
Downloaded from: http://eprints.bbk.ac.uk/10304/

Usage Guidelines

Please refer to usage guidelines at http://eprints.bbk.ac.uk/policies.html or alternatively contact lib-eprints@bbk.ac.uk.
Temporalising \textit{OWL 2 QL}*

A. Artale\textsuperscript{1}, R. Kontchakov\textsuperscript{2}, F. Wolter\textsuperscript{3} and M. Zakharyaschev\textsuperscript{2}

\textsuperscript{1}Faculty of Computer Science, Free University of Bozen-Bolzano, Italy
\textsuperscript{2}Department of Computer Science and Information Systems
Birkbeck, University of London, U.K.
\textsuperscript{3}Department of Computer Science, University of Liverpool, U.K.

Abstract. We design a temporal description logic, \textit{TQL}, that extends
the standard ontology language \textit{OWL 2 QL}, provides basic means for
temporal conceptual modelling and ensures first-order rewritability of
conjunctive queries for suitably defined data instances with validity time.

1 Introduction

In this paper, we investigate the possibility of extending the current W3C stan-
dard language \textit{OWL 2 QL} for ontology-based data access (OBDA) with tempo-
ral operators in a way preserving first-order rewritability of conjunctive queries.
Our ultimate aim is to understand the feasibility of OBDA for temporal data.
In applications, instance data is often time-dependent: employment contracts
come to an end, parliaments are elected, children are born. Temporal data can
be modelled by pairs consisting of facts and their validity time; for example,
\textit{givesBirth}(diana, william, 1982). To query data with validity time, it would be
useful to employ an ontology that provides a conceptual model for both static
and temporal aspects of the domain of interest. Thus, when querying the fact
above, one could use the knowledge that, if \( x \) gives birth to \( y \), then \( x \) becomes
a mother of \( y \) from that moment on:

\[ \Diamond_p \textit{givesBirth} \sqsubseteq \textit{motherOf}, \]

where \( \Diamond_p \) reads ‘sometime in the past.’ \textit{OWL 2 QL} does not support temporal
conceptual modelling and, rather surprisingly, no attempt has yet been made to
lift OBDA based on query rewriting to temporal ontologies and data.

Temporal extensions of DLs have been investigated since 1993; see [9, 2, 17]
for surveys and [8, 6, 14, 5] for more recent developments. Moreover, temporalised
\textit{DL-Lite} logics (the logical underpinning of \textit{OWL 2 QL}) have been constructed for
temporal conceptual data modelling [3]. But unfortunately, none of the existing
temporal DLs supports first-order rewritability.

The aim of this paper is to design a temporal DL that contains \textit{OWL 2 QL},
provides basic means for temporal conceptual modelling and, at the same time,

* This paper is an abridged version of the paper accepted for IJCAI 2013; omitted
proofs can be found in [4].
ensures first-order rewritability of conjunctive queries (for suitably defined data instances with validity time). The temporal extension TQL of OWL 2 QL we present here is interpreted over sequences T(n), n ∈ Z, of standard DL structures reflecting possible evolutions of data. TBox axioms are interpreted globally, that is, are assumed to hold in all of the T(n), but the concepts and roles they contain can vary in time. ABox assertions (temporal data) are time-stamped unary (for concepts) and binary (for roles) predicates that hold at the specified moments of time. Concept (role) inclusions of TQL generalise OWL 2 QL inclusions by allowing intersections of basic concepts (roles) in the left-hand side, possibly prefixed with temporal operators ⊓ (sometime in the past) or ⊓ (sometime in the future). Among other things, one can express in TQL that a concept/role name is rigid (or time-independent), persistent in the past/future or instantaneous. For example, ⊓ Person ⊑ Person states that the concept Person is rigid, ⊓ Person ⊑ hasName says that the role hasName is persistent in the future, while ⊓ givesBirth ⊓ ⊓ givesBirth ⊑ ⊥ implies that givesBirth is instantaneous. Inclusions such as ⊓ Lecturer ⊓ ⊓ Lecturer ⊑ Lecturer represent convexity (or existential rigidity) of concepts or roles. However, in contrast to most existing temporal DLs, we cannot use temporal operators in the right-hand side of inclusions (e.g., to say that every student will eventually graduate: Student ⊑ ⊓ Student.Graduate).

