Essential and Mandatory Part-Whole Relations in Conceptual Data Models

Alessandro Artale and C. Maria Keet

Free University of Bozen-Bolzano, IT – {artale,keet}@inf.unibz.it

Abstract. A recurring problem in conceptual modelling and ontology development is the representation of part-whole relations, with a requirement to be able to distinguish between essential and mandatory parts. To solve this problem, we formally characterize the semantics of these shareability notions by resorting to the temporal conceptual model $\mathcal{ER}_{VT}$ and its formalization in the description logic $\mathcal{DLR}_{\mathcal{US}}$.

1 Introduction

Modeling part-whole relations and aggregations have been investigated and experimented with from various perspectives and this has resulted in advances and better problem identification [2, 5, 7, 14, 17, 18, 20]. Several issues, such as transitivity and types of part-whole relations, are being addressed successfully, whereas other sub-topics, such as *life cycle semantics* of parts and wholes, remain largely still open with alternative approaches [8, 14, 18].

In particular, the issue of essential versus mandatory parts and wholes is a long-standing open problem [2]. Informally, it indicates the difference between the usual mandatory participation where a part is part of *some* whole (or vv.) but can become part of another whole at a later time (e.g., a heart of a human body), versus necessarily being part of only *the same* whole (e.g., the crucial paragraph in a scientific article that contains the novel contribution is essential). This has been addressed in part by [5, 14, 18] for UML class diagrams—in [14] modalities such as $\Box$ for necessity and ε for existence are used while constraints expressed in the UML's Object Constraint Language are proposed in [5]. A common perspective that emerges from these works is the need to add a temporal dimension to the part-whole relation. An approach to temporalize the part-whole relation is to turn a part-of predicate into a ternary relation, such that we have p part of w at time t: $part\_of(p, w, t)$ [8, 21].

The formalization proposed here builds on previous efforts to formalize temporal conceptual models based on a modal temporal logic. Namely, we rely on a previous work to define the $\mathcal{ER}_{VT}$ model [3], a temporal Extended Entity-Relationship (EER) model equipped with both a textual and a graphical syntax and based on a model-theoretic semantics. This paper extends $\mathcal{ER}_{VT}$ with the notion of *mandatory* and *essential* parts by *fully temporalizing the part-whole relation and its participating classes*. As a byproduct, we also define the notion of *status relations* that constrains the temporal evolution of a relation. Furthermore, since $\mathcal{ER}_{VT}$ can be mapped into the temporal description logic $\mathcal{DLR}_{\mathcal{US}}$

[3], we show how $\mathcal{DLR}_{\mathcal{US}}$ can capture in a succinct way the newly introduced constructors (status relations, mandatory and essential parts).

This paper essentially provides a formalization of a conceptual model able to capture the differences between essential and mandatory parts while showing the foundation for representing the evolution of relations along their life cycles. The formal model-theoretic semantics and the corresponding $\mathcal{DLR}_{\mathcal{US}}$ axiomatization are general enough to be applied to other (temporal) conceptual modelling and ontology languages.

The remainder of the paper is organised as follows. We start with technical details of $\mathcal{DLR}_{\mathcal{US}}$ and $\mathcal{ER}_{VT}$ in section 2. In section 3 we introduce status relations and prove the properties of essential parts and wholes. Last, we close with conclusions in section 4.

2 Temporal Data Models

To capture mandatory and essential parts and represent them in a conceptual modelling language we introduce here representation languages able to capture time varying information. In particular, we consider the temporal EER $\mathcal{ER}_{VT}$ [3, 4] together with its mapping to the temporal DL $\mathcal{DLR}_{\mathcal{US}}$ [3]. $\mathcal{ER}_{VT}$ has a model-theoretic semantics giving a precise meaning to any corresponding icon in the graphical diagrams. $\mathcal{ER}_{VT}$ is based on the general principle that with an UML/EER/ORM to DL transformation [6, 10, 11, 16] we can provide the sought-after genericity together with the possibility to reason on top of conceptual data models.

This section summarizes both $\mathcal{DLR}_{\mathcal{US}}$ and $\mathcal{ER}_{VT}$. They are both used in the following sections when the basic conceptual data model is extended to capture particular properties of part-whole relations.

The temporal description logic $\mathcal{DLR}_{\mathcal{US}}$. The temporal description logic $\mathcal{DLR}_{\mathcal{US}}$ [3] combines the propositional temporal logic with the *Since* and *Until* operators and the (non-temporal) description logic $\mathcal{DLR}$ [9] that serves as common foundational language for various conceptual data modeling languages [11]. $\mathcal{DLR}_{\mathcal{US}}$ can be regarded as an expressive fragment of the first-order temporal modal logic $L^{\{\textbf{since, until}\}}$ [12, 15].

The basic syntactical types of $\mathcal{DLR}_{\mathcal{US}}$ are *classes* and n-ary *relations* ($n \geq 2$). Starting from a set of *atomic classes* (denoted by CN), a set of *atomic relations* (denoted by RN), and a set of *role symbols* (denoted by U), we can define inductively (complex) class and relation expressions (see upper part of Fig. 1), where the binary constructors ($\sqcap, \sqcup, \mathcal{U}, \mathcal{S}$) are applied to relations of the same arity, i, j, k, n are natural numbers, $i \leq n$, j does not exceed the arity of R, and all the Boolean constructors are available for both class and relation expressions. The selection expression $U_i/n : C$ denotes an n-ary relation whose i-th argument ($i \leq n$), named U_i, is of type C. If it is clear from the context, we omit n and write $(U_i : C)$. The projection expression $\exists^{\lessgtr k}[U_j]R$ is a generalisation with cardinalities of the projection operator over argument U_j

$$
\begin{aligned}
C \to\ & \top \mid \bot \mid CN \mid \neg C \mid C_1 \sqcap C_2 \mid \exists^{\leqslant k}[U_j]R \mid \\
& \Diamond^+ C \mid \Diamond^- C \mid \Box^+ C \mid \Box^- C \mid \oplus C \mid \ominus C \mid C_1 \,\mathcal{U}\, C_2 \mid C_1 \,\mathcal{S}\, C_2 \\
R \to\ & \top_n \mid RN \mid \neg R \mid R_1 \sqcap R_2 \mid U_i/n : C \mid \\
& \Diamond^+ R \mid \Diamond^- R \mid \Box^+ R \mid \Box^- R \mid \oplus R \mid \ominus R \mid R_1 \,\mathcal{U}\, R_2 \mid R_1 \,\mathcal{S}\, R_2
\end{aligned}
$$

