The formalisation of ORM2 and its encoding in OWL2

Enrico Franconi and Alessandro Mosca

Free University of Bozen-Bolzano, Italy
lastname@inf.unibz.it

Abstract. The *Object Role Modelling* language (ORM2) is a conceptual modelling language similar to UML and ER, adopted by Visual Studio, the integrated development environment from Microsoft. This paper introduces a new linear syntax and complete set-theoretic semantics for a generalisation of ORM2, which can be shown correctly embedding the original proposal. A provably correct encoding of the core fragment of ORM2 (similarly expressive to UML class diagrams) in the $\mathcal{ALCQI}$ description logic is presented. On the basis of these results, a systematic critique of alternative approaches to the formalisation of ORM2 in (description) logics published so far is provided. A prototype has been implemented providing a backend for the automated support of consistency and entailment checks for ORM2 conceptual schemas along with its translation into $\mathcal{ALCQI}$ knowledge bases.

1 Introduction

Automated support to enterprise modelling has increasingly become a subject of interest for organisations seeking solutions for storage, distribution and analysis of knowledge about business processes [1], and the main expectation from automated solutions built upon these approaches is the ability to automatically determine consistency of a business model, so that they can be further exploited to build information systems and relational databases that are coherent with the intended domain business logic. Common approaches for describing business and the information used by that business are the rule-based approach [1], and the ontology-based approach [2]. Where the first one consists in identifying and articulating the rules that define the structure and control the operation of an enterprise [3], the second approach seeks to model basic business logic and meta-knowledge about business domain using ontologies. ORM2 ('Object Role Modelling 2') is a graphical fact-oriented approach for modelling, transforming, and querying business domain information, which allows for a verbalisation in language readily understandable by non-technical users. ORM2 is at the core of the OGM standard SBVR language ('Semantics of Business Vocabulary and Business Rules'), and of the conceptual modelling language for database design in Microsoft Visual Studio .NET. In particular, the Neumont ORM Architect (NORMA) tool is an open source plug-in to Microsoft Visual Studio providing the most complete support for the ORM2 notation. Previous version of ORM was supported as the ORM Source Model Solution in Microsoft Visio (Halpin et al. 2003) and, finally, VisioModeler is also a tool freely available from Microsoft's MSDN Web site: It allows to create ORM models under Windows (version earlier than Vista) and map them to a range of database management systems [4].

The NIAM language ('Natural-language Information Analysis Method'), ancestor ORM, has been equipped with an FOL-based semantics for the first time in 1989 [5]. Since then, despite the remarkable evolution in terms of expressivity and graphical notation that ORM2 has experienced, much less attention has been paid in the consequent development of appropriate formal foundations for the modelling language. In particular, the lack of formal foundations for ORM2 had two major consequences: (i) it prevented the identification of formalisms able to capture the expressivity of ORM2, and the possibility of exploiting the reasoning services of these formalisms to support the modelling activity itself; and (ii) it resolved in the last years into a plethora of formally unjustified, and often misleading, publications dedicated to automatic translation of ORM2 into other formal modelling languages (most notably the W3C OWL2 language). This paper can be considered as the first attempt to fill this gap for ORM2, and it is part of the well known tradition in Knowledge Representation and Reasoning whose aim is to provide logic-based foundations, as well as sound and complete reasoning services, to conceptual modelling languages like, for example, for (E)ER and UML Class Diagrams (see, among others, [6,7,8,9,10]. The paper addresses the main problem of providing a logic formalism, equipped with sound and complete reasoning services, that captures the expressiveness of ORM2. The paper also provides the identification of a 'practical' fragment of ORM2 whose computational complexity of reasoning is tractable. The first contribution of the paper is thus the introduction of a completely new linear syntax and a set-theoretic semantics for ORM2 matching the usage patterns in the community. The new syntax can be used to express the full set of ORM2 graphical symbols introduced in [4]. The new semantics has been proved to be equivalent with the original FOL semantics of NIAM, up to the differences in the expressivity of the two languages (NIAM expressiveness is properly included in the one of ORM2, indeed). Now, due to the intrinsic undecidability of FOL, relying on a FOL theorem prover does not represent in general an effective way to equip a highly expressive conceptual modelling language with a completely automated reasoning service, not even in the case of ORM2. Therefore, the second contribution of the paper is driven by a practical objective. On the basis of well known results developed in the Description Logics (DLs) community, we identified a 'core' fragment of ORM2 that can be translated in a sound a complete way into the ExpTime-complete logic $\mathcal{ALCQI}$ [11], through n-ary relations *reification*. The $\mathcal{ALCQI}$ logic is actually the most expressive formalism that is supported by the current state-of-the-art of DL reasoners. On the basis of the results presented in the paper, a first prototype, built on top of available DL reasoners, has been implemented, which provides an automated support for schema consistency, entity/relations consistency check, and entailment verification for user-defined ORM2 statements.

The rest of the paper is organised as follows: Section 2 is about the introduction, through examples, of the ORM2 graphical notation and intended semantics in the framework of the fact modelling approach; Section 3 introduces the new linear syntax, together with a complete set-theoretic semantics for a generalisation of ORM2. Several examples are use there in order to clarify the reasoning tasks that can automatised via the proposed formal semantics, as well as to show the flexibility of the new linear syntax in providing the schemas encoding. The first-order logic translation of the introduced set-theoretic semantics is provided in Section 4, while the encoding of the introduced semantics into the DL logic $\mathcal{ALCQI}$ is the main topic of Section 5. There, a 'practical fragment' of OWL2 is explicitly introduced and it is shown that the correctness of the encoding for this fragment is enough to preserve soundness and completeness of the introduce reasoning tasks. An extensive critique of alternative approaches to the formalisation of ORM2 in (description) logics published so far is finally provided in Section 6. Section 7 gives an overview of the implemented reasoning support prototype, by means of an example of its usage. All the graphical examples in the paper have been drawn using NORMA.

2 Fact-oriented modelling in ORM2

'Fact-oriented modelling' began in the early 1970s as a conceptual modelling approach that views the world in terms of simple facts about objects and the roles they play [4].

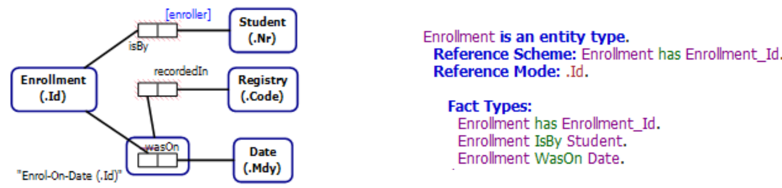


Fig. 1. Example of an ORM2 schema, together with the fragment of its verbalisation.

The building blocks of this approach are represented by *facts* (i.e. assertions that are taken to be true in the domain of interest) about *objects playing roles* (e.g. 'Alice is enrolled in the Computer Science program', 'Mary works for the Department of Philosophy'). Basic ORM2 objects are: **entities** (e.g. a particular house or a car) and **values** (e.g. *character string* or *number*). Moreover, entities and values are described in terms of the **types** they belong to: A type (e.g. House, Car) is a set of possible instances. Each entity in the domain of interest is an instance of a particular type. In order to avoid ambiguity among the possible instances of a given type, entities are identified also by means of a particular *reference mode* and a value. A reference mode (e.g. countryCode, securityNumber) specifies the way in which a value refers to an entity (e.g. 'The person with security number '285' is born in 'US'). The full specification of an entity type, together with a reference mode and associated values gives rise to a *reference schema* (e.g. *Entity Type*: Person, *Reference Mode*: surname, *Value*: 'Stone'). ORM2 also admits the possibility of treating relationships among objects as object itself. Once a relation has been transformed into an object type, this last is said to be the **objectification** of the relation. The roles played by the entities in a given domain are introduced by means of logical **predicates**; each predicate (or relation) has a given set of roles according to its arity. Each role is connected to exactly one object type, indicating that the role is played only by the possible instances of that type (notice that, unlike ER, ORM2 makes no use of 'attributes' in its base models). Given an n-ary predicate R, the predicate is decomposed into $R.a_1, \ldots, R.a_n$ roles, and each role is linked to a types $O_1, \ldots, O_n$. Now, turning into the ORM2 graphical notation, let us consider the example in Fig. 1. The schema includes:

1. Four entity types Enrollment, Student, Date, and Enrol-On-Date;
2. three binary predicates isBy, wasOn, and recordedIn;
3. a user-defined role name [enroller], for the role played by Student
4. a reference mode for each entity, .Id, .Nr, and .Mdy.

As for the reference modes, the semantics can be explained by fully expanding their representation. In the case of the Student entity type in Fig. 2, a new relation has and a value type StudentNr have been inserted. Then, a **mandatory participation** constraint ('Each Student has *at least one* StudentNr'), and an **internal uniquess** constraint ('Each Student has *at most one* StudentNr') agree on the same role. Together with the uniqueness constraint on the other role of has, the diagram forces an *injection* between Student and StudentNr instances, that is, 'For each Student there exists *exactly one* StudentNr' *and* 'Each StudentNr is associated to *exactly one* Student' (i.e. the relation is both functional and inverse functional).

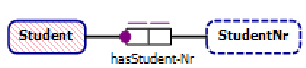


Fig. 2. ORM2 *reference schema* explained. Mandatory, graphically represented by a dot, and uniqueness constraints, represented by a continuos bar above the roles, are introduced.

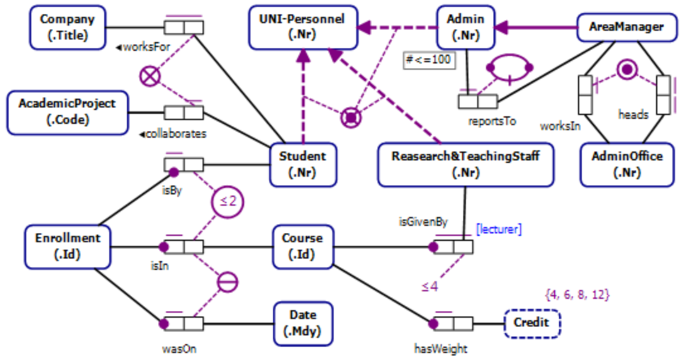


Fig. 3. A conceptual schema including an instantiation of most of the ORM2 constraints.

According to the ORM2 design procedure, after the specification of the relevant **object types** (i.e. entity and value types), predicates, static *constraints* must be considered - the rest of this section is devoted to an informal introduction of their graphical representation and intended semantics. Let us consider the example in Fig. 3 that, in addiction to the elements introduced in 1 and 2, includes:

1. **Subtyping** links (depicted as thick arrows) indicating 'isa' relationships among types, and a constraint combination, called **partition**, made of an **exclusive** constraint (a circled 'X' for 'Research&TeachingStaff, Admin, Student' are *mutually disjoint*), and a **total** constraint (a circled dot for 'Research&TeachingStaff, Admin, Student completely cover their common super-type').
2. **Internal frequency occurrence** constraint indicating that *if* an instance of Research&TeachingStaff plays the role of being lecturer in the relation isGivenBy, it plays the role at most 4 times. A frequency occurrence may span over more than one role, and frequency *ranges* can be specified.
3. An **external frequency occurrence** that applies to roles played by Student and Course, meaning that 'Students are allowed to enrol in the same course *at most twice*'.
4. An **external uniqueness** constraint between the role played by Course in isIn and the role played by Date in wasOn, saying that 'For each combination of Course and Date, *at most one* Enrollment isIn that Course and wasOn that Date'.

Notice that uniqueness and frequency occurrence constraints may be specialised as 'internal' or 'external' constraints. External are those constraints that apply to roles from different predicates, and their semantics requires the specification of *join path* [4]. Join paths are used specify how the involved predicates should be concatenated, e.g. the type Enrollment in the example is used to navigate from the predicate isIn to the predicate isBy. Therefore, since the result of performing a join over the type Enrollment is a ternary table, the external frequency constraint forbids this table from having more than 2 triples with identical combinations of Student and Course (e.g. if {`(Stone,AI,enr123)`,`(Stone,AI,enr234)`} are in the table, the update add(`Stone,AI,enr345`) is forbidden)[1].

5. A disjunctive mandatory 'circled dot', called **inclusive-or**, linking the roles played by AreaManager indicating that 'Each area manager *either* works in *or* heads (or *both*)'.
6. An **object cardinality** constraint forcing the number of the Admin instances to be less or equal to 100. **Role cardinality** constraints, applied to role instances, are also part of ORM2.

[1] The actual release of NORMA does not allow the specification of arbitrary join paths. Nonetheless, according to the ORM Foundation, this feature is currently under development.

7. An **object type value** constraint indicating which values are allowed in Credit. **Role value** constraints can be also expressed to indicate which values are allowed to the instances playing a given role.
8. An **exclusion** constraint (depicted as circled 'X') between the roles played by Student in the relations worksFor and collaborates, expressing the fact that no student can play *both* these roles. Exclusion constraint can also span over arbitrary sequences of roles. The combination of exclusion and inclusive-or constraints gives rise to **exclusive-or** constraints meaning that each instance in the attached entity type plays *exactly one* of the attached roles. Exclusion constraints, together with **subset** and **equality**, are called *set-comparison* constraints.
9. A **ring** constraint expressing that the relation reportsTo is *asymmetric*.

3 ORM2 from a formal perspective

The modelling activity in ORM2 is supported by several tools that provide user friendly graphical interfaces to build complex conceptual schema in real world application domains. The tools perform syntactic check on the graphical notation, warning for not-admitted combinations of basic elements and constraints, and driving the modelling activity coherently with the ORM2 *conceptual schema design procedure* [4]. Nonetheless, the ability to avoid the definition of syntactically correct schemas that resolve to be semantically inconsistent is currently left to expertise and skill of the modeller itself, since none of the available design tools offers automated reasoning support on specific combinations of constraints provided by the user. It is well known that, due to design mistakes or to over-constraining, a conceptual schema may be syntactically correct and, nonetheless, (i) it may not admit any instantiation (i.e. the entire schema cannot be populated without the violation of some of the constraints), or (ii) it may admit only a partial instantiation (i.e. some entity or value types/relations, but not all of them, are forced to be empty). *Schema consistency*, *consistency of an object type*, and the fact that some constraints may be already present in a schema as *implicit consequences*, are typical properties of a conceptual schema that, once checked, significantly improve the quality of the schema giving to the modeller precise information to refining the schema by relaxing some constraints, or removing some entity types and relations. Now, the automated verification of these properties over a schema strictly depends on the possibility to perform reasoning and make inferences on it by means of a semantic-based logic representation of the schema itself.

