The Performance of TimeER Model by Description Logics

Nguyen Viet Chanh(✉) and Hoang Quang

Faculty of Information Technology – College of Sciences, Hue University, 77 Nguyen Hue Street, Hue City, Vietnam
chanhkhmt@gmail.com, hquang@hueuni.edu.vn

Abstract. The relationship between Description Logic (DL) and database is quite close. Indeed, the needs for the building of the systems that can manage both the database and the knowledge representation are really necessary. A description-logic based on the knowledge representation system not only allows the knowledge management, but also provides a standard framework which is considered to be very close to the language used to represent the Entity-Relationship model (ER model). On the other hand, the temporal ER model is used to model the time aspects of the conceptual database schema. Thus, the problem of the use of description logic to express temporal ER models is really useful for modeling the conceptual data models. Based on the temporal DL, Alessandro Artale et al. (2011) presented temporal ER schemas and integrity constraints in the form of complex inclusion dependencies. The paper approaches the representation method of Alessandro Artale and proposes mapping multi-valued attributes in the temporal ER model to DL. Description logic application in TimeER modeling

Keywords: ER model · Temporal ER model · Description logic · Temporal description logic

1 Introduction

In recent years, Description Logic has usually been mentioned as an effective knowledge representation method. Description Logic is applied in varied fields, it is considered as languages representing knowledge and inference. In particular applications, they can use description logic, the application domain's knowledge specified by the concepts and relationships.

During the past time, the description logic has been used in many fields such as: software technology, configuration setting, electronic library systems, information system, semantic web, natural language processing, and database administration…

Description logic has a quite close relationship with database. In fact, it is really necessary to build a system that is able to represent description logic knowledge while it still allows database administration. Database administration systems resolve date integrity issue and administers a large amount of data, while description logic knowledge base representation system manages knowledge. In addition, description

P.C. Vinh and V. Alagar (Eds.): ICCASA 2015, LNICST 165, pp. 282–297, 2016.
DOI: 10.1007/978-3-319-29236-6_28

logic provides a standard frame considered to be close to the languages is used to model data as Entity– Relationship model.

In temporal database, it has launched many different data models, each model has its certain advantages and disadvantages. Temporal ER model (Entity model– Relationship with time factor) is a model using for modeling temporal database in the concept That temporal ER model is performed as diagrams (Figs. 1 and 4) and has developed the time factor in the database schema, that is valid time and transaction time, makes us easily see the change of data at different times. Temporal ER models has been researched such as: TERM, RAKE, MOTAR, TEER, STEER, ERT, TimeER,….

In the other side, Entity– Relationship (ER) model has temporal component used to model temporal aspects of the conceptual database schema, such as valid time– a time that event happens is right in practice, and transaction time– a time that event is stored in database. Temporal ER model has two main approaches proposed by researchers are: Implicit approach (Fig. 1) and explicit approach (Fig. 4), which are used to support modeling temporal ER models, then to represent temporal integrity constraints. Different versions of ER model have been proposed to model temporal concepts of models at concept level. This modeling has provided some formalization methods and expansions in temporal ER model. However, there are some complex constraints which can not be represented in temporal ER model, and temporal ER model has many different versions which have some inconsistent representation symbols, they causes many difficulties for designers in designing database.

Based on description logic with temporal factor, Alessandro Artale and partners [1] have represented temporal ER schemes and integrity constraints by formalizing inclusion dependencies with inclusion axioms. This study, in addition to the introduction of a method performed by the authors, we would like to propose multi-valued attributes representation on temporal ER models by representation logic. In this paper, we present a method of performing the ER models representation by describing the logic time with the temporal factor and indicate the result through modeling temporal ER model with description logic. Finally, this is the conclusion.

2 Modeling Temporal ER Model by Description Logic

Representing a temporal ER model in the description logic is performed through defining a conversion function Φ from temporal ER model to knowledge base ALCQIT.

Modeling is performed as the following. All the names of entities and relationships in temporal ER scheme are switched in correspondence with the names of the concepts in ALCQIT. The names of the domains are corresponding with additional concepts in separated pair. The attributes of entity sets and the role of relationships in corresponding ER model are names of roles in ALCQIT and with limited number to clear that the attribute is single-valued, in case of the multi-valued attributes, this limited number will be removed. IS-A relationship among entity sets or relationships is modeled by using term axioms. Number version constraints in temporal ER model are represented by number of words in ALCQIT. Natures of temporal in ER model are represented in correspondence with temporal operators in ALCQIT [2].

As mentioned above, there are two approaches in building a temporal ER model: implicit approach and explicit approach. Therefore, in order to model temporal ER models, we need to perform this representation on each particular approach. However, this research only focuses on modeling with implicit ER model.

Firstly, we consider switching an ER model (regardless of integrity constraints) to ALCQIT knowledge base as the following.