$$
\begin{aligned}
\top^{\mathcal{I}(t)} &= \Delta^{\mathcal{I}} \\
\bot^{\mathcal{I}(t)} &= \emptyset \\
CN^{\mathcal{I}(t)} &\subseteq \top^{\mathcal{I}(t)} \\
(\neg C)^{\mathcal{I}(t)} &= \top^{\mathcal{I}(t)} \setminus C^{\mathcal{I}(t)} \\
(C_1 \sqcap C_2)^{\mathcal{I}(t)} &= C_1^{\mathcal{I}(t)} \cap C_2^{\mathcal{I}(t)} \\
(\exists^{\leqslant k}[U_j]R)^{\mathcal{I}(t)} &= \{\, d \in \top^{\mathcal{I}(t)} \mid \sharp\{\langle d_1, \ldots, d_n\rangle \in R^{\mathcal{I}(t)} \mid d_j = d\} \leqslant k\} \\
(C_1 \,\mathcal{U}\, C_2)^{\mathcal{I}(t)} &= \{\, d \in \top^{\mathcal{I}(t)} \mid \exists v > t.(d \in C_2^{\mathcal{I}(v)} \wedge \forall w \in (t, v).d \in C_1^{\mathcal{I}(w)})\} \\
(C_1 \,\mathcal{S}\, C_2)^{\mathcal{I}(t)} &= \{\, d \in \top^{\mathcal{I}(t)} \mid \exists v < t.(d \in C_2^{\mathcal{I}(v)} \wedge \forall w \in (v, t).d \in C_1^{\mathcal{I}(w)})\} \\
(\top_n)^{\mathcal{I}(t)} &\subseteq (\Delta^{\mathcal{I}})^n \\
RN^{\mathcal{I}(t)} &\subseteq (\top_n)^{\mathcal{I}(t)} \\
(\neg R)^{\mathcal{I}(t)} &= (\top_n)^{\mathcal{I}(t)} \setminus R^{\mathcal{I}(t)} \\
(R_1 \sqcap R_2)^{\mathcal{I}(t)} &= R_1^{\mathcal{I}(t)} \cap R_2^{\mathcal{I}(t)} \\
(U_i/n : C)^{\mathcal{I}(t)} &= \{\, \langle d_1, \ldots, d_n\rangle \in (\top_n)^{\mathcal{I}(t)} \mid d_i \in C^{\mathcal{I}(t)}\} \\
(R_1 \,\mathcal{U}\, R_2)^{\mathcal{I}(t)} &= \{\, \langle d_1, \ldots, d_n\rangle \in (\top_n)^{\mathcal{I}(t)} \mid \\
&\qquad \exists v > t.(\langle d_1, \ldots, d_n\rangle \in R_2^{\mathcal{I}(v)} \wedge \forall w \in (t, v).\, \langle d_1, \ldots, d_n\rangle \in R_1^{\mathcal{I}(w)})\} \\
(R_1 \,\mathcal{S}\, R_2)^{\mathcal{I}(t)} &= \{\, \langle d_1, \ldots, d_n\rangle \in (\top_n)^{\mathcal{I}(t)} \mid \\
&\qquad \exists v < t.(\langle d_1, \ldots, d_n\rangle \in R_2^{\mathcal{I}(v)} \wedge \forall w \in (v, t).\, \langle d_1, \ldots, d_n\rangle \in R_1^{\mathcal{I}(w)})\} \\
(\Diamond^+ R)^{\mathcal{I}(t)} &= \{\langle d_1, \ldots, d_n\rangle \in (\top_n)^{\mathcal{I}(t)} \mid \exists v > t.\, \langle d_1, \ldots, d_n\rangle \in R^{\mathcal{I}(v)}\} \\
(\oplus R)^{\mathcal{I}(t)} &= \{\langle d_1, \ldots, d_n\rangle \in (\top_n)^{\mathcal{I}(t)} \mid \langle d_1, \ldots, d_n\rangle \in R^{\mathcal{I}(t+1)}\} \\
(\Diamond^- R)^{\mathcal{I}(t)} &= \{\langle d_1, \ldots, d_n\rangle \in (\top_n)^{\mathcal{I}(t)} \mid \exists v < t.\, \langle d_1, \ldots, d_n\rangle \in R^{\mathcal{I}(v)}\} \\
(\ominus R)^{\mathcal{I}(t)} &= \{\langle d_1, \ldots, d_n\rangle \in (\top_n)^{\mathcal{I}(t)} \mid \langle d_1, \ldots, d_n\rangle \in R^{\mathcal{I}(t-1)}\}
\end{aligned}
$$

Fig. 1. Syntax and semantics of $\mathcal{DLR}_{\mathcal{US}}$.

of relation R; the classical projection is $\exists^{\geq 1}[U_j]R$. It is also possible to use the pure argument position version of the language by replacing role symbols U_i with their corresponding position numbers i. The model-theoretic semantics of $\mathcal{DLR}_{\mathcal{US}}$ assumes a flow of time $\mathcal{T} = \langle \mathcal{T}_p, < \rangle$, where $\mathcal{T}_p$ is a set of time points and $<$ a binary precedence relation on $\mathcal{T}_p$, which is assumed to be isomorphic to $\langle \mathbb{Z}, < \rangle$. The language of $\mathcal{DLR}_{\mathcal{US}}$ is interpreted in *temporal models* over $\mathcal{T}$, which are triples of the form $\mathcal{I} \doteq \langle \mathcal{T}, \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}(t)} \rangle$, where $\Delta^{\mathcal{I}}$ is non-empty set of objects (the *domain* of $\mathcal{I}$) and $\cdot^{\mathcal{I}(t)}$ an *interpretation function* such that, for every $t \in \mathcal{T}$ (a shortcut for $t \in \mathcal{T}_p$), every class C, and every n-ary relation R, we have $C^{\mathcal{I}(t)} \subseteq \Delta^{\mathcal{I}}$ and $R^{\mathcal{I}(t)} \subseteq (\Delta^{\mathcal{I}})^n$. The semantics of class and relation expressions is defined in the lower part of Figure 1, where $(u, v) = \{w \in \mathcal{T} \mid u < w < v\}$. For classes, the temporal operators $\Diamond^+$ (some time in the future), $\oplus$ (at the next moment), and their past counterparts can be defined via $\mathcal{U}$ and $\mathcal{S}$: $\Diamond^+ C \equiv \top \mathcal{U} C$, $\oplus C \equiv \bot \,\mathcal{U}\, C$, etc. The operators $\Box^+$ (always in the future) and $\Box^-$ (always in the past) are the duals of $\Diamond^+$ (some time in the future) and $\Diamond^-$ (some time in the past), respectively, i.e. $\Box^+ C \equiv \neg \Diamond^+ \neg C$ and $\Box^- C \equiv \neg \Diamond^- \neg C$, for both classes and relations. The operators $\Diamond^*$ (at some moment) and its dual $\Box^*$ (at all moments) can be defined for both classes and relations as $\Diamond^* C \equiv C \sqcup \Diamond^+ C \sqcup \Diamond^- C$ and $\Box^* C \equiv C \sqcap \Box^+ C \sqcap \Box^- C$, respectively. A *knowledge base* is a finite set Σ of $\mathcal{DLR}_{\mathcal{US}}$ axioms of the form $C_1 \sqsubseteq C_2$ and $R_1 \sqsubseteq R_2$, and with R_1 and R_2 being relations of the same arity. An interpretation $\mathcal{I}$ satisfies $C_1 \sqsubseteq C_2$ ($R_1 \sqsubseteq R_2$) if and only if the interpretation of C_1 (R_1) is included in the interpretation of C_2 (R_2) at all time, i.e. $C_1^{\mathcal{I}(t)} \subseteq C_2^{\mathcal{I}(t)}$ ($R_1^{\mathcal{I}(t)} \subseteq R_2^{\mathcal{I}(t)}$), for all $t \in \mathcal{T}$.