With this goal in mind, this section presents a linear syntax that fully covers the set of graphical symbols of ORM2. For each construct ϕ in the syntax, its corresponding set-theoretic semantics expressed in relational algebra is also introduced in table 3 (where O denotes an object type). The signature $\mathcal{S}$ of the linear ORM2 syntax is made of:

- A set $\mathcal{E}$ of *entity type* symbols;
- a set $\mathcal{V}$ of *value type* symbols;
- a set $\mathcal{R}$ of *relation* symbols;
- a set $\mathcal{A}$ of *role* symbols;
- a set $\mathcal{D}$ of *domain* symbols, and
- a set Λ of pairwise disjoint sets of values;
- for each $D \in \mathcal{D}$, an injective extension function $\Lambda_{(\cdot)} : \mathcal{D} \rightarrow \Lambda$ associating each domain symbol D to an extension Λ_D;
- a binary relation $\varrho \subseteq \mathcal{R} \times \mathcal{A}$ linking role symbols to relation symbols. We take the pair $R.a$ as the atomic elements of the syntax, and we call it *localised role*. Given a relation symbol R, $\varrho_R = \{R.a | R.a \in \varrho\}$ is the set of localised roles with respect to R; $arity(R) = |\varrho_R|$ is the arity of the relation R;

– for each relation symbol R, a bijection $\tau_R \colon \varrho_R \to [1..|\varrho_R|]$ mapping each element in ϱ_R to an element in the finite sequence of natural numbers $[1..|\varrho_R|]$. We also define $\tau = \bigcup_{R \in \mathcal{R}} \tau_R$. The mapping τ_R guarantees a correspondence between role components and argument positions in a relation, so that we can freely choose between an 'attribute-based' and a 'positional-based' representation.

Now, given the signature $\mathcal{S}$, an **ORM2 *conceptual schema*** Σ over $\mathcal{S}$ includes a finite combination of the constructs in table 3. The list of constraints graphically introduced Section2 can now be re-formulated using the new syntax, where: (1) TYPE is for linking a role to its object type; (2) FREQ indicates the frequency occurrence applied to a role sequence; (3) MAND is for mandatory participation; (4) $\mathsf{R\text{-}SET_H}$ is for the family of set-comparison constraints; (5) $\mathsf{O\text{-}SET_H}$ is for the family of subtyping constraints; (6) O-CARD and R-CARD indicate object and role cardinality respectively; and (7) OBJ expresses the objectification of a relation, and associates a name to the resulting objectified type; (8) RING is for ring constraints; and (9) V-VAL enumerate the values that are in a value type and role, respectively.

The linear syntax does not only provide a way to fully represent the ORM2 graphical notation but, in some respect, it represents also a genuine generalisation of it. In particular, as the FREQ and MAND are concerned, external and internal forms of the constraints are represented by means of different specialisations of the same constructs; the FREQ construct can now be applied to arbitrary role sequences no matter about the arity of relations involved, and the same holds for the $\mathsf{R\text{-}SET_H}$ constructs. Moreover, additional sequences of role pairs (see $\bowtie_{\mathbf{R}}$, and $\bowtie_{\mathbf{S}}$) are among the arguments of both FREQ and $\mathsf{R\text{-}SET_H}$, and used to specify the roles where the joins must be computed. $\mathsf{R\text{-}SET_H}$ constraints are equipped with a function μ that fixes the mapping between the constrained roles. No specific construct has been added to represent uniqueness constraints, since they can be naturally viewed as frequency occurrence constraints with a fixed range of $\mathsf{min} = 1, \mathsf{max} = 1$. Moreover, several constraints that appear among the primitive symbols in ORM2, can now be easily derived by combining (and specialising) the constructs of the linear syntax as shown in table 3 (note that, in the case of the **exclusive-or** constraint, since no join operation is needed, we simply omitted to include this information in the specification of the $\mathsf{R\text{-}SET_{Exc}}$. Note that, as in the case of the simple $\mathsf{R\text{-}SET_H}$ constraints, the actual version of the exclusive-or could be further generalized by the introduction of join paths specification). In particular, the 'strict' version of the subtyping relation, that is assumed as primitive in [4], is seen here as a derived constraint: Given a non-strict semantics for the subtyping relation, the strict one can be represented by a combination of partition, cardinality constraint, and the introduction of a a new fresh object type symbol ('equality' can also be expressed using a similar pattern, where the cardinality of the new introduced symbol is zero).

Table 2 shows how the new introduced syntax can be used to encode conceptual schemas that have been originally specified in graphical terms. Role names result from the concatenation of 'relation name', 'dot', and 'name of the attached type'. New fresh suffixes are introduced whenever more than one role in a relation is attached to the same type (e.g. see reportsTo in the example).

The semantics of a conceptual schema Σ is formally specified through the notion of interpretation. Let Σ be an ORM2 conceptual schema over a signature $\mathcal{S}$, an interpretation for Σ is a triple $\mathcal{I} = \langle \Delta^{\mathcal{I}}, (\cdot)^{\mathcal{I}}, \text{ID}^{\mathcal{I}} \rangle$, where

– $\Delta^{\mathcal{I}}$ is a set, the interpretation domain, properly including each $\Lambda_D \in \Lambda$;
– $(\cdot)^{\mathcal{I}}$ is a total function such that:
  (i) For each $E \in \mathcal{E}$, $E^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \setminus \bigcup_{D_j \in \mathcal{D}} \Lambda_{D_j}$;

Table 1. Derived constraints.

Uniqueness:	$\mathsf{FREQ}(\{R^1.a_{11},\ldots,R^1.a_{1n},\ldots,R^k.a_{k1},\ldots,R^k.a_{km}\},\Join_{\mathbf{R}},\langle 1,1\rangle)$
Role value:	$\mathsf{TYPE}(R.a,V^*)$ where V^* is a new fresh value type symbol $\mathsf{V\text{-}VAL}(V^*)\subseteq\{v_1^D,\ldots,v_n^D\}$
Equality:	$\mathsf{R\text{-}SET_{Sub}}((\{R^1.a_{11},\ldots,R^1.a_{1n},\ldots,R^k.a_{k1},\ldots,R^k.a_{km}\},\Join_{\mathbf{R}}),$ $(\{S^1.b_{11},\ldots,S^1.b_{1v},\ldots,S^q.b_{q1},\ldots,S^q.b_{qw}\},\Join_{\mathbf{S}}),\mu)$ $\mathsf{R\text{-}SET_{Sub}}((\{S^1.b_{11},\ldots,S^1.b_{1v},\ldots,S^q.b_{q1},\ldots,S^q.b_{qw}\},\Join_{\mathbf{S}})$ $(\{R^1.a_{11},\ldots,R^1.a_{1n},\ldots,R^k.a_{k1},\ldots,R^k.a_{km}\},\Join_{\mathbf{R}}),\mu^-)$
Exclusive-Or:	$\mathsf{MAND}(\{R^1.a_{11},\ldots,R^1.a_{1n},\ldots,R^k.a_{k1},\ldots,R^k.a_{km}\},O)$ $\mathsf{R\text{-}SET_{Exc}}((\{R^1.a_{11},\ldots,R^1.a_{1n}\}),(\{R^2.a_{21},\ldots,R^2.a_{2n}\}),\mu_1)$ $\mathsf{R\text{-}SET_{Exc}}((\{R^1.a_{11},\ldots,R^1.a_{1n}\}),(\{R^3.a_{31},\ldots,R^3.a_{3n}\}),\mu_2),\cdots,$ $\mathsf{R\text{-}SET_{Exc}}((\{R^{k-1}.a_{k-11},\ldots,R^{k-1}.a_{k-1n}\}),(\{R^k.a_{k1},\ldots,R^k.a_{kn}\}),\mu_k)$
Partition:	$\mathsf{O\text{-}SET_{Tot}}(\{O_1,\ldots,O_n\},O)$ $\mathsf{O\text{-}SET_{Ex}}(\{O_1,\ldots,O_n\},O)$
Strict Subtyping:	$\mathsf{O\text{-}SET_{Tot}}(\{O_1,O^*\},O)$ $\mathsf{O\text{-}SET_{Ex}}(\{O_1,O^*\},O)$ $\mathsf{O\text{-}CARD}(O^*)=(1,\mathsf{inf})$ where O^* is a new fresh object type symbol

(ii) For each $V\in\mathcal{V}$, $V^{\mathcal{I}}\subseteq\Lambda_{D_i}$, for some $\Lambda_{D_i}\in\Lambda$;
(iii) For each $R\in\mathcal{R}$, $R^{\mathcal{I}}\subseteq\{\langle o_1,\ldots,o_{|\varrho_R|}\rangle\mid o_i\in\Delta^{\mathcal{I}},\ \text{for } i=1,\ldots,|\varrho_R|\}$.

- $\textsc{id}^{\mathcal{I}}\colon\bigcup_{R\in\mathcal{R}}R^{\mathcal{I}}\to\Delta^{\mathcal{I}}$ is a total injective function mapping each tuple in the interpretation of a symbol R in $\mathcal{R}$, with $|\varrho_R|\geq 2$, with a unique identifier. Given a tuple t, we call $\textsc{id}^{\mathcal{I}}(t)$ the 'objectification' of t.

Intuitively, the main components of an interpretation are the *interpretation domain* and the *interpretation function*. The interpretation function associates to each entity type E a subset $E^{\mathcal{I}}$, to each value type V a subset of values $V^{\mathcal{I}}$ that are in the extension Λ_{D_i} for some D_i, and to each relation R of arity n a subset $R^{\mathcal{I}}$ of tuples of length n. An interpretation $\mathcal{I}$ of a schema Σ is called a **legal database state** if it satisfies the conditions in table 3. The notion of legal database state $\mathcal{I}$ can now be used to formally characterised the reasoning tasks introduced above:

- **Strong consistency**. A conceptual schema Σ is *consistent* if there exist a legal database state $\mathcal{I}$ such that $O^{\mathcal{I}}\neq\varnothing$ *for every* $O\in\mathcal{E}\cup\mathcal{V}$.
- **Object type (relation) consistency**. An object type $O\in\mathcal{E}\cup\mathcal{V}$ ($R\in\mathcal{R}$) is *consistent* w.r.t. a schema Σ if there exist a legal database state for Σ s.t. $O^{\mathcal{I}}\neq\varnothing$ ($R^{\mathcal{I}}\neq\varnothing$).
- **Constraint entailment**. A constraint ϕ *is entailed* by a conceptual schema Σ if and only if each interpretation $\mathcal{I}$ that is legal database state for Σ is also a legal database state for ϕ, denoted by $\Sigma\vDash\phi$.

As for the mutual reducibility of the above reasoning tasks, it is known that concept consistency and logical entailment generalise concept satisfiability and concept subsumption [12]. Note also that as far as ORM2 is concerned, the **Partial schema consistency** service (where, 'A conceptual schema Σ is *partially consistent* if there exist a legal database state $\mathcal{I}$ such that $O^{\mathcal{I}}\neq\varnothing$ *for some* $O\in\mathcal{E}\cup\mathcal{V}$') becomes a non-sense. By means of the object cardinality constraint, it is always possibile to draw a conceptual schema that has no model, and such that there exists at least one non-empty object (e.g., think about two disjoint object types that are declared to be subtypes of a common object type whose cardinality is stated to be greater than 1).

The examples below show how the introduced semantics can be used to automatise consistency check and constraint entailment tasks, respectively.

Example 1. Let us take from 3 the fragment made of: (i) TYPE({reportsTo.sub},Admin), TYPE({reportsTo.obj},AreaManager), and (ii) MAND({reportsTo.sub},Admin). Then, let us add to this fragment a new entity type ICT, and a ternary relation supports with: TYPE(supports.first,ICT), TYPE(supports.second,AreaManager), TYPE(supports.third,Date), MAND({supports.first},ICT), MAND({supports.second},AreaManager), and

$$\text{R-SET}_{\text{Exc}}((\{\text{reportsTo.sub}\}, \{\text{reportsTo.obj}\}), (\{\text{supports.first}\}, \{\text{supports.second}\}), \{\bullet\})$$

Table 2. Constraints C1-C11 below represent a fragment of 3. On the right side, the way the graphical notation is captured by the linear syntax, given the signature specified in the first rows. The symbol {•} has been introduced here to stress that the required information is redundant.

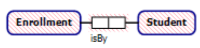
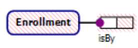
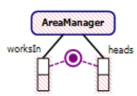
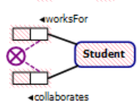
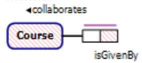
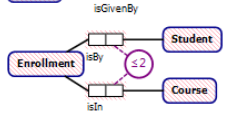
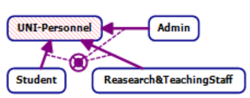
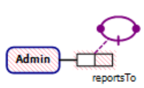
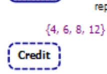
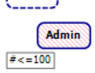
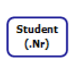
		ENTITYTYPES:{Enrollement,Student,...} VALUETYPES:{Credit,Student-Nr,...} RELATIONS:{isIn,isBy,worksFor,collaborates,hasStudent-Nr,...}
C1.		TYPE(isBy.enrollment,Enrollment) TYPE(isBy.student,Student)
C2.		MAND({isBy.enrollment},Enrollment)
C3.		MAND({worksIn.areaManager,heads.areaManager},AreaManager)
C4.		$\text{R-SET}_{\text{Exc}}$(({worksFor.student},{•}),({collaborates.student},{•}) {(worksFor.student,collaborates.student)})
C5.		FREQ({isGivenBy.course,isGivenBy.student},{•},⟨1, 1⟩)
C6.		FREQ({isBy.student,isIn.course}, {isBy.enrollment,isIn.enrollment},⟨1, 2⟩)
C7.		$\text{O-SET}_{\text{Tot}}$({R&TStaff,Student,Admin},UNI-Personnel) O-SET_{Ex}({R&TStaff,Student,Admin},UNI-Personnel)
C8.		$\text{RING}_{\text{Asym}}$(reportsTo.sub, reportsTo.obj)
C9.		V-VAL(Credit)={4,6,8,12}
C10.		O-CARD(Admin)=(0,100)
C11.		TYPE(hasStudent-Nr.student,Student) TYPE(hasStudent-Nr.student-Nr,Student-Nr) FREQ({hasStudent-Nr.student},{•}, ⟨1, 1⟩) FREQ({hasStudent-Nr.student-Nr},{•}, ⟨1, 1⟩) MAND({hasStudent-Nr.student},Student)

Table 3. Linear Syntax and Semantics table.

