Temporal ER Modelling with Description Logics

Alessandro Artale[1] and Enrico Franconi[2]

[1] Department of Computation, UMIST, Manchester, UK,
artale@co.umist.ac.uk
[2] Department of Computer Science, University of Manchester, UK,
franconi@cs.man.ac.uk,
http://www.cs.man.ac.uk/~franconi/

Abstract. Recent efforts in the Conceptual Modelling community have been devoted to properly capturing time-varying information. Various temporally enhanced Entity-Relationship (ER) models have been proposed that are intended to model the temporal aspects of database conceptual schemas. This work gives a logical formalisation of the various properties that characterise and extend different temporal ER models which are found in literature. The formalisation we propose is based on Description Logics (DL), which have been proved useful for a logical reconstruction of the most popular conceptual data modelling formalisms. The proposed DL has the ability to express both enhanced temporal ER schemas and integrity constraints in the form of complex inclusion dependencies. Reasoning in the devised logic is decidable, thus allowing for automated deductions over the whole conceptual representation, which includes both the ER schema and the integrity constraints over it.

1 Introduction

Temporal Databases are databases that store historical information, i.e., essentially past and present data. Temporally enhanced Entity-Relationship (ER) models have been developed to conceptualise the temporal aspects of database schemas, namely *valid time* – when a fact holds, i.e., it is true in the representation of the world – and *transaction time* – which records the history of database states rather than the world history. In this paper we only consider the modelling of the validity time which is supported by almost all the temporal ER models – differently from the transaction time.

In the temporal ER community two different main modelling approaches have been devised to provide temporal support. The *implicit* approach hides the temporal dimension in the interpretation structure of the ER constructs. Thus, a temporal ER model does not include any new specific temporal construct with respect to a standard ER model. Each ER construct is always interpreted with a temporal semantics, so that instances of temporal entities or relationships are always potentially time-varying objects. The *explicit* approach, on the other hand, retains the non-temporal semantics for the conventional ER constructs, while adding new syntactical constructs for representing temporal entities and

relationships and their temporal interdependencies. The advantage of the explicit approach is the so called *upward compatibility*: the meanings of conventional (legacy) ER diagrams when used inside a temporal model remains unchanged. This is crucial, for example, in modelling data warehouses or federated databases, where sources may be a collection of both temporal and legacy databases.

A logical formalisation is introduced in this paper that can cover both the implicit and the explicit approaches. The idea is to provide a formalisation for implicit temporal ER models, enriched with the ability to express a powerful class of temporal integrity constraints. While instances of ER entities or relationships are potentially time-varying objects, integrity constraints can impose restrictions in the temporal validity of such objects. The formalisation is powerful enough that it is possible to explicitly state as integrity constraints the distinction between time-varying and snapshot (i.e., time invariant) constructs. In this way, an ER diagram may contain both temporal and non-temporal information, providing the ability to capture the explicit approach.

The formalisation presented in this paper is based on an expressive temporal Description Logic. Advantages of using Description Logics are their high expressivity combined with desirable computational properties – such as decidability, soundness and completeness of deduction procedures. The core is the Description Logic $\mathcal{ALCQI}$, which is able to capture conventional ER models and has the ability to express a powerful class of inclusion dependency constraints. The core language is then extended with a tense logic. Temporal integrity constraints can be expressed in this combined logic, called $\mathcal{ALCQIT}$. Reasoning in the whole framework is a decidable task, allowing for a complete calculus for temporal integrity constraints.

The paper is organised as follows. Section 2 introduces the temporally enhanced ER models, in both the implicit and explicit approaches. Section 3 first introduces the non-temporal $\mathcal{ALCQI}$ description logic, and then its extension by a standard tense modal logic. Section 4 will show how temporal Entity-Relationship schemas can be encoded into the temporal description logic, how additional temporal integrity constraints can be imposed on schemas, and how it is possible to reason in this framework. The final sections describe how integrity constraints can encode time-varying and snapshot constructs. Final remarks on possible extensions conclude the paper.

2 The Temporal ER Model

In this Section we introduce the temporally enhanced ER model and its semantics. We first consider standard ER diagrams.

Basic elements of ER schemas are *entities*, denoting a set of objects called *instances*, and *relationships*, denoting a set of *tuples* made by the instances of the different entities involved in the relationship. Since the same entity can be involved in the same relationship more than once, participation of entities in relationships is represented by means of *ER-roles*, to which a unique name is assigned. ER-roles can have *cardinality constraints* to limit the number instances

of an entity involved in the relationship. Both entities and relationships can have *attributes*, i.e., properties whose value belong to some predefined domain – e.g., `Integer`, `String`. Additionally, the considered ER model includes taxonomic relationships to state inclusion assertions between entities and between relationships. ER schemas are usually built using graphical tools. As showed in Figure 1, entities are represented by boxes, while diamonds are used for relationships. Attributes are ovals, while ER-roles are depicted by connecting relationships to each involved entity; in the figure, ER-roles do not have a name. Cardinality constraints on ER-roles are depicted by labelling the edges with the desired numeric constraints. An ISA relationship is shown with an oriented edge – from the more general to the more specific term – e.g., `Manager` ISA `Employee` in Figure 1.

So far we have considered a standard ER diagram, i.e., a diagram where no explicit temporal constructs appear. According to the implicit approach, a temporally enhanced ER diagram does not have any specific temporal construct, since it is intended that every construct has always a temporal interpretation. Thus, the syntax of the temporal model is the same as the standard one, and the temporal dimension is considered only at the semantical level.

