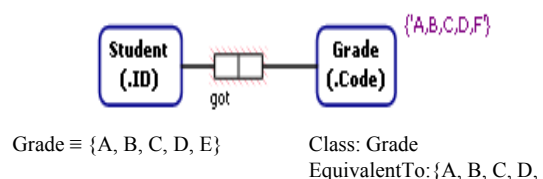


Fig. 2. Mapping Value Constraint from ORM2 to OWL2 based on SROIQ

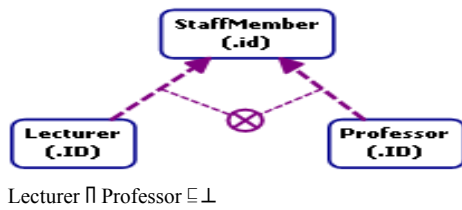
Verbalizing Value constraint is represented as follows:

The possible value of Grade_Code in Grade has Grade_Code is 'A, B, C, D, F'

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:

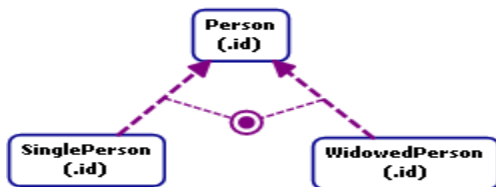
- Exclusive (disjoint): superclass instance may belong to at most one of the subclasses and the subclasses are disjoint. Exclusive constraint is mapped to both SROIQ and OWL2 as shown in Fig (3.a).
- Inclusive-Or (total): superclass instance belongs to at least one of the subclasses and is mapped to both SROIQ and OWL2 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 both SROIQ and OWL2 as shown in Fig (3.c).



Lecturer \sqcap Professor $\sqsubseteq \perp$

Class: Lecturer
 SubClassOf: StaffMember
 Class: Professor
 SubClassOf: StaffMember
 DisjointClasses: Lecturer, Professor

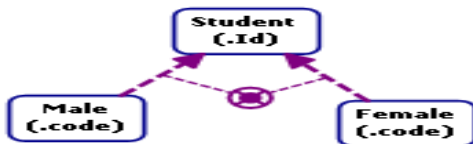
(a)



Person \equiv SimplePerson \sqcup WidowedPerson

Class: SinglePerson
 SubClassOf: Person
 Class: WidowedPerson
 SubClassOf: Person
 Class: Person
 EquivalentTo: SinglePerson or WidowedPerson

(b)



Male \sqcap Female $\sqsubseteq \perp$
 Student \equiv Male \sqcup Female

Class: Male
 SubClassOf: Student
 Class: Female
 SubClassOf: Student
 Class: Student
 DisjointUnionOf: Male, Female

(c)

Fig. 3. Mapping Subtyping Constraints from ORM2 to OWL2 based on SROIQ

Exclusive constraint between subtypes is verbalized as follows:

For each StaffMember, at most one of the following holds:
that StaffMember is some Professor;
that StaffMember is some Lecturer.

Inclusive OR constraint between subtypes is Verbalized as follows:

Each Person is some WidowedPerson or is some SinglePerson.
Each SinglePerson is an instance of Person.
Each WidowedPerson is an instance of Person.

Exclusive OR constraint between subtypes is Verbalized as follows:

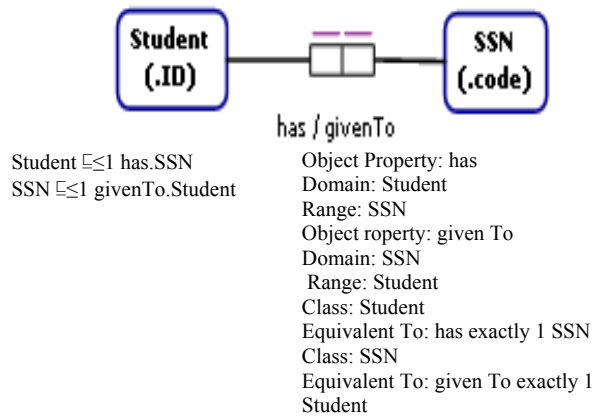
For each Student, exactly one of the following holds:
that Student is some Female;
that Student is some Male.