The temporal conceptual model $\mathcal{ER}_{VT}$ The temporal EER model $\mathcal{ER}_{VT}$ is briefly introduced (see [3] for full details). $\mathcal{ER}_{VT}$ supports timestamping for classes, attributes, and relationships, and is equipped with both a textual and a graphical syntax along with a model-theoretic semantics as a temporal extension of the EER semantics [11].

Definition 1 ($\mathcal{ER}_{VT}$ Conceptual Data Model). *An $\mathcal{ER}_{VT}$ conceptual data model is a tuple: $\Sigma = (\mathcal{L}, \text{REL}, \text{ATT}, \text{CARD}, \text{ISA}, \text{DISJ}, \text{COVER}, \text{S}, \text{T}, \text{KEY})$, such that: $\mathcal{L}$ is a finite alphabet partitioned into the sets: $\mathcal{C}$ (*class *symbols), $\mathcal{A}$ (*attribute *symbols), $\mathcal{R}$ (*relationship *symbols), $\mathcal{U}$ (*role *symbols), $\mathcal{D}$ (*domain *symbols) and*

1. *The set $\mathcal{C}$ of class symbols is partitioned into a set $\mathcal{C}^S$ of* Snapshot classes *(marked with an* S*), a set $\mathcal{C}^M$ of* Mixed classes *(unmarked classes), and a set $\mathcal{C}^T$ of* Temporary classes *(marked with a* T*); likewise for $\mathcal{R}$.*
2. ATT *is a function that maps a class symbol in $\mathcal{C}$ to an $\mathcal{A}$-labeled tuple over $\mathcal{D}$,* $\text{ATT}(C) = \langle A_1 : D_1, \ldots, A_h : D_h \rangle$.
3. REL *is a function that maps a relationship symbol in $\mathcal{R}$ to an $\mathcal{U}$-labeled tuple over $\mathcal{C}$,* $\text{REL}(R) = \langle U_1 : C_1, \ldots, U_k : C_k \rangle$*, and k is the* arity *of R.*
4. CARD *is a function $\mathcal{C} \times \mathcal{R} \times \mathcal{U} \mapsto \mathbb{N} \times (\mathbb{N} \cup \{\infty\})$ denoting cardinality constraints. We denote with* $\text{CMIN}(C, R, U)$ *and* $\text{CMAX}(C, R, U)$ *the first and second component of* CARD.
5. ISA *is a binary relationship* $\text{ISA} \subseteq (\mathcal{C} \times \mathcal{C}) \cup (\mathcal{R} \times \mathcal{R})$. ISA *between relationships is restricted to relationships with the same arity.*
6. $\text{DISJ}, \text{COVER}$ *are binary relations over $2^{\mathcal{C}} \times \mathcal{C}$, describing disjointness and covering partitions over a group of* ISA *that share the same superclass.*
7. S, T *are binary relations over $\mathcal{C} \times \mathcal{A}$ containing, respectively, the snapshot and temporary attributes of a class;*
8. KEY *is a function,* $\text{KEY} : \mathcal{C} \to \mathcal{A}$*, that maps a class symbol in $\mathcal{C}$ to its key attribute.*

The model-theoretic semantics associated with the $\mathcal{ER}_{VT}$ modelling language adopts the snapshot representation of temporal conceptual data models [12]—following the snapshot paradigm, $\mathcal{T}_p$ is a set of time points and $<$ is a binary precedence relation on $\mathcal{T}_p$, the flow of time $\mathcal{T} = \langle \mathcal{T}_p, < \rangle$ is assumed to be isomorphic to either $\langle \mathbb{Z}, < \rangle$ or $\langle \mathbb{N}, < \rangle$)

Definition 2 ($\mathcal{ER}_{VT}$ Semantics). *Let Σ be an $\mathcal{ER}_{VT}$ schema. A* temporal database state *for the schema Σ is a tuple $\mathcal{B} = (\mathcal{T}, \Delta^{\mathcal{B}} \cup \Delta^{\mathcal{B}}_D, \cdot^{\mathcal{B}(t)})$, such that: $\Delta^{\mathcal{B}}$ is a nonempty set of abstract objects disjoint from $\Delta^{\mathcal{B}}_D$; $\Delta^{\mathcal{B}}_D = \bigcup_{D_i \in \mathcal{D}} \Delta^{\mathcal{B}}_{D_i}$ is the set of basic domain values used in the schema Σ; and $\cdot^{\mathcal{B}(t)}$ is a function that for each $t \in \mathcal{T}$ maps:*