The first-order semantics for the temporally enhanced ER model – in the implicit approach – is the standard and intuitive choice. We consider as a starting point the non temporal semantics introduced in [Calvanese *et al.*, 1998]. Interpretations of a temporal ER diagram $\mathcal{D}$ are those temporally dependent *database states* $\mathcal{B}(t) = (\Delta^{\mathcal{B}}, \cdot^{\mathcal{B}(t)})$ such that: each domain symbol D is mapped to a subset $D^{\mathcal{B}(t)}$ of the corresponding domain interpretation at time point t; each entity symbol E is mapped to $E^{\mathcal{B}(t)} \subseteq \Delta^{\mathcal{B}}$ at time point t; each attribute symbol A is mapped to $A^{\mathcal{B}(t)} : \Delta^{\mathcal{B}} \rightarrow \bigcup_i D_i^{\mathcal{B}(t)}$ at time point t; each n-ary relationship, with both ER-roles and attributes, denotes a set of labelled n-tuples over $(\Delta^{\mathcal{B}} \bigcup_i D_i^{\mathcal{B}(t)})^n$ at time point t. A *legal database state* should satisfy the following conditions: If $E_1 \text{ISA} E_2$, then $E_1^{\mathcal{B}(t)} \subseteq E_2^{\mathcal{B}(t)}$ for every t; if A is an attribute of entity E with domain D, then $\forall e \in E^{\mathcal{B}(t)}.A^{\mathcal{B}(t)}(e) \in D^{\mathcal{B}(t)}$ for every t; for each relationship R with $E_1, \ldots, E_k$ entities involved by means of ER-roles $U_1, \ldots, U_k$, and attributes $A_1, \ldots, A_l$ with domains $D_1, \ldots, D_l$, then every instance tuple of R at time t has the form $[U_1 : e_1, \ldots, U_k : e_k, A_1 : a_1, \ldots, A_l : a_l]$, where $e_i \in E_i^{\mathcal{B}(t)}$ and $a_j \in D_j^{\mathcal{B}(t)}$; for each minimum and maximum cardinality constraint, n, m respectively, in a ER-role U relating a relationship R with and entity E, then $n \leq \#\{r \in R^{\mathcal{B}(t)} \mid r[U] \in E^{\mathcal{B}(t)}\} \leq m$ for every t.

Let us consider the example ER diagram of Figure 1; this diagram is the running example considered in the survey paper [Gregersen and Jensen, 1999]. As we have noticed before, the implicit approach does not consider the temporal constructs related to the validity time of entities and relationships (see, e.g., the TEER model in [Elmasri and Navathe, 1994]). Thus, the example diagram should be modified, since there are some of those disallowed constructs. The *Profit* relationship becomes an attribute of the entity *Department*, the *Salary* relationship becomes an attribute of the entity *Employee*, the *Work-period* entity disappears, since it just denotes the validity time of the relationship *Works-for*.

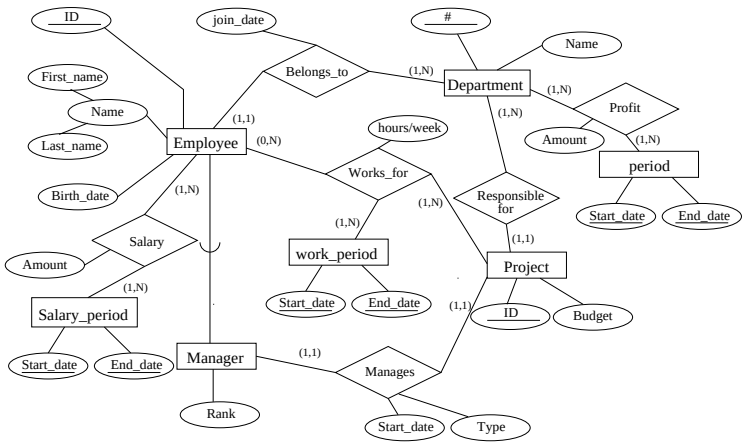

Fig. 1. The reference temporal ER diagram

The resulting diagram is such that *every* construct has its own validity time, conforming to the above temporal semantics.

We consider now an enhancement of the temporal ER model by means of integrity constraints. In this section just an informal introduction will be given; the exact characterisation of integrity constraints over a temporal ER diagram will be given in Section 4. The following integrity constraints may be imposed over the example ER diagram:

- managers are the only employees who do not work for a project (she/he just manages it);
- a manager becomes qualified after a period when she/he was just an employee[1].

These constraints can not be expressed directly by means of ER constructs; the logical formalisation presented in the following sections has been explicitly devised for expressing a large class of (temporal) constraints. The presence of the above constraints limits the number of legal database states, since not all the unconstrained databases conform to the newly introduced constraints. It is also easy to see that the legal database states conforming to the enriched schema, which includes both the ER diagram and the integrity constraints, also conform to the following constraints:

- for every project, there is at least an employee who is not a manager,
- each manager worked in a project before managing some (possibly different) project

[1] Let us assume that an entity called *Qualified* appears somewhere else in the ER diagram.

Please note that these deductions are not trivial, since from the ER schema the cardinality constraints do not impose that employees necessarily work in a project.

Explicit Approach

More complex and challenging becomes the case where an *explicit* approach to provide temporal support is adopted. In this case new constructs are added to represent the temporal dimension of the model. At the cost of adding new constructs, this approach has the advantage of preserving the atemporal meaning of conventional (legacy) ER schemas when embedded into temporal ER diagrams: this property is called *upward compatibility*. This crucial property is not realizable within the standard implicit temporal approach. Indeed, if the implicit approach has the attractive of leaving unchanged the original ER model, "this approach rules out the possibility of designing non-temporal databases or databases where some part of a database is non-temporal and the rest is temporal" [Gregersen and Jensen, 1999]. In particular, we suppose that both entities and relationships in a explicit temporal ER model can be either unmarked, in what case they are considered *snapshot* constructs (i.e., each of their instances has a global lifetime, as in the case they derive from a legacy diagram), or explicitly *temporary* marked (i.e., each of their instances has a temporary lifetime).

