An Arabic Verbalization Template
for ORM conceptual models and rules

Mustafa Jarrar, Vrije Universiteit Brussel, Belgium. (Contact Author)
Maria Keet, Free University of Bozen-Bolzano, Italy.

A technical report of the article[1]: *Jarrar, M., Keet, C.M., Dongilli, P.* ***Multilingual verbalization of ORM conceptual models and axiomatized ontologies****. [Submitted].*
URL: http://www.starlab.vub.ac.be/staff/mustafa/orm/verbalization/

Abstract. In the above-mentioned article we describe a novel approach to support *multilingual* verbalization of logical theories, axiomatizations, and other specifications such as business rules. This engineering solution is demonstrated with the Object Role Modeling language, although its underlying principles can be reused with other conceptual models and formal languages, such as Description Logics, to improve its understandability and usability by the domain expert. The engineering solution for multilingual verbalization is characterized by its flexibility, extensibility and maintainability of the verbalization templates, which allow for easy augmentation with other languages than the 11 currently supported.

This report presents the Arabic verbalization template file. Given an ORM schema (or an ORM-ML file), and given the verbalization template, an Arabic verbalization of the rules and fact types (in the schema) is generated automatically. A comprehensive example of an ORM schema and its corresponding verbalization is generated and given in this report.

1 Introduction

In the above-mentioned article, we present a novel approach to support multilingual verbalization of logical theories, formal axiomatizations, and other specifications such as business rules, ontologies, etc. We demonstrate our approach by providing a flexible and extensible verbalization template for the Object Role Modeling language. This template can be easily customized and translated into other human languages. *This technical report provides the Arabic verbalization of the ORM models and rules.* The verbalization of several other languages (including, but not limited to: German, Italian, English, Russian, Dutch, Spanish, French, Chinese, Vietnamese, and Lithuanian) can be found at the above-mentioned URL.

The underlying principles of our approach can be reused for other conceptual models and formal languages, such as Description Logics. The objective was to define a template parameterized over a given set of rules, models, or axioms, with as output fixed-syntax pseudo natural language sentences. A simple example is the following: the formal rule

$$\forall x\,(\mathrm{Book}(x) \rightarrow \exists y\,(\mathrm{ISBN}(y) \wedge \mathrm{Has}(x,y)))$$

can be translated into

It is mandatory that each **Book Has** an **ISBN**.

[1] For Citation use: *Jarrar, M., Keet, C.M.: An Arabic Verbalization Template for ORM conceptual models and rules. A technical report of the article: Jarrar, M., Keet, C.M., Dongilli, P. Multilingual verbalization of ORM conceptual models and axiomatized ontologies. [Submitted].*

In this way, we enable domain experts themselves to build and/or validate the formal specifications of their domains, without having to know that these sentences are formal axioms; i.e. the underpinning logics and reasoning services are hidden from the user. Our approach with the provided templates can be reused in modeling business rules, ontologies, knowledge bases, etc. See [H04] for a similar approach to ORM business rules verbalization.

In the following section, we present an ORM example followed by verbalization of all rules in this ORM schema. These verbalizations are generated *automatically*, according to the Arabic verbalization template presented in section 3. This approach is fully implemented and supported in the DogmaModeler ontology modeling tool [J05]. It is worth noting that DogmaModeler's automated verbalization has been used by tens of lawyers to build the Customer Complaint Ontology [J05][JVM03].

Remark on Modality: Our verbalization template can be adapted easily according to the application/reasoning scenario, whether it is used as integrity constraints, derivation/inference rules, business rules, etc. For example, the above mandatory constraint can be verbalized in different ways, such as: 1) Each **Book** must **Has** at least one **ISBN**. 2) Each **Book Has** some **ISBN** values. 3) If there is a **Book** then it **Has** an **ISBN** value. 4) A **Book** that does not **Has** an **ISBN** is not allowed. 5) If a **Book** does not **Has** an **ISBN** value then….