A. Mapping Role Constraints from ORM2 to OWL

A number of constraints are applied to role facts as uniqueness, mandatory, set comparison, frequency, and exclusion and ring constraints. In the following subsections we will introduce the mapping of these constraints to their equivalent OWL2 and SROIQ constructs.

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 both OWL2 and SROIQ DL.



Student ≤ 1 has.SSN
 SSN ≤ 1 givenTo.Student

Object Property: has
 Domain: Student
 Range: SSN
 Object property: given To
 Domain: SSN
 Range: Student
 Class: Student
 Equivalent To: has exactly 1 SSN
 Class: SSN
 Equivalent To: given To exactly 1 Student

Fig. 4. Mapping 1: 1 Relationship from ORM2 to OWL2 based on SROIQ

Verbalizing Internal Uniqueness Constraint (1:1) is represented as follows:

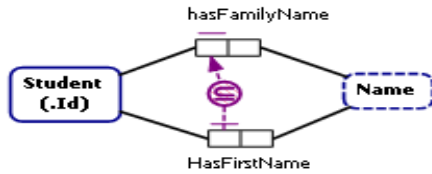
Student has SSN.
Each Student has at most one SSN.
Each SSN givenTo at most one Student.

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 OWL2 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 both OWL2 and SROIQ DL.



Classroom $\equiv \exists$ has. Projector
 Class: Classroom
 Equivalent To: has some Projector

Fig. 5. Mapping Mandatory Constraint from ORM2 to OWL2 based on SROIQ

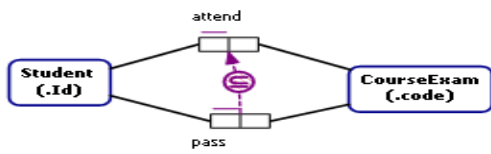
Verbalizing Mandatory Constraint is represented as follows:

EachClassroom has someProjector.

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 both OWL2 and SROIQ DL.



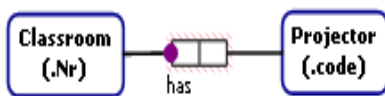
Pass \sqsubseteq attend

ObjectProperty: attend
 Domain: Student
 Range: CourseExam
 ObjectProperty: pass
 SubPropertyOf: attend
 Domain: Student
 Range: CourseExam

Fig. 6. Mapping Subset Constraints from ORM2 to OWL2 based on SROIQ

Verbalizing Subset Constraint between a set of binary predicates is represented as follows:

If someStudent pass someCourseExam then thatStudent attend thatCourseExam.



\exists HasFirstName.name \sqsubseteq \exists HasFamilyName. Name
 Class: Student
 EquivalentTo: HasFirstName some name
 SubClassOf: HasFamilyName some name

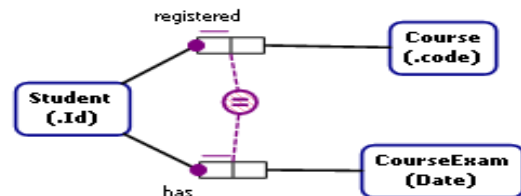
Fig.7.Mapping Subset Constraints between single roles from ORM2 OWL2 based on SROIQ

TO map the subset constraint between single roles from ORM2 to OWL2, we define the first role population as a subclass of the second role population as illustrated in Fig (7).

Verbalizing Subset Constraint between a set of binary predicates is represented as follows:

If someStudentHasSecondNamesomename then thatStudentHasFirstNamesomename.

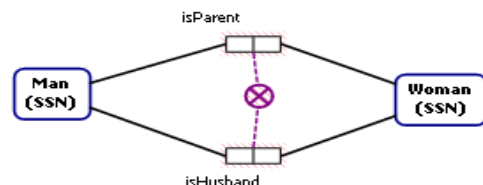
Equality Constraint implies that for a certain business domain, the population of one role must be equal to the population of the second role. Equality constraint can be substituted by two subset constraints in opposite directions. Mandatory constraint must be applied to the roles of the relation before using the equality constraints. If one role is mandatory and the other is optional, then equality constraints cannot be applied. Fig (8) shows an example of mapping equality constraint between a pair of roles to both SROIQ and OWL2.



