

# Modello italiano di verbalizzazione per modelli concettuali e regole ORM

**Mustafa Jarrar**, Vrije Universiteit Brussel, Belgium. (Contact Author)

**Paolo Dongilli**, Free University of Bozen-Bolzano, Italy.

**Maria Keet**, Free University of Bozen-Bolzano, Italy.

---

Documentazione tecnica relativa all'articolo<sup>1</sup>: *Jarrar, M., Keet, C.M., Dongilli, P. Multilingual verbalization of ORM conceptual models and axiomatized ontologies. [Submitted].*  
URL: <http://www.starlab.vub.ac.be/stuff/mustafa/orm/verbalization/>

---

**Sommario.** Nell'articolo sopra menzionato descriviamo un nuovo approccio di supporto alla verbalizzazione multilingue di teorie logiche, assiomatizzazioni ed altre specifiche quali ad esempio business rules. Questa soluzione ingegneristica viene illustrata utilizzando il linguaggio Object Role Modeling, sebbene i principi su cui si basa possano essere riutilizzati con altri modelli concettuali e linguaggi formali quali Description Logics, al fine di migliorarne la comprensibilità e l'usabilità dall'esperto di dominio. La soluzione ingegneristica per la verbalizzazione multilingue è caratterizzata dalla sua flessibilità, estensibilità e mantenibilità dei modelli di verbalizzazione che permettono una semplice estensione ad altre lingue oltre alle 11 già supportate.

Questa documentazione tecnica presenta il modello italiano di verbalizzazione. Dato uno schema ORM (o un file ORM-ML) e dato un modello di verbalizzazione, la verbalizzazione italiana delle regole e dei tipi di fatti (nello schema) vengono generati automaticamente. Un esempio completo di schema ORM viene generato ed illustrato in questo documento unitamente alla sua verbalizzazione.

## 1 Introduzione

Nell'articolo sopra menzionato descriviamo un nuovo approccio capace di supportare la verbalizzazione multilingue di teorie logiche, assiomatizzazioni ed altre specifiche quali ad esempio business rules, ontologie ecc. Illustriamo il nostro approccio fornendo un modello di verbalizzazione flessibile ed estensibile per il linguaggio Object Role Modeling. Questo template può essere facilmente personalizzato e tradotto in altre lingue. La presente documentazione tecnica fornisce una verbalizzazione italiana dei modelli e delle regole ORM. La verbalizzazione in diverse altre lingue (incluso arabo, russo, spagnolo, olandese, tedesco, francese, lituano e non limitatamente a queste) può essere trovata al link sopra indicato.

I principi basilari del nostro approccio possono essere riutilizzati per altri modelli concettuali a linguaggi formali quali Description Logics. L'obiettivo è stato quello di definire un modello parametrizzato rispetto ad un dato insieme di regole, modelli o assiomi, avente come output frasi a sintassi prefissata in pseudo linguaggio naturale. Segue un semplice esempio: la regola

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

può essere tradotta come

**Ogni Libro deve avere un ISBN.**

In questo modo diamo la possibilità agli stessi esperti di dominio di costruire e/o validare le specifiche formali dei loro domini senza la necessità di sapere che queste frasi sono assiomi formali; ciò significa che la logica sottintesa ed i servizi di reasoning vengono nascosti all'utente. Il nostro approccio con i modelli messi a disposizione può essere riusato per modellizzare business rules, ontologie, basi di conoscenza ecc. Si veda [H04] per un simile approccio alla verbalizzazione di business rules ORM.

Nella prossima sezione presentiamo un esempio ORM seguito dalla verbalizzazione di tutte le regole dello schema ORM. Queste verbalizzazioni vengono generate *automaticamente*, seguendo il modello italiano di verbalizzazione presentato nella terza sezione. Questo approccio è interamente implementato e supportato

---

<sup>1</sup> Per riferimenti bibliografici: *Jarrar, M., Dongilli, P., Keet, C.M.: An Italian 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].*

dal tool di modellizzazione ontologica DogmaModeler [J05]. Vale la pena di far notare che la verbalizzazione automatica di DogmaModeler è stata usata da decine di giuristi per costruire la Customer Complaint Ontology [J05][JVM03].