Syntax	Semantics
$\mathsf{TYPE} \subseteq \varrho \times (\mathcal{E} \cup \mathcal{V})$	If $\mathsf{TYPE}(R.a, O) \in \Sigma$ then $\Pi_{R.a} R^{\mathcal{I}} \subseteq O^{\mathcal{I}}$
$\mathsf{FREQ} \subseteq \wp(\varrho) \times (\wp(\varrho) \times \wp(\varrho)) \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$	If $\mathsf{FREQ}(\{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, \bowtie_{\mathbf{R}}, \langle \mathsf{min}, \mathsf{max} \rangle) \in \Sigma$ then $\Pi_{\varrho^{\mathsf{C}}}(R^{1^{\mathcal{I}}} \bowtie_{\mathbf{R}} \cdots \bowtie_{\mathbf{R}} R^{k^{\mathcal{I}}}) \subseteq \{\overline{x} \mid \mathsf{min} \leq \sharp\{\sigma_{\overline{x}=\varrho^{\mathsf{C}}}(R^{1^{\mathcal{I}}} \bowtie_{\mathbf{R}} \cdots \bowtie_{\mathbf{R}} R^{k^{\mathcal{I}}})\} \leq \mathsf{max}\}$
where: (1) $\varrho^{\mathsf{C}} = \{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}$, and $\overline{x} = \varrho^{\mathsf{C}}$ iff $R^1.a_{11} = x^1_{\tau(R^1.a_{11})}, \dots, R^k.a_{km} = x^k_{\tau(R^k.a_{km})}$ (2) $\bowtie_{\mathbf{R}} = \{\dots, \langle R^i.a_{i\mathbf{v}} = R^j.a_{j\mathbf{w}} \rangle, \dots\}$, with $i \neq j$ and $1 \leq i, j \leq k$, is the finite set of role pairs where the joins must be computed (e.g. given sequence of n relations, $\mathbf{R}$, $\mid \bowtie_{\mathbf{R}} \mid = n - 1$), and $R^x.a_{x\mathbf{y}} \in \varrho^x_R$ for any $R^x \in \mathcal{R}$	
$\mathsf{MAND} \subseteq \wp(\varrho) \times (\mathcal{E} \cup \mathcal{V})$	If $\mathsf{MAND}(\{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, O) \in \Sigma$ then $O^{\mathcal{I}} \subseteq \Pi_{R^1.a_{11}} R^{1^{\mathcal{I}}} \cup \dots \cup \Pi_{R^1.a_{1n}} R^{1^{\mathcal{I}}} \cup \dots \cup \Pi_{R^k.a_{k1}} R^{k^{\mathcal{I}}} \cup \dots \cup \Pi_{R^k.a_{km}} R^{k^{\mathcal{I}}}$
$\mathsf{R\text{-}SET}_{\mathsf{H}} \subseteq ((\wp(\varrho) \times (\wp(\varrho) \times \wp(\varrho))) \times (\wp(\varrho) \times (\wp(\varrho) \times \wp(\varrho))) \times (\mu\colon \varrho \to \varrho)$	• If $\mathsf{R\text{-}SET}_{\mathsf{Sub}}((\{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, \bowtie_{\mathbf{R}}),$ $(\{S^1.b_{11}, \dots, S^1.b_{1v}, \dots, S^q.b_{q1}, \dots, S^q.b_{qw}\}, \bowtie_{\mathbf{S}}, \mu) \in \Sigma$ then $\Pi_{\varrho^{\mathsf{C_A}}}(R^{1^{\mathcal{I}}} \bowtie_{\mathbf{R}} \cdots \bowtie_{\mathbf{R}} R^{k^{\mathcal{I}}}) \subseteq \Pi_{\varrho^{\mathsf{C_B}}}(S^{1^{\mathcal{I}}} \bowtie_{\mathbf{S}} \cdots \bowtie_{\mathbf{S}} S^{q^{\mathcal{I}}})$ • If $\mathsf{R\text{-}SET}_{\mathsf{Exc}}((\{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, \bowtie_{\mathbf{R}}),$ $(\{S^1.b_{11}, \dots, S^1.b_{1v}, \dots, S^q.b_{q1}, \dots, S^q.b_{qw}\}, \bowtie_{\mathbf{S}}, \mu) \in \Sigma$ then $\Pi_{\varrho^{\mathsf{C_A}}}(R^{1^{\mathcal{I}}} \bowtie_{\mathbf{R}} \cdots \bowtie_{\mathbf{R}} R^{k^{\mathcal{I}}}) \cap \Pi_{\varrho^{\mathsf{C_B}}}(S^{1^{\mathcal{I}}} \bowtie_{\mathbf{S}} \cdots \bowtie_{\mathbf{S}} S^{q^{\mathcal{I}}}) = \varnothing$
where: (1) $\varrho^{\mathsf{C_A}} = \{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}$, and $\varrho^{\mathsf{C_B}} = \{S^1.b_{11}, \dots, S^1.b_{1v}, \dots, S^q.b_{q1}, \dots, S^q.b_{qw}\}$ (2) μ is a partial bijection s.t. for any $\langle \varrho^{\mathsf{C_A}}, \varrho^{\mathsf{C_B}}, \mu \rangle \in \mathsf{R\text{-}SET}_{\mathsf{H}}$, we have $\varrho^{\mathsf{C}_A} = \{R.a \mid \mu(R.a) \in \varrho^{\mathsf{C}_B}\}$, and (3) $\mathsf{H} = \{\mathsf{Sub}, \mathsf{Exc}\}$	
$\mathsf{O\text{-}SET}_{\mathsf{H}} \subseteq \wp(\mathcal{E} \cup \mathcal{V}) \times \mathcal{E} \cup \mathcal{V}$ where $\mathsf{H} = \{\mathsf{Isa}, \mathsf{Tot}, \mathsf{Ex}\}$	• If $\mathsf{O\text{-}SET}_{\mathsf{Isa}}(\{O_1, \dots, O_n\}, O) \in \Sigma$ then $O_i^{\mathcal{I}} \subseteq O^{\mathcal{I}}$ for $1 \leq i \leq n$ • If $\mathsf{O\text{-}SET}_{\mathsf{Tot}}(\{O_1, \dots, O_n\}, O) \in \Sigma$ then $O^{\mathcal{I}} \subseteq \bigcup_{i=1}^{n} O_i^{\mathcal{I}}$ • If $\mathsf{O\text{-}SET}_{\mathsf{Ex}}(\{O_1, \dots, O_n\}, O) \in \Sigma$ then $\mathsf{O\text{-}SET}_{\mathsf{Isa}}(\{O_1, \dots, O_n\}, O) \in \Sigma$ and $O_i^{\mathcal{I}} \cap O_j^{\mathcal{I}} = \varnothing$ for any $1 \leq i < j \leq n$
$\mathsf{O\text{-}CARD} \subseteq (\mathcal{E} \cup \mathcal{V}) \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$	If $\mathsf{O\text{-}CARD}(O) = (\mathsf{min}, \mathsf{max}) \in \Sigma$ then $\mathsf{min} \leq \sharp\{o \mid o \in O^{\mathcal{I}}\} \leq \mathsf{max}$
$\mathsf{R\text{-}CARD} \subseteq \wp(\varrho) \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$	If $\mathsf{R\text{-}CARD}(R.a) = (\mathsf{min}, \mathsf{max}) \in \Sigma$ then $\mathsf{min} \leq \sharp\{o \mid o \in \Pi_{R.a} R^{\mathcal{I}}\} \leq \mathsf{max}$
$\mathsf{OBJ} \subseteq \mathcal{R} \times (\mathcal{E} \cup \mathcal{V})$	If $\mathsf{OBJ}(R, O) \in \Sigma$ then $\mathrm{ID}^{\mathcal{I}}(R^{\mathcal{I}}) = O^{\mathcal{I}}$
$\mathsf{RING}_{\mathsf{J}} \subseteq \wp(\varrho \times \varrho)$ where $\mathsf{J} = \{\mathsf{Irr}, \mathsf{Asym}, \mathsf{Trans}, \mathsf{Intr}, \mathsf{Antisym}, \mathsf{Acyclic}, \mathsf{Sym}, \mathsf{Ref}\}$	If $\mathsf{RING}_{\mathsf{J}}(R.a, R.b) \in \Sigma$ then $\Pi_{(R.a, R.b)} R^{\mathcal{I}}$ is *irreflexive, asymmetric, transitive, intransitive, antisymmetric, acyclic, symmetric, reflexive*
$\mathsf{V\text{-}VAL}\colon \mathcal{V} \to \wp(\Lambda_D)$ for some $\Lambda_D \in \Lambda$	If $\mathsf{V\text{-}VAL}(V) = \{v_1^D, \dots, v_n^D\} \in \Sigma$ then $V^{\mathcal{I}} = \{v_1^D, \dots, v_n^D\}$ for some D

An interpretation $\mathcal{I}$ is a model of the new schema if:

1. $\Pi_{\mathsf{reportsTo.sub}}\mathsf{ReportsTo}^{\mathcal{I}} \cap \Pi_{\mathsf{supports.first}}\mathsf{Supports}^{\mathcal{I}} = \varnothing$ [R-SET$_{\mathsf{Exc}}$]
2. $\mathsf{Admin}^{\mathcal{I}} \subseteq \Pi_{\mathsf{reportsTo.sub}}\mathsf{ReportsTo}^{\mathcal{I}}$ [TYPE]
3. $\mathsf{ICT}^{\mathcal{I}} \subseteq \Pi_{\mathsf{supports.first}}\mathsf{Supports}^{\mathcal{I}}$ [TYPE]

Let us suppose now that the constraint O-SET$_{\mathsf{Isa}}$(ICT,Admin) is added to the schema. In order to be a model, $\mathcal{I}$ must also satisfy the condition:

4. $\mathsf{ICT}^{\mathcal{I}} \subseteq \mathsf{Admin}^{\mathcal{I}}$ that, together with 2., implies
5. $\mathsf{ICT}^{\mathcal{I}} \subseteq \Pi_{\mathsf{reportsTo.sub}}\mathsf{ReportsTo}^{\mathcal{I}}$ but then, by 3. and 4.
6. $\Pi_{\mathsf{reportsTo.sub}}\mathsf{ReportsTo}^{\mathcal{I}} \cap \Pi_{\mathsf{supports.first}}\mathsf{Supports}^{\mathcal{I}} \neq \varnothing$

which contradicts our assumption 1. Therefore, O-SET$_{\mathsf{Isa}}$(ICT,Admin) causes the entity type ICT to be inconsistent. But then, the relation supports also becomes inconsistent, and, due to the mandatory participation, the same happens to AreaManager, to the relation reportsTo and to the associated Admin. This simple argument proves that the schema is *partially consistent*, i.e. it admits a model where everything is empty except Date.

Example 2. Again from the example 3, let us select the schema made of:
O-SET$_\bigstar$({R&TStaff,Student,Admin},UNI-Personnel), where $\bigstar = \{\mathsf{Ex}, \mathsf{Tot}\}$. Then let us add the new entity type LazyPeople with: (i) O-SET$_{\mathsf{Ex}}$({R&TStaff,LazyPeople},UNI-Personnel) and (ii) O-SET$_{\mathsf{Ex}}$({Admin,LazyPeople},UNI-Personnel). Then, an interpretation $\mathcal{I}$ is a legal database state of the schema if:

1. $\bigstar^{\mathcal{I}} \subseteq \mathsf{UNI\text{-}Personnel}^{\mathcal{I}}$, where $\bigstar = \{\mathsf{Student}, \mathsf{Admin}, \mathsf{R\&TStaff}, \mathsf{LazyPeople}\}$
2. $\mathsf{UNI\text{-}Personell}^{\mathcal{I}} \subseteq \mathsf{R\&TStaff}^{\mathcal{I}} \cup \mathsf{Admin}^{\mathcal{I}} \cup \mathsf{Student}^{\mathcal{I}}$
3. the involved entity types are pairwise disjoint, in particular:
   $\mathsf{LazyPeople}^{\mathcal{I}} \cap \mathsf{R\&TStaff}^{\mathcal{I}} = \varnothing$, and $\mathsf{LazyPeople}^{\mathcal{I}} \cap \mathsf{Admin}^{\mathcal{I}} = \varnothing$

Now, let us consider the new constraint O-SET$_{\mathsf{Isa}}$({LazyPeople},Student). An interpretation satisfies it if $\mathsf{LazyPeople}^{\mathcal{I}} \subseteq \mathsf{Student}^{\mathcal{I}}$, but this is actually what the conditions 1-3 imply. Therefore, it turns out that all the interpretations that are models of the schema are also model of O-SET$_{\mathsf{Isa}}$({LazyPeople},Student), namely, the constraint is *entailed* by the schema.

4 FOL encoding of ORM2 conceptual schema

The FOL semantics for ORM2 is based on a signature $\mathcal{S}_{\mathsf{FOL}}$ that perfectly matches the one of the linear syntax:

(i) $E_1, E_2, \ldots, E_n$ 1-ary predicates for *entity types*;
(ii) $V_1, V_2, \ldots, V_m$ 1-ary predicates for *value types*;
(iii) $D_1, D_2, \ldots, D_l$ 1-ary predicates for *domain symbols*;
(iv) $R_1, R_2, \ldots, R_k$ n-ary predicates for *relations*;
(v) a countable set of *constants* $d_1, d_2, \ldots$;
(vi) a set $\textsc{id}^2, \ldots, \textsc{id}^{n_{max}}$ of functions, $n_{max} = \mathsf{max}\{|\varrho_R||R \in \mathcal{R}\}$.

The FOL encoding of the ORM2 semantics introduced in the previous section is then as follows:

– Background domain axioms:

$$\forall x.E_i(x) \rightarrow \neg(D_1(x) \vee \cdots \vee D_l(x)), \text{ for } 1 \leq i \leq n \quad (1)$$

$$\forall x.V_i(x) \rightarrow D_j(x), \text{ for } 1 \leq i \leq m \quad (2)$$

$$\forall x.D_i(x) \leftrightarrow (x = d_1 \vee x = d_2 \vee \ldots), \text{ for all } d_i \in \Lambda_{D_i} \quad (3)$$

$$\forall x_1, \ldots, x_n, z_1, \ldots, z_n.\textsc{id}(\overline{\mathbf{x}}) = \textsc{id}(\overline{\mathbf{z}}) \leftrightarrow \overline{\mathbf{x}} = \overline{\mathbf{s}}, \text{ for } n = 1, \ldots, n_{max} \quad (4)$$

The above set of *background axioms* is needed in order to force the interpretation of the symbols in the FOL knowledge bases (KBs) to be correct w.r.t. the intended semantics of the corresponding ORM2 symbols. In particular, axiom (1) forces the interpretation of each entity type to be disjoint from the interpretation of the domain symbols; axiom (2) says that objects in the interpretation of a value type must be also in the interpretation of a specific domain symbol; axiom (3) forces the interpretation of a domain symbols to be among the set of values predefined by $\Lambda_{(\cdot)}$, while axiom (4) captures the injective nature of each ID function and the fact that tuples of different length will never agree on the same identifier. We also add the axioms $a \neq b$, for any pair of distinct constants a and b (UNA). A FO interpretation is a **model** (or, a 'legal database state') for an ORM2$^{\mathsf{plus}}$ schema if it satisfies the background axioms and the corresponding FOL KB built as described in tables 5 and 4. We can prove that, when the schema is restricted to a NIAM schema, the models of the corresponding ORM2$^{\mathsf{plus}}$ schema are the same as the FO models of the NIAM schema as specified in [5].