2.1 Modeling Implicit Temporal ER Model

2.1.1 Switch Implicit Temporal ER Model to Knowledge Base

Consider temporal ER model in implicit approach as the following Fig. 1

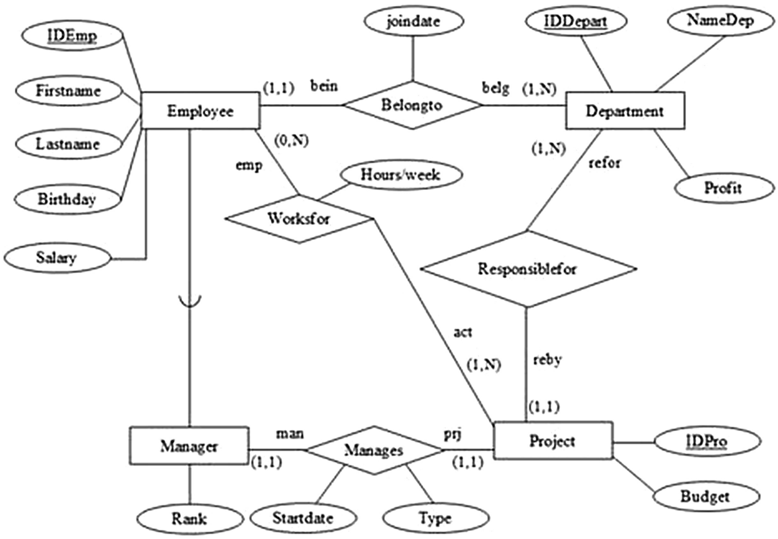

Fig. 1. A temporal ER model by implicit approach.

Given an ER model $\mathcal{D}$. Then, knowledge base Σ is called a switch from scheme $\mathcal{D}$ through the function $\Phi(\mathcal{D})$, if Σ contains 3 following sets:

- Set of elementary concepts $\Phi(A)$ corresponding with each range name, entity name and relationship name A in ER model $\mathcal{D}$;
- Set of elementary concepts $\Phi(P)$ corresponding with attributes name and roles name of a relationship P in ER model $\mathcal{D}$;
- Set of term axioms of Σ includes the following components:
  - Each IS-A relationship between two sets of entities E_1, E_2 (or two corresponding relationships R_1, R_2) with E_1 *Isa* E_2 (or R_1 *Isa* R_2) in $\mathcal{D}$ then we have the following term axiom:

$$\Phi(E_1) \sqsubseteq \Phi(E_2) \ (\text{or}\, \Phi(R_1) \sqsubseteq \Phi(R_2))$$

- Each set of entities E with attributes $A_1,...,A_h$ corresponding with value range $D_1,...,D_h$ then term axiom:

$$\Phi(E) \sqsubseteq \forall\Phi(A_1).\Phi(D_1) \sqcap \ldots \sqcap \forall\Phi(A_h).\Phi(D_h) \sqcap \\ (= 1\Phi(A_1)) \sqcap \ldots \sqcap (= 1\Phi(A_h))$$

- Each relationship R with attributes $A_1,...,A_h$ corresponding with value range $D_1,...,D_h$ then we have the following term axiom:

$$\Phi(R) \sqsubseteq \forall\Phi(A_1).\Phi(D_1) \sqcap \ldots \sqcap \forall\Phi(A_h).\Phi(D_h) \sqcap \\ (= 1\Phi(A_1)) \sqcap \ldots \sqcap (= 1\Phi(A_h))$$

- Each relationship R level k between sets of entities $E_1,...,E_k$ with R is connected by k roles $U_1,...,U_k$ then we have following term axiom:

$$\Phi(R) \sqsubseteq \forall\Phi(U_1).\Phi(E_1) \sqcap \ldots \sqcap\ = \forall\Phi(U_k).\Phi(E_k) \sqcap \\ (= 1\Phi(U_1)) \sqcap \ldots \sqcap (= 1\Phi(U_k))$$

- To the value *n, m* corresponding with the value (min, max) in number version constraint, on the role U connecting relationship R and set of entities E, and:
  - If $n \neq 0$ then we have the following term axiom:

$$\Phi(E) \sqsubseteq (\geq n(\Phi(U_i))^{-}.\Phi(R))\ \text{with}\ i \in \{1,\ldots,k\}, k\ \text{is level.}$$

  - If $m \neq \infty$ then we have the following term axiom:

$$\Phi(E) \sqsubseteq (\leq m(\Phi(U_i))^{-}.\Phi(R))\ \text{with}\ i \in \{1,\ldots,k\}, k\, islevel.$$

- Each pair of symbol X_1, X_2 that:
  - $X_1 \in D; X_2 \in E \cup D; X_1 \neq X_2$, or:
  - $X_1 \in R; X_2 \in E \cup R$; X_1 và X_2 with different levels, then we have the following term axiom: $\Phi(X_1) \sqsubseteq \neg\Phi(X_2)$, with D is the name of value range; R is set of relationships name and E is set of enities name.

$$\Phi(E) \sqsubseteq (= 1 \sim {}^{*}\Phi(A)) \\ \top \sqsubseteq (\leq 1(\Phi(A))^{-}.\Phi(E)$$

- To each attribute A which is a key attribute of entity set E then we have the following term axiom:

$$\Phi(\mathrm{E}) \sqsubseteq (= 1 \sim {}^{*}\Phi(\mathrm{A})) \\ \top \sqsubseteq (\leq 1(\Phi(\mathrm{A}))^{-}.\Phi(\mathrm{E})$$