- *Every basic domain symbol* D_i *into a set* $D_i^{\mathcal{B}(t)} = \Delta^{\mathcal{B}}_{D_i}$.
- *Every class* C *to a set* $C^{\mathcal{B}(t)} \subseteq \Delta^{\mathcal{B}}$*—thus* objects *are instances of classes.*
- *Every relationship* R *to a set* $R^{\mathcal{B}(t)}$ *of* $\mathcal{U}$*-labeled tuples over* $\Delta^{\mathcal{B}}$*—i.e. let* R *be an n-ary relationship connecting the classes* $C_1, \ldots, C_n$, $\textsc{rel}(R) = \langle U_1 : C_1, \ldots, U_n : C_n\rangle$, *then,* $r \in R^{\mathcal{B}(t)} \rightarrow (r = \langle U_1 : o_1, \ldots, U_n : o_n\rangle \wedge \forall i \in \{1, \ldots, n\}.o_i \in C_i^{\mathcal{B}(t)})$. *We adopt the convention:* $\langle U_1 : o_1, \ldots, U_n : o_n\rangle \equiv \langle o_1, \ldots, o_n\rangle$, *when* $\mathcal{U}$*-labels are clear from the context.*
- *Every attribute* A *to a set* $A^{\mathcal{B}(t)} \subseteq \Delta^{\mathcal{B}} \times \Delta^{\mathcal{B}}_D$, *such that, for each* $C \in \mathcal{C}$, *if* $\textsc{att}(C) = \langle A_1 : D_1, \ldots, A_h : D_h\rangle$, *then,* $o \in C^{\mathcal{B}(t)} \rightarrow (\forall i \in \{1, \ldots, h\}, \exists a_i. \langle o, a_i\rangle \in A_i^{\mathcal{B}(t)} \wedge \forall a_i.\langle o, a_i\rangle \in A_i^{\mathcal{B}(t)} \rightarrow a_i \in \Delta^{\mathcal{B}}_{D_i})$.

$\mathcal{B}$ *is said a* legal temporal database state *if it satisfies all of the constraints expressed in the schema, i.e. for each* $t \in \mathcal{T}$*:*

- *For each* $C_1, C_2 \in \mathcal{C}$, *if* $C_1 \ \textsc{isa}\ C_2$, *then,* $C_1^{\mathcal{B}(t)} \subseteq C_2^{\mathcal{B}(t)}$.
- *For each* $R_1, R_2 \in \mathcal{R}$, *if* $R_1 \ \textsc{isa}\ R_2$, *then,* $R_1^{\mathcal{B}(t)} \subseteq R_2^{\mathcal{B}(t)}$.
- *For each cardinality constraint* $\textsc{card}(C, R, U)$, *then:* $o \in C^{\mathcal{B}(t)} \rightarrow \textsc{cmin}(C, R, U) \leq \#\{r \in R^{\mathcal{B}(t)} \mid r[U] = o\} \leq \textsc{cmax}(C, R, U)$.
- *For* $C, C_1, \ldots, C_n \in \mathcal{C}$, *if* $\{C_1, \ldots, C_n\} \ \textsc{disj}\ C$, *then,* $\forall i \in \{1, \ldots, n\}.C_i \ \textsc{isa}\ C \wedge \forall j \in \{1, \ldots, n\},\ j \neq i.C_i^{\mathcal{B}(t)} \cap C_j^{\mathcal{B}(t)} = \emptyset$.
- *For* $C, C_1, \ldots, C_n \in \mathcal{C}$, *if* $\{C_1, \ldots, C_n\} \ \textsc{cover}\ C$, *then,* $\forall i \in \{1, \ldots, n\}.C_i \ \textsc{isa}\ C \wedge C^{\mathcal{B}(t)} = \bigcup_{i=1}^{n} C_i^{\mathcal{B}(t)}$.
- *For each snapshot class* $C \in \mathcal{C}^S$, *then,* $o \in C^{\mathcal{B}(t)} \rightarrow \forall t' \in \mathcal{T}.o \in C^{\mathcal{B}(t')}$.
- *For each temporary class* $C \in \mathcal{C}^T$, *then,* $o \in C^{\mathcal{B}(t)} \rightarrow \exists t' \neq t.o \notin C^{\mathcal{B}(t')}$.
- *For each snapshot relationship* $R \in \mathcal{R}^S$, *then,* $r \in R^{\mathcal{B}(t)} \rightarrow \forall t' \in \mathcal{T}.r \in R^{\mathcal{B}(t')}$.
- *For each temporary relationship* $R \in \mathcal{R}^T$, *then,* $r \in R^{\mathcal{B}(t)} \rightarrow \exists t' \neq t.r \notin R^{\mathcal{B}(t')}$.
- *For each class* $C \in \mathcal{C}$, *if* $\textsc{att}(C) = \langle A_1 : D_1, \ldots, A_h : D_h\rangle$, *and* $\langle C, A_i\rangle \in \textsc{s}$, *then,* $(o \in C^{\mathcal{B}(t)} \wedge \langle o, a_i\rangle \in A_i^{\mathcal{B}(t)}) \rightarrow \forall t' \in \mathcal{T}.\langle o, a_i\rangle \in A_i^{\mathcal{B}(t')}$.
- *For each class* $C \in \mathcal{C}$, *if* $\textsc{att}(C) = \langle A_1 : D_1, \ldots, A_h : D_h\rangle$, *and* $\langle C, A_i\rangle \in \textsc{t}$, *then,* $(o \in C^{\mathcal{B}(t)} \wedge \langle o, a_i\rangle \in A_i^{\mathcal{B}(t)}) \rightarrow \exists t' \neq t.\langle o, a_i\rangle \notin A_i^{\mathcal{B}(t')}$.
- *For each* $C \in \mathcal{C}, A \in \mathcal{A}$ *such that* $\textsc{key}(C) = A$, *then,* A *is a snapshot attribute–i.e.* $\langle C, A_i\rangle \in \textsc{s}$*— and* $\forall a \in \Delta^{\mathcal{B}}_D.\#\{o \in C^{\mathcal{B}(t)} \mid \langle o, a\rangle \in A^{\mathcal{B}(t)}\} \leq 1$.

Given such a set-theoretic semantics for $\mathcal{ER}_{VT}$, some relevant modelling notions such as satisfiability, subsumption, and derivation of new constraints by means of logical implication have been defined rigorously [4]. Finally, we briefly summarize how $\mathcal{DLR}_{\mathcal{US}}$ is able to capture temporal schemas expressed in $\mathcal{ER}_{VT}$—see [3] for more details.