3 The Temporal Description Logic

We introduce very briefly in this section the $\mathcal{ALCQIT}$ temporal DL, which is obtained by combining a standard tense logic and the standard non-temporal $\mathcal{ALCQI}$ DL [Calvanese *et al.*, 1999].

The basic types of the DL are *concepts*, *roles*, and *features*. According to the syntax rules at the left of Figure 2, $\mathcal{ALCQI}$ *concepts* (denoted by the letters C and D) are built out of *primitive concepts* (denoted by the letter A), *roles* (denoted by the letter R, S), and *primitive features* (denoted by the letter f); roles are built out of *primitive roles* (denoted by the letter P) and *primitive features*.

We define the *meaning* of concepts as sets of individuals and the meaning of roles as sets of pairs of individuals. A temporal structure $\mathcal{T} = (\mathcal{P}, <)$ is assumed, where $\mathcal{P}$ is a set of time points and $<$ is a strict linear order on $\mathcal{P}$. Formally, an $\mathcal{ALCQIT}$ *temporal interpretation* over $\mathcal{T}$ is a triple $\mathcal{I} \doteq \langle \mathcal{T}, \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}(t)} \rangle$, consisting of a set $\Delta^{\mathcal{I}}$ of individuals (the *domain* of $\mathcal{I}$) and a function $\cdot^{\mathcal{I}(t)}$ (the *interpretation function* of $\mathcal{I}$) mapping, for each $t \in \mathcal{P}$, every concept to a subset of $\Delta^{\mathcal{I}}$, every role to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and every feature to a partial function from $\Delta^{\mathcal{I}}$ to $\Delta^{\mathcal{I}}$, such that the equations at the right of Figure 2 are satisfied.

A *knowledge base* is a finite set Σ of *terminological axioms* of the form $C \sqsubseteq D$. An interpretation $\mathcal{I}$ over a temporal structure $\mathcal{T} = (\mathcal{P}, <)$ satisfies a terminological axiom $C \sqsubseteq D$ if $C^{\mathcal{I}(t)} \subseteq D^{\mathcal{I}(t)}$ for every $t \in \mathcal{P}$. A knowledge base Σ is *satisfiable* in the temporal structure $\mathcal{T}$ if there is a temporal interpretation $\mathcal{I}$

$$
\begin{array}{lll}
C, D \rightarrow & A \mid & \\
& \top \mid & \Delta^{\mathcal{I}} \\
& \bot \mid & \emptyset \\
& \neg C \mid & \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}(t)} \\
& C \sqcap D \mid & C^{\mathcal{I}(t)} \cap D^{\mathcal{I}(t)} \\
& C \sqcup D \mid & C^{\mathcal{I}(t)} \cup D^{\mathcal{I}(t)} \\
& \forall R.C \mid & \{i \in \Delta^{\mathcal{I}} \mid \forall j.\, R^{\mathcal{I}(t)}(i,j) \Rightarrow C^{\mathcal{I}(t)}(j)\} \\
& \exists R.C \mid & \{i \in \Delta^{\mathcal{I}} \mid \exists j.\, R^{\mathcal{I}(t)}(i,j) \wedge C^{\mathcal{I}(t)}(j)\} \\
& f\!\uparrow \mid & \Delta^{\mathcal{I}} \setminus \mathrm{dom}\, f^{\mathcal{I}(t)} \\
& f : C \mid & \{i \in \mathrm{dom}\, f^{\mathcal{I}(t)} \mid C^{\mathcal{I}(t)}(f^{\mathcal{I}(t)}(i))\} \\
& \geq n\, R.C \mid & \{i \in \Delta^{\mathcal{I}} \mid \sharp\{j \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}(t)}(i,j) \wedge C^{\mathcal{I}(t)}(j)\} \geq n\} \\
& \leq n\, R.C \mid & \{i \in \Delta^{\mathcal{I}} \mid \sharp\{j \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}(t)}(i,j) \wedge C^{\mathcal{I}(t)}(j)\} \leq n\} \\
& C\,\mathcal{U}\,D \mid & \{i \in \Delta^{\mathcal{I}} \mid \exists v.\, v > t \wedge D^{\mathcal{I}(v)}(i) \wedge \forall w.\, (t < w < v) \rightarrow C^{\mathcal{I}(w)}(i)\} \\
& C\,\mathcal{S}\,D \mid & \{i \in \Delta^{\mathcal{I}} \mid \exists v.\, v < t \wedge D^{\mathcal{I}(v)}(i) \wedge \forall w.\, (v < w < t) \rightarrow C^{\mathcal{I}(w)}(i)\} \\
& \Diamond^{+} C \mid & \{i \in \Delta^{\mathcal{I}} \mid \exists v.\, v > t \wedge C^{\mathcal{I}(v)}(i)\} \\
& \Diamond^{-} C \mid & \{i \in \Delta^{\mathcal{I}} \mid \exists v.\, v < t \wedge C^{\mathcal{I}(v)}(i)\} \\
& \Box^{+} C \mid & \{i \in \Delta^{\mathcal{I}} \mid \forall v.\, v > t \rightarrow C^{\mathcal{I}(v)}(i)\} \\
& \Box^{-} C & \{i \in \Delta^{\mathcal{I}} \mid \forall v.\, v < t \rightarrow C^{\mathcal{I}(v)}(i)\} \\
R, S \rightarrow & P \mid & \\
& f \mid & \\
& R^{-1} \mid & \{(i,j) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid R^{\mathcal{I}(t)}(j,i)\} \\
& R|_C \mid & R^{\mathcal{I}(t)} \cap (\Delta^{\mathcal{I}} \times C^{\mathcal{I}(t)}) \\
& R \circ S & R^{\mathcal{I}(t)} \circ\, S^{\mathcal{I}(t)}
\end{array}
$$