Table 4. Linear Syntax (■) and FOL Semantics (□)

■ $\mathsf{O\text{-}SET_H} \subseteq \wp(\mathcal{E} \cup \mathcal{V}) \times \mathcal{E} \cup \mathcal{V}$ where $\mathsf{H} = \{\mathsf{Isa}, \mathsf{Tot}, \mathsf{Ex}\}$ □ • If $\mathsf{O\text{-}SET_{Isa}}(\{O_1, \dots, O_n\}, O) \in \Sigma$ then $\forall y.O_i(y) \rightarrow O(y)$ for all $i = 1, \dots, n$ • If $\mathsf{O\text{-}SET_{Tot}}(\{O_1, \dots, O_n\}, O) \in \Sigma$ then $\begin{cases} \forall y.O_i(y) \rightarrow O(y) \\ \forall y.O(y) \rightarrow O_1(y) \vee \cdots \vee O_n(y), \text{ for all } i = 1, \dots, n \end{cases}$ • If $\mathsf{O\text{-}SET_{Ex}}(\{O_1, \dots, O_n\}, O) \in \Sigma$ then $\begin{cases} \forall y.O_1(y) \rightarrow O(y) \wedge \neg O_2(y) \wedge \cdots \wedge \neg O_n(y) \\ \forall y.O_2(y) \rightarrow O(y) \wedge \neg O_3(y) \wedge \cdots \wedge \neg O_{n-1}(y) \\ \dots \\ \forall y.O_{n-1}(y) \rightarrow O(y) \wedge \neg O_1(y) \\ \forall y.O_n(y) \rightarrow O(y) \end{cases}$
■ $\mathsf{O\text{-}CARD} \subseteq (\mathcal{E} \cup \mathcal{V}) \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$ □ If $\mathsf{O\text{-}CARD}(O) = (\mathsf{min}, \mathsf{max})) \in \Sigma$ then $\exists^{\geq \mathsf{min}} y.O(y) \wedge \exists^{\leq \mathsf{max}} y.O(y)$
■ $\mathsf{R\text{-}CARD} \subseteq \wp(\varrho) \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$ □ If $\mathsf{R\text{-}CARD}(R.a) = (\mathsf{min}, \mathsf{max})) \in \Sigma$ then $\exists^{\geq \mathsf{min}} x_{\tau(R.a)}.R(x_1 \dots x_{\tau(R.a)} \dots x_n) \wedge \exists^{\leq \mathsf{max}} x_{\tau(R.a)}.R(x_1 \dots x_{\tau(R.a)} \dots x_n)$
■ $\mathsf{OBJ} \subseteq \mathcal{R} \times (\mathcal{E} \cup \mathcal{V})$ □ If $\mathsf{OBJ}(R, O) \in \Sigma$ then $\forall x.O(x) \leftrightarrow \exists \overline{y}.R(\overline{y}) \wedge \mathrm{ID}^{\lvert \varrho_R \rvert}(\overline{y}) = x$
■ $\mathsf{RING_J} \subseteq \wp(\varrho \times \varrho)$ where $\mathsf{J} = \{\mathsf{Irr}, \mathsf{Asym}, \mathsf{Trans}, \mathsf{Intr}, \mathsf{Antisym}, \mathsf{Acyclic}, \mathsf{Sym}, \mathsf{Ref}\}$ □ E.g. If $\mathsf{RING_{Irr}}(R.a, R.b) \in \Sigma$ then $\forall x_{\tau(R.a)}, x_{\tau(R.b)}.R(x_{\tau(R.a)}, x_{\tau(R.b)}) \rightarrow \neg R(x_{\tau(R.b)}, x_{\tau(R.a)})$
■ $\mathsf{V\text{-}VAL}\colon \mathcal{V} \rightarrow \wp(\Lambda_D)$ for some $\Lambda_D \in \Lambda$ (where $\Lambda_{(\cdot)}$ associates an extension to each domain symbol) □ If $\mathsf{V\text{-}VAL}(V) = \{d_1, \dots, d_n\} \in \Sigma$ then $\forall x.V(x) \rightarrow (x = d_1) \vee \cdots \vee (x = d_n)$

Let Σ^{FOL} be the FOL knowledge base over the signature $\mathcal{S}_{\mathsf{FOL}}$ resulting from the encoding above. Now, it is easy to see that an interpretation satisfies an ORM2 schema if and only if it satisfies the corresponding FOL knowledge base Σ^{FOL}. Therefore, we have that the following holds:

Theorem 1. *Let Σ be an ORM2 conceptual schema and Σ^{FOL} the FOL knowledge base constructed according to the mapping above. Then every (interpretation that is a) legal database state of Σ is a model of Σ^{FOL}, and viceversa.*

In order to make this explicit, it is useful to split the proof in two parts: first, show that the FOL background axioms exactly represent the conditions for an interpretation $\mathcal{I}$ of a conceptual schema as introduced in Section 3, then consider separately each ORM2 construct with its corresponding FOL assertion. Notice that the ORM2 conceptual schema Σ and the corresponding FOL knowledge base Σ^{FOL} agree over the same signature.

Table 5. Linear Syntax (■) and FOL Semantics (□) table.

■ $\mathsf{TYPE} \subseteq \varrho \times (\mathcal{E} \cup \mathcal{V})$

□ If $\mathsf{TYPE}(R.a, O) \in \Sigma$ then $\forall x_1 \ldots x_{\tau(R.a)} \ldots x_n . R(x_1, \ldots, x_{\tau(R.a)}, \ldots, x_n) \rightarrow O(x_{\tau(R.a)})$

■ $\mathsf{FREQ} \subseteq \wp(\varrho) \times (\wp(\varrho) \times \wp(\varrho)) \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$

□ If $\mathsf{FREQ}(\{R^1.a_{11}, \ldots, R^1.a_{1n}, \ldots, R^k.a_{k1}, \ldots, R^k.a_{km}\}, \bowtie_{\mathbf{R}}, (\mathsf{min}, \mathsf{max})) \in \Sigma$ then

$$\forall \overline{y}[\exists \overline{x}^1 \ldots \overline{x}^k (\bigwedge_{j=1}^{k} R^j(\overline{x}^j) \wedge \bigwedge_{\mathbf{i1}=1}^{n} (x^1_{\tau(R^1.a_{1\mathbf{i1}})} = y_{1\mathbf{i1}}) \wedge \cdots \wedge \bigwedge_{\mathbf{ik}=1}^{m} (x^1_{\tau(R^1.a_{1\mathbf{ik}})} = y_{1\mathbf{ik}}) \wedge \bigwedge_{\bowtie_{\mathbf{R}}} (x^{r^+}_{\tau(R^{r^+}.a_{r^+\mathbf{v}_r})} = x^{r^-}_{\tau(R^{r^-}.a_{r^-\mathbf{w}_r})}))] \rightarrow$$

$$\exists^{\geq \mathsf{min}; \leq \mathsf{max}} \overline{y}[\exists \overline{x}^1 \ldots \overline{x}^k (\bigwedge_{j=1}^{k} R^j(\overline{x}^j) \wedge \bigwedge_{\mathbf{i1}=1}^{n} (x^1_{\tau(R^1.a_{1\mathbf{i1}})} = y_{1\mathbf{i1}}) \wedge \cdots \wedge \bigwedge_{\mathbf{ik}=1}^{m} (x^1_{\tau(R^1.a_{1\mathbf{ik}})} = y_{1\mathbf{ik}}) \wedge (\bigwedge_{\bowtie_{\mathbf{R}}} x^{r^+}_{\tau(R^{r^+}.a_{r^+\mathbf{v}_r})} = x^{r^-}_{\tau(R^{r^-}.a_{r^-\mathbf{w}_r})}))]$$

where:

(1) $\bowtie_{\mathbf{R}} = \{\ldots, \langle R^i.a_{i\mathbf{v}} = R^j.a_{j\mathbf{w}} \rangle, \ldots\}$, with $i \neq j$ and $1 \leq i, j \leq k$, is the finite set of join (role) pairs (given k relations $\mathbf{R}$, $| \bowtie_{\mathbf{R}} | = k - 1$)

(2) $R^i.a_{i\mathbf{x}} \in \varrho_{R^i}$ for any $R^i \in \mathcal{R}$, and

(3) the equalities in $\bigwedge_{\bowtie_{\mathbf{R}}}$ are specified according to $\bowtie_{\mathbf{R}}$

(e.g. given $\overline{x}^1, \overline{x}^2, \overline{x}^3$ s.t. $R^1(\overline{x}^1), R^2(\overline{x}^2), R^3(\overline{x}^3)$, if $\bowtie_{\mathbf{R}} = \{(R^1.a, R^2.b), (R^2.c, R^3.d)\}$ then $\bigwedge_{\bowtie_{\mathbf{R}}} =_{\mathsf{def}} (x^1_{\tau(R^1.a)} = x^2_{\tau(R^2.b)}) \wedge (x^2_{\tau(R^2.c)} = x^3_{\tau(R^3.d)})$

■ $\mathsf{MAND} \subseteq \wp(\varrho) \times (\mathcal{E} \cup \mathcal{V})$

□ If $\mathsf{MAND}(\{R^1.a_{11}, \ldots, R^1.a_{1n}, \ldots, R^k.a_{k1}, \ldots, R^k.a_{km}\}, O) \in \Sigma$ then

$$\forall y[A(y) \rightarrow \bigvee_{i=1}^{n} \exists \overline{z}^i (R^1(\overline{z}^i) \wedge (z^i_{\tau(R^1.a_{1i})} = y) \vee \cdots \vee \bigvee_{j=1}^{m} \exists \overline{z}^j . R^k(\overline{z}^j) \wedge (z^j_{\tau(R^k.a_{kj})} = y))]$$

■ $\mathsf{R\text{-}SET}_{\mathsf{H}} \subseteq (\wp(\varrho) \times (\wp(\varrho) \times \wp(\varrho))) \times (\wp(\varrho) \times (\wp(\varrho) \times \wp(\varrho))) \times (\mu\colon \varrho \rightarrow \varrho)$ where $\mathsf{H} = \{\mathsf{Sub}, \mathsf{Exc}\}$

□ • If $\mathsf{R\text{-}SET}_{\mathsf{Sub}}((\{R^1.a_{11}, \ldots, R^1.a_{1n}, \ldots, R^k.a_{k1}, \ldots, R^k.a_{km}\}, \bowtie_{\mathbf{R}}), (\{S^1.b_{11}, \ldots, S^1.b_{1v}, \ldots, S^q.b_{q1}, \ldots, S^q.b_{qw}\}, \bowtie_{\mathbf{S}}, \mu) \in \Sigma$ then

$$\forall \overline{y}[\exists \overline{x}^1 \ldots \overline{x}^k (\bigwedge_{j=1}^{k} R^j(\overline{x}^j) \wedge \bigwedge_{\mathbf{i1}=1}^{n} (x^1_{\tau(R^1.a_{1\mathbf{i1}})} = y_{1\mathbf{i1}}) \wedge \cdots \wedge \bigwedge_{\mathbf{ik}=1}^{m} (x^1_{\tau(R^1.a_{1\mathbf{ik}})} = y_{1\mathbf{ik}}) \wedge \bigwedge_{\bowtie_{\mathbf{R}}} (x^{r^+}_{\tau(R^{r^+}.a_{r^+\mathbf{v}_r})} = x^{r^-}_{\tau(R^{r^-}.a_{r^-\mathbf{w}_r})})) \rightarrow$$

$$\exists \overline{z}^1 \ldots \overline{z}^q (\bigwedge_{i=1}^{q} S^i(\overline{z}^i) \wedge \bigwedge_{\mathbf{i1}=1}^{n} (z^{f_{\mu(1\mathbf{i1})}}_{\tau(\mu(R^1.a_{1\mathbf{i1}}))} = y_{1\mathbf{i1}}) \wedge \cdots \wedge \bigwedge_{\mathbf{ik}=1}^{m} (z^{f_{\mu(1\mathbf{ik})}}_{\tau(\mu(R^1.a_{1\mathbf{ik}}))} = y_{1\mathbf{ik}}) \wedge \bigwedge_{\bowtie_{\mathbf{S}}} (z^{s^+}_{\tau(S^{s^+}.b_{s^+\mathbf{v}_s})} = z^{s^-}_{\tau(S^{s^-}.b_{s^-\mathbf{w}_s})}))]$$

• If $\mathsf{R\text{-}SET}_{\mathsf{Exc}}((\{R^1.a_{11}, \ldots, R^1.a_{1n}, \ldots, R^k.a_{k1}, \ldots, R^k.a_{km}\}, \bowtie_{\mathbf{R}}), (\{S^1.b_{11}, \ldots, S^1.b_{1v}, \ldots, S^q.b_{q1}, \ldots, S^q.b_{qw}\}, \bowtie_{\mathbf{S}}, \mu) \in \Sigma$ then

$$\forall \overline{y}[\exists \overline{x}^1 \ldots \overline{x}^k (\bigwedge_{j=1}^{k} R^j(\overline{x}^j) \wedge \bigwedge_{\mathbf{i1}=1}^{n} (x^1_{\tau(R^1.a_{1\mathbf{i1}})} = y_{1\mathbf{i1}}) \wedge \cdots \wedge \bigwedge_{\mathbf{ik}=1}^{m} (x^1_{\tau(R^1.a_{1\mathbf{ik}})} = y_{1\mathbf{ik}}) \wedge \bigwedge_{\bowtie_{\mathbf{R}}} (x^{r^+}_{\tau(R^{r^+}.a_{r^+\mathbf{v}_r})} = x^{r^-}_{\tau(R^{r^-}.a_{r^-\mathbf{w}_r})})) \rightarrow$$

$$\neg(\exists \overline{z}^1 \ldots \overline{z}^q (\bigwedge_{i=1}^{q} S^i(\overline{z}^i) \wedge \bigwedge_{\mathbf{i1}=1}^{n} (z^{f_{\mu(1\mathbf{i1})}}_{\tau(\mu(R^1.a_{1\mathbf{i1}}))} = y_{1\mathbf{i1}}) \wedge \cdots \wedge \bigwedge_{\mathbf{ik}=1}^{m} (z^{f_{\mu(1\mathbf{ik})}}_{\tau(\mu(R^1.a_{1\mathbf{ik}}))} = y_{1\mathbf{ik}}) \wedge \bigwedge_{\bowtie_{\mathbf{S}}} (z^{s^+}_{\tau(S^{s^+}.b_{s^+\mathbf{v}_s})} = z^{s^-}_{\tau(S^{s^-}.b_{s^-\mathbf{w}_s})})))]$$

where:

(1) given $\varrho^{\mathsf{C_A}} = \{R^1.a_{11}, \ldots, R^1.a_{1n}, \ldots, R^k.a_{k1}, \ldots, R^k.a_{km}\}$, and $\varrho^{\mathsf{C_B}} = \{S^1.b_{11}, \ldots, S^1.b_{1v}, \ldots, S^q.b_{q1}, \ldots, S^q.b_{qw}\}$,
μ is a partial bijection s.t. for any $\langle \varrho^{\mathsf{C_A}}, \varrho^{\mathsf{C_B}}, \mu \rangle \in \mathsf{R\text{-}SET}_{\mathsf{H}}$, we have $\varrho^{\mathsf{C}_A} = \{R.a | \mu(R.a) \in \varrho^{\mathsf{C}_B}\}$, and