Definition 3 (Mapping $\mathcal{ER}_{VT}$ into $\mathcal{DLR}_{\mathcal{US}}$). *Let* $\Sigma = (\mathcal{L}, \textsc{rel}, \textsc{att}, \textsc{card}, \textsc{isa}, \textsc{disj}, \textsc{cover}, \textsc{s}, \textsc{t}, \textsc{key})$ *be an* $\mathcal{ER}_{VT}$ *schema. The* $\mathcal{DLR}_{\mathcal{US}}$ *knowledge base,* $\mathcal{K}$, *mapping* Σ *is as follows.*

- *For each* $A \in \mathcal{A}$, *then,* $A \sqsubseteq \texttt{From}{:}\top \sqcap \texttt{To}{:}\top \in \mathcal{K}$*;*
- *If* $C_1 \ \textsc{isa}\ C_2 \in \Sigma$, *then,* $C_1 \sqsubseteq C_2 \in \mathcal{K}$*; If* $R_1 \ \textsc{isa}\ R_2 \in \Sigma$, *then,* $R_1 \sqsubseteq R_2 \in \mathcal{K}$*;*
- *If* $\textsc{rel}(R) = \langle U_1 {:} C_1, \ldots, U_k {:} C_k\rangle \in \Sigma$, *then* $R \sqsubseteq U_1 {:} C_1 \sqcap \ldots \sqcap U_k {:} C_k \in \mathcal{K}$*;*
- *If* $\textsc{att}(C) = \langle A_1 : D_1, \ldots, A_h : D_h\rangle \in \Sigma$, *then,* $C \sqsubseteq \exists[\texttt{From}]A_1 \sqcap \ldots \sqcap \exists[\texttt{From}]A_h \sqcap \forall[\texttt{From}](A_1 \rightarrow \texttt{To} : D_1) \sqcap \ldots \sqcap \forall[\texttt{From}](A_h \rightarrow \texttt{To} : D_h) \in \mathcal{K}$*;*

– *If* $\text{CARD}(C, R, U) = (m, n) \in \Sigma$*, then,* $C \sqsubseteq \exists^{\geq m}[U]R \sqcap \exists^{\leq n}[U]R \in \mathcal{K}$*;*
– *If* $\{C_1, \ldots, C_n\}$ DISJ $C \in \Sigma$*, then* $\mathcal{K}$ *contains:* $C_1 \sqsubseteq C \sqcap \neg C_2 \sqcap \ldots \sqcap \neg C_n$*;* $C_2 \sqsubseteq C \sqcap \neg C_3 \sqcap \ldots \sqcap \neg C_n$*; ...* $C_n \sqsubseteq C$*; If* $\{C_1, \ldots, C_n\}$ COVER $E \in \Sigma$*, then* $\mathcal{K}$ *contains:* $C_1 \sqsubseteq C$*; ...* $C_n \sqsubseteq C$*;* $C \sqsubseteq C_1 \sqcup \ldots \sqcup C_n$*;*
– *If* $\text{KEY}(C) = A$*, then,* $\mathcal{K}$ *contains:* $C \sqsubseteq \exists^{=1}[\mathtt{From}]\Box^* A$*;* $\top \sqsubseteq \exists^{\leq 1}[\mathtt{To}](A \sqcap [\mathtt{From}] : C)$*;*
– *If* $C \in \mathcal{C}^S$*, then,* $C \sqsubseteq (\Box^* C) \in \mathcal{K}$ *(similar for* $R \in \mathcal{R}^S$*);*
– *If* $C \in \mathcal{C}^T$*, then,* $C \sqsubseteq (\Diamond^* \neg C) \in \mathcal{K}$ *(similar for* $R \in \mathcal{R}^T$*);*
– *If* $\langle C, A \rangle \in \text{S}$*, then,* $C \sqsubseteq \forall[\mathtt{From}](A \rightarrow \Box^* A) \in \mathcal{K}$*;*
– *If* $\langle C, A \rangle \in \text{T}$*, then,* $C \sqsubseteq \forall[\mathtt{From}](A \rightarrow \Diamond^* \neg A) \in \mathcal{K}$*.*

In the next sections we extend the formalism presented here to capture essential and sharable parts.

3 Modeling Mandatory and Essential Parts

This section presents a formalization of the notion of *mandatory* and *essential* part-whole relations. To formalize such properties of part-whole relations we will resort to the formalism introduced in the previous section and extend $\mathcal{ER}_{VT}$ with the possibility to capture such part-whole properties, while the description logic $\mathcal{DLR}_{\mathcal{US}}$ will present a corresponding axiomatization for them. A basic building block to achieve the desired formalization is the notion of *status relations*. The formalization of status relations is an original contribution of this paper. They are in analogy with status classes addressed by [4] and will be useful for modeling essential part-whole relations. We therefore start by introducing status relations and then we proceed to formalizing mandatory and essential parts and wholes.

Status Relations. *Status relations* extend the notion of *status classes* [22, 13, 4] to statuses for relations. Status classes—formalized in [4]—constrain the evolution of an instance's membership in a class along its lifespan. According to [22, 4], status modeling includes up to four different statuses *scheduled, active, suspended, disabled*, each one entailing different constraints. Likewise, we have four different statuses for relations, too: *scheduled, active, suspended, disabled*. Each one is illustrated with an example before we proceed to the formalization.

– Scheduled: a relation is scheduled if its instantiation is known but its membership will only become effective some time later. Objects in its participating classes must be either scheduled, active, or suspended; e.g., a new pillar for the Sagrada Familia's interior is scheduled to become part of that church.
– Active: the status of a relation is active if the particular relation fully instantiates the type-level relation and only active classes can participate into an active relation; e.g., the Mont Blanc mountain is part of the Alps mountain range, and the country Republic of Ireland is part of the European Union.
– Suspended: to capture a temporarily inactive relation; e.g., an instance of a CarEngine is removed from the instance of a Car it is part of for purpose of maintenance. At the moment of suspension, the participating objects must be active, but can upon suspension of the relation be either active or suspended.

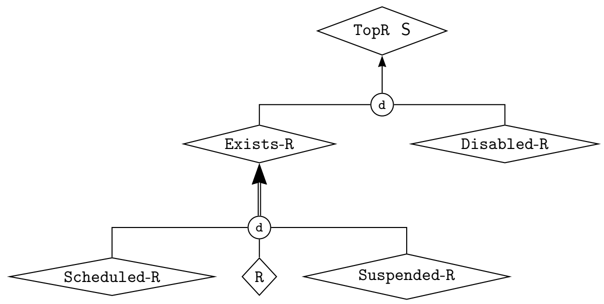

Fig. 2. Status relations (from status classes in [4]).

– Disabled: to model expired relations that never again can be used; e.g., to represent the donor of an organ who has donated that organ and one wants to keep track of who donated what to whom. Participating objects can be member of the active, suspended or disabled class.