Fig. 2. $\mathcal{ALC}\,\mathcal{QIT}$ and its semantics.

over $\mathcal{T}$ which satisfies every axiom in Σ; in this case $\mathcal{I}$ is called a *model* over $\mathcal{T}$ of Σ. Checking for KB satisfiability is deciding whether there is at least one model for the knowledge base. Σ *logically implies* an axiom $C \sqsubseteq D$ in the temporal structure $\mathcal{T}$ (written $\Sigma \models C \sqsubseteq D$) if $C \sqsubseteq D$ is satisfied by every model over $\mathcal{T}$ of Σ. In this latter case, the concept C is said to be *subsumed* by the concept D in the knowledge base Σ and the temporal structure $\mathcal{T}$. Concept subsumption can be reduced to concept satisfiability since C is subsumed by D in Σ if and only if $(C \sqcap \neg D)$ is unsatisfiable in Σ.

Reasoning in $\mathcal{ALC}\,\mathcal{QI}$ (i.e., deciding knowledge base satisfiability and logical implication) is decidable, and it has been proven to be an EXPTIME-complete problem [Calvanese *et al.*, 1999]. The tense-logical extension of $\mathcal{ALC}\,\mathcal{QI}$ has been inspired by the works of [Schild, 1993,Artale and Franconi, 1998,Artale and Franconi, 1999,Wolter and Zakharyaschev, 1998b]. The following theorem states that reasoning in $\mathcal{ALC}\,\mathcal{QIT}$ is decidable:

Theorem 1 (Decidability of $\mathcal{ALC}\,\mathcal{QIT}$). *The problems of checking knowledge base satisfiability and logical implication are decidable for $\mathcal{ALC}\,\mathcal{QIT}$ over a linear, unbounded, and discrete temporal structure (like the natural numbers).*

The proof is based on a reduction to the decidable language introduced in [Wolter and Zakharyaschev, 1998a]. The exact computational complexity of reasoning in $\mathcal{ALCQIT}$ is still unknown; the lower bound is EXPTIME-hard.

As an example let us consider the axiom describing the situation in which any living mortal has mortal parents, has only living children (if any), remains alive until it will die, and at some point in the past was born:

$$\begin{aligned}\texttt{Mortal} \sqcap \texttt{LivingBeing} \sqsubseteq\ & \texttt{has-father} : \texttt{Mortal} \sqcap \texttt{has-mother} : \texttt{Mortal} \sqcap \\ & (\forall \texttt{has-father}^{-1}.(\texttt{Mortal} \sqcap \texttt{LivingBeing}) \sqcup \\ & \ \forall \texttt{has-mother}^{-1}.(\texttt{Mortal} \sqcap \texttt{LivingBeing})) \sqcap \\ & \texttt{LivingBeing}\ \mathcal{U}\ \Box^{+}\neg\texttt{LivingBeing} \sqcap \\ & \texttt{LivingBeing}\ \mathcal{S}\ \Box^{-}\neg\texttt{LivingBeing}\end{aligned}$$

4 Encoding the Implicit Temporal ER Model

Now, it is shown how an ER schema with implicit representation of time can be expressed in a $\mathcal{ALCQIT}$ knowledge base whose models correspond with legal database states of the ER schema – allowing for reasoning services such as satisfiability of a schema or the computation of logically implied integrity constraints. It is important to emphasise the fact that in this approach the integrity constraints are part of the schema, and that reasoning is carried on by taking in complete account all the information contained in the schema.

Let us first consider the translation between an ER diagram (without considering the integrity constraints) and an $\mathcal{ALCQIT}$ knowledge base as follows.

Definition 1 (Translation). *An ER diagram $\mathcal{D}$ is translated into a corresponding knowledge base Σ where each domain, entity or relationship symbol corresponds to an atomic concept, and each attribute or ER-role symbol corresponds to an atomic feature, in such a way that the following axioms hold:*

- *For each* ISA *link between two entities E, F (resp. two relationships R, S) in $\mathcal{D}$, add to Σ the terminological axiom:*
  $E \sqsubseteq F$ *(resp.* $R \sqsubseteq S$*)*
- *For each attribute A in $\mathcal{D}$ with domain D of an entity E (resp. of a relationship R), add to Σ the terminological axiom:*
  $E \sqsubseteq A : D$ *(resp.* $R \sqsubseteq A : D$*)*
- *For each relationship R in $\mathcal{D}$ relating n entities $E_1 \ldots E_n$ by means of the ER-roles*[2] *$P^R_{E_1} \ldots P^R_{E_n}$, add to Σ the terminological axiom:*
  $R \sqsubseteq (P^R_{E_1} : E_1) \sqcap \ldots \sqcap (P^R_{E_n} : E_n)$
- *For each minimum cardinality constraint $n \neq 0$ in a ER-role P^R_E in $\mathcal{D}$ relating a relationship R with and entity E, add to Σ the terminological axiom:*
  $E \sqsubseteq \geq n\,(P^R_E)^{-1}.\,R$

[2] We assume that a unique name is given within a relationship to each ER-role, representing a specific participation of an entity in the relationship.

– *For each maximum cardinality constraint* $n \neq \infty$ *in a ER-role* P_E^R *in* $\mathcal{D}$ *relating a relationship* R *with and entity* E*, add to* Σ *the terminological axiom:*
$E \sqsubseteq \leq n\,(P_E^R)^{-1}.\,R$ □

From the definition 1 it is clear that n-ary relationships are *reified* in the translated description logic knowledge base, i.e., they become concepts with n special feature names – the ER-roles – denoting the n arguments of the n-ary relationship. Moreover, note that $\mathcal{ALCQIT}$ temporal operators do not appear in the translation, since we are in the implicit approach.