Has \equiv registered

ObjectProperty: registered?
 Domain: Student
 Range: Course
 ObjectProperty: has
 EquivalentTo: registered
 Domain: Student
 Range: CourseExam

Fig. 8. Mapping Equality Constraints from ORM2 to OWL2 based on SROIQ



Dis (isHusband, isParent)

ObjectProperty: isHusband
 Domain: Man
 Range: Woman
 ObjectProperty: isParent
 Domain: Man
 Range: Woman
 DisjointWith: isHusband

Fig. 9. Mapping Exclusion Constraint from ORM2 to OWL2 based on SROIQ

Verbalizing Exclusion Constraint between different predicates is represented as follows:

For eachMan and Woman, at most one of the following holds:

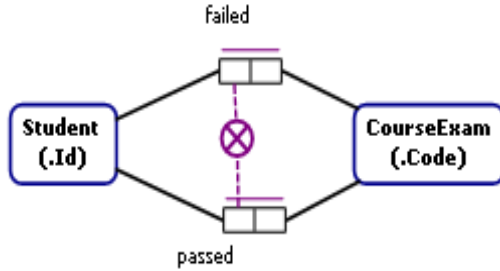
**thatManisParentthatWoman;
 thatManishusbandthatWoman.**

Verbalizing Equality Constraint between a set of binary predicates is represented as follows:

For each Student and CourseExam,
that Student has that CourseExam if and only if
that Student registered some Course that is that CourseExam.

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 both SROIQ and OWL2.

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



Student $\sqsubseteq \neg (\exists \text{ failed. CourseExam} \sqcap \exists \text{ passed. CourseExam})$
 Class: Student
 SubClassOf: not (failed some CourseExam and passed some CourseExam)

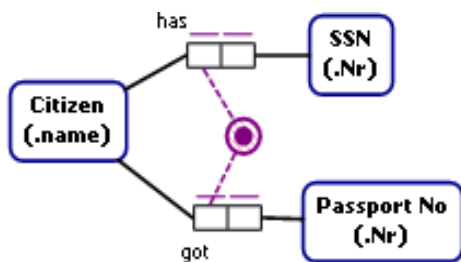
Fig. 10. Mapping Exclusion Constraint between single roles from ORM2 OWL2 based on SROIQ

Verbalizing Exclusion Constraint between single roles in different predicates is represented as follows:

For each Student, at most one of the following holds:
that Student passed some CourseExam;
that Student failed some CourseExam.

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 both OWL2 and SROIQ as shown in Fig (11).



Citizen $\equiv \exists \text{ has.SSN} \sqcup \exists \text{ got. Passport No}$
 Class: Citizen
 Equivalent To: has some SSN or got some Passport No

Fig. 11. Mapping Inclusive-Or Constraint from ORM2 to OWL2 based on SROIQ

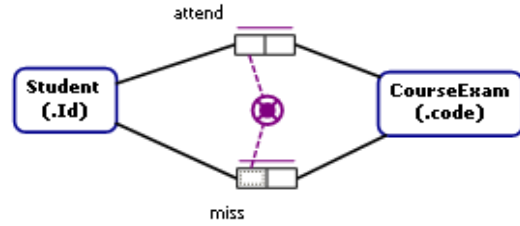
Verbalizing Inclusive Or Constraint between different predicates is represented as follows:

Each Citizen got some Passport No or has some SSN.

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 both SROIQ and OWL2.