Note sulla modalità: il nostro modello di verbalizzazione può essere semplicemente adattato a seconda dello scenario applicativo e ragionativo, a seconda che sia usato come vincoli di integrità, regole di derivazione/inferenza, business rules ecc. Per esempio, il vincolo obbligatorio di cui sopra può essere verbalizzato in diversi modi quali ad esempio: 1) Ogni **libro deve avere** almeno un **ISBN**. 2) Ogni **libro ha** dei valori **ISBN**. 3) Se esiste un **libro** allora ha un valore **ISBN**. 4) Un **libro** che non ha un **ISBN** non è ammesso. 5) Se un **libro non ha** un **ISBN**, allora ...

## 2 Esempio di schema ORM

Illustriamo nel seguente diagramma la maggiorparte delle regole previste da ORM. Il nostro articolo [JKD06] descrive i dettagli tecnici di come il nostro approccio di verbalizzazione è implementato; si veda [H01] per avere ulteriori informazioni su ORM e [J05] per conoscere il tool DogmaModeler usato per costruire e verbalizzare automaticamente i modelli ORM.

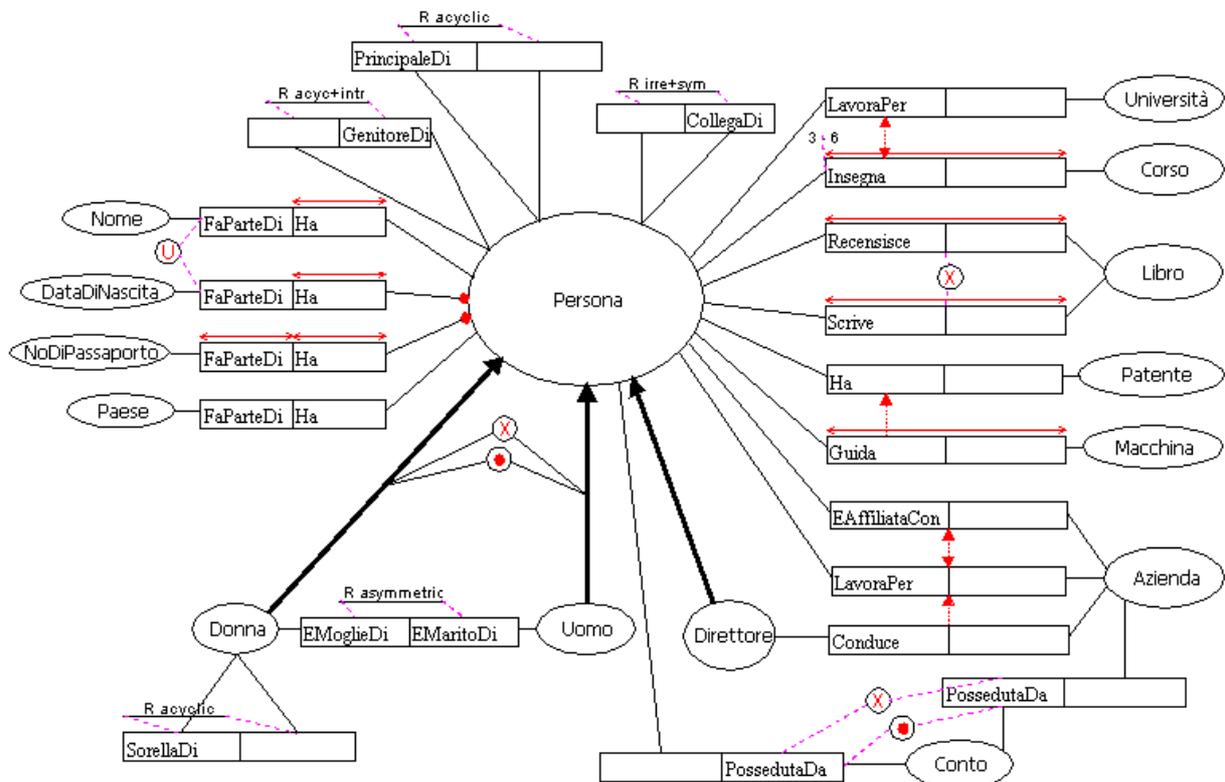


Fig. 1. Esempio di regole ORM in italiano.