Temporal integrity constraints are expressed by means of additional terminological axioms in Σ. Recall that a terminological axiom states an inclusion between concepts. Thus, an integrity constraint is any kind of inclusion dependency expressible in the full temporal DL $\mathcal{ALCQIT}$.

Definition 2 (Inclusion Dependencies). *An integrity constraint for an ER diagram* $\mathcal{D}$ *is any inclusion dependency which can be expressed in the corresponding knowledge base* Σ *by means of a terminological axiom of the kind* $C \sqsubseteq D$*, where atomic concepts appearing in* C, D *correspond to domain, entity or relationship symbols in* $\mathcal{D}$ *and atomic features appearing in* C, D *correspond to ER-role symbols in* $\mathcal{D}$*.*

Based on the results of [Calvanese *et al.*, 1994,Calvanese *et al.*, 1998], we have proved that the translation is correct, in the sense that there is a precise correspondence between legal database states of $\mathcal{D}$ and models of the derived knowledge base Σ. The existence of this correspondence is such that, whenever the problem of checking an ER schema against a property has a specific solution, then the corresponding reasoning problem in the DL has a corresponding solution, and vice-versa. Thus, it is possible to exploit standard reasoning procedures in the DL for checking properties of the ER schema. The reasoning problems we are mostly interested in are *consistency* of a ER schema – which is mapped to a satisfiability problem in the corresponding DL knowledge base – and *logical implication* within a ER schema – which is mapped to a logical implication problem in the corresponding DL knowledge base.

The proof is based by establishing the existence of two mappings from legal database states of $\mathcal{D}$ to models of Σ and vice-versa. Informally speaking, the mapping is possible since the temporal semantics of $\mathcal{ALCQIT}$ is such that every object is given a temporal interpretation, the same way a validity time is always associated to every tuple (or instance of entity) in the implicit approach. The existence of the mappings ensures that, whenever Σ is satisfiable, it is possible to build a corresponding non-empty legal database state, and vice-versa.

As a final remark, it should be noted that the high expressivity of DL constructs can capture an extended version of the basic ER model, which includes not only taxonomic relationships, but also arbitrary boolean constructs to represent so called generalized hierarchies with disjoint unions; entity definitions by means of either necessary or sufficient conditions or both, and integrity constraints expressed by means of generalised axioms [Calvanese *et al.*, 1998].

Example

Let us consider the example introduced in Section 2. We first translate the fragment of the ER diagram (Figure 1) involving the entities *Project, Employee, Manager* and the relationship *Works-for* in the description logic knowledge base Σ_{ER}:

$\texttt{WORKS-FOR} \sqsubseteq \texttt{has-prj} : \texttt{Project} \sqcap \texttt{has-emp} : \texttt{Employee}$

$\texttt{Project} \sqsubseteq \exists \texttt{has-prj}^{-1}.\texttt{WORKS-FOR}$

$\texttt{Manager} \sqsubseteq \texttt{Employee}$

We then encode the integrity constraints, which are expressed by means of terminological axioms in a knowledge base Σ_{IC}:

- Managers are the only employees who do not work for a project:
  $\texttt{Manager} \sqsubseteq \forall \texttt{has-emp}^{-1}.\neg\texttt{WORKS-FOR}$
  $\forall \texttt{has-emp}^{-1}.\neg\texttt{WORKS-FOR} \sqsubseteq \texttt{Manager}$
  The constraints state that managers are exactly employees which are not involved in the *Works-for* relation.
- A manager becomes qualified after a period when she/he was just an employee:
  $\texttt{Manager} \sqsubseteq \texttt{Qualified}\ \mathcal{S}\ (\texttt{Employee} \sqcap \neg\texttt{Manager})$
  The constraint states that all managers are qualified after they have been at the same time employees and not managers.

It turns out that the following integrity constraints are logically implied from $\Sigma_{ER} \cup \Sigma_{IC}$:

- For every project, there is at least an employee who is not a manager:
  $\Sigma_{ER} \cup \Sigma_{IC} \models \texttt{Project} \sqsubseteq \exists(\texttt{has-prj}^{-1} \circ \texttt{has-emp}).\neg\texttt{Manager}$
  The constraint states that every project is such that there exists somebody working for it who is not a manager.
- A manager worked in a project before managing some (possibly different) project:
  $\Sigma_{ER} \cup \Sigma_{IC} \models \texttt{Manager} \sqsubseteq \Diamond^{-}\exists(\texttt{has-emp}^{-1} \circ \texttt{has-prj}).\texttt{Project}$
  The constraint states that every manager should have worked some time in the past for a project.

Moreover, if we change in Σ_{ER} the minimum cardinality of the participation of employees to the *Works-for* relationship to one (i.e., we make it a mandatory participation):

$\texttt{Employee} \sqsubseteq \exists \texttt{has-emp}^{-1}.\texttt{WORKS-FOR}$

then, even if Σ_{ER} is satisfiable, $\Sigma_{ER} \cup \Sigma_{IC}$ is an unsatisfiable knowledge base, because of the first integrity constraint. For the abovementioned theorem, no legal DB state exists for the ER schema including the constraints.

5 Encoding the Explicit Temporal ER Model

In the following it will be showed how the proposed formalisation can encode explicit temporal ER models by simply imposing specific constraints defining snapshot and temporary constructs, thus maintaining the required upward compatibility.

5.1 Snapshot Vs. Temporary Entities

The description logic $\mathcal{ALCQIT}$ is able to capture explicit temporal ER models by first applying the translation given in the previous Section, and then adding precise axioms to distinguish between snapshot and temporal constructs. In the following, axioms for entities are illustrated. In the next Section, the analogous for relationships will be showed.