Dis($\text{miss}, \text{attend}$)
 Student $\sqsubseteq \exists \text{ attend. CourseExam} \sqcup \exists \text{ miss. CourseExam}$
 ObjectProperty: attend
 Domain: Student
 Range: CourseExam
 ObjectProperty: miss
 Domain: Student
 Range: CourseExam
 DisjointProperties: miss, attend
 Class: Student
 SubClassOf: attend some CourseExam or miss some CourseExam

Fig. 12. Mapping Exclusive OR Constraint between single roles from ORM2 to OWL2 based on SROIQ

Verbalizing Exclusive Or Constraint between different predicates is represented as follows:

For each Student, exactly one of the following holds:
that Student attend some CourseExam;
that Student miss some CourseExam.

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 (\geq, \leq) constructs and to OWL syntax using the (max, min and exactly) as shown in Fig (13).



Reviewer $\equiv \leq 4 \text{ reviews Publication}$
 Class: Reviewer
 Equivalent To: reviews max 4 Publication

Fig. 13. Mapping Internal Frequency Constraint over Single Role from ORM2 to OWL2 based on SROIQ

Verbalizing Value Object type is represented as follows:

Each Reviewer reviews at most 4 instances of Publication.

External Frequency Constraint is applied to roles from two or more different predicates. There are neither SROIQ nor OWL2 constructs available to map this constraint.

8) Ring Constraints

Ring constraint is defined in [22] 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 both OWL2 and SROIQ.
- Irreflexive: relation on a set where no element is related to itself. Fig (14.b) shows an example of mapping Irreflexive constraint to both OWL2 and 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 both OWL2 and SROIQ.
- Asymmetric: if a relationship holds then its inverse cannot hold. Fig (14.d) shows an example of mapping Asymmetric constraint to both OWL2 and SROIQ.
- Ant symmetric: 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.

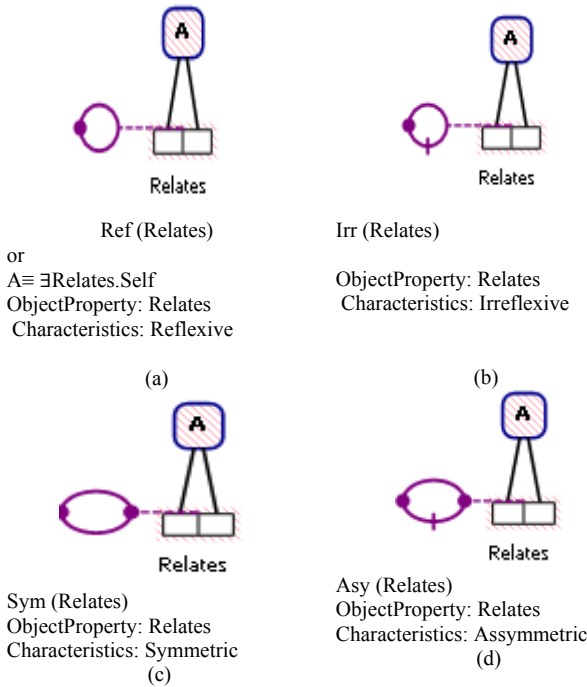


Fig. 14. Ring Constraints in SROIQ and OWL2

Verbalizing Reflexive Ring Constraint is represented as follows:

A relates A.

Verbalizing Irreflexive Ring Constraint is represented as follows:

No A relates the same A.

Verbalizing Symmetric Ring Constraint is represented as follows:

If A_1 relates A_2 then A_2 relates A_1 .

Verbalizing Asymmetric Ring Constraint is represented as follows:

If A_1 relates A_2 then it is impossible that A_2 relates A_1 .

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 and OWL2. Property chain features 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 is Grand Parent/ is Parent Of relation. In is Grand Parent / is Parent Of relation a property chain is used to represent the population of is Grand Parent as all individuals that are linked by a chain of exactly two is Parent Of properties. For class person, we assume we have three instances P1, P2 and P3. So if (P1)is Parent Of(P2) and (P2) is Parent Of(P3), then P1 is Grand Parent 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 P3 to P1. Derivation rule will be then used to define the relation between is Parent Of and is Grand Parent properties where is Grand Parent property will be derived from is Parent Of property. Fig (15) shows an example of mapping acyclic and intransitive ring constraint from ORM2 to both SROIQ DL and OWL2.