Status relations apply only to temporal relations (i.e. either temporary or mixed relations, see Definition 1). We assume that active relations involve only active classes and the name of a relation denotes already its active status—i.e. $\texttt{Active-R} \equiv \texttt{R}$. Disjointness and ISA constraints among the four status relations are analogous to those for status classes and can be represented in $\mathcal{ER}_{VT}$ as illustrated in Fig. 2. In addition to hierarchical constraints, the following constraints hold (where $R \sqsubseteq U_1 : C_1 \sqcap \ldots \sqcap U_n : C_n$):

(REXISTS) *Existence persists until Disabled.*
$\texttt{Exists-R} \sqsubseteq \Box^+(\texttt{Exists-R} \sqcup \texttt{Disabled-R})$
(RDISAB1) *Disabled persists.* $\texttt{Disabled-R} \sqsubseteq \Box^+\texttt{Disabled-R}$
(RDISAB2) *Disabled was Active in the past.* $\texttt{Disabled-R} \sqsubseteq \Diamond^-\texttt{R}$
(RSUSP1) *Suspended was Active in the past.* $\texttt{Suspended-R} \sqsubseteq \Diamond^-\texttt{R}$
(RSUSP2) *Suspended involve Active or Suspended Classes.*
$\texttt{Suspended-R} \sqsubseteq \texttt{U}_\texttt{i} : (\texttt{C}_\texttt{i} \sqcup \texttt{Suspended-C}_\texttt{i}),\ i = 1, \ldots, 2$
(RSCH1) *Scheduled will eventually become Active.* $\texttt{Scheduled-R} \sqsubseteq \Diamond^+\texttt{R}$
(RSCH2) *Scheduled can never follow Active.* $\texttt{R} \sqsubseteq \Box^+\neg\texttt{Scheduled-R}$

In the following, with Σ_{st} we denote the above set of $\mathcal{DLR}_{\mathcal{US}}$ axioms that formalize status relations. In analogy with the logical implications holding for status classes [4], we can derive the following ones for status relations.

Proposition 1 (Status Relations: Logical Implications). *Given the set of axioms* Σ_{st} (REXISTS-RSCH2)*, an n-ary relation (where* $n \geq 2$*)* $R \sqsubseteq U_1 : C_1 \sqcap \ldots \sqcap U_n : C_n$*, the following logical implications hold:*

(RACT) *Active will possible evolve into Suspended or Disabled.*
$\Sigma_{st} \models \texttt{R} \sqsubseteq \Box^+(\texttt{R} \sqcup \texttt{Suspended-R} \sqcup \texttt{Disabled-R})$
(RDISAB3) *Disabled will never become active anymore.*
$\Sigma_{st} \models \texttt{Disabled-R} \sqsubseteq \Box^+\neg\texttt{R}$

(RDISAB4) *Disabled classes can participate only in disabled relations.*
$\Sigma_{st} \models \texttt{Disabled-C}_\texttt{i} \sqcap \Diamond^- \exists[\texttt{U}_\texttt{i}]\texttt{R} \sqsubseteq \exists[\texttt{U}_\texttt{i}]\texttt{Disabled-R}$

(RDISAB5) *Disabled relations involve active, suspended, or disabled classes.*
$\texttt{Disabled-R} \sqsubseteq \texttt{U}_i : (\texttt{C}_i \sqcup \texttt{Suspended-C}_i \sqcup \texttt{Disabled-C}_i)$, *for all* $i = 1, \ldots, n$.

(RSCH3) *Scheduled persists until active.*
$\Sigma_{st} \models \texttt{Scheduled-R} \sqsubseteq \texttt{Scheduled-R}\,\mathcal{U}\,\texttt{R}$

(RSCH4) *Scheduled cannot evolve directly to Disabled.*
$\Sigma_{st} \models \texttt{Scheduled-R} \sqsubseteq \oplus \neg\texttt{Disabled-R}$

(RSCH5) *Scheduled relations do not involve disabled classes.*
$\texttt{Scheduled-R} \sqsubseteq \texttt{U}_\texttt{i} : \neg\texttt{Disabled-C}_\texttt{i}$, *for all* $i = 1, \ldots, n$.

Proof We prove here (RDISAB4) as the others are similar to what already proved in [4]. Let $o_i \in \texttt{Disabled-C}_\texttt{i}^{\mathcal{B}(\texttt{t})}$ and $r = \langle o_1, \ldots, o_i, \ldots, o_n \rangle \in R^{\mathcal{B}(t')}$ for some $t' < t$. Then, by (RACT), $r \in (\texttt{R} \sqcup \texttt{Suspended-R} \sqcup \texttt{Disabled-R})^{\mathcal{B}(t)}$. Since active relations, by assumption, can involve just active classes, then $r \notin R^{\mathcal{B}(t)}$. However, by (RSUSP2), $r \notin \texttt{Suspended-R}^{\mathcal{B}(t)}$. Thus, $r \in \texttt{Disabled-R}^{\mathcal{B}(t)}$. ❑

Lifespan and related notions. The lifespan of an object with respect to a class describes the temporal instants (and thus intervals) where the object can be considered a member of that class. We can distinguish between the following notions: EXISTENCESPAN_C, LIFESPAN_C, ACTIVESPAN_C, BEGIN_C, BIRTH_C, and DEATH_C depending on the status of the class the object is member of. We briefly report here their definition as presented in [4].

$\text{EXISTENCESPAN}_C(o) = \{t \in \mathcal{T} \mid o \in \texttt{Exists-C}^{\mathcal{B}(t)}\}$
$\text{LIFESPAN}_C(o) = \{t \in \mathcal{T} \mid o \in \texttt{C}^{\mathcal{B}(t)} \cup \texttt{Suspended-C}^{\mathcal{B}(t)}\}$
$\text{ACTIVESPAN}_C(o) = \{t \in \mathcal{T} \mid o \in \texttt{C}^{\mathcal{B}(t)}\}$
$\text{BEGIN}_C(o) = \texttt{min}(\text{EXISTENCESPAN}_C(o))$
$\text{BIRTH}_C(o) = \texttt{min}(\text{ACTIVESPAN}_C(o)) \equiv \texttt{min}(\text{LIFESPAN}_C(o))$
$\text{DEATH}_C(o) = \texttt{max}(\text{LIFESPAN}_C(o))$

For atemporal classes,
$\text{EXISTENCESPAN}_C(o) \equiv \text{LIFESPAN}_C(o) \equiv \text{ACTIVESPAN}_C(o) \equiv \mathcal{T}$.
This concludes the preliminaries. In the next section we will use the notions introduced so far for representing essential parts-whole relations.