(2) $f_{\mu(x\mathbf{y})} = z$ iff $\mu(R^x.a_{x\mathbf{y}}) \in \varrho_{S^z}$

5 Encoding in $\mathcal{ALCQI}$

With the main aim of relying on available reasoning tools to reason in an effective way on ORM2 schemas, we present here the encoding in the logic $\mathcal{ALCQI}$ for which tableaux-based reasoning algorithms with a tractable computational complexity have been developed [11]. $\mathcal{ALCQI}$ corresponds to the basic DL $\mathcal{ALC}$ equipped with *qualified cardinality restrictions* and *inverse roles*, and it can also be viewed as a fragment of OWL2. The difficulty implied by the absence of n-ary relations has been overcome by means of *reification*: For each relation R with arity $n \geq 2$, a new atomic concept A_R and n functional roles $\tau(R.a_1), \ldots, \tau(R.a_n)$ one for each component of R. Due to the tree-model property of $\mathcal{ALCQI}$, the reification process provides a sound and complete translation w.r.t. concept satisfiability, such that each instance of the new introduced concept is a representative of one and only one tuple of R. Given as such, the tree-model property is enough to preserve the *correctness* of the $\mathcal{ALCQI}$ encoding, as well as of the introduced reasoning services over ORM2. Besides reification, we also know that the expressiveness of $\mathcal{ALCQI}$ does not allow to fully capture the semantics of the ORM2 constraints in table 3. In particular, $\mathcal{ALCQI}$ does not admit neither arbitrary set-comparison assertions on relations (only comparison between entire role sequences, and between pairs of single roles, are allowed), nor external uniqueness or uniqueness involving more than one role (only *unary keys* for relations are allowed), nor arbitrary frequency occurrence constraints (only *qualified number restrictions* are allowed). The analysis of these restrictions thus led to identification of a fragment of ORM2, called ORM2$^{\mathsf{zero}}$, that is maximal with respect to the expressiveness of $\mathcal{ALCQI}$, and still expressive enough to capture the most frequent usage patterns of the modelling community.

Table 6. $\mathcal{ALCQI}$ encoding.

Background domain axioms:	$E_i \sqsubseteq \neg(D_1 \sqcup \cdots \sqcup D_l)$ for $i \in \{1,\ldots,n\}$ $V_i \sqsubseteq D_j$ for $i \in \{1,\ldots,m\}$, and some j with $1 \leq j \leq l$ $D_i \sqsubseteq \sqcap_{j=i+1}^{l} \neg D_j$ for $i \in \{1,\ldots,l\}$ $\top \sqsubseteq A_{\top_1} \sqcup \cdots \sqcup A_{\top_{n_{max}}}$ $\top \sqsubseteq (\leq 1 i.\top)$ for $i \in \{1,\ldots,n_{max}\}$ $\forall i.\bot \sqsubseteq \forall i+1.\bot$ for $i \in \{1,\ldots,n_{max}\}$ $A_{\top_n} \equiv \exists 1.A_{\top_1} \sqcap \cdots \sqcap \exists n.A_{\top_1} \sqcap \forall n+1.\bot$ for $n \in \{2,\ldots,n_{max}\}$ $A_R \sqsubseteq A_{\top_n}$ for each atomic relation R of arity n $A \sqsubseteq A_{\top_1}$ for each atomic concept A
$\mathsf{TYPE}(R.a, O)$	$\exists \tau(R.a)^{-}.A_R \sqsubseteq O$
$\mathsf{FREQ}^{-}(R.a, \langle \mathsf{min}, \mathsf{max} \rangle)$	$\exists \tau(R.a)^{-}.A_R \sqsubseteq\ \geq \mathsf{min}\ \tau(R.a)^{-}.A_R \sqcap \leq \mathsf{max}\ \tau(R.a)^{-}.A_R$
$\mathsf{MAND}(\{R^1.a_1, \ldots, R^1.a_n,$ $\ldots, R^k.a_1, \ldots, R^k.a_m\}, O)$	$O \sqsubseteq \exists \tau(R^1.a_1)^{-}.A_{R^1} \sqcup \cdots \sqcup \exists \tau(R^1.a_n)^{-}.A_{R^1} \sqcup \cdots \sqcup$ $\exists \tau(R^k.a_1)^{-}.A_{R^k} \sqcup \cdots \sqcup \exists \tau(R^k.a_m)^{-}.A_{R^k}$
$^{(A)}$ $\mathsf{R\text{-}SET}^{-}_{\mathsf{Sub}}(A, B)$ $^{(A)}$ $\mathsf{R\text{-}SET}^{-}_{\mathsf{Exc}}(A, B)$	$A_R \sqsubseteq A_S$ $\quad$ $^{(A)}\ A = \{R.a_1, \ldots, R.a_n\}, B = \{S.b_1, \ldots, S.b_n\}$ $A_R \sqsubseteq A_{\top_n} \sqcap \neg A_S$
$^{(B)}$ $\mathsf{R\text{-}SET}^{-}_{\mathsf{Sub}}(A, B)$ $^{(B)}$ $\mathsf{R\text{-}SET}^{-}_{\mathsf{Exc}}(A, B)$	$\exists \tau(R.a_i)^{-}.A_R \sqsubseteq \exists \tau(S.b_j)^{-}.A_S$ $\quad$ $^{(B)}\ A = \{R.a_i\}, B = \{S.b_j\}$ $\exists \tau(R.a_i)^{-}.A_R \sqsubseteq A_{\top_n} \sqcap \neg \exists \tau(S.b_j).A_S$
$\mathsf{O\text{-}SET}_{\mathsf{Isa}}(\{O_1, \ldots, O_n\}, O)$	$O_1 \sqcup \cdots \sqcup O_n \sqsubseteq O$
$\mathsf{O\text{-}SET}_{\mathsf{Tot}}(\{O_1, \ldots, O_n\}, O)$	$O \sqsubseteq O_1 \sqcup \cdots \sqcup O_n$
$\mathsf{O\text{-}SET}_{\mathsf{Ex}}(\{O_1, \ldots, O_n\}, O)$	$O_1 \sqcup \cdots \sqcup O_n \sqsubseteq O$ and $O_i \sqsubseteq \sqcap_{j=i+1}^{n} \neg O_j$ for each $i = 1, \ldots, n$
$\mathsf{OBJ}(R, O)$	$O \equiv A_R$

Let $\mathsf{ORM2^{zero}} = \{\mathsf{TYPE}, \mathsf{FREQ^-}, \mathsf{MAND}, \mathsf{R\text{-}SET^-}, \mathsf{O\text{-}SET_{Isa}}, \mathsf{O\text{-}SET_{Tot}}, \mathsf{O\text{-}SET_{Ex}}, \mathsf{OBJ}\}$ be the fragment of ORM2 where: (i) $\mathsf{FREQ^-}$ can be applied to only one role at time, and (ii) $\mathsf{R\text{-}SET^-}$ applies either to a pair of relations of the same arity or to two single roles. The encoding of the semantics of $\mathsf{ORM2^{zero}}$ shown in table 5 makes use of the following $\mathcal{S}^{\mathcal{ALCQI}}$ signature:

- A set $E_1, E_2, \ldots, E_n$ of concepts for *entity types*;
- a set $V_1, V_2, \ldots, V_m$ of concepts for *value types*;
- a set $A_{R_1}, A_{R_2}, \ldots, A_{R_k}$ of concepts for objectified n-ary *relations*;
- a set $D_1, D_2, \ldots, D_l$ of concepts for *domain symbols*;
- $1, 2, \ldots, n_{max} + 1$ roles, where $n_{max} = \max\{|\varrho_R| | R \in \mathcal{R}\}$

Given the encoding above, the $\mathcal{ALCQI}$ KBs corresponding to the examples 1. and 2. in Section 3 are as follows (where reified relations are prefixed with 'R-', and the concept $\mathsf{A}_{\top_2}$ is the top element in the hirarchy of reified binary relations):

Example 1.

$$\begin{aligned}
\exists\tau(\mathsf{reportsTo.sub})^-.\mathsf{R\text{-}reportsTo} &\sqsubseteq \mathsf{Admin}\\
\exists\tau(\mathsf{reportsTo.obj})^-.\mathsf{R\text{-}reportsTo} &\sqsubseteq \mathsf{AreaManager}\\
\mathsf{Admin} &\sqsubseteq \exists\tau(\mathsf{reportsTo.sub})^-.\mathsf{R\text{-}reportsTo}\\
\exists\tau(\mathsf{supports.first})^-.\mathsf{R\text{-}supports} &\sqsubseteq \mathsf{ICT}\\
\exists\tau(\mathsf{supports.second})^-.\mathsf{R\text{-}supports} &\sqsubseteq \mathsf{AreaManager}\\
\exists\tau(\mathsf{supports.third})^-.\mathsf{R\text{-}supports} &\sqsubseteq \mathsf{Date}\\
\mathsf{ICT} &\sqsubseteq \exists\tau(\mathsf{supports.first})^-.\mathsf{R\text{-}supports}\\
\mathsf{AreaManager} &\sqsubseteq \exists\tau(\mathsf{supports.second})^-.\mathsf{R\text{-}supports}\\
\exists\tau(\mathsf{reportsTo.sub})^-.\mathsf{R\text{-}reportsTo} &\sqsubseteq \mathsf{A}_{\top_2} \sqcap \neg\exists\tau(\mathsf{supports.first}).\mathsf{R\text{-}supports}
\end{aligned}$$

Example 2.

$$\begin{aligned}
\mathsf{UNI\text{-}Personnel} &\sqsubseteq \mathsf{R\&TStaff} \sqcup \mathsf{Admin} \sqcup \mathsf{Student} \sqcup \mathsf{LazyPeople}\\
\mathsf{R\&TStaff} &\sqsubseteq \neg\mathsf{Admin} \sqcup \neg\mathsf{Student}\\
\mathsf{Admin} &\sqsubseteq \neg\mathsf{Student}\\
\mathsf{LazyPeople} &\sqsubseteq \neg\mathsf{R\&TStaff} \sqcup \neg\mathsf{Admin}
\end{aligned}$$

The correctness of the introduced encoding is guaranteed by the following theorem:

Theorem 2. *Let Σ^{zero} be an $\mathsf{ORM2^{zero}}$ conceptual schema and $\Sigma^{\mathcal{ALCQI}}$ the $\mathcal{ALCQI}$ knowledge base constructed as described above. Then an object type O is consistent in Σ^{zero} if and only if the corresponding concept O is satisfiable w.r.t. $\Sigma^{\mathcal{ALCQI}}$.*

Proof. [$\rightarrow$] Given a model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, (\cdot)^{\mathcal{I}}, \mathrm{ID}^{\mathcal{I}} \rangle$ of the expressions in Σ^{zero} and such that $O^{\mathcal{I}} \neq \varnothing$, we can always build a model $\mathcal{J} = \langle \Delta^{\mathcal{J}}, (\cdot)^{\mathcal{J}} \rangle$ for $\Sigma^{\mathcal{ALCQI}}$ such that $O^{\mathcal{J}} \neq \varnothing$ as follows:

- $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}}$
- $O^{\mathcal{J}} = O^{\mathcal{I}}$ for each $O \in \Sigma^{\mathcal{ALCQI}}$
- $A_R^{\mathcal{J}} = \{\mathrm{ID}^{\mathcal{I}}(d_1, \ldots, d_n) | (d_1, \ldots, d_n) \in R^{\mathcal{I}}\}$ for each concept A_R corresponding to a relation $R \in \Sigma^{\mathsf{zero}}$, and $\tau(R.a_i)^{\mathcal{J}} = \{(\mathrm{ID}^{\mathcal{I}}(d_1, \ldots, d_n), d_i) | (d_1, \ldots, d_n) \in R^{\mathcal{I}}\}$ for each new $\mathcal{ALCQI}$ functional role $\tau(R.a_i)$, representing the i-th component of R

By construction, one can now trivially conclude that $O^{\mathcal{I}} = O^{\mathcal{J}} \neq \varnothing$. As for the rest of the expressions in Σ^{zero}, it must be verified that for all $\mathcal{I}$ that are model of Σ^{zero}, there is a $\mathcal{J}$ that is a model of the corresponding $\mathcal{ALCQI}$ knowledge base. Let us show hereafter the case of the FREQ constraint. The constraint is represented in the linear syntax by means of the assertion $\mathsf{FREQ}^-(R.a, \langle\mathsf{min}, \mathsf{max}\rangle)$ having the following semantics:

$$\Pi_{R.a}(R^{\mathcal{I}}) \subseteq \{x | \mathsf{min} \leq \sharp\{\sigma_{x=R.a}(R^{\mathcal{I}})\} \leq \mathsf{max}\}$$

The corresponding $\mathcal{ALCQI}$ translation, making use of the reification, is as follows:

$$\exists \tau(R.a)^-.A_R \sqsubseteq \geq \mathsf{min}\ \tau(R.a)^-.A_R \sqcap \leq \mathsf{max}\ \tau(R.a)^-.A_R$$

An interpretation $\mathcal{I}$ is a model of the $\mathsf{ORM2}$ assertion above if for all $o \in \Delta^{\mathcal{I}}$ such that $o = t(R.a)$ for some $t \in R^{\mathcal{I}}$, there are n tuples $t_1, \ldots, t_n \in R^{\mathcal{I}}$, with $\mathsf{min} \leq n \leq \mathsf{max}$, such that $o = t_k(R.a)$ for $1 \leq k \leq n$ (i.e. if an object o participates as $R.a$-th element in $R^{\mathcal{I}}$, it must participate at least min and at most max times). Now, an interpretation $\mathcal{J}$, built from $\mathcal{I}$ as above, is such that whenever $\text{ID}^{\mathcal{I}}(t) \in A_R^{\mathcal{J}}$ with $(\text{ID}^{\mathcal{I}}(t), o) \in \tau(R.a)^{-\mathcal{J}}$, then there are $\text{ID}^{\mathcal{I}}(t_1), \ldots, \text{ID}^{\mathcal{I}}(t_n) \in A_R^{\mathcal{J}}$ with $(\text{ID}^{\mathcal{I}}(t_i), o) \in \tau(R.a)^{\mathcal{J}}$ for $1 \leq i \leq n$. Hence, $\mathcal{J}$ is a model of the corresponding $\mathcal{ALCQI}$ assertion above.