2 Example of an ORM Schema

We illustrate in one diagram most types of rules supported in ORM. Our article [JKD06] describes technical details on *how* our verbalization approach is implemented, see [H01] to know more about ORM, and [J05] to know about the DogmaModeler tool that we use to build and automatically verbalize ORM models.

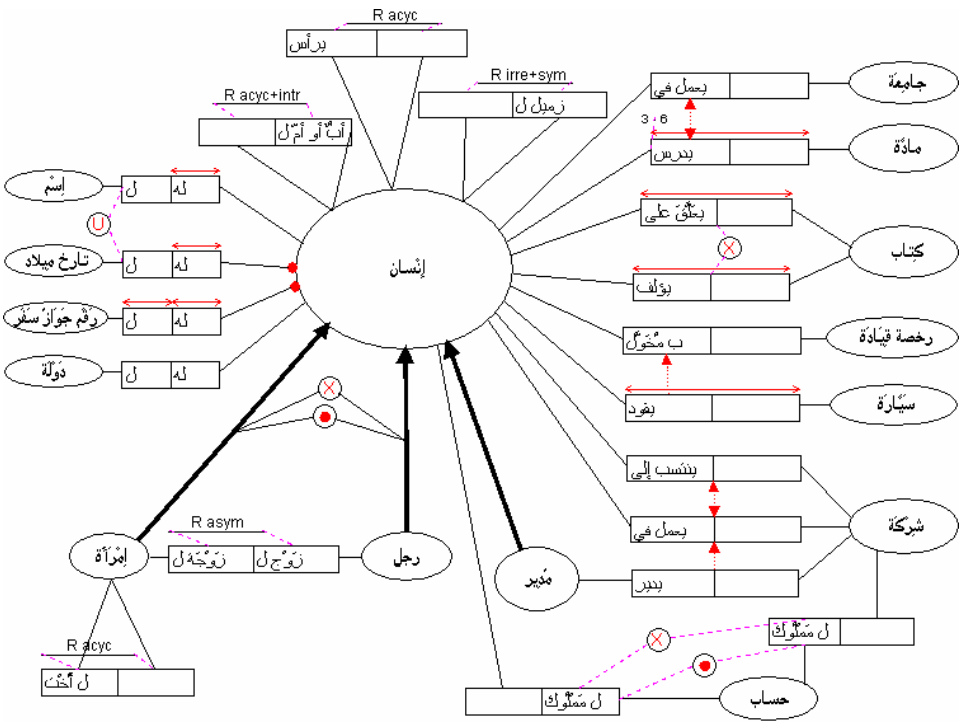

Fig. 1. Example of ORM rules, in Arabic.

The rules in the above ORM example are verbalized automatically (according to the verbalization template presented in section 2):