- If the entity set E is the generalization of separate entity sets $E_1,...,E_n$ then it can be switched to the following term axiom:

$$\begin{aligned}
\Phi(E) &\sqsubseteq \Phi(E_1) \sqcup \ldots \sqcup \Phi(E_n) \\
\Phi(E_1) &\sqsubseteq \Phi(E) \sqcap \neg\Phi(E_2) \sqcap \neg\Phi(E_3) \sqcap \ldots \sqcap \neg\Phi(E_n) \\
\Phi(E_2) &\sqsubseteq \Phi(E) \sqcap \neg\Phi(E_3) \sqcap \neg\Phi(E_4) \sqcap \ldots \sqcap \neg\Phi(E_n) \\
\Phi(E_{n-1}) &\sqsubseteq \Phi(E) \sqcap \neg\Phi(E_n) \\
\Phi(E_n) &\sqsubseteq \Phi(E)
\end{aligned}$$

- Each entity set E with attribute $A_1,...,A_p$, $A_{p+1},...,A_h$ corresponding with value ranges $D_1,....,D_p$, $D_{p+1},...,D_h$, in which $A_1,..,A_p$ are single-valued attributes and $A_{p+1},...,A_h$ are multi-valued attributes, then we have the following axiom:

$$\begin{gathered}
\Phi(E) \sqsubseteq \forall\Phi(A_1).\Phi(D_1) \sqcap \ldots \sqcap \forall\Phi(A_p).\Phi(D_p) \sqcap \\
(\geq 1\Phi(A_{p+1}).\Phi(D_{p+1})) \sqcap \ldots \sqcap (\geq 1\Phi(A_h).\Phi(D_h)) \sqcap (= 1\Phi(A_1)) \sqcap \ldots \\
\sqcap (= 1\Phi(A_p)) \sqcap (\neg \geq 1\Phi(A_{p+1}).\neg\Phi(D_{p+1})) \sqcap \ldots \sqcap (\neg \geq 1\Phi(A_h).\neg\Phi(D_h))
\end{gathered}$$

- If an entity set E with attribute A which is a compound attribute with components $A_1,...,A_p$ then we have the term axiom for representation as following:

$$\begin{gathered}
\Phi(E) \sqsubseteq \forall\Phi(A).(\forall\Phi(A_1).\Phi(D_1) \sqcap \ldots \sqcap \forall\Phi(A_p).\Phi(D_p) \sqcap \\
(= 1\Phi(A_1)) \sqcap \ldots \sqcap (= 1\Phi(A_p))) \sqcap (= 1\Phi(A))
\end{gathered}$$

Example 1.

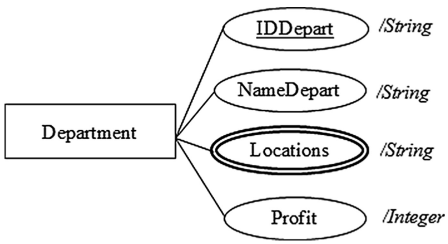

Fig. 2. Example of multi-valued attribute

In the above example, entity set *Department* with attribute *Locations* is multi-valued attribute, we have the representing term axiom:

$$\begin{gathered}
Department \sqsubseteq \forall IDDepart.String \sqcap \forall NameDepart.String \sqcap \\
(\geq 1 Locations.String) \sqcap \forall Profit.Integer \sqcap (= 1 IDDepart) \sqcap \\
(= 1 NameDepart) \sqcap (=1 Profit)) \sqcap (\neg \geq 1 Locations.\neg String)
\end{gathered}$$

Example 2.

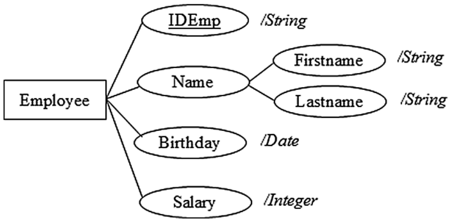

Fig. 3. Example of compound attribute

In Fig. 3 the entity set *Employee* with the attribute *Name* has a compound attribute containing two elementary attributes *FirstName* and *LastName*, we will have a representation term axiom:

$$Employee \sqsubseteq IDEmp.String \sqcap \forall Name.(\forall Firstname.String \sqcap Lastname.String \sqcap (= 1Firstname) \sqcap (= 1Lastname)) \sqcap \forall Birthday.Date \sqcap \forall Salary.Integer \sqcap (= 1IDEmp) \sqcap (= 1Name) \sqcap (= 1Birthday) \sqcap (= 1Salary)$$

In addition, this constraint of temporal integrity is represented in description logic by adding term axioms in …. A term axiom represents an inclusion among the concepts. Therefore, a integrity constraint is an inclusive dependence form represented in temporal description logic ALCQIT.

2.1.2 Inclusive Dependence

An integrity constraint for an ER model $\mathcal{D}$ is an inclusive dependence that can be represented in a knowledge base in correspondence with Σ by a term axiom in a form of C$\sqsubseteq$D, in which elementary concept appears in C, D corresponding with the name of the domain, entity set or relationship in $\mathcal{D}$.