A snapshot entity is axiomatised by the following constraint:

$$E \sqsubseteq (\Box^+ E) \sqcap (\Box^- E) \qquad (\textit{Snapshot axiom})$$

expressing that whenever the entity is true it is necessarily true in every past and future time point. Indeed, instances of snapshot entities have necessarily a global lifetime. On the other hand, a temporary entity is axiomatised by the following constraint:

$$E \sqsubseteq (\Diamond^+ \neg E) \sqcup (\Diamond^- \neg E) \qquad (\textit{Temporary axiom})$$

asserting that there must be a past or future time point where the entity does not hold. Indeed, instances of temporary entities have necessarily a limited lifetime.

With respect to our running example, the `Department` entity may be considered a snapshot entity, since it is unlikely that the organisational structure of an enterprise changes in time, while the entity `Manager` may be considered a temporary entity, since managers may change in time.

Using the reasoning capabilities of $\mathcal{ALCQIT}$ it is possible to support the database designer to discover relevant schema properties. As an example of the logical implications holding in a diagram making use of both snapshot and temporary entities, let us consider the interaction between entities via ISA links. Let us suppose that there is an ISA link between a snapshot entity E_1 and a temporary entity E_2. This temporal ER diagram is translated into the following unsatisfiable knowledge base:

$$E_1 \sqsubseteq (\Box^+ E_1) \sqcap (\Box^- E_1)$$
$$E_2 \sqsubseteq (\Diamond^+ \neg E_2) \sqcup (\Diamond^- \neg E_2)$$
$$E_1 \sqsubseteq E_2$$

Thus, a snapshot entity can not be a subclass of a temporary entity, this is true also whenever such a kind of taxonomic relation is derived in the temporal ER model. This can be formally explained by observing that if this ISA relationship would be valid there will be an instance, say a, such that a is of type E_1 and

E_2 at a certain point t_0 – let represent this fact by means of the following set notation: $\{a : E_1, a : E_2\}_{t_0}$. As specified by the temporary axiom for E_2, there must be a time point, say t_1, such that a is not of type E_2 – $\{a : \neg E_2\}_{t_1}$. On the other hand, since E_1 is a snapshot entity, a is of kind E_1 at all time points, and in particular at time t_1 – $\{a : \neg E_2, a : E_1\}_{t_1}$. Due to the subclass relationship, this would imply that a is of type E_2 at t_1 – $\{a : \neg E_2, a : E_1, a : E_2\}_{t_1}$. Then, both a is of type E_2 and a is *not* of type E_2 holds at t_1, which is a contradiction.

From these considerations it is easy to understand why the following implications hold:

$$\{E_2 \sqsubseteq (\Diamond^+ \neg E_2) \sqcup (\Diamond^- \neg E_2),\ E_1 \sqsubseteq E_2\ \} \models E_1 \sqsubseteq (\Diamond^+ \neg E_1) \sqcup (\Diamond^- \neg E_1)$$
$$\{E_1 \sqsubseteq (\Box^+ E_1) \sqcap (\Box^- E_1),\ E_1 \sqsubseteq E_2\ \} \models E_2 \sqsubseteq (\Box^+ E_2) \sqcap (\Box^- \neg E_2)$$

i.e., necessarily, every subclass of a temporary entity must be temporary; and a superclass of a snapshot entity must be a snapshot entity. Conversely, nothing can be said with respect to subclasses of snapshot entities. For example, a schema where a temporary entity is a subclass of a snapshot entity is consistent.

An incorrect ER schema can be the result of disjoint subclasses – i.e., a partitioning. A schema is inconsistent if exactly one of a whole set of snapshot disjoint subclasses is temporary [McBrien *et al.*, 1992]. Without loss of generality, let us illustrate the case where E_1, E_2 are disjoint subclasses of the entity E, with E_1 snapshot and E_2 temporary, then such an ER schema is inconsistent. Indeed, the knowledge base corresponding to this ER schema is unsatisfiable:

$$E \sqsubseteq E_1 \sqcup E_2, \quad E_1 \sqsubseteq E \sqcap \neg E_2, \quad E_2 \sqsubseteq E \qquad \text{(disjoint subclass axioms)}$$
$$E_1 \sqsubseteq (\Box^+ E_1) \sqcap (\Box^- E_1) \qquad \text{(snapshot axiom)}$$
$$E_2 \sqsubseteq (\Diamond^+ \neg E_2) \sqcup (\Diamond^- \neg E_2) \qquad \text{(temporary axiom)}$$

This can showed by proving that the entity E_2 has no models. Let suppose, by absurd, that a is an instance of E_2 at time t_0, then, due to the subclass relationship, a is also of kind E – $\{a : E_2, a : E\}_{t_0}$. Since E_2 is a temporary entity there must be t_1 such that $\{a : \neg E_2\}_{t_1}$. As we showed previously, a superclass of a snapshot entity must be a snapshot entity. Then, since E_1 is a snapshot entity, also E must be snapshot, and at t_1 we have $\{a : \neg E_2, a : E\}_{t_1}$. Due to the disjointness subclass axioms, necessarily a is of kind E_1 at t_1 – $\{a : E_1, a : \neg E_2, a : E\}_{t_1}$. But E_1 is a snapshot entity, then a is of kind E_1 at t_0, too. At the end, at time t_0 we have: $\{a : E_1, a : E_2, a : E\}_{t_0}$, which is a contradiction since E_1, E_2 are disjoint entities. The following is an immediate consequence of the above inconsistent schema:

$$\{\ E \sqsubseteq E_1 \sqcup E_2, \quad E_1 \sqsubseteq E \sqcap \neg E_2, \quad E_2 \sqsubseteq E,$$
$$E_1 \sqsubseteq (\Box^+ E_1) \sqcap (\Box^- E_1) \quad \} \models E_2 \sqsubseteq (\Box^+ E_2) \sqcap (\Box^- E_2)$$

i.e., an ER schema with exactly one entity whose temporal behaviour is unknown among a whole set of snapshot disjoint subclasses, will imply that this entity is necessarily snapshot.