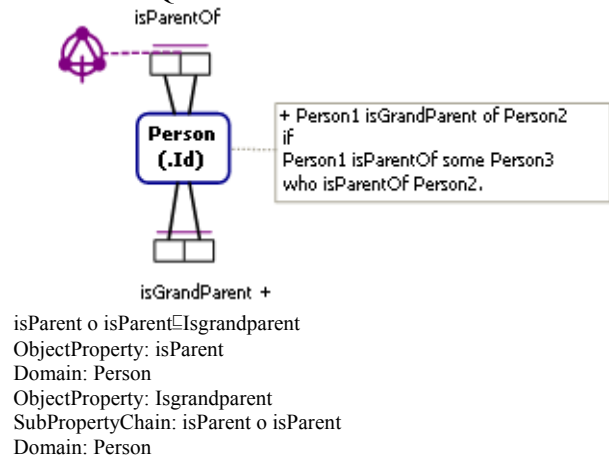


Fig. 15. Mapping SubPropertyChain Construct from ORM2 to OWL2 based on SROIQ

Verbalizing Property Chain is represented as follows:

Person is parent of Person.

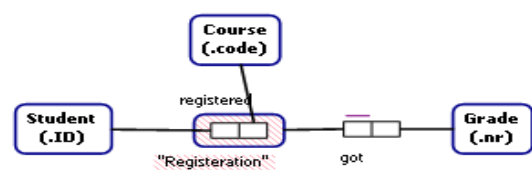
No Person may cycle back to itself via one or more traversals through Person isParentOf Person.

If $Person_1$ isParentOf $Person_2$ and $Person_2$ isParentOf $Person_3$ then it is impossible that $Person_1$ isParentOf $Person_3$.

Person isGrandParent Person.

Derivation Rule: + Person1 is a grandparent of Person2 if Person1 is a parent of some Person3 who is a parent of Person2.

A. Mapping Objectification Constrains from ORM2 to SROIQ



Class: Registration
EquivalentTo: Student registered some course
ObjectProperty: got
Domain: Registration
Range: Grade

Fig. 16. Mapping Objectification from ORM2 to OWL2

Verbalizing Objectification is represented as follows:

Registration is an entity type.

Registration objectifies "Student registered Course".

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. No Available SROIQ constructs to map this constraint however Fig (16) shows an example of mapping Objectification from ORM2 to OWL2 constructs.

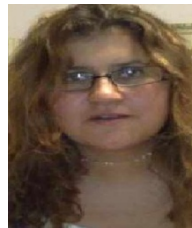
V. CONCLUSIONS AND FUTURE WORK

Ontology is gaining more importance due to its influence in enhancing data representation on the web. In this paper we used description logics as our basis to present a number of rules for mapping ORM2 constructs to its equivalent OWL2 syntax. We used in our research ORM2 since it is unlike other graphical modeling techniques in its expressivity and flexibility in representing most of the business semantic. It is an attribute-free data modeling technique and can be verbalized into natural language sentences. Manchester OWL syntax is used as our ontology language for its user-friendly syntax as it is easy to write, read and edit Manchester ontologies. As for the description logics we used 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 and OWL2 constructs based on 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, OWL2 and SROIQ constructs.

REFERENCES

- [1] T. Berners-Lee, M. Fischetti. "Weaving the Web: The Past, Present and Future of the World Wide Web by its Inventor". *Britain: Orion Business*. ISBN 0-7528-2090-7, 1999.
- [2] T. Gruber. "A translation approach to portable ontology specifications". In: *Knowledge Acquisition*. 5: 199-220.1993.
- [3] S. Bechjofer, F. van Harmelen, J. Hendler, I. Horrocks, D.L. McGuinness, P.F. Page-Schneider, L.A. Stein. "OWL Web Ontology Language Reference W3C Recommendation 10", *W3C Working Group*, 2004.
- [4] M. Horridge, N. Drummond, J. Goodwin, A. Rector, R. Stevens, and H. H. Wang. "The Manchester OWL Syntax". *OWL Experiences and Directions Workshop*, 2006.
- [5] Horridge, Matthew; Patel-Schneider, Peter F. "OWL 2 Web Ontology Language Manchester Syntax". W3C OWL 2 Web Ontology Language. World Wide Web Consortium. <http://www.w3.org/TR/owl2-manchester-syntax/>. (27 October 2009 - Retrieved 18 April 2010).
- [6] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. F. Patel-Schneider. "The Description Logic Handbook: Theory, Implementation and Applications". *Cambridge University Press*, 2003.
- [7] F. Baader and U. Sattler. "An overview of tableau algorithms for description logics". *Studia Logica*, 69(1):5-40, 2001.