There is a correspondence in switching between valid database status of $\mathcal{D}$ and models of deduced knowledge base. The appearance of this correspondence drags on correspondence between solutions for checking a nature in ER model and corresponding deduction in description logic and vice versa. Thus, it can use deduction operations in description logic to check a nature of ER schema [4].

Example 3. Consider the example shown in Fig. 2, coding integrity constraints is represented by term axioms in a knowledge base Σ_{IC} as following:

$$Manager \sqsubseteq Qualified\ \mathcal{S}\ (Employee \sqcap \neg Manager)$$

The above constraint shows that all of managers are eligible after a period of being a staff.

In fact, integrity constraints are logical deductions from $\Sigma_{ER} \cup \Sigma_{IC}$, for example:

$$\Sigma_{ER} \cup \Sigma_{IC} \models Project \sqsubseteq \exists(act^{-} \circ emp).\neg Manager$$

The above constraint shows that each project exists a staff who is not a manager working for it.

$$\Sigma_{ER} \cup \Sigma_{IC} \models Manager \sqsubseteq {}^{-}\exists(emp^{-} \circ act).Project$$

The above constraint represents that a manager needs to have a temporal of the work in the past for a project (maybe another project).

2.2 Modeling Explicit Temporal ER Model

As we know, temporal ER model with explicit approach remains non-temporal semantic meaning for normal ER models, in the other side, it also implements new structures which allow to represent entity sets, temporal relationships and temporal dependences between them.

In this part, we propose a formalization approach to model explicit temporal ER model by using simple constraints to define temporal and non-temporal structures, thus it remains upward compatibility. Temporal description logic ALCQIT can represent explicit ER model, at first by applying switching principles in the previous part (modeling explicit temporal ER model) and then adding axioms to distinguish temporal and non-temporal structures. Below are some presentations of additional axioms for coding this model.

2.2.1 Entity Set and Temporal and Non-Temporal Relationship

As stated above, for an explicit temporal ER model, entity sets and relationships include non-temporal structure and temporal structure. Therefore, when modeling, we have more additional axioms to clear the following structure:

- Each non-temporal entity set E is represented by the following axiom:

$$\Phi(E) \sqsubseteq (\sim {}^{+}\Phi(E)) \sqcap (\sim {}^{-}\Phi(E)), \text{ that means } \Phi(E) \equiv \sim^{*}\Phi(E)$$

  The above axiom shows that entity set is right whenever that entity set must be right at any point in the past and the future. Indeed, non-temporal entity sets have an overall living temporal.
- In the other side, if the entity set E is a temporal entity set, it will be represented:

$$\Phi(E) \sqsubseteq ({}^{+}\neg\Phi(E)) \sqcup ({}^{-}\neg\Phi(E))$$

  The axiom represents that there is a point in the past or in the future when entities exist. Indeed, temporal entity sets have a limit of living temporal of entity set.

  Similar to entity sets, the relationships also have axioms to distinguish temporal and non-temporal structures.

- Each non-temporal relationship R level k between entity sets $E_1,\ldots,E_k$ that R is connected by k roles $U_1,\ldots,U_k$ is represented by the following term axioms:

$$\Phi(R) \sqsubseteq (\sim{}^{+}\Phi(R)) \sqcap (\sim{}^{-}\Phi(R)) - \text{ that means } \Phi(R) \equiv \sim{}^{*}\Phi(R);$$
$$\Phi(R) \sqsubseteq (= 1 \sim{}^{*}\Phi(U_1)) \sqcap \ldots \sqcap (= 1 \sim{}^{*}\Phi(U_k))$$

- If the relationship R is a temporal relationship, it will be distinguished by the following term axiom:

$$\Phi(R) \sqsubseteq ({}^{+}\neg\Phi(R)) \sqcup ({}^{-}\neg\Phi(R)) \sqcup \neg((= 1 \sim{}^{*}\Phi(U_1)) \sqcap \ldots \sqcap (= 1 \sim{}^{*}\Phi(U_k)))$$

Example 4. Consider temporal ER model in Fig. 4.

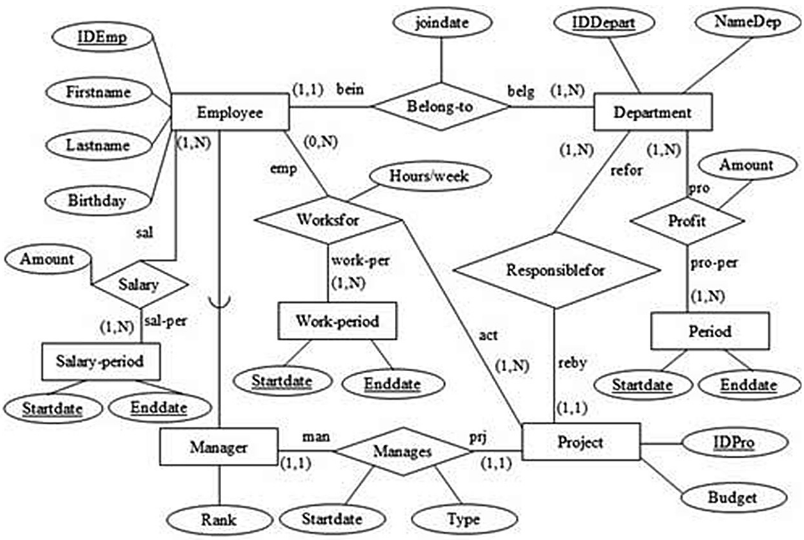

Fig. 4. An example of temporal ER model [4]