I vincoli/regole nell'esempio ORM mostrato sopra vengono verbalizzate automaticamente (concordemente al modello di verbalizzazione presentato nella seconda sezione):

- [Mandatory] Ogni Persona deve Ha almeno un NoDiPassaporto.
- [Mandatory] Ogni Persona deve Ha almeno un DataDiNascita.
- [Mandatory] Ogni Conto deve essere PossedutaDa Azienda o PossedutaDa Persona.
- [Uniqueness] Ogni Persona deve Ha al massimo un(a) DataDiNascita.
- [Uniqueness] Ogni Persona deve Ha al massimo un(a) Nome.
- [Uniqueness] Ogni Persona deve Ha al massimo un(a) NoDiPassaporto.
- [Uniqueness] Ogni NoDiPassaporto deve FaParteDi al massimo un(a) Persona.
- [Uniqueness] E' possibile che un(a) Persona Insegna più di un(a) Corso , e viceversa.
- [Uniqueness] E' possibile che un(a) Persona Recensisce più di un(a) Libro , e viceversa.
- [Uniqueness] E' possibile che un(a) Persona Scrive più di un(a) Libro , e viceversa.

-[Uniqueness] E' possibile che un(a) Persona Guida più di un(a) Macchina , e viceversa.

-[Uniqueness] La combinazione di { Nome e DataDiNascita } deve essere riferito a al massimo un(a) Persona.

-[Exclusive] Ogni Persona deve essere o Uomo o Donna.

-[Totality] Ogni Persona deve essere almeno o Donna o Uomo.

-[Subset] Se un(a) Persona Guida un(a) Macchina, in questo caso il/lo/la Persona in questione Ha un(a) Patente.

-[Subset] Se un(a) Direttore Conduce un(a) Azienda, in questo caso il/lo/la Direttore in questione LavoraPer questo(a) Azienda.

-[Equality] Un(a) Persona LavoraPer un(a) Università se e solo se questo(a) Persona Insegna un(a) Corso.

-[Equality] Un(a) Persona EAffiliataCon un(a) Azienda se e solo se questo(a) Persona LavoraPer questo(a) Azienda.

-[Exclusion] Nessun(a) Conto PossedutaDa un(a) Azienda ed anche PossedutaDa Persona.

-[Exclusion] Nessun(a) Persona Scrive un(a) Libro ed anche Recensisce lo/la stesso(a) Libro..

-[Value] Gli oggetti possibili di Paese sono: {Belgio, Francia, Germania }.

-[Irreflexive] Nessun Persona CollegaDi se stesso.

-[Symmetric] Se Persona X CollegaDi Persona Y, deve essere viceversa.

-[Acyclic] Un(a) Persona non può essere diretto (o indiretto per una catena) PrincipaleDi se stesso .

-[Acyclic] Un(a) Donna non può essere diretto (o indiretto per una catena) SorellaDi se stesso .

-[Asymmetric] Se Donna X EMoglieDi Donna Y, viceversa è impossibile .

-[Intransitive] Se Persona X GenitoreDi Persona Y, e Y GenitoreDi Z, in quel caso non è possibile che X GenitoreDi Z.

-[Frequency] Se Persona Insegna un(a) Corso, in questo caso il/lo/la Persona in questione Insegna almeno 3 ed al massimo 6 Corso.

### 3 Il modello italiano di verbalizzazione

Il modello è mostrato in sintassi XML ed è implementato nel tool DogmaModeler per supportare le verbalizzazione italiana di modelli ORM. Ulteriori dettagli su questo approccio possono essere trovati in [JKD06] mentre si faccia riferimento a [J05] per una descrizione di 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/verbalizati
on/'>

<ORMNLMeta>
  <Meta name="DC.Title" content="Italian verbalization template (Ver0.1)"/>
  <Meta name="DC.Version" content="0.1"/>
  <Meta name="DC.Creator" content="Mustafa Jarrar"/>
  <Meta name="DC.Contributor" content="Maria Keet"/>
  <Meta name="DC.Language" content="Italian"/>
</ORMNLMeta>

<ORMNLBody>

<FactType xsi:type="FactType" >
<Text>Un(a)</Text>
<Object index="0" />
<Role index="0" />
<Text></Text>
<Role index="1" />
<Text> un(a)</Text>
<Object index="1" />
</FactType>

<Constraint xsi:type="Mandatory">
  <Text> -[Mandatory] Ogni</Text>
  <Object index="0"/>
  <Text>deve</Text>
  <Role index="0"/>
  <Text>almeno un</Text>
  <Object index="1"/>
</Constraint>

```