- [8] T. Halpin, "ORM 2", On the Move to Meaningful Internet Systems 2005: OTM 2005 Workshops, eds R. Meersman, Z. Tari, P. Herrero et al., Springer LNCS 3762, pp. 676-87, 2005.
- [9] M. Curland and T. Halpin. "Model Driven Development with NORMA". *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International*, 2007.
- [10] E. Jürgen; W. Tobias. "Interoperability Services for Models and Ontologies". In: *Databases and Information Systems VI*. Oppress. Bd. 224. 2010.
- [11] I. Myroshnichenko, M. C. Murphy, "Mapping ER Schemas to OWL Ontologies," *icsc*, pp.324-329, 2009 *IEEE International Conference on Semantic Computing*, 2009.
- [12] M. Jarrar. "Towards Automated Reasoning on ORM Schemes. - Mapping ORM into the DLR_idf description logic". *Proceedings of the 26th International Conference on Conceptual Modeling (ER 2007)*. Springer, LNCS, Volume 4801, Pages (181-197), 2007.
- [13] M. Jarrar. "Mapping ORM into the SHOIN/OWL Description Logic-Towards a Methodological and Expressive Graphical Notation for Ontology Engine". *OTM workshops, Springer-Verlag*, Vol. LNCS 4805, Portugal, p.pp.729-741 .2007.
- [14] D. B. Bach, R. Meersman "Import/Export of OWL ontologies into/from DOGMA" Master's thesis, *Vrije Universiteit Brussel*, 2006, 2007.
- [15] D. Berardi, D. Calvanese, and G. De Giacomo. "Reasoning on UML Class Diagrams". *Artificial Intelligence*, 2005.
- [16] F. Franconi, G. Ng. "The ICOM Tool for Intelligent Conceptual Modelling". *7th Workshop on Knowledge Representation meets Databases (KRDB'00)*, 2000.
- [17] R. Colomb, A. Gerber and M. Lawley, "Issues in Mapping Metamodels in the Ontology Development Metamodel", *Proceedings of the 1st International Workshop on the Model-Driven Semantic Web. The First International Workshop on the Model-Driven Semantic Web, Monterey, California (September, 2004)*.
- [18] S. Upadhyaya, P. Kumar, "ERONTO: a tool for extracting ontologies from extended E/R diagrams", *ACM symposium on Applied computing SAC '05*, March 2005.
- [19] T. Halpin. "Object-Role Modeling (ORM/NIAM)", *Handbook on Architectures of Information Systems*, 2nd edition, Springer, Heidelberg, pp. 81-103, 2006.
- [20] M. Jackson. "Object Role Modelling and Conceptual Database Design", *Proc. Visual Developer's Academy*, Oxford, 1996.
- [21] I. Horrocks, O. Kutz, U. In KR. Sattler. "The Even More Irresistible SROIQ". In *Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR2006)*, 2006.
- [22] T. Halpin, "Information Modeling and Relational Databases", M. Kaufmann, San Francisco. 2001.



Heba M. Wagih received the B.S. degree in Informatics and Computer Sciences from Ain Shams University in 2003. During 2008-2011, she is preparing the M.S. degree in Information Systems in the Arab Academy for Science, Technology and Maritime Transport. All of her researches are concerned with the semantic web technologies and conceptual modeling techniques.



Doaa S. Elzanfal 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 Koutan 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 Johns 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),