In Fig. 4. The entity set *Department* can be considered as a non-temporal entity set because organizational structure of the business does not change over temporal, while the entity set *Manager* can be considered as a temporal entity set because the manager can change over temporal. Thus we have the distinguishing axiom which is temporal or non-temporal entity set added when representing the entity set *Department* and *Manager* as following:

$$Department \sqsubseteq (\sim{}^{+}Department) \sqcap (\sim{}^{-}Department)$$
$$Manager \sqsubseteq ({}^{+}\neg Manager) \sqcup ({}^{-}\neg Manager)$$

With considered example in Figs. 1 and 4; the relationship *Worksfor* is a temporal relationship, the relationship *Responsiblefor* is a non-temporal relationship, therefore we have the following distinguishing axioms:

- Non-temporal relationship *Responsiblefor*

$$Responsiblefor \sqsubseteq (\sim^{+} Responsiblefor) \sqcap (\sim^{-} Responsiblefor);$$
$$Responsiblefor \sqsubseteq (= 1 \sim^{*} reby) \sqcap (= 1 \sim^{*} refor)$$

- Temporal relationship *Worksfor*

$$Worksfor \sqsubseteq (^{+} \neg Worksfor) \sqcup (^{-} \neg Worksfor) \sqcup$$
$$\neg((= 1 \sim^{*} act) \sqcap (= 1 \sim^{*} emp))$$

Using of deduction ability of ALCQIT can support database designer to identify matching natures with temporal ER scheme:

- An entity subset of a temporal entity set is a temporal entity set.
- An entity subset of a non-temporal entity set and an entity father-set of a temporal entity set or implicit temporal entity set may be a non-temporal entity set or temporal entity set or implicit temporal entity set.
- An entity father-set of a non-temporal entity set is a non-temporal entity set.
- A schema is inconsistent if one entity set of all separate subsets is a temporal entity set.
- Entity sets taking part in non-temporal relationships can be non-temporal entity sets or implicit temporal entity sets.
- Entity sets taking part in temporal relationships or implicit relationships with temporal factor can be non-temporal entity sets or implicit temporal entity sets or temporal entity sets.

For instance, we consider the following example to see the correction of the scheme organization for using both temporal entity set and non-temporal entity set, we consider the interactive between entity sets by IS-A relations. Assuming that there is an IS-A relation between a time less entity set E_1 and a temporal entity set E_2. The temporal ER model switched to the following knowledge base is not satisfied:

$$\Phi(E_1) \sqsubseteq (\sim^{+} \Phi(E_1)) \sqcap (\sim^{-} \Phi(E_1))$$
$$\Phi(E_2) \sqsubseteq (^{+} \neg \Phi(E_2)) \sqcup (^{-} \neg \Phi(E_2))$$
$$\Phi(E_1) \sqsubseteq \Phi(E_2)$$

Thus, a non-temporal entity set cannot be a subset of a temporal entity set, this is always true with taxonomic relation included in temporal ER model. This can be explained by an observation: if the relation IS-A has a representation called a, for example: a is representation of E_1 and E_2 at a certain instance t_0– is represented by the following symbol set: $\{a : E_1, a : E_2\}_{t_0}$. According to the statement of temporal axiom for E_2, at an instance of $t1$, the representation a is not E_2 - $\{a : \neg E_2\}_{t_1}$. In the other side, because E_1 is a non-temporal entity set, the representation a is E_1 at any instance, and especially at the instance of t_1 - $\{a : \neg E_2, a : E_1\}_{t_1}$. According to layer relation, it will be shown that a is E_2 at t_1 - $\{a : \neg E_2, a : E_1, a : E_2\}_{t_1}$. This shows that both a of E_2 and a are not E_2 at t_1, this is a contradiction.

Based on those comments, it is easy to understand the reasons of the following consequences:

$$\Phi(E_2) \sqsubseteq \{({}^{+}\neg\Phi(E_2)) \sqcup ({}^{-}\neg\Phi(E_2)), \Phi(E_1) \sqsubseteq \Phi(E_2)\} \vDash \Phi(E_1) \sqsubseteq ({}^{+}\neg\Phi(E_1)) \sqcup ({}^{-}\neg\Phi(E_1))$$

$$\Phi(\mathrm{E}_1) \sqsubseteq \{(\sim{}^{+}\Phi(\mathrm{E}_1)) \sqcap (\sim{}^{-}\Phi(\mathrm{E}_1)), \Phi(\mathrm{E}_1) \sqsubseteq \Phi(\mathrm{E}_2)\} \vDash \Phi(\mathrm{E}_2) \sqsubseteq (\sim{}^{+}\Phi(\mathrm{E}_2)) \sqcap (\sim{}^{-}\Phi(\mathrm{E}_2))$$

that means, all entity subsets of temporal entity set are temporal entity sets and an entity father-set of a non-temporal entity set is a non-temporal entity set [4].