-[Mandatory] كل إنسان له رَقم جَوَازُ سَفَر واحد على الاقل
-[Mandatory] كل إنسان له تارخ ميلاد واحد على الاقل
-[Mandatory] كل حساب يجب ان يكون مملوك لانسان او مملوك لشركة
-[Uniqueness] كل انسان له تاريخ ميلاد واحد على الاكثر
-[Uniqueness] كل انسان له اسم واحد على الاكثر
-[Uniqueness] كل انسان له رقم جواز سفر واحد على الاكثر
-[Uniqueness] كل رقم جواز سفر لانسان واحد على الاكثر
-[Uniqueness] كل انسان يمكن ان يدرس اكثر من مادة والعكس صحيح
-[Uniqueness] كل انسان يمكن ان يؤلف اكثر من كتاب والعكس صحيح
-[Uniqueness] كل انسان يمكن ان يعلق على اكثر من كتاب والعكس صحيح
-[Uniqueness] كل انسان يمكن ان يقود اكثر من سيارة والعكس صحيح
-[Uniqueness] اتحاد كل من تاريخ ميلاد واسم يشير الى انسان واحد على الاكثر
-[Exclusive] كل انسان يمكن ان يكون اما رجل او اِمْرَأة
-[Totality] كل انسان يجب ان يكون رجل او اِمْرَأة
-[Subset] اذا انسان يقود سيارة فان هذا الانسان مخول برخصة سياقة
-[Subset] اذا مديريديرشركة فان هذا المديريعمل في هذة الشركة
-[Equality] كل انسان يعمل في جامعة اذا و فقط اذا هذا الانسان يدرس مادة
-[Equality] كل انسان منسوب لشركة اذا و فقط اذا هذا الانسان يعمل في هذة الشركة
-[Exclusion] لا يمكن ان يكون حساب مملوك لا نسان و في نفس الوقت مملوك لشركة
-[Exclusion] لا يمكن ان يكون انسان يعلق على كتاب و في نفس الوقت يؤلف ذلك كتاب
-[Value] القيم الممكنة ل دولة هي:{ بلجيكا, فرنسا, المانيا}
-[Irreflexive] لا يجوز لانسان ان يكون زميل لنفسه
-[Symmetric] اذا انسان س زميل ل ص,فان العكس بالعكس
-[Acyclic] لايمكن لانسان ان يكون (بطريقة مباشرة او غير مباشرة) اب او ام لنفسه
-[Acyclic] لايمكن لانسان ان يكون (بطريقة مباشرة او غير مباشرة) مشرف على نفسه
-[Asymmetric] اذا انسان س زوجة لانسان ص, فان العكس غير صحيح
-[Intransitve] اذا انسان س اب او ام لانسان ص, و ص اب او ام لانسان ج, فانه لايمكن ان يكون س اب او ام ل ج
-[Frequency] اذا الانسان يدرس مادة, فان هذا الانسان يجب ان يدرس بين 2 الى 3 مادة