[$\leftarrow$] Given that $\mathcal{ALCQI}$ has the tree-model property, we know that if a concept C is satisfiable w.r.t. the $\Sigma^{\mathcal{ALCQI}}$ then there exists a tree-like model $\mathcal{J}$ such that $C \neq \varnothing$. Now, given a model $\mathcal{J} = \langle \Delta^{\mathcal{J}}, (\cdot)^{\mathcal{J}} \rangle$ of the expressions in $\Sigma^{\mathcal{ALCQI}}$ and such that $O^{\mathcal{J}} \neq \varnothing$, we can build an interpretation $\mathcal{I} = \langle \Delta^{\mathcal{I}}, (\cdot)^{\mathcal{I}}, \text{ID}^{\mathcal{I}} \rangle$ for Σ^{zero} such that $O^{\mathcal{I}} \neq \varnothing$ as follows:

- $\Delta^{\mathcal{I}} = \bigcup_{O \in \{\mathcal{E} \cup \mathcal{V}\}} O^{\mathcal{J}}$, where $\mathcal{E} \cup \mathcal{V}$ are the entity and value types in Σ^{zero}
- $O^{\mathcal{I}} = O^{\mathcal{J}}$ for all $O \in \Sigma^{\mathsf{zero}}$
- $R^{\mathcal{I}} = \{(d_1, \ldots, d_n) | \exists d \in A_R^{\mathcal{J}}, \bigwedge_{i=1}^{n} (d, d_i) \in \tau(R.a_i)^{-\mathcal{J}}\}$ for each concept A_R corresponding to a relation R in Σ^{zero} and $n \geq 2$
- $\text{ID}^{\mathcal{I}} = \bigcup_{R \in \mathcal{R}} A_R^{\mathcal{J}}$

Observe that, since $\mathcal{J}$ is a tree-like model, the fact that there is only one object in an objectified relation A_R representing a given tuple in R is guaranteed. As in the previous case, in the following we detail the proof for the restricted version of the frequency occurrence constraint. Given a model $\mathcal{J}$ for the $\mathcal{ALCQI}$ assertion above, each object $o \in \Delta^{\mathcal{J}}$ that is related via the functional role $\tau(R.a)^{-\mathcal{J}}$ to an object $o' \in A_R^{\mathcal{J}}$, is actually related to n objects $o'_1, \ldots, o'_n \in A_R^{\mathcal{J}}$, with $\mathsf{min} \leq n \leq \mathsf{max}$, i.e. $(o'_i, o) \in \tau(R.a)^{-\mathcal{J}}$, for $1 \leq i \leq n$. Now, the interpretation $\mathcal{I}$ built from $\mathcal{J}$ as above, populates the relation $R^{\mathcal{I}}$ with n tuples $t_1, \ldots, t_n$ corresponding to the objects in A_R, and such that $o = t_i(R.a)$ for each $1 \leq i \leq n$. According to the fact that $\mathcal{J}$ is a tree-like model, it is always possible to exclude the case where there is more than one tuple in $R^{\mathcal{I}}$ for each object in $A_R^{\mathcal{J}}$. Hence, $\mathcal{I}$ satisfies the $\mathsf{ORM2}$ assertion above. Let us finally observe that if we consider a knowledge base containing only the background axioms (corresponding to an empty $\mathsf{ORM2}$ schema), a model $\mathcal{J}$ of it is such that the interpretation of each entity type is disjoint from the interpretation of each domain symbol, and each value type is instantiated with a subset of the objects in the interpretation of a domain symbol. Therefore, an interpretation $\mathcal{I}$ built from $\mathcal{I}$ as shown above, populates each entity type with objects from the domain that are not in the extension of any domain symbol, and populates each value type with objects that are in the extension of some domain symbol, thus enforcing the conditions introduced for an $\mathsf{ORM2}$ interpretation in Section ref:ORM2.

5.1 Encoding in $\mathcal{DLR}$

We report in this section the encoding of the $\mathsf{ORM2}^{\mathsf{zero}}$ fragment in the description logic $\mathcal{DLR}$ [13,14]. The distinctive feature of $\mathcal{DLR}$ consists in the possibility of representing

n-ary relations, and the intention of capturing conceptual, as well as object-oriented data models has been the main reason for its development. Notice that verifying $\mathcal{DLR}$ knowledge base satisfiability and logical implication can be done in the ExpTime computational complexity class.

Given the $\mathcal{S}_{\mathcal{DLR}}$ signature made of:

(i) $E_1, E_2, \ldots, E_n$ concepts for *entity types*;
(ii) $V_1, V_2, \ldots, V_m$ concepts for *value types*;
(iii) $D_1, D_2, \ldots, D_l$ concepts for *domain symbols*;
(iv) $R_1, R_2, \ldots, R_k$ n-ary ($n \geq 2$) roles for *relations*;

The $\mathcal{DLR}$ translation of the FOL background axioms is straightforward:

– Background domain axioms:

$$E \sqsubseteq \neg(D_1 \sqcup \cdots \sqcup D_l) \quad (5)$$
$$V \sqsubseteq D_i, \text{ for some } D_i \quad (6)$$
$$D_1 \sqsubseteq \neg D_2 \quad (7)$$
$$\ldots$$
$$D_{n-1} \sqsubseteq \neg D_n$$

In what follows, we introduce the $\mathcal{DLR}$ axioms corresponding to the $\mathsf{ORM2}^{\mathsf{zero}}$ constraints.

– $\mathsf{TYPE}(R.a, O)$
$$\exists[\$\tau(R.a)]R \sqsubseteq O$$
– $\mathsf{FREQ}^-(R.a, \langle \mathsf{min}, \mathsf{max} \rangle)$
$$\exists[\$\tau(R.a)]R \sqsubseteq\, \geq \mathsf{min}[\$\tau(R.a)]R \sqcap \leq \mathsf{max}[\$\tau(R.a)]R$$
– $\mathsf{MAND}(\{R^1.a^1_1, \ldots, R^1.a^1_n, \ldots, R^k.a^k_1, \ldots, R^k.a^k_m\}, O)$
$$O \sqsubseteq \exists[\$\tau(R^1.a^1_1)]R^1 \sqcup \cdots \sqcup \exists[\$\tau(R^1.a^1_n)]R^1 \sqcup \cdots \sqcup \exists[\$\tau(R^k.a^k_1)]R^k \sqcup \cdots \sqcup \exists[\$\tau(R^k.a^k_m)]R^k$$
– $\mathsf{R\text{-}SET}^-_{\mathsf{H}}(A, B)$
  – (i) If $A = \{R.a_1, \ldots, R.a_n\}$, $B = \{S.b_1, \ldots, S.b_n\}$, and $n = |\varrho_R| = |\varrho_S|$:

$$\mathsf{R\text{-}SET}^-_{\mathsf{Sub}}(A, B) \qquad R \sqsubseteq S$$
$$\mathsf{R\text{-}SET}^-_{\mathsf{Exc}}(A, B) \qquad R \sqsubseteq \neg S$$

  – (ii) If $A = \{R.a_i\}$, $B = \{S.b_j\}$:

$$\mathsf{R\text{-}SET}^-_{\mathsf{Sub}}(A, B) \qquad \exists[\$\tau(R.a_i)]R \sqsubseteq \exists[\$\tau(S.b_j)]S$$
$$\mathsf{R\text{-}SET}^-_{\mathsf{Exc}}(A, B) \qquad \exists[\$\tau(R.a_i)]R \sqsubseteq \neg\exists[\$\tau(S.b_j)]S$$

– $\mathsf{O\text{-}SET}_{\mathsf{Isa}}(O_1, \ldots, O_n, O)$
$$O_1 \sqcup \cdots \sqcup O_n \sqsubseteq O$$
– $\mathsf{O\text{-}SET}_{\mathsf{Tot}}(O_1, \ldots, O_n, O)$
$$O \sqsubseteq O_1 \sqcup \cdots \sqcup O_n$$

– $\mathsf{O\text{-}SET}_{\mathsf{Ex}}(O_1, \ldots, O_n, O)$

$$\begin{aligned} O_1 \sqcup \cdots \sqcup O_n \sqsubseteq & O \\ O_1 \sqsubseteq & \neg O_2 \\ & \cdots \\ O_{n-1} \sqsubseteq & \neg O_n \end{aligned}$$

The following theorem shows the correctness of the encoding:

Theorem 3. *Let Σ^{zero} be conceptual model in the signature $\mathcal{S}^{\mathsf{zero}}$ and $\Sigma^{\mathcal{DLR}}$ the $\mathcal{DLR}$ knowledge base constructed according to the mapping above. Then every model of Σ^{zero} is a model of $\Sigma^{\mathcal{DLR}}$, and viceversa.*

Let us finally note that, given the $\mathcal{DLR}$ axioms above, an equivalent $\mathcal{ALCQI}$ knowledge base $\alpha(\Sigma^{\mathcal{DLR}})$ can be defined by applying α to all the assertions in $\Sigma^{\mathcal{DLR}}$ [12](where i is a role symbol):

$$\begin{aligned} \alpha(\top_n) &= A_{\top_n} & \alpha(\top_1) &= A_{\top_1} \\ \alpha(P) &= A_P & \alpha(A) &= A \\ \alpha(\neg R) &= A_{\top_n} \sqcap \neg\alpha(R) & \alpha(\neg C) &= A_{\top_1} \sqcap \neg\alpha(C) \\ \alpha(R^1 \sqcap R_2) &= \alpha(R^1) \sqcap \alpha(R_2) & \alpha(C_1 \sqcap C_2) &= \alpha(C_1) \sqcap \alpha(C_2) \\ \alpha(i/n : C) &= A_{\top_n} \sqcap \forall i.\alpha(C) \\ \alpha(\exists[i]R) &= \exists i^-.\alpha(R) \\ \alpha(\leq k[i]R) &= (\leq k i^-.\alpha(R)) \\ \alpha(L_1 \sqsubseteq L_2) &= \alpha(L_1) \sqsubseteq \alpha(L_2) \end{aligned}$$

6 Related works

6.1 The Halpin's FOL formalisation

In the 1989, an FOL formalisation of the semantics of the NIAM language has been proposed by T. Halpin in its *A Logical Analysis of information Systems: static aspects of the data-oriented perspective*. The main aim of the thesis was to provide designers with 'a formal basis for reasoning about conceptual schema and making design choices' [5,15]. After several revisions and expansions, in 1990 NIAM became the basis of the ORM language. Up to now, the Halpin's thesis can be considered, modulo the differences between NIAM and ORM2, as the only available attempt to provide a FOL-based semantics to ORM2. Given an FOL translation of the fragment of our semantics conditions corresponding to NIAM, we have been able to formally compare the two semantics, and to conclude for their complete equivalence. This result confirms that the our 'logic-independent' semantics actually has a perfect match with the one intended for ORM.

In what follows, we mainly go through the content of Section 4. of [5] ('Specifying NIAM conceptual schemas in KL'), and compares the Halpin's formalisation with the one presented in the paper. The language 'KL' is a tailored version of the first-order language with identity, containing a definite set of predicates and function constants with fixed interpretation that simulate a partition of the domain into five classes: *Strings*, *Numbers*, *Described objects*, *Pairs*, and the special symbol 'nil'.

Uniqueness. Statements TUC3 and TUC5 are introduced in order to express the simple case of an inter-predicate UCs spanning over $n-1$ roles of an n-ary relation R, and the UCs spanning over a single role of n binary relations, respectively.

$$\forall x_1,\dots,x_i,\dots,x_n,y.(R(x_1,\dots,x_i,\dots,x_n)\wedge \\ R(x_1,\dots,y,x_{i+1},\dots,x_n)\rightarrow x_i=y) \quad \text{(TUC3)}$$

$$\forall x_1,x_2,y_1,\dots,y_n.(R^1(x_1,y_1)\wedge\dots\wedge R_n(x_1,y_n)\wedge \\ R^1(x_2,y_1)\wedge\dots\wedge R_n(x_2,y_n))\rightarrow x_1=x_2) \quad \text{(TUC5)}$$

According to [4, p.272], we treated uniqueness constraints (UCs) as a special case of frequency occurrence constraints. Our FREQ construct provides a genuine generalisation of the Halpin's expressions. In particular, the possibility of using the FREQ construct to express UCs spanning over an arbitrary number of roles and relations, no matter about the arities of the involved relations, is not covered by the Halpin's formalisation.

Mandatory. 'A role is mandatory if, in every interpretation of the conceptual schema, it must be played by all the instances of its object type that are mentioned in the interpretation' [5]. TMR2 covers the simple case where the mandatory role may occur at any position of an n-ary relation, while TMR4 is meant to express mandatory constraints over a disjunction of roles coming from different relations.

$$\forall x_i[A(x_i)\rightarrow\exists x_1,\dots,x_{i-1},x_{i+1},\dots,x_n.R(x_1,\dots,x_n)] \quad \text{(TMR2)}$$

$$\forall x[A(x)\rightarrow\exists x_1,\dots,x_{m-1}(R_1(x_1,\dots,x_{i_1-1},x,x_{i_1+1},\dots,x_{m_1}) \\ \vee\dots\vee \\ R_n(x_1,\dots,x_{i_n-1},x,x_{i_n+1},\dots,x_{m_n}))] \quad \text{(TMR4)} \\ \text{where } m=\mathsf{max} \text{ of } m_1,\dots,m_n$$

No relevant difference between our encoding and TMR4 can be observed: Both formalizations allow the possibility to select roles from relations with (possibly) different arities.

Frequency. 'A frequency constraint (FC) of n on a role or role sequence means any given instantiation of the role (sequence) occurs n times (in that relation). [...] A FC of $n;m$ on a role (sequence) means that each instantiation of the role (sequence) occurs at least n and at most m times' [5]. The most general case of a frequency constraint is represented by the expression TFC5: The involved relation has arbitrary arity u, and the constraint spans over r roles.

$$\forall\overline{x}.[R(\overline{x})\rightarrow\exists^{n;m}\overline{z}.(R(\overline{z})\wedge z_{i_1}=x_{i_1}\wedge\dots\wedge z_{i_r}=x_{i_r})] \quad \text{(TFC5)}$$

The most notable difference between TFC5 and our FREQ construct is that the latter is general enough to express frequency constraints spanning over an arbitrary number of roles coming from a selection of n relations with (possibly) different arities, while TFC5 takes into account only one relation.

Sub-typing. Sub-typing constraints basically corresponds to *is-a* relation among object types: Given two object types, A and B, we say that A is a subtype of B if, for each interpretation, the population of A instances is *properly* included in the population of B

instances, formalised in KL as: $\forall x(B(x) \rightarrow A(x))$. In [5] this was the only kind of sub-typing considered, the one we called $\mathsf{O\text{-}SET_{Isa}}$. According to the extensions provided in [4], we also introduced 'mutually exclusive' ($\mathsf{O\text{-}SET_{Ex}}$), and 'total' ($\mathsf{O\text{-}SET_{Tot}}$) sub-typing, and we combined them in order to express the case of mutually disjoint subtypes that totally cover their super-type ('partition' sub-typing).