```

<Constraint xsi:type="Backward Mandatory">
  <Text> -[M] Per ogni</Text>
  <Object index="0"/>
  <Text>c'è almeno un</Text>
  <Object index="1"/>
  <Text>che</Text>
  <Role index="1"/>
  <Text>questo (a)</Text>
  <Object index="0"/>
</Constraint>

<Constraint xsi:type="Disjunctive Mandatory">
  <Text> -[Mandatory] Ogni</Text>
  <Object index="0"/>
  <Text>deve essere</Text>
  <Role index="0"/>
  <Object index="1"/>
  <Loop index="1" >
    <Text>o</Text>
    <Role index="n"/>
    <Object index="n"/>
  </Loop>
</Constraint>

<Constraint xsi:type="Uniqueness">
  <Text> -[Uniqueness] Ogni</Text>
  <Object index="0"/>
  <Text>deve</Text>
  <Role index="0"/>
  <Text>al massimo un (a)</Text>
  <Object index="1"/>
</Constraint>

<Constraint xsi:type="Backward Uniqueness">
  <Text> -[Uniqueness] Per ogni</Text>
  <Object index="0"/>
  <Text>deve essere al massimo un (a)</Text>
  <Object index="1"/>
  <Text>che</Text>
  <Role index="1"/>
  <Text>questo</Text>
  <Object index="0"/>
</Constraint>

<Constraint xsi:type="Many Uniqueness">
  <Text> -[Uniqueness] E' possibile che un (a)</Text>
  <Object index="0"/>
</Role>
  <Text> più di un (a) </Text>
  <Object index="1"/>
  <Text>, e viceversa</Text>
</Constraint>

<Constraint xsi:type="External Uniqueness">
  <Text> -[Uniqueness] La combinazione di {</Text>
  <Object index="1"/>
  <Loop index="1">
    <Text>e</Text>
    <Object index="n"/>
  </Loop>
  <Text>} deve essere riferito a al massimo un (a) </Text>
  <Object index="0"/>
</Constraint>

<Constraint xsi:type="Subtype">
  <Text> -[Subtype] Ogni oggetto </Text>
  <Object index="child"/>
  <Text>è anche un oggetto di </Text>
  <Object index="parent"/>
</Constraint>

```

```

<Constraint xsi:type="Value">
  <Text> -[Value] Gli oggetti possibili di </Text>
  <Object index="0"/>
  <Text> sono :{</Text>
  <Value index="0"/>
  <Loop index="1">
    <Text>,</Text>
    <Value index="n"/>
  </Loop>
  <Text> }</Text>
</Constraint>

<Constraint xsi:type="Exclusive">
  <Text> -[Exclusive] Ogni</Text>
  <Object index="0"/>
  <Text>deve essere o </Text>
  <Object index="1"/>
  <Loop index="1">
    <Text>o</Text>
    <Object index="n"/>
  </Loop>
</Constraint>

<Constraint xsi:type="Total">
  <Text> -[Totality] Ogni</Text>
  <Object index="0"/>
  <Text>deve essere almeno o </Text>
  <Object index="1"/>
  <Loop index="1">
    <Text>o</Text>
    <Object index="n"/>
  </Loop>
</Constraint>

<Constraint xsi:type="Partition">
  <Text> -[Partition] Ogni</Text>
  <Object index="0"/>
  <Text>e' almeno uno di </Text>
  <Object index="1"/>
  <Loop index="1">
    <Text>o</Text>
    <Object index="n"/>
  </Loop>
  <Text>,<Text>, pero non tutti</Text>
</Constraint>

<Constraint xsi:type="Subset">
  <Text> -[Subset] Se un(a)</Text>
  <Object index="0"/>
  <Role index="child"/>
  <Text>un(a)</Text>
  <Object index="child"/>
  <Text>,<Text>, in questo caso il/lo/la</Text>
  <Object index="0"/>
  <Text>in questione</Text>
  <Role index="parent"/>
  <Text>un(a)</Text>
  <Object index="parent"/>
</Constraint>