3 The Arabic Verbalization Template

The template is presented in an XML syntax, it is being implemented in the DogmaModeler tool to support the Arabic verbalization of ORM models. More details about this approach can be found [JKD06], and refer to [J05] about DogmaModeler.

```
<?xml version='1.0' encoding='UTF-8'?> <ORMSchema xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='http://www.starlab.vub.ac.be/staff/mustafa/orm/verbalization/'>

<ORMNLMeta>
 <Meta name="DC.Title" content="Arabic verbalization template"/>
 <Meta name="DC.Version" content="0.2"/>
```

```
 <Meta name="DC.Creator" content="Mustafa Jarrar"/>
<Meta name="DC.Language" content="Arabic"/>
 </ORMNLMeta>
<ORMNLBody>

<FactType xsi:type="FactType" >
<Object index="0" />
<Role index="0"  />
<Text>/</Text>
<Role index="1"  />
<Object index="1" />
 </FactType>

<Constraint xsi:type="Mandatory"  >
 <Text> كل </Text>
 <Object index="0"  />
 <Role index="0"  />
 <Object index="1"  />
 <Text> واحد على الاقل</Text>
</Constraint>

<Constraint xsi:type="Backward Mandatory"  >
 <Text>لكل</Text>
 <Object index="0"  />
<Text>يوجد</Text>
<Object index="1"  />
<Text>واحد على الاقل</Text>
<Role index="1"  />
 <Text> هذا</Text>
 <Object index="0"  />
</Constraint>

<Constraint xsi:type="Disjunctive Mandatory">
 <Text> -[Mandatory]  كل </Text>
 <Object index="0"/>
 <Text>يجب ان يكون </Text>
 <Role index="0"/>
 <Object index="1"/>
 <Loop index="1" >
  <Text>او </Text>
  <Role index="n"/>
  <Object index="n"/>
 </Loop>
</Constraint>

<Constraint xsi:type="Uniqueness">
 <Text> -[Uniqueness]  لكل </Text>
 <Object index="0"/>
 <Role index="0"/>
 <Object index="1"/>
 <Text>واحد على الاكثر  </Text>
</Constraint>

<Constraint xsi:type="Backward Uniqueness"  >
<Text>لكل </Text>
<Object index="0" />
<Text> يوجد </Text>
<Object index="1"  />
<Text> واحد على الاكثر</Text>
<Role index="1"  />
 <Text> هذا</Text>
 <Object index="0"  />
```

```
</Constraint>

<Constraint xsi:type="Many Uniqueness" >
<Text>كل </Text>
<Object index="0" />
<Text>يمكن ان </Text>
<Role index="0"></Role>
<Text> اكثر من </Text>
<Object index="1"  />
<Text> والعكس صحيح </Text>
</Constraint>

<Constraint xsi:type="External Uniqueness" >
<Text>اتحاد كل من</Text>
<Object index="1"  />
<Loop index="1">
<Text>و</Text>
<Object index="n"  />
</Loop>
<Text>يشير الى</Text>
<Object index="0"   />
<Text>واحد على الاكثر </Text>
</Constraint>
<Constraint xsi:type="Subtype" >
<Text>كل</Text>
<Object index="child" />
<Text>هو</Text>
<Object index="parent"  />
 </Constraint>

<Constraint xsi:type="Value">
 <Text> -[Value]  اليقيم الممكنة ل </Text>
 <Object index="0"/>
 <Text>}:هي </Text>
 <Value index="0"/>
 <Loop index="1">
  <Text>,</Text>
  <Value index="n"/>
 </Loop>
 <Text> {</Text>
 </Constraint>

<Constraint xsi:type="Subtype" >
  <Text>كل</Text>
  <Object index="child" />
  <Text>هو</Text>
  <Object index="parent"  />
</Constraint>

<Constraint xsi:type="Exclusive" >
<Text>كل</Text>
<Object index="0"/>
<Text> يمكن ان يكون اما</Text>
<Object index="1"/>
<Loop index="1">
  <Text>او</Text>
  <Object index="n"/>
</Loop>
 </Constraint>
```

```
<Constraint xsi:type="Total" >
 <Text>كل </Text>
<Object index="0" />
<Text>  يجب ان يكون </Text>
<Object index="1"  />
<Loop index="1" >
  <Text> او </Text>
  <Object index="n" />
</Loop>
</Constraint>

<Constraint xsi:type="Partition" >
<Text>كل </Text>
<Object index="0" />
<Text> يجب ان يكون اما </Text>
<Object index="1"  />
<Loop index="1" >
  <Text> او </Text>
  <Object index="n" />
</Loop>
</Constraint>

<Constraint xsi:type="Subset" >
<Text>اذا </Text>
<Object  index="0" />
<Role index="child"  />
<Object index="child" />
<Text> فان هذا </Text>
<Object index="0"  />
<Role index="parent"  />
<Object index="parent" />
</Constraint>

<Constraint xsi:type="Subset FactType">
 <Text> -[Subset]  اذا </Text>
 <Object index="0"/>
 <Role index="child"/>
 <Object index="child"/>
 <Text> فان هذا  </Text>
 <Object index="1"  />
 <Role index="parent"/>
 <Text> هذة </Text>
 <Object index="parent"/>
</Constraint>

<Constraint xsi:type="Equality" >
<Text>كل </Text>
<Object index="0"  />
<Role index="first"  />
<Object index="first"  />
<Text>اذا و فقط اذاهذا ال </Text>
<Object index="0"  />
<Role index="second"  />
<Object index="second"  />
</Constraint>

<Constraint xsi:type="Equality FactType" >
<Text>كل </Text>
<Object index="0"  />
<Role index="0"  />
```

```
<Object index="1"  />
<Text>اذا و فقط اذاهذا ال</Text>
<Object index="1"  />
<Role index="second"  />
<Text>هذة ال</Text>
<Object index="second"  />
</Constraint>

<Constraint xsi:type="Exclusion" >
<Text>لا يمكن ان يكون  </Text>
<Object index="0"  />
<Role index="first"  />
<Object index="first"  />
<Text> و في نفس الوقت </Text>
<Role index="second"  />
<Object index="second"  />
</Constraint>

<Constraint xsi:type="Exclusion FactType" >
<Text>لا يمكن ان يكون  </Text>
<Object index="0"  />
<Role index="first"  />
<Object index="first"  />
<Text>و في نفس الوقت  </Text>
<Role index="second"  />
<Text>ذلك </Text>
<Object index="second"  />
 </Constraint>

<Constraint xsi:type="Frequency">
 <Text> -[Frequency]  اذا ال</Text>
 <Object index="0"/>
 <Role index="0"/>
 <Object index="1"/>
 <Role index="0"/>
 <Text>فان هذا ال  </Text>
 <Object index="0"/>
 <Text>يجب ان  </Text>
 <Role index="0"/>
 <Text>بين  </Text>
 <Minimum/>
 <Text> الى </Text>
 <Maximum/>
 <Role index="0"/>
</Constraint>

<Constraint xsi:type="Irreflexive" >
<Text>لا يجوز ل  </Text>
<Object index="0"/>
<Text> ان يكون </Text>
<Role index="0"/>
<Text> لنفسه </Text>
</Constraint>

<Constraint xsi:type="Symmetric"  >
<Text>اذا</Text>
<Object index="0"/>
<Text> س</Text>
<Role index="0"/>
<Object index="0"/>
<Text> ص</Text>
```

```
<Text> فانه العكس بالعكس</Text>
</Constraint>

<Constraint xsi:type="Asymmetric">
 <Text> -[Symmetric] اذا</Text>
 <Object index="0"/>
 <Text>س </Text>
 <Role index="0"/>
 <Object index="0"/>
 <Text> </Text>
 <Text> ص, فان العكس غير صحيح </Text>
</Constraint>

<Constraint xsi:type="Acyclic">
 <Text> -[Acyclic] لايمكن ل</Text>
 <Object index="0"/>
 <Text>(بطريقة مباشرة او غير مباشرة) ان يكون </Text>
 <Role index="0"/>
 <Text> نفسه</Text>
</Constraint>

<Constraint xsi:type="Transitve">
 <Text> -[Intransitve] اذا </Text>
 <Object index="0"/>
 <Text>س </Text>
 <Role index="0"/>
 <Object index="0"/>
 <Text> ص, و ص </Text>
 <Role index="0"/>
 <Text>ج, فانه لايمكن ان يكون س </Text>
 <Role index="0"/>
 <Text>ج </Text>
</Constraint>

</ORMNLBody>
</ORMSchema>
```