Set comparison. This kind of constraints must be intended to hold between roles sequence from different relations. TSC4, TEC4, TXC4 (and TXC6) are the most general expressions in [5] for subset, equality, and exclusion constraints, respectively. TXC6 covers the case where more than two relations are involved, with only one selected role per relation.

$$\forall x_1, \ldots, x_n.[\exists \overline{y}.(R(\overline{y}) \wedge x_1 = y_{i_1} \wedge \cdots \wedge x_n = y_{i_n}) \rightarrow \exists \overline{z}.(S(\overline{z}) \wedge x_1 = z_{j_1} \wedge \cdots \wedge x_n = z_{j_n}))] \tag{TSC4}$$

$$\forall x_1, \ldots, x_n.[\exists \overline{y}.(R(\overline{y}) \wedge x_1 = y_{i_1} \wedge \cdots \wedge x_n = y_{i_n}) \equiv \exists \overline{z}.(S(\overline{z}) \wedge x_1 = z_{j_1} \wedge \cdots \wedge x_n = z_{j_n}))] \tag{TEC4}$$

$$\forall x_1, \ldots, x_n.\neg[\exists \overline{y}.(R(\overline{y}) \wedge x_1 = y_{i_1} \wedge \cdots \wedge x_n = y_{i_n}) \wedge \exists \overline{z}.(S(\overline{z}) \wedge x_1 = z_{j_1} \wedge \cdots \wedge x_n = z_{j_n}))] \tag{TXC4}$$

$$\forall x.\neg(x \in R_1.i_1 \wedge R_2.i_2 \vee x \in R_1.i_1 \wedge R_3.i_3 \vee \cdots \vee x \in R_{n-1}.i_{n-1} \wedge R_n.i_n) \tag{TXC6^2}$$

Notice that: (i) the family of our R-SET constructs admits the possibility of comparing arbitrary role sequences, where roles are selected from different relations, and (ii) the involved relations may have arbitrary arities. This means that our $\mathsf{R\text{-}SET_{Exc}}$ is a genuine generalisation of TXC6 (we admit the possibility of selecting more than one role in the same relation at the same time), and it covers TXC4 as a special case. The flexibility of the R-SET is guaranteed by the presence of the partial bijection μ in the constraint expression. Equality constraints can obviously be expressed using $\mathsf{R\text{-}SET_{Sub}}$ in both directions. Recent developments of the ORM language (see [4]), have imposed stricter conditions in the definition of the role sequences that can be compared by means of the set-comparison constraints, on one hand, and they have increased the range of application of the constraints, on the other. Stricter conditions have been imposed in the sense that the selected roles in the two sequences must have now 'compatible types': Roles that are mapped one into the other in the constraint must be typed by the same object type, or by two distinct object types having a common (direct or indirect) supertype. More flexibility has been gained by allowing the possibility of selecting more than a single role per relation, from an arbitrary number of relations at the same time. A (conceptual) join mechanism to relate together the different relations involved in the definition of a single role sequence is considered, and the presence of explicit *join paths* along these relations is now part of the requirements in the assertion of a set-comparison constraint (once needed). Finally, join paths are allowed to be specified only along roles (of distinct relations) that agree on the same (direct) type. As one would expect, this version of the set-comparison constraints is almost identical to the one presented in our formalisation, except that we did not impose any restrictions neither in the definition of the mapping μ (roles in the mapping may be typed by distinct object types), nor in the definition of the join path (our 'join anchors' may be typed by distinct object types).

[2] If R is n-ary, $x \in R.i =_{def} \exists x_1, \ldots, x_n(R(x_1, \ldots, x_n) \wedge x = x_i)$.

Nesting. 'An objectified relationship [is] a relationship that is treated as an object which itself plays roles' [5]. The proposed formalisation relies on the Halpin's ontological institution of the 'pairs' objects: A pair is the result of the application of the special function pair to two generic objects (see p.3-24). Any ordered n-tuple of arity above 1 (e.g. an object $(x_1, \ldots, x_n)$ is then captured as the pair $(x_1, (x_2, \ldots, x_n)))$. When the result of objectification is named, the FOL semantics is as follows:

$$\forall x.[A(x) \equiv \exists x_1, \ldots, x_n.(R(x_1, \ldots, x_n) \wedge x = (x_1, \ldots, x_n)] \qquad \text{(TN3)}$$

It is easy to prove that TN3 is semantically equivalent to the introduced formalisation. Nonetheless, more recently, [16] introduces a (different) formalisation of the objectification where the function pair is absent, and the principle of having a unique identifier for each reified tuple in the domain of interpretation is definitively lost.

6.2 Mapping ORM2 to the DL $\mathcal{DLR}_{ifd}$

In the last few years, several papers addressed the issue of encoding ORM2 conceptual schema into DL knowledge bases [17,18,19,20]. Among those proposals, [17] can be taken as the only one going through the encoding with a formal perspective. In particular, [17] pretends to start from the Halpin's FOL semantics, and introduces an encoding of a fragment of ORM2 into the logic $\mathcal{DLR}_{ifd}$, an extension of $\mathcal{DLR}$ with *identification assertions* on concepts, and *functional dependencies* assertions on relations [21]. Except for the presence of uniqueness constraints spanning over arbitrary sequence of n roles of the same relation, and external uniqueness over 2 roles, that are represented in the paper by means of suitable identification assertions, ORM2$^{\text{zero}}$ and the fragment identified in [17] agree on the same expressive power. In general, the paper suffers from the presence of several imprecisions, redundancy, and syntactical mistakes that makes the proposed mapping solutions not always clearly understandable (see, for instance, the introduction of the identification assertions for the uniqueness constraints representation). Moreover, some of the proposed solutions look simply incorrect w.r.t. the intended semantics specified in [5], such as in the case where 'objectification' is simply treated as 'relation reification' in DL, and where the *optionality* principle that is crucial in the semantics of frequency occurrence constraints in [5] simply disappeared in the proposed $\mathcal{DLR}_{ifd}$ mapping. In what follows, we go into the details of the proposed mapping (text in boldface represents the names used in [17] to denote elements and constraints).

Object type. Here the concept of 'object type' is introduced, and mapped to a $\mathcal{DLR}_{ifd}$ concept. The FOL characterisation, that the author ascribes to the Halpin's seminal work, is meaningless: The semantics of the FOL expression, taken as such, establishes that any element in the domain should be in the interpretation of a concept C, that is obviously unreasonable.

Named value type ('not constrained' case).'Value types' ('such as types of character strings' [4]) are mapped by means of a complex axiom, in the same way the presence of an attribute a of a certain type T for a class C in a UML diagram is captured in [10]: A new binary relation a is introduced, and the axiom specifies that for each instance c of the concept C, all object related to c by a, are instances of T. According to this mapping there is no intrinsic difference between 'object types, and 'value types'. In particular, value types are simply object types with a binary relation attached linking their instances with the instances of a so-called 'concrete domain' T.

Unary,Binary, and n-ary relations. They are devoted to the introduction, and mapping, of the ORM2 relations. Except for the presence of 'unary' relations, the formalisation proposed by Keet is equivalent to one we introduced in the paper. Nonetheless, our approach assumes the 'roles' as first class citizens, and exploits the TYPE constraint to link each role to the respective concept (in our framework, any role is 'localised' by means of ϱ). Differently from [17], where unary relations are technically mapped using binary relations whose second component is typed by a 'auxiliary new introduced filler object- or value type', we ruled out them from our encoding into $\mathcal{DLR}$ (unary relations are perfectly covered, on the other hand, by our FOL encoding).

Object type. The proposed FOL formalisation, where elements of the concept C only appear as playing the first component of the n-ary relation R, is not in line with the DL mapping (where the instances of the concept C participate in the i-th component of R). Moreover, given points 3, 4, and 5, the introduction of this kind of axioms is redundant: The types of the roles are introduced there as relation definitions. The fact that no concept may appear in isolation in the context of an ORM2 conceptual schema is not captured by the formalisation at point 6 (quantification over concepts is not expressible in $\mathcal{DLR}$). Finally, the FOL statement introduced here is meant to represent the 'mandatory participation' that is correctly encoded at Point 8 below.

Mandatory and Disjunctive mandatory. There is no relevant difference between our formalisation and the one proposed by Keet in the case of mandatory constraints. Nonetheless, since 'disjunctive mandatory' can be considered as a special case of the mandatory constraint, we introduced only one construct and we admitted the presence of an arbitrary set of roles.

Uniqueness. The first difference here relates to our choice of treating the uniqueness constraint as special case of frequency occurrence, as suggested in [4]. Moreover, the plain version of $\mathcal{DLR}$ we used does not allow the possibility of expressing identification assertions ('**id**'). However, even considering the encoding into $\mathcal{DLR}_{ifd}$, the proposed formalisation is asyntactic, on one hand, and incorrect, on the other. In particular, it is known that 'identification assertions' (id) do not apply to relation symbols and, moreover, considering R as the reification of original relation, the components $r_1, \ldots, r_i$ in the id statement should represent new introduced functional roles, rather than the components of the original relation. And finally, since uniqueness constraints basically mimic functional dependencies, in the light of the encoding in $\mathcal{DLR}_{ifd}$ the use of the so-called 'functional dependency assertions' [21] would have represented a more natural choice here.

Role frequency. The formalisation of uniqueness constraint correctly covers the case where some of the instances of the involved concept might not play the role at all, but this does not hold for the role frequency constraint formalisation. According to the semantics of the two proposed role frequency formalisations, each instance of the involved concept must play the specified roles a number a of times, with $a \geq 1$. Given as such, they do not agree on the *optionality* principle discussed in [4].

Proper subtypes, Total covering, Exclusive subtypes. The 'exhaustive subtype', as introduced in [4], does not admit the possibility of having more instances in the union of the subtypes than the instances in the unique super-type, but the formalisation proposed in [17] is slightly weaker than this, and does not exclude that this could be the case. On the other hand, by adopting the equivalence symbol in our formalisation, we ruled out that possibility. Moreover, we treated the 'exclusive subtypes, total' (called 'partition' in [4]) as a derived construct.

Subset, Set-equality, Role exclusion. Except from what is probably a typo (the proper way of representing the objects that participate in a given component i of a relation R

is by using the quantifier, like in $\exists[\$i]R$), we proposed the same formalisation for subset, set-equality, and role exclusion 'over two roles'. Notice that the set-equality is redundant, once the subset constraint has been introduced.

Joins. We did not introduce any special construct to formalise 'join'(s), since a join can be simply expressed using the constructs that are already taken as primitive.

Objectification. The way objectification is introduced in [17] is the one is usually taken 'reification' in the DL literature, where a new concept name and a set of functional roles is introduced according to arity of the original relation. Nonetheless, we decided not to introduce this kind of formalisation simply because this is not the way objectification is intended in the present ORM2 literature.

Finally, observe that, in the absence of an Abox logical implication of inclusion assertions (i.e. assertions in the TBox having the form $C_1 \sqsubseteq C_2$ and $R_1 \sqsubseteq R_2$, where C_1 and C_2 are concepts, and R_1 and R_2 are relations of the same arity) can be verified without considering 'identification' and 'functional dependency' assertions [21].

As regards to [18], we mostly rely here on the extensive review already made by Keet in [17]. Starting from this, it should be also noticed that subsequent attempts, focused on the possibilities of encoding ORM2 into the the web ontology language OWL2 [19,20], suffer from the same formal inconsistencies and limitations of [18]. In particular, [18] is misleading with respect to the underlying DL formalism: distinct extensions of the adopted logic (e.g. $\mathcal{DLR}$ plus $\mathcal{DLR}$-*Lite*), distinct DL languages (e.g. $\mathcal{DLR}$, plus $\mathcal{DLR}$-*Lite*, plus $\mathcal{SROIQ}$, plus 'role composition' operator) are there arbitrary mixed together. No special semantics is provided by [18] in correspondence with these combinations, nor theorems showing the complexity of reasoning with them. Unfortunately, this lack of formality, together with the introduction of syntactic elements that are simply not allowed in DL (e.g. '$\not\sqsubseteq$'), makes the proposed formalisation difficult to understand, not applicable in practice, and logically incoherent. Nonetheless, there are several others critical points that negatively affect [18]: (i) the introduction of 'special' DL concepts with the aim of encoding *data types* is improper in the sense that the semantics of these concepts cannot, by definition, capture the intended one (i.e. the interpretation of a DL concept is a set of objects, and not of values), and this choice also has an impact on (ii) the way unary relations are treated in $\mathcal{DLR}$, i.e. by introducing a new auxiliary concept called 'BOOLEAN' whose semantics is not formally specified. Again, the intuition is correct but, the formal translation of it lacks of precision.

In [22], a list of 9 'unsatisfiability constraint patterns' is introduced with the aim of supporting the automatic detection of unsatisfiable concepts and roles. The patterns discussed in the paper represent a subset only of all the possible sources of inconsistency that can occur in a conceptual schema, and the absence of any formal semantics behind the way the authors deal with them makes the approach non-systematic, *incomplete*, and *ad hoc*. Instead of proposing a logic-based theoretical background for the encoding of an interesting ORM fragment, as we did, the authors decided to proceed in a non formal way focusing on what they call 'the most common unsatisfiability cases in practice' [22, p.3]: In particular, there is no reasoning procedure behind the intuitive justification suggested by the authors for the unsatisfiability of the patterns.

As regards to the patterns themselves, it should be noticed at least one of them, the one concerning 'strict subset' relation, is inconsistent with the interpretation conveyed in [5] and [4], while the first one, assuming *by default* in the semantics (and not only as a best practice in modelling) the mutually exclusiveness of all the entity types in a schema, it is simply the result of a standalone decision, whose logical consequence in terms of reasoning are not further discussed in the paper. This said, it should be clear that the [22] approach to the automatic detection of conceptual schema unsatisfiability is in perfect contradiction with our approach. Moreover, our approach not only covers all the cases presented in [22] but it is also *correct* with respect to the $\mathsf{ORM2}^{\mathsf{zero}}$ fragment, and provides formal justifications for schema unsatisfiability. This is the main reason why we do not see any 'complementarity' between the two approaches, as rather claimed by the authors in the concluding section of [22].

Similar to [22], a paper focused on the encoding of ORM2 in OWL has been recently published [23]. The paper introduces a set of informal 'rules' devoted to the mapping of a subset of the ORM2 constructs into OWL. Unfortunately, the paper is misleading in several respects (for instance: (i) the OWL EquivalentTO, instead of the SubClassOf, is erroneously introduced several times; (ii) optionality of uniqueness constraints is definitively lost). In general, the paper covers a fragment that is smaller than $\mathsf{ORM2}^{\mathsf{zero}}$, and the proposed mapping mostly remains formally unjustified.

Finally, in [24] an incomplete encoding of OWL-DL into ORM2 is presented. In the paper, the authors claim that 'universal restrictions' of the form $A \sqsubseteq \forall R.B$ cannot be translated in ORM2 in a way that preserves the semantics of the original constructs. But, this is not the case: A viable translation into ER has been introduced in [9], where covering (i.e. total subtyping) and disjointness (i.e. exclusive subtyping) between relationships are used, and a second one into UML can be found in [10], making use of reification of roles. Both translations can be straightforwardly rephrased into ORM2.