<Constraint xsi:type="Subset FactType">
  <Text> -[Subset] Se un(a)</Text>
  <Object index="0"/>
  <Role index="child"/>
  <Text>un(a)</Text>
  <Object index="child"/>
  <Text>,<Text>, in questo caso il/lo/la</Text>
  <Object index="0" />
  <Text>in questione</Text>
  <Role index="parent"/>
  <Text>questo(a)</Text>

```

```

<Object index="parent"/>
</Constraint>

<Constraint xsi:type="Equality">
<Text> -[Equality] Un(a) </Text>
<Object index="0"/>
<Role index="first"/>
<Text> un(a) </Text>
<Object index="first"/>
<Text>se e solo se</Text>
<Text>questo(a) </Text>
<Object index="0"/>
<Role index="second"/>
<Text> un(a) </Text>
<Object index="second"/>
</Constraint>

<Constraint xsi:type="Equality FactType">
<Text> -[Equality] Un(a) </Text>
<Object index="0"/>
<Role index="First"/>
<Text>un(a)</Text>
<Object index="First"/>
<Text>se e solo se</Text>
<Text>questo(a)</Text>
<Object index="1"/>
<Role index="Second"/>
<Text>questo(a)</Text>
<Object index="Second"/>
</Constraint>

<Constraint xsi:type="Exclusion">
<Text> -[Exclusion] Nessun(a) </Text>
<Object index="0"/>
<Role index="first"/>
<Text>un(a)</Text>
<Text> </Text>
<Object index="first"/>
<Text> ed anche </Text>
<Role index="second"/>
<Text> </Text>
<Object index="second"/>
</Constraint>

<Constraint xsi:type="Exclusion FactType">
<Text> -[Exclusion] Nessun(a)</Text>
<Object index="0"/>
<Role index="first"/>
<Text>un(a)</Text>
<Object index="first"/>
<Text>ed anche</Text>
<Role index="second"/>
<Text>lo/la stesso(a)</Text>
<Object index="second"/>
</Constraint>

<Constraint xsi:type="Frequency">
<Text> -[Frequency] Se </Text>
<Object index="0"/>
<Role index="0"/>
<Text>un(a)</Text>
<Object index="1"/>
<Role index="0"/>
<Text>, in quel caso il/lo/la</Text>
<Object index="0"/>
<Text>in questione</Text>
<Role index="0"/>
<Text>almeno </Text>
<Minimum/>
<Text> ed al massimo </Text>
<Maximum/>

```

```

<Role index="0"/>
<Text>i/e</Text>
</Constraint>

<Constraint xsi:type="Irreflexive">
<Text> -[Irreflexive] Nessun</Text>
<Object index="0"/>
<Role index="0"/>
<Text> se stesso</Text>
</Constraint>

<Constraint xsi:type="Symmetric">
<Text> -[Symmetric] Se </Text>
<Object index="0"/>
<Text>X</Text>
<Role index="0"/>
<Object index="0"/>
<Text>Y</Text>
<Text>, deve essere viceversa</Text>
</Constraint>

<Constraint xsi:type="Asymmetric">
<Text> -[Asymmetric] Se</Text>
<Object index="0"/>
<Text> X</Text>
<Role index="0"/>
<Text> </Text>
<Object index="0"/>
<Text> Y, viceversa è impossibile</Text>
</Constraint>

<Constraint xsi:type="Acyclic">
<Text> -[Acyclic] Un(a)</Text>
<Object index="0"/>
<Text> non può essere diretto (o indiretto per una catena)</Text>
<Role index="0"/>
<Text> se stesso</Text>
</Constraint>

<Constraint xsi:type="Transitive">
<Text> -[Intransitive] Se</Text>
<Object index="0"/>
<Text>X</Text>
<Role index="0"/>
<Object index="0"/>
<Text>Y, and Y</Text>
<Role index="0"/>
<Text> Z, in quel caso non e' possibile che X</Text>
<Role index="0"/>
<Text>Z</Text>
</Constraint>

</ORMNLBody>
</ORMSchema>

```

## Ringraziamenti

Siamo in debito con Andriy Lisovoy che ci ha aiutati per lo sviluppo di DogmaModeler e con Hai Nguyen Hoang che ci ha aiutati nella prima implementazione del modulo di verbalizzazione nel corso della sua tesi di master. Questo lavoro è finanziato in parte dal progetto europeo Knowledge Web NoE (IST-2004-507482).

## Bibliografia

- [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.