Mandatory and essential parts and wholes. Recollecting Guizzardi's contribution on the formalization of the difference between mandatory and essential parts and wholes we can say that: a part is *mandatory* if the whole cannot exist without it (in a symmetric way we can define *mandatory wholes*); a part is *essential* if it is mandatory and cannot change without destroying the whole (analogously, *essential whole*). Furthermore, we say that a part is *exclusive* if it can be part of at most one whole (similarly for *exclusive wholes*). In this section we provide a formalization using $\mathcal{DLR}_{\mathcal{US}}$ axioms of such mandatory, essential and exclusive parts and wholes. Fig. 3-A shows the various temporal relations that can hold between a whole its essential part (vv. in Fig. 3-B). Let `partOf`

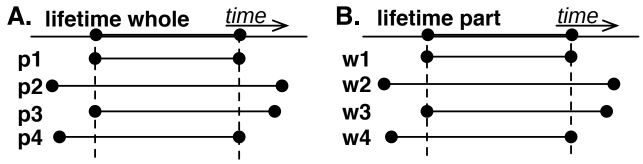

Fig. 3. Lifelines of essential or mandatory parts w.r.t. the whole (A) and vv (B).

$\sqsubseteq$ `part:P` $\sqcap$ `whole:W` be a generic part-whole relation, the following $\mathcal{DLR}_{\mathcal{US}}$ axioms give a formalization of *mandatory* and *exclusive* parts and wholes:

(ManP)	$\texttt{W} \sqsubseteq \exists[\texttt{whole}]\texttt{partOf}$	*Mandatory Part*
(ManW)	$\texttt{P} \sqsubseteq \exists[\texttt{part}]\texttt{partOf}$	*Mandatory Whole*
(ExlP)	$\texttt{P} \sqsubseteq \exists^{\leq 1}[\texttt{part}]\texttt{partOf}$	*Exclusive Part*
(ExlW)	$\texttt{W} \sqsubseteq \exists^{\leq 1}[\texttt{whole}]\texttt{partOf}$	*Exclusive Whole*

To capture essential parts and wholes, in addition to the above axioms, we will use appropriate subsets of the following axioms.

(ConPO)	$\texttt{Suspended-partOf} \sqsubseteq \bot$	*Continuous Parts*
(DisP)	$\texttt{Disabled-partOf} \sqsubseteq \texttt{part} : \texttt{Disabled-P}$	*Disabled Part*
(DisW)	$\texttt{Disabled-partOf} \sqsubseteq \texttt{whole} : \texttt{Disabled-W}$	*Disabled Whole*
(SchPO)	$\texttt{partOf} \sqsubseteq \Diamond^{-}\texttt{Scheduled-partOf}$	*Scheduled Part-Whole*
(SchP)	$\texttt{Scheduled-partOf} \sqsubseteq \texttt{part} : \texttt{Scheduled-P}$	*Scheduled Part*
(SchW)	$\texttt{Scheduled-partOf} \sqsubseteq \texttt{whole} : \texttt{Scheduled-W}$	*Scheduled Whole*

We can now show that the above axiomatization is sufficient to represent the various forms of essential parts as shown in Fig. 3(p1-p4).

Theorem 1 (Essential Parts). *Let* `partOf` $\sqsubseteq$ `part:P` $\sqcap$ `whole:W` *be a generic part-whole relation satisfying* Σ_{st}*, then,*
1. *p2 holds if* (ManP), (ConPO), (DisW) *hold;*
2. *p4 holds if* (ManP), (ConPO), (DisW), (DisP) *hold;*
3. *p3 holds if* (ManP), (ConPO), (DisW), (SchPO), (SchP) *hold;*
4. *p1 holds if* (ManP), (ConPO), (DisW), (DisP), (SchPO), (SchP) *hold.*

Proof Let $o_w \in \texttt{W}^{\mathcal{B}(t_0)}$ with $t_0 = \textsc{Birth}_W(o_w)$, then by (ManP), $\exists o_p \in \texttt{P}^{\mathcal{B}(t_0)}$ and $\langle o_p, o_w\rangle \in \texttt{partOf}^{\mathcal{B}(t_0)}$.

Case p2. To prove that p2 of Fig. 3 holds we prove that $\textsc{ActiveSpan}_W(o_w) \subseteq \textsc{ActiveSpan}_P(o_p)$, i.e. $\textsc{Birth}_W(o_w) \geq \textsc{Birth}_P(o_p)$ and $\textsc{Death}_W(o_w) \leq \textsc{Death}_P(o_p)$. Now, let $t_0 < t' < \textsc{Death}_W(o_w)$, then by (RAct) and (ConPO) either $\langle o_p, o_w\rangle \in \texttt{partOf}^{\mathcal{B}(t')}$ or $\langle o_p, o_w\rangle \in \texttt{Disabled-partOf}^{\mathcal{B}(t')}$. The last case cannot happen since, by (DisW), $o_w \in \texttt{Disabled-W}^{\mathcal{B}(t')}$ but by assumption $o_w \in \texttt{W}^{\mathcal{B}(t')}$. Thus, since by assumption active relations involve only active classes, then $o_p \in \texttt{P}^{\mathcal{B}(t')}$ for all t' s.t. $t_0 < t' < \textsc{Death}_W(o_w)$. For $t_1 \geq \textsc{Death}_W(o_w)$ and $t_2 \leq \textsc{Birth}_W(o_w)$ none of the axioms constraints the lifespan of o_p.
Thus, $\textsc{Death}_W(o_w) \leq \textsc{Death}_P(o_p)$, and $\textsc{Birth}_W(o_w) \geq \textsc{Birth}_P(o_p)$.

Case p1. To prove that the case p1 of Fig. 3 holds we should prove that $\text{ActiveSpan}_W(o_w) = \text{ActiveSpan}_P(o_p)$, i.e. $\text{Birth}_W(o_w) = \text{Birth}_P(o_p)$ and $\text{Death}_W(o_w) = \text{Death}_P(o_p)$. As for case p2, since (ConPO) and (DisW) hold, then, $o_p \in \mathtt{P}^{\mathcal{B}(\mathtt{t}')}$ for all t' s.t. $t_0 < t' < \text{Death}_W(o_w)$. Now, let $t_1 = \text{Death}_W(o_w)$, then, by (RDisab4) and (RAct), $\langle o_p, o_w\rangle \in \mathtt{Disabled\text{-}partOf}^{\mathcal{B}(t_1)}$, and, by (DisP), $o_p \in \mathtt{Disabled\text{-}P}^{\mathcal{B}(t_1)}$. Thus, $\text{Death}_W(o_w) = \text{Death}_P(o_p)$. Now, by absurd, let's assume that $o_p \in P^{\mathcal{B}(t_2)}$, with $t_2 < t_0$. By (SchPO), there is a $t' < t_0$ s.t. $\langle o_p, o_w\rangle \in \mathtt{Scheduled\text{-}partOf}^{\mathcal{B}(t')}$ and, by (SchP), $o_p \in \mathtt{Scheduled\text{-}P}^{\mathcal{B}(t')}$. Then, $t_2 \neq t'$. Also, $t_2 \not< t'$ since, by (Sch2), an active class cannot evolve into its scheduled status. Finally, $t_0 > t_2 \not> t'$ since, by (RSch3) $\langle o_p, o_w\rangle \in \mathtt{Scheduled\text{-}partOf}^{\mathcal{B}(t_2)}$ and, by (SchP), $o_p \in \mathtt{Scheduled\text{-}P}^{\mathcal{B}(t_2)}$. Thus, $\text{Birth}_W(o_w) = \text{Birth}_P(o_p)$.