7 Automated Reasoning Support Tool

With the main goal of providing automated reasoning services facilitating the conceptual modelling activity, a prototype of $\mathsf{ORM2}^{\mathsf{zero}}$ modelling support tool has been implemented for NORMA. The prototype takes an ORM2 schema produced by NORMA as input, and encodes it into the linear syntax using an XSLT script. By relying on existing OWL2 reasoners (e.g. HermiT, FaCT++), the tool provides the following functionality:

- **Implicit constraints deduction**. Derived implicit $\mathsf{ORM2}^{\mathsf{zero}}$ constraints, including inconsistent object types and fact types, are displayed in distinct pop-up windows. The computation is complete, but only cognitively relevant constraints are visualised, e.g. redundant transitive links are not visualised.
- **Translation into OWL2 ontology**. In order to facilitate web-data exchange and to make conceptual schemas readily accessible to automated processes, the prototype features a translator from $\mathsf{ORM2}^{\mathsf{zero}}$ schema into OWL2 ontology, which can then be saved in various formats.

Let us now illustrate the essential functionality of the prototype using the example introduced on fig. 4. Its encoding into the linear syntax is as follows:

$$\begin{aligned}
\mathsf{ENTITYTYPES} :& \{\mathsf{PhoneCall}, \mathsf{MobileCall}, \mathsf{PhonePoint}, \mathsf{Cell}, \mathsf{Landline}, \mathsf{HomePoint}\}\\
\mathsf{VALUETYPES} :& \{\mathsf{PhoneCall_Id}, \mathsf{PhonePoint_\sharp}\}\\
\mathsf{RELATIONS} :& \{\mathsf{HasOriginFrom}, \mathsf{HasDestinationTo}, \mathsf{HasMOriginFrom}, \mathsf{HasPhoneCall_Id},\\
& \mathsf{HasPhonePoint_\sharp}\}
\end{aligned}$$

$$
\begin{array}{rl}
\mathsf{TYPE}(\mathsf{HasOriginFrom.1}, \mathsf{PhoneCall}) & \mathsf{MAND}(\{\mathsf{HasOriginFrom.1}\}, \mathsf{PhoneCall}) \\
 & \mathsf{FREQ}(\{\mathsf{HasOriginFrom.1}\}, \{\}, (1,1)) \\
\mathsf{TYPE}(\mathsf{HasOriginFrom.2}, \mathsf{PhonePoint}) & \\
\mathsf{TYPE}(\mathsf{HasDestinationTo.1}, \mathsf{PhoneCall}) & \mathsf{MAND}(\{\mathsf{HasDestinationTo.1}\}, \mathsf{PhoneCall}) \\
\mathsf{TYPE}(\mathsf{HasDestinationTo.2}, \mathsf{PhonePoint}) & \\
\mathsf{TYPE}(\mathsf{HasMOriginFrom.1}, \mathsf{MobileCall}) & \\
\mathsf{TYPE}(\mathsf{HasMOriginFrom.2}, \mathsf{Cell}) & \\
\mathsf{TYPE}(\mathsf{HasPhoneCall\_Id.1}, \mathsf{PhoneCall}) & \mathsf{MAND}(\{\mathsf{HasPhoneCall\_Id.1}\}, \mathsf{PhoneCall}) \\
 & \mathsf{FREQ}(\{\mathsf{HasPhoneCall\_Id.1}\}, \{\}, (1,1)) \\
\mathsf{TYPE}(\mathsf{HasPhoneCall\_Id.2}, \mathsf{PhoneCall\_Id}) & \mathsf{FREQ}(\{\mathsf{HasPhoneCall\_Id.2}\}, \{\}, (1,1)) \\
\mathsf{TYPE}(\mathsf{HasPhonePoint\_\sharp.1}, \mathsf{PhonePoint}) & \mathsf{MAND}(\{\mathsf{HasPhonePoint\_\sharp.1}\}, \mathsf{PhonePoint}) \\
 & \mathsf{FREQ}(\{\mathsf{HasPhonePoint\_\sharp.1}\}, \{\}, (1,1)) \\
\mathsf{TYPE}(\mathsf{HasPhonePoint\_\sharp.2}, \mathsf{PhonePoint\_\sharp}) & \mathsf{FREQ}(\{\mathsf{HasPhonePoint\_\sharp.2}\}, \{\}, (1,1)) \\
\mathsf{O\text{-}SET}_{\mathsf{Isa}}(\{\mathsf{MobileCall}\}, \mathsf{PhoneCall}) & \\
\mathsf{O\text{-}SET}_{\mathsf{Tot}}(\{\mathsf{Landline}, \mathsf{Cell}\}, \mathsf{PhonePoint}) & \\
\mathsf{O\text{-}SET}_{\mathsf{Ex}}(\{\mathsf{Landline}, \mathsf{Cell}\}, \mathsf{PhonePoint}) & \\
\mathsf{O\text{-}SET}_{\mathsf{Ex}}(\{\mathsf{HomePoint}, \mathsf{Landline}\}, \mathsf{PhonePoint}) &
\end{array}
$$

$$
\begin{array}{l}
\mathsf{R\text{-}SET}_{\mathsf{Sub}}(\{\mathsf{HasMOriginFrom.1}, \mathsf{HasMOriginFrom.2}\}, \{\}), (\{\mathsf{HasOriginFrom.1}, \mathsf{HasOriginFrom.2}\}, \{\}), \\
\quad \{(\mathsf{HasMOriginFrom.1}, \mathsf{HasOriginFrom.1}), (\mathsf{HasMOriginFrom.2}, \mathsf{HasOriginFrom.2})\}
\end{array}
$$

Among the key constraints, the schema involves the uniqueness constraint imposed on the origin of a phone call as well as the hierarchical constraints describing the nature of possible phone points. Therefore, with a single click we can obtain the following relevant deductions for the given conceptual schema: $\mathsf{FREQ}(\{\mathsf{hasMOriginFrom.1}\}, \{\}, (1,1))$ and $\mathsf{O\text{-}SET}_{\mathsf{Isa}}(\{\mathsf{HomePoint}\}, \mathsf{Cell})$, i.e. it is true that any home point is also a cell point, and each mobile call may have an origin from at most one cell point.

In order to understand why this is true, consider the following. The class of home points is a sub class of all the phone points, and it is disjoint from the class of landline points. Since any phone point is either a cell point or a landline point, then any home point should necessarily be a cell point. The hasMOriginFrom binary relation is included in the hasOriginFrom binary relation. Since each call participates exactly once as first argument to the hasOriginFrom, if we take a generic sub class of calls, such as the class of mobile calls, and a sub relationship of the hasOriginFrom relation, such as hasMOriginFrom, then we can conclude that necessarily each mobile call participates at most once as first argument to the hasMOriginFrom relation. The full list of the inferred constraints is displayed in the ORM2 *Inference Browser* window while selected deductions are illustrated by relevant fragments of the inferred schema in pop-up windows over the initial schema.

8 Conclusions

In this paper we introduced a linear syntax and a complete set-theoretic semantics for the ORM2 conceptual modelling language. A decidable, and computationally tractable, fragment of ORM2 has been clearly identified and mapped into the DL logic $\mathcal{ALCQI}$. A first reasoning support prototype for ORM2 has been implemented which enables consistency and entailment checks for the defined fragment of ORM2. Future theoretic works will be mainly focused on the extension of the $\mathsf{ORM2}^{\mathsf{zero}}$ towards the identification of a more expressive, still decidable, ‘object role’ modelling language. The practical objectives

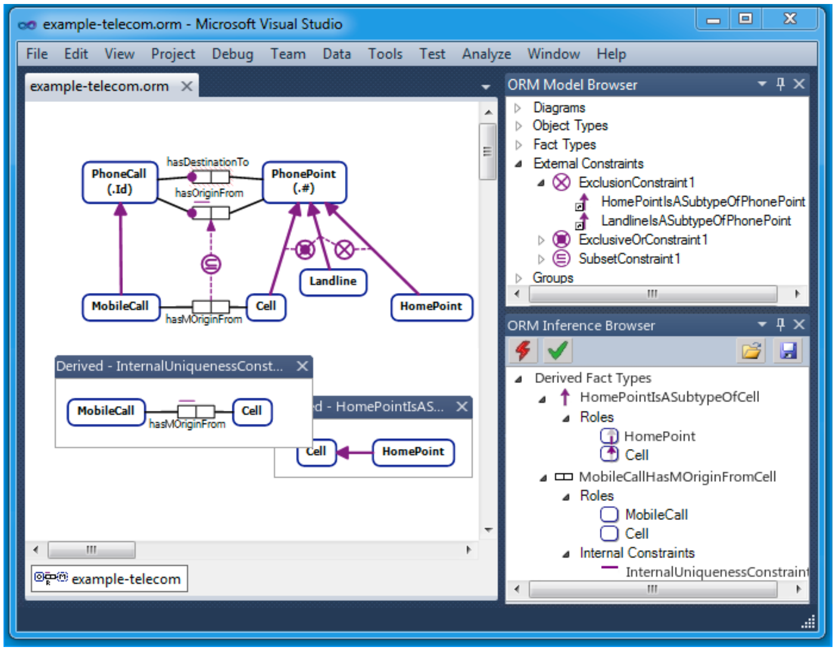


Fig. 4. Graphical interface of the prototype

of the research will be directed towards full integration of the prototype into third-party solutions providing graphical user interface for designing ORM2 conceptual schemas (e.g. NORMA plugin for Microsoft Visual Studio).

References

1. Ceravolo, P., Fugazza, C., Leida, M.: Modeling semantics of business rules. In: Proceedings of the Inaugural IEEE International Conference On Digital Ecosystems and Technologies (IEEE-DEST). (February 2007)
2. Zhao, G., Gao, Y., Meersman, R.: An ontology-based approach to business modelling. In: Proceedings of the International Conference of Knowledge Engineering and Decision Support (ICKEDS2004). (2004) 213–221
3. The Business Rules Group: Defining business rules. What are they really? Technical report, The Business Rules Group (2001)
4. Halpin, T., Morgan, T.: Information Modeling and Relational Databases: From Conceptual Analysis to Logical Design. 2nd edn. Morgan Kaufmann (2001)
5. Halpin, T.: A Logical Analysis of Information Systems: Static Aspects of the Data-oriented Perspective. PhD thesis, Department of Computer Science, University of Queensland (1989)
6. Evans, A.S.: Reasoning with UML class diagrams. In: Proceedings of the Second IEEE Workshop on Industrial Strength Formal Specification Techniques. WIFT '98, Washington, DC, USA, IEEE Computer Society (1998)
7. Calvanese, D., Lenzerini, M., Nardi, D.: Description logics for conceptual data modeling. In Chomicki, J., Saake, G., eds.: Logics for Databases and Information Systems, Kluwer (1998) 229–263
8. Artale, A., Calvanese, D., Kontchakov, R., Ryzhikov, V., Zakharyaschev, M.: Complexity of reasoning in entity relationship models. In: Proc. of the 2007 Int. Workshop on Description Logics (DL 2007). (2007)

9. Artale, A., Calvanese, D., Kontchakov, R., Ryzhikov, V., Zakharyaschev, M.: Reasoning over extended ER models. In Parent, C., Schewe, K.D., Storey, V.C., Thalheim, B., eds.: ER. Volume 4801 of Lecture Notes in Computer Science., Springer (2007) 277–292
10. Berardi, D., Cali, A., Calvanese, D., Giacomo, G.D.: Reasoning on UML class diagrams. ARTIFICIAL INTELLIGENCE **168** (2003)
11. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F., eds.: The description logic handbook: theory, implementation, and applications. Cambridge University Press, New York, NY, USA (2003)
12. Calvanese, D., De Giacomo, G., Nardi, D., Lenzerini, M. In: Reasoning in expressive description logics. Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands (2001) 1581–1634
13. Calvanese, D., De Giacomo, G., Lenzerini, M., Nardi, D., Rosati, R.: Description logic framework for information integration. In: Proc. of the 6th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'98). (1998) 2–13
14. Calvanese, D., De Giacomo, G., Lenzerini, M.: On the decidability of query containment under constraints. In: Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems. PODS '98, New York, NY, USA, ACM (1998) 149–158
15. Verheijen, G., Bekkum, J.V.: NIAM, an information analysis method. In: Proc. of the IFIP TC-8 Conference on Comparative Review of Information System Methodologies (CRIS 82), North-Holland (1982)
16. Halpin, T.: Objectification. In Castro, J., Teniente, E., eds.: Proceedings of the CAiSE05 Workshops, vol.1. (2005) 519–532
17. Keet, M.: Mapping the Object-Role Modeling language ORM2 into description logic language $\mathcal{DLR}_{ifd}$. Technical Report KRDB07-2, KRDB Research Centre, Faculty of Computer Science, Free University of Bozen-Bolzano (2007)
18. Jarrar, M.: Towards automated reasoning on ORM schemes. mapping ORM into the $\mathcal{DLR}_{ifd}$ description logic. In: Proceedings of the 26th International Conference on Conceptual Modeling (ER 2007). Volume 4801 of LNCS., Springer (November 2007) 181–197
19. Jarrar, M.: Mapping ORM into the $\mathcal{SHOIN}$/OWL description logic - towards a methodological and expressive graphical notation for ontology engineering. In: OTM 2007 workshops: Proceedings of the International Workshop on Object-Role Modeling (ORM'07). Volume 4805 of LNCS., Springer (November 2007) 729–741
20. Hodrob, R., Jarrar, M.: ORM to OWL2 DL mapping. In: International Conference on Intelligent Semantic Web: Applications and Services, ACM (2010)
21. Calvanese, D., De Giacomo, G., Lenzerini, M.: Identification constraints and functional dependencies in description logics. In: Proceedings of the 17th international joint conference on Artificial intelligence - Volume 1, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (2001) 155–160
22. Jarrar, M., Heymans, S.: Unsatisfiability reasoning in orm conceptual schemes. In: Current Trends in Database Technology - EDBT 2006: Proceeding of the IFIP-2.6 International Conference on Semantics of a Networked. Volume 4254 of LNCS., Springer (March 2006) 517–534
23. Wagih, H.M., ElZanfaly, D.S., Kouta, M.M.: Mapping Object Role Modeling 2 schemes to OWL2 ontologies. In Ting, Z., ed.: Proceedings of the 3rd IEEE International Conference on Computer Research and Development (ICCRD), Shanghai, China, March 11-13, 2011. Volume 4., IEEE Press (2011) 126–132
24. Bach, D.B., Meersman, R., Spyns, P., Trog, D.: Mapping OWL-DL into ORM/RIDL. In: Proceedings of the 2007 OTM - Volume Part I. OTM'07, Berlin, Heidelberg, Springer-Verlag (2007) 742–751