2.2.2 Attributes with Time Factor

At different instances, an entity set may have different values for the same attribute. These attributes are combined by a valid time, in other words, they are attributes with time factor. Therefore, so as to model attributes with time factor, there are some more term axioms to distinguish attributes with and without time factor besides applying switching principles for attribute given in the part of modeling a model.

An entity set E (corresponding with R) with attributes $A_1,\ldots,A_h$ and with:

- Each attribute A_i (so that $\mathrm{A_i} \in \{\mathrm{A}_1, \ldots, \mathrm{A_h}\}$) is a non-temporal attribute of the entity set E (corresponding with the relationship R) then the term axiom is added as following:

$$\Phi(E) \sqsubseteq (= 1 \sim {}^{*}\Phi(A_i))$$

equal to $\quad \Phi(R) \sqsubseteq (= 1 \sim {}^{*}\Phi(A_i))$

- Each attribute A_i (so that $A_i \in \{A_1, \ldots, A_h\}$) is a temporal attribute of the entity attribute E (corresponding with the relationship R) then the term axiom is added as following:

$$\Phi(E) \sqsubseteq \neg(= 1 \sim {}^{*}\Phi(A_i))$$

equal to $\quad \Phi(R) \sqsubseteq \neg(= 1 \sim {}^{*}\Phi(A_i))$

Example 5. In the entity set *Employee* in Figs. 1 and 4, we see that attributes *FirstName*, *LastName*, *Birthday* are non-temporal attributes, *Salary* is temporal attribute, so we have the distinguishing axioms as following:

$$Employee \sqsubseteq (= 1 \sim {}^{*}FirstName);$$
$$Employee \sqsubseteq (= 1 \sim {}^{*}LastName);$$
$$Employee \sqsubseteq (= 1 \sim {}^{*}Birthday);$$
$$Employee \sqsubseteq \neg(= 1 \sim {}^{*}Salary)$$

2.2.3 Coding Time Number Version Constraint

For temporal ER model with explicit approach, besides number versions constraint *(min, max)* assigned a role to limit number of entity of an entity set that is allowed to take part in through roles of relationship, there is also a constraint of living temporal of entity set with relationship represented by the pair of number version *(minL, maxL)* on the role from entity set to relationship. With the meaning that during the time of the exist of an entity, each entity $e \in E$ will have relationship with minimum of *minL* element and maximum of *maxL* element of relationship *R*.

Assuming that the values *n, m* are corresponding with the values *(minL, maxL)* in number version constraint of living temporal of entity set to relationship, and:

- If n ≠ 0 then we have the following term axiom:

$$\Phi(E) \sqsubseteq (\geq n(\Phi(U_i))^{-}.^{*}\Phi(R) \text{ with } i \in \{1, \ldots, k\} \text{ and } k \text{ is the level.}$$

- If m ≠ ∞ then we have the following term axiom:

$$\Phi(E) \sqsubseteq (\leq m(\Phi(U_i))^{-}.^{*}\Phi(R)) \text{ with } i \in \{1, \ldots, k\} \text{ and } k \text{ is level.}$$

Example 6. Assuming that during exist the manager has managed 1 project at minimum and 5 projects at maximum. With this constraint of living temporal we have the following representation:

$$Manager \sqsubseteq (\geq 1 man^{-}.^{*}Manages) \sqcap (\leq 5 man^{-}.^{*}Manages)$$

3 Application of Description Logic in Modeling TimeER Model

In this part, the paper will proceed modeling TimeER model with description logic. As above introduction, TimeER model (Fig. 5) is a temporal ER model in explicit approach. Therefore, in order to model this model we need to use switching principles for explicit temporal ER model, in other words, we have to apply both switch definition in 2.1 and implement axioms to distinguish temporal and non-temporal structures in 2.2 (Table 1).

Considering temporal ER model represented in Fig. 4. If we represent this model with TimeER model, we have the following scheme:

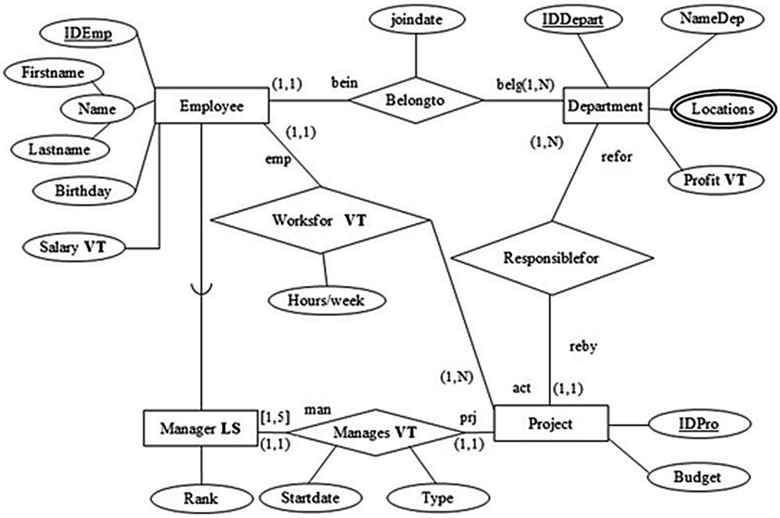

Fig. 5. An example of TimeER model

With TimeER model in Fig. 5, assuming that attributes of entity sets and relationships in this model are corresponding with the domains as following:

Modeling TimeER model (Fig. 5) by switching function Φ from TimeER model to knowledge base Σ. We have knowledge base received from this switch including:

Table 1. Corresponding domains for attribute in Fig. 5

Entity set or relationship	Attribute and corresponding domain
Employee	*{IDEmp: String, FirstName: String, LastName: String, Birthday: Date, Salary: Integer}*
Manager	*{Rank: String}*
Department	*{IDDepart: String, NameDep: String, Locations: String, Profit: Integer}*
Project	*{IDPro: String, Budget: Integer}*
Belongto	*{joindate: Date}*
Worksfor	*{Hours/week: String}*
Manages	*{Startdate: Date, Type: String}*

Set of elementary concepts $\Phi(A) = \{String, Integer, Date, Employee, Manager, Department, Project, Belongto, Worksfor, Responsiblefor, Manages\}$

Set of elementary roles $\Phi(P) = \{IDEmp, FirstName, LastName, Name, Birthday, Salary, joindate, IDDepart, NameDep, Locations, Profit, Hours/week, IDPro, Budget, Startdate, Type, Rank, emp, act, bein, belg, refor, reby, man, prj\}$

Set of term axioms of Σ as the following:

- Entity sets and their attributes

$$Employee \sqsubseteq \forall IDEmp.String \sqcap \forall Name.(\forall Firstname.String \sqcap \forall Lastname.String \sqcap (=1 Firstname) \sqcap (=1 Lastname)) \sqcap \forall Birthday.Date \sqcap \forall Salary.Integer \sqcap (=1 IDEmp) \sqcap (=1 Name) \sqcap (=1 Birthday) \sqcap (=1 Salary)$$

$$Department \sqsubseteq \forall IDDepart.String \sqcap \forall NameDepart.String \sqcap (\geq 1 Locations.String) \sqcap \forall Profit.Integer \sqcap (=1 IDDepart) \sqcap (=1 NameDepart) \sqcap (=1 Profit)) \sqcap (\neg \geq 1 Locations.\neg String)$$

$$Project \sqsubseteq \forall IDPro.String \sqcap \forall Budget.Integer \sqcap (=1 IDPro) \sqcap (=1 Budget)$$

$$Manager \sqsubseteq Employee \sqcap \forall Rank.String \sqcap (=1 Rank)$$

- Relationships and their attributes

$$Belongto \sqsubseteq \forall joindate.Date \sqcap (=1 joindate)$$

$$Worksfor \sqsubseteq \forall Hours/week.String \sqcap (=1 Hours/week)$$

$$Manages \sqsubseteq \forall Startdate.Date \sqcap \forall Type.String \sqcap (=1 Startdate) \sqcap (=1 Type)$$

- Relationships with connecting role between these relationships and entity sets

$$Belongto \sqsubseteq \forall bein.Employee \sqcap \forall belg.Department \sqcap (=1 bein) \sqcap (=1 belg)$$

$$Workfor \sqsubseteq \forall emp.Employee \sqcap \forall act.Project \sqcap (=1 emp) \sqcap (=1 act)$$

$$Responsiblefor \sqsubseteq \forall refor.Department \sqcap \forall reby.Project \sqcap (=1 refor) \sqcap (=1 reby)$$

$$Manages \sqsubseteq \forall man.Manager \sqcap \forall prj.Project \sqcap (=1 man) \sqcap (=1 prj)$$

- Non-temporal constraints

$$Employee \sqsubseteq (=1 bein^{-}.Belongto) \sqcap (=1 emp^{-}.Worksfor)$$

$$Project \sqsubseteq (=1 prj^{-}.Manages) \sqcap (=1 reby^{-}.Responsiblefor) \sqcap (\geq 1 act^{-}.Worksfor)$$

$$Department \sqsubseteq (\geq 1 belg^{-}.Belongto) \sqcap (\geq 1 refor^{-}.Responsiblefor)$$

$$Manager \sqsubseteq (=1 man^{-}.Manages)$$

- Keys of entity sets

$$Employee \sqsubseteq (= 1^{*} IDEmp)$$
$$\top \sqsubseteq (\leq 1 IDEmp^{-}.Employee)$$
$$Department \sqsubseteq (= 1^{*} IDDepart)$$
$$\top \sqsubseteq (\leq 1 IDDepart^{-}.Department)$$
$$Project \sqsubseteq (= 1^{*} IDPro)$$
$$\top \sqsubseteq (\leq 1 IDPro^{-}.Project)$$

- Distinguishing temporal and non-temporal entity sets

$$Employee \sqsubseteq (\sim^{+} Employee) \sqcap (\sim^{-} Employee)$$
$$Project \sqsubseteq (\sim^{+} Project) \sqcap (\sim^{-} Project)$$
$$Department \sqsubseteq (\sim^{+} Department) \sqcap (\sim^{-} Department)$$
$$Manager \sqsubseteq (^{+} \neg Manager) \sqcup (^{-} \neg Manager)$$