Cases p4, p3 can be easily obtained from the above. ❑

A similar result can be proved for essential wholes.

Theorem 2 (Essential Wholes). *Let* `partOf` $\sqsubseteq$ `part:P` $\sqcap$ `whole:W` *be a generic part-whole relation satisfying* Σ_{st}*, then,*

1. *w2 holds if* (ManW), (ConPO), (DisP) *hold;*
2. *w4 holds if* (ManW), (ConPO), (DisP), (DisW) *hold;*
3. *w3 holds if* (ManW), (ConPO), (DisP), (SchPO), (SchW) *hold;*
4. *w1 holds if* (ManW), (ConPO), (DisP), (DisW), (SchPO), (SchW) *hold.*

Thus, from the axiomatization presented above, the essential parts and wholes in a part-whole relation are always active and cannot be suspended and when the strict case is allowed (i.e. either p1 or w1 holds) then they are either *both* member of their respective Scheduled class, or both Active, or both member of their respective Disabled classes. Hence, a change of membership from one of the two objects implies *instantaneous* change ("$\bigoplus$") of the other in the same type of status class. In the literature, essential parts are often considered also exclusive [14, 18]. Our modeling of essential parts and wholes can be easily extended by adding to the axiomatization of Theorems 1-2 either (ExlP) or (ExlW), depending whether we want to capture exclusive essential parts or wholes.

4 Conclusions

We proposed a solution to the modelling problem of representing mandatory and essential parts and wholes by characterizing the semantics of these shareability notions by resorting to the $\mathcal{ER}_{VT}$ temporal conceptual data modelling language augmented with *status relations* and its formalization into the temporal DL $\mathcal{DLR}_{\mathcal{US}}$.

Several issues have still to be addressed. We are currently elaborating on the presented approach to also cater for concurrent and sequential sharing of parts and their interaction with various types of part-whole relations. From the reasoning point of view, while reasoning on $\mathcal{DLR}_{\mathcal{US}}$ is known to be undecidable, we are studying whether by weakening the EER expressiveness (e.g. forbidding covering and at-most cardinalities and allowing only binary relations) we can use a simpler and thus decidable temporal DL as presented in [1].

References

1. Artale, A., Kontchakov, R., Wolter, C.L.F., Zakharyaschev, M. Temporalising tractable description logics. In *14th Int. Symposium on Temporal Representation and Reasoning (TIME07)*. IEEE Computer Society, 2007.
2. Artale, A., Franconi, E., Guarino, N. & Pazzi, L. Part-Whole Relations in Object-Centered Systems: an Overview. *DKE*, 1996, 20(3):347-383.
3. 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, Springer-Verlag. 2003.
4. Artale, A., Parent, C. & Spaccapietra, S. Evolving objects in temporal information systems. *AMAI*, 2007, 50(1-2), 5-38.
5. Barbier, F., Henderson-Sellers, B., Le Parc-Lacayrelle, A. & Bruel, J.-M. Formalization of the whole-part relationship in the Unified Modelling Language. *IEEE Trans. on Softw. Eng.*, 2003, 29(5):459-470.
6. Berardi, D., Calvanese, D. & De Giacomo, G. Reasoning on UML class diagrams. *AI*, 2005, 168(1-2):70-118.
7. Bittner, T. & Donnelly, M. Computational ontologies of parthood, componenthood, and containment, In: *Proc. of IJCAI05*. Kaelbling, L. (ed.). pp382-387.
8. Bittner, T. & Donnelly, M. A temporal mereology for distinguishing between integral objects and portions of stuff. In: *Proc. of AAAI'07*, 287-292.
9. Calvanese, D. & De Giacomo, G. Expressive description logics. In: *The DL Handbook*, Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (Eds). Cambridge University Press, 2003. pages 178-218.
10. Calvanese, C., De Giacomo, G. & Lenzerini, M. On the decidability of query containment under constraints. In: *Proc. of PODS'98*, 149-158.
11. Calvanese, D., Lenzerini, M. & Nardi, D. Unifying class-based representation formalisms. *JAIR*, 11:199-240, 1999.
12. Chomicki, J. & Toman, D. Temporal logic in information systems. In: J. Chomiki, G. Saake (Eds.). *Logics for databases and information systems*, ch. 1. Kluwer, 1998.
13. Etzion, O., Gal, A., & Segev, A. Extended update functionality in temporal databases. In O. Etzion, S. Jajodia, S. Sripada, (Eds.), Temporal Databases - Research and Practice, LNCS, pp56-95. Springer-Verlag, 1998.
14. Guizzardi, G. *Ontological foundations for structural conceptual models*. PhD Thesis, Telematica Institute, Twente University, Enschede, the Netherlands. 2005.
15. Hodgkinson, I.M., Wolter, F. & Zakharyaschev, M. Decidable fragments of first-order temporal logics. *Annals of pure and applied logic*, 106, 85-134, 2000.
16. Keet, C.M. Prospects for and issues with mapping the Object-Role Modeling language into $\mathcal{DLR}_{ifd}$. *Proc. of DL'07*, CEUR-WS, Vol-250, 331-338.
17. Keet, C.M. & Artale, A. Representing and Reasoning over a Taxonomy of Part-Whole Relations. *Applied Ontology*, to appear.
18. Motschnig-Pitrik, R. & Kaasbøll, J. Part-Whole Relationship Categories and Their Application in Object-Oriented Analysis. *IEEE Trans. KDE*, 1999, 11(5):779-797.
19. Object Management Group. *Unified Modeling Language: Superstructure*. v2.0. formal/05-07-04. http://www.omg.org/cgi-bin/doc?formal/05-07-04.
20. Sattler, U. A concept language for an engineering application with part-whole relations. In: *Proc. of DL'95*, 119-123.
21. Smith, B., Ceusters, W., Klagges, B., Köhler, J., *et al.*. Relations in biomedical ontologies. *Genome Biology*, 2005, 6:R46.
22. Spaccapietra, S., Parent, C. & Zimanyi, E. Modeling time from a conceptual perspective. In: *Proc. of CIKM98*, 1998.