5.2 Snapshot Vs. Temporary Relationships

The case for relationships is more complex. Temporary relationships are captured by enforcing the *temporary axiom* on relationships – in a way analogous to the case of temporary entities:

$$R \sqsubseteq (\Diamond^+ \neg R) \sqcup (\Diamond^- \neg R) \qquad (\textit{Temporary axiom})$$

To capture snapshot relationships, in addition to the *snapshot axiom*, we need to force each ER-role to be time invariant. For this purpose, the so called *global features* are needed. They are features whose value does not depend on time: we will indicate such particular kind of feature by prefixing the feature name with a "$\star$". Atomic global features are interpreted as partial functions independent from time: $\forall t, v \in \mathcal{P}.\, \star g^{\mathcal{I}(t)} = \star g^{\mathcal{I}(v)}$.

Using global features instead of generic features for ER-roles defining a relationship results in a *homogeneous relationship* – i.e., a relationship with tuples whose values are valid at the same time period. Homogeneous relationships are encoded by means of the following axiom:

$$R \sqsubseteq (\star P^R_{E_1} : E_1) \sqcap \ldots \sqcap (\star P^R_{E_n} : E_n) \qquad (\textit{Homogeneity axiom})$$

Snapshot relationships are necessarily global and homogeneous relationships. Thus, if R is a snapshot relationship involving the entities $E_1, \ldots, E_n$, the following axioms should be added to Σ:

$$R \sqsubseteq (\Box^+ R) \sqcap (\Box^- R) \qquad (\textit{Snapshot axiom})$$
$$R \sqsubseteq (\star P^R_{E_1} : E_1) \sqcap \ldots \sqcap (\star P^R_{E_n} : E_n) \qquad (\textit{Homogeneity axiom})$$

The two axioms are such that whenever a tuple belongs to a snapshot relationship, then the very same tuple is assumed to belong to the relationship at every time.

According to the running example, all relationships – with the exception of the `Responsible-for` relationships – have to be modelled as temporary relationships.

The interaction between temporal and snapshot constructs can result in an inconsistent ER schema that can be checked and discarded automatically. This is case when a snapshot relationship R involves a temporary entity. Indeed, the following knowledge base is unsatisfiable:

$$\begin{aligned} R &\sqsubseteq (\Box^+ R) \sqcap (\Box^- R) \\ R &\sqsubseteq (\star P^R_{E_1} : E_1) \sqcap \ldots \sqcap (\star P^R_{E_n} : E_n) \\ E_i &\sqsubseteq (\Diamond^+ \neg E_i) \sqcup (\Diamond^- \neg E_i) \end{aligned}$$

i.e., snapshot relationships cannot have temporary entities as participants. This can be formally proved by showing that such a relationship has no models. Let us suppose, by absurd, that a is an instance of R at time t_0 then a is also related to an instance of E_i via the ER-role $\star P^R_{E_i}$, i.e., $\{a : R, a \star P^R_{E_i} b, b : E_i\}_{t_0}$. Since

E#	ENAME	DEPARTMENT	SALARY
121	Tom	$\langle[10,12), Sales\rangle$	$\langle[10,15), 20K\rangle$
		$\langle[14,18), Mktg\rangle$	$\langle[15,17), 25K\rangle$
			$\langle[17,20), 30K\rangle$
133	Ann	$\langle[25,30), Sales\rangle$	$\langle[25,30), 35K\rangle$
133	$John$	$\langle[18,now), Toys\rangle$	$\langle[18,now), 42K\rangle$

Fig. 3. The employee heterogeneous relation

E_i is temporary then there must be a time t_1 such that b is not of type E_i – $\{b : \neg E_i\}_{t_1}$. On the other hand, since R is a snapshot relation, a is an instance of R also at time t_1 which is always related via the same instance b to the entity E_i, since the ER roles are global features – $\{a : R, a \star P^R_{E_i} b, b : E_i\}_{t_1}$. This result in a contradiction since b was supposed not to be of type E_i at t_1. To obtain this inconsistency result the use of global features instead of conventional features is crucial. Indeed, mapping ER-roles of snapshot relationships using conventional features would allow the same R instance to be related to different ER-roles at different times, then invalidating the above expected inference.

On the other hand, temporary relationships admit snapshot entities since the entity instances participate in the relationship only for a temporary time – i.e., during the validity time of the relationship.

6 Heterogeneous Relationships

In the temporal relational database community, a relation is called *heterogeneous* when its attribute values have different period of existence [Tansel and Tin, 1998]. Conversely, *homogeneity* impose that the whole tuple is associated with a unique time stamp. The Figure 3 shows a heterogeneous employee relation as presented in the paper [Tansel and Tin, 1998]. The attributes DEPARTMENT and SALARY have time stamps represented as intervals with (possible) different values at different time points – as is the case for the tuple associated with *Tom*.