- Distinguishing temporal and non-temporal relationships

$$Belongto \sqsubseteq (\sim^{+} Belongto) \sqcap (\sim^{-} Belongto)$$
$$Belongto \sqsubseteq (= 1 \sim^{*} bein) \sqcap (= 1 \sim^{*} belg)$$
$$Responsiblefor \sqsubseteq (\sim^{+} Responsiblefor) \sqcap (\sim^{-} Responsiblefor)$$
$$Responsiblefor \sqsubseteq (= 1 \sim^{*} reby) \sqcap (= 1 \sim^{*} refor)$$
$$Workfor \sqsubseteq (^{+} \neg Workfor) \sqcup (^{-} \neg Workfor) \sqcup \neg((= 1^{*} act) \sqcap (= 1^{*} emp))$$
$$Manages \sqsubseteq (^{+} \neg Manages) \sqcup (^{-} \neg Manages) \sqcup \neg((= 1 \sim^{*} man) \sqcap (= 1 \sim^{*} prj))$$

- Attributes with and without time factor

$$Employee \sqsubseteq (= 1 \sim^{*} Name).((= 1 \sim^{*} FirstName) \sqcap (= 1 \sim^{*} LastName)) \sqcap (= 1 \sim^{*} Birthday)$$
$$Employee \sqsubseteq \neg(= 1 \sim^{*} Salary)$$
$$Department \sqsubseteq (= 1 \sim^{*} NameDep) \sqcap (= 1 \sim^{*} Locations)$$
$$Department \sqsubseteq \neg(= 1 \sim^{*} Profit)$$
$$Project \sqsubseteq (= 1 \sim^{*} Budget)$$
$$Manager \sqsubseteq (= 1 \sim^{*} Rank)$$
$$Belongto \sqsubseteq (= 1 \sim^{*} joindate)$$
$$Worksfor \sqsubseteq (= 1 \sim^{*} Hours/week)$$
$$Manages \sqsubseteq (= 1 \sim^{*} Startdate) \sqcap (= 1 \sim^{*} Type)$$

- Constraint of living time

$$Manager \sqsubseteq (\geq 1 man^{-}.^{*}Manages) \sqcap (\leq 5 man^{-}.^{*}Manages)$$

4 Conclusion

In this paper, we have shown the representation of implicit and explicit TimeER models with temporal description logic. In addition, we have implemented representation multi-valued and compound attribute on TimeER models with inclusive axioms in description logic. The paper will also present an application of description logic in modeling TimeER model with the above approaches.

Besides, in practice there are still many extending issues in theory of description logic and its applications in the field of the database. Therefore, according to this research, we will continue to use description logic to perform database models having the time factor, which are relational database model and object-oriented database model.

References

1. Artale, A., Kontchakov, R., Ryzhikov, V., Zakharyaschev, M.: Tailoring temporal description logics for reasoning over temporal conceptual models. In: Tinelli, C., Sofronie-Stokkermans, V. (eds.) FroCoS 2011. LNCS, vol. 6989, pp. 1–11. Springer, Heidelberg (2011)
2. Artale, A., Franconi, E., Wolter, F., Zakharyaschev, M.: Reasoning over conceptual schemas and queries in temporal database. In: Proceedings of the 9th Italian Database Conference (2002)
3. Artale, A., Franconi, E.: Reasoning with enhanced temporal entity-relationship models. In: Proceedings of the International Workshop on Spatio-Temporal Data Models and Languages. IEEE Computer Society Press, Also in Proceedings of the 6th International Workshop on Knowledge Representation meets Databases (KRDB 1999), and in Proceedings of the 1999 International Workshop on Description Logics (DL 1999) (1999)
4. Calvanes, D., Lenzerini, M., Nardi, D.: Description logics for conceptual data modeling. In: Chomicki, J., Saake, G. (eds.) Logics for Databases and Information Systems. Kluwer Academic Publisher (1998)
5. Artale, A., Franconi, E.: Temporal ER modelling with description logics. In: Akoka, J., Bouzeghoub, M., Comyn-Wattiau, I., Métais, E. (eds.) ER 1999. LNCS, vol. 1728, pp. 81–95. Springer, Heidelberg (1999)
6. Artale, A., Franconi, E., Mandreoli, F.: Description logics for modelling dynamic information. In: Chomicki, J., van der Meyden, R., Saake, G. (eds.) Logics for Emerging Applications of Databases. LNCS, pp. 239–275. Springer, Heidelberg (2003)
7. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Shneiter, P.F.: The Description Logic Handbook: Theory, Implementation and Applications, 2nd edn. Cambridge University Press, Cambridge (2007)

8. Artale, A., Franconi, E.: Foundations of temporal conceptual data models. In: Borgida, A.T., Chaudhri, V.K., Giorgini, P., Yu, E.S. (eds.) Conceptual Modeling: Foundations and Applications. LNCS, vol. 5600, pp. 10–35. Springer, Heidelberg (2009)
9. Rudolph, S.: Foundations of Description Logics. Karlsruhe Institute of Technology, Germany (2011)
10. Hoang, Q., Van Nguyen, T.: Extraction of a temporal conceptual model from a relational database. Int. J. Intell. Inf. Database Syst. **7**(4), 340–355 (2013)