Acknowledgments

We are in debt to Andriy Lisovoy who helped in the implementation of DogmaModeler, and to Hai Nguyen Hoang who helped in the first implementation of the verbalization component during his Master thesis. This work is partially supported by the EU Knowledge Web NoE project (IST-2004-507482).

References

[JKD06] Jarrar, M., Keet, C.M., Dongilli, P. Multilingual verbalization of ORM conceptual models and axiomatized ontologies. [Submitted].

[J05] Jarrar, M.: Towards Methodological Principles for Ontology Engineering. PhD thesis, Vrije Universiteit Brussel, 2005.

[JVM03] Jarrar, M., Verlinden, R., Meersman, R.: Ontology-based Customer Complaint Management. In: Jarrar M., Salaun A., (eds.): Proceedings of the workshop on regulatory ontologies and the modeling of complaint regulations, Catania, Sicily, Italy. Springer Verlag LNCS. Vol. 2889. November (2003) pp. 594–606

[H01] Halpin, T.: Information Modeling and Relational Databases. 3rd ed. Morgan-Kaufmann. (2001)

[H04] Halpin, T.: Business Rule Verbalization. In Doroshenko, A., Halpin, T., Liddle, S., Mayr H. (eds): Information Systems Technology and its Applications, 3rd International Conference (ISTA'2004), LNI 48 GI ISBN 3-88579-377-6, (2004) pp:39-52.