The translation 1 of ER-relationships into $\mathcal{ALCQIT}$ gives rise to a relationship with (possibly) heterogeneous values. However, relationships can be forced to have *homogeneous* tuples [Gadia, 1988] by using global features to map the ER-roles – as we showed in Section 5.2. Combining both global and generic features we can represent heterogeneous relationships with both time-varying and time-independent attributes. For example, the employee relation in Figure 3, where E# and ENAME are time-independent attributes, has the following mapping:

$$R \sqsubseteq (\star \texttt{E\#} : \texttt{ID}) \sqcap (\star \texttt{ENAME} : \texttt{Name}) \sqcap (\texttt{DEPARTMENT} : \texttt{Dept}) \sqcap (\texttt{SALARY} : \texttt{Amount})$$

[McBrien *et al.*, 1992] enforce temporal constraints on the validity time of instances of entities involved in heterogeneous relationships. The original ERT model [Theodoulidis *et al.*, 1991] has been extended to include *historical* marks (H-mark) by imposing particular temporal constraints on the validity time of the different ER-roles. Different temporal constraints give rise to different H-marked relationships. For example, a relationship between entities E_1 and E_2 is H-marked *past* if the instances of E_2 involved in the relationship hold at intervals before the intervals where the instances of E_1 hold. Both historical past and historical future relationships can be formalized with the following axioms:

$$R \sqsubseteq (P^R_{E_2} : E_2 \sqcap \neg P^R_{E_1} : E_1) \sqcap (P^R_{E_2} : E_2 \; \mathcal{U} \; P^R_{E_1} : E_1) \text{ (Historical past axiom)}$$
$$R \sqsubseteq (P^R_{E_1} : E_1 \sqcap \neg P^R_{E_2} : E_2) \sqcap (P^R_{E_1} : E_1 \; \mathcal{U} \; P^R_{E_2} : E_2) \text{ (Historical future axiom)}$$

i.e., as far as historical past is concerned, the attribute which involve the entity $E_2 - P^R_{E_2} -$ is true *until* the attribute involving $E_1 - P^R_{E_1} -$ holds.

7 Conclusions

This preliminary work gives a logical formalisation of a temporal ER model, which has the ability to express both enhanced temporal ER schemas and (temporal) integrity constraints in the form of general axioms imposed on the schema itself. The formal language we have proposed is a member of the family of Description Logics, and it has a decidable reasoning problem, thus allowing for automated deduction over the whole conceptual representation. We have also shown how the integrity constraints can encode the distinction between time-varying and snapshot constructs.

This work is just at the beginning. The most promising research direction to be explored is to better characterise the expressivity of temporal integrity constraints in order to axiomatise several extensions as proposed in the literature of temporal ER models.

Acknowledgements

The work presented in this paper was partially supported by the "Foundations of Data Warehouse Quality" (*DWQ*) European ESPRIT IV Long Term Research (LTR) Project 22469.

References

[Artale and Franconi, 1998] Alessandro Artale and Enrico Franconi. A temporal description logic for reasoning about actions and plans. *Journal of Artificial Intelligence Research*, 9:463–506, 1998.

[Artale and Franconi, 1999] Alessandro Artale and Enrico Franconi. Temporal description logics. In L. Vila, P. vanBeek, M. Boddy, M. Fisher, D. Gabbay, A. Galton, and R. Morris, editors, *Handbook of Time and Temporal Reasoning in Artificial Intelligence*. MIT Press, 1999. To appear.

[Calvanese *et al.*, 1994] Diego Calvanese, Maurizio Lenzerini, and Daniele Nardi. A unified framework for class-based representation formalisms. In *Proc. of the 4 th International Conference on Principles of Knowledge Representation and Reasoning*, Bonn, Germany, May 1994.

[Calvanese *et al.*, 1998] D. Calvanese, M. Lenzerini, and D. Nardi. Description logics for conceptual data modeling. In Jan Chomicki and Günter Saake, editors, *Logics for Databases and Information Systems*. Kluwer, 1998.

[Calvanese *et al.*, 1999] Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Daniele Nardi. Reasoning in expressive description logics. In Alan Robinson and Andrei Voronkov, editors, *Handbook of Automated Reasoning*. Elsevier Science Publishers, Amsterdam, 1999. To appear.

[Elmasri and Navathe, 1994] R. Elmasri and S. B. Navathe. *Fundamentals of Database Systems*. Benjamin/Cummings, 2nd edition, 1994.

[Gadia, 1988] S.K. Gadia. A homogeneous relational model and query languages for temporal databases. *ACM Transactions On Database Systems*, 13:418–448, 1988.

[Gregersen and Jensen, 1999] H Gregersen and J. S. Jensen. Temporal Entity-Relationship models - a survey. *IEEE Transaction on Knowledge and Data Engineering*, 11(3):464–497, 1999.

[McBrien *et al.*, 1992] P. McBrien, A.H. Seltveit, and B. Wangler. An Entity-Relationship model extended to describe historical information. In *Proc. of CISMOD'92*, pages 244–260, Bangalore, India, 1992.

[Schild, 1993] Klaus D. Schild. Combining terminological logics with tense logic. In *Proceedings of the 6th Portuguese Conference on Artificial Intelligence, EPIA'93*, October 1993.

[Tansel and Tin, 1998] A.U. Tansel and E. Tin. Expressive power of temporal relational query languages and temporal completeness. In O. Etzion, S. Jajodia, and S. Sripada, editors, *Temporal Databases - Research and Practice*, Lecture Notes in Computer Science, pages 129–149. Springer-Verlag, 1998.

[Theodoulidis *et al.*, 1991] C. Theodoulidis, P. Loucopoulos, and B. Wangler. A conceptual modelling formalism for temporal database applications. *Information Systems*, 16(3):401–416, 1991.

[Wolter and Zakharyaschev, 1998a] F. Wolter and M. Zakharyaschev. Temporalizing description logics. In *Proceedings of FroCoS'98*, Amsterdam, NL, 1998.

[Wolter and Zakharyaschev, 1998b] Frank Wolter and Michael Zakharyaschev. Satisfiability problem in description logics with modal operators. In A.Cohn, L.Schubert, and S.C.Shapiro, editors, *Proc. of the 6 th International Conference on Principles of Knowledge Representation and Reasoning*, pages 512–523, Trento, Italy, June 1998. Morgan Kaufmann.