In conjunctive queries (CQs) over TQL knowledge bases, we allow time-stamped predicates together with atoms of the form (τ < τ′) or (τ = τ′), where τ, τ′ are temporal constants denoting integers or variables ranging over integers.

Our main result is that, given a TQL TBox T and a CQ q, one can construct a union q’ of CQs such that the answers to q over T and any temporal ABox A can be computed by evaluating q’ over A extended with the temporal precedence relation < between the moments of time in A. For example, the query motherOf(x, y, t) over (1) can be rewritten as

\[ \text{motherOf}(x, y, t) \lor \exists t' \left( (t' < t) \land \text{givesBirth}(x, y, t') \right) \].

Note that the addition of the transitive relation < to the ABox is unavoidable: without it, there is no first-order rewriting even for the simple example above [16].

From a technical viewpoint, one of the challenges we are facing is that, in contrast to known OBDA languages with CQ rewritability (including fragments of datalog+ [7]), witnesses for existential quantifiers outside the ABox are not independent from each other but interact via the temporal precedence relation. For this reason, a reduction to known languages seems to be impossible and a novel approach to rewriting has to be found. Note that straightforward temporal extensions of TQL lose first-order rewritability. For example, query answering over the ontology \{\text{Student} ⊑ ⊓ Student.Graduate\} is shown to be non-tractable.

**Related Work.** In addition to research on temporals DLs, the Semantic Web community has developed a variety of extensions of RDF/S and OWL with validity time [19, 20, 13]. The focus of this line of research is on representing and querying time-stamped RDF triples or OWL axioms. In contrast, in our language only instance data are time-stamped, while the ontology is extended with
constraints that describe how concepts and roles can change over time. In the temporal DL literature, a similar distinction has been discussed as the difference between temporalised axioms and temporalised concepts/roles; the expressive power of the respective languages is incomparable [9, 6]. To show rewritability we will use the notion of boundedness of recursion. This connection between first-order definability and boundedness is well known from the datalog and logic literature, where boundedness has been investigated extensively [10, 18, 15]. Boundedness for datalog programs on linear orders was investigated in [12]; the results are different from ours since the linear order is the only predicate symbol of the datalog programs considered and no further restrictions (comparable to ours) are imposed.

2 TQL: a Temporal Extension of OWL 2 QL

Roles $S$ and concepts $C$ of TQL are defined by the grammar:

\[
\begin{align*}
R & ::= \bot \mid P_i \mid P_i^\neg, \quad S ::= R \mid S_1 \cap S_2 \mid \Diamond_p S \mid \Diamond_S, \\
B & ::= \bot \mid A_i \mid \exists R, \quad C ::= B \mid C_1 \cap C_2 \mid \Diamond_p C \mid \Diamond_p C,
\end{align*}
\]

where $P_i$ is a role name, $A_i$ a concept name ($i \geq 0$), and $\Diamond_p$ and $\Diamond_S$ are temporal operators ‘sometime in the past’ and ‘sometime in the future,’ respectively. We call roles and concepts of the form $R$ and $B$ basic. A TQL TBox, $\mathcal{T}$, is a finite set of concept and role inclusions of the form $C \subseteq B$, $S \subseteq R$ which are assumed to hold globally (over the whole timeline). Note that the $\Diamond_p$-free fragment of TQL is an extension of the description logic DL-Lite$_{\text{horn}}^3$ [1] with role inclusions of the form $R_1 \cap \cdots \cap R_n \subseteq R$; it properly contains OWL 2 QL (the missing role constraints can be safely added to the language). A TQL ABox, $\mathcal{A}$, is a (finite) set of atoms $P_i(a, b, n)$ and $A_i(a, n)$, where $a, b$ are individual constants and $n \in \mathbb{Z}$ a temporal constant. The set of individual constants in $\mathcal{A}$ is denoted by $\text{ind}(\mathcal{A})$, and the set of temporal constants by $\text{tem}(\mathcal{A})$. A TQL knowledge base (KB) is a pair $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, where $\mathcal{T}$ is a TBox and $\mathcal{A}$ an ABox.

A temporal interpretation, $\mathcal{I}$, is given by the ordered set $(\mathbb{Z}, <)$ of time points and standard (atemporal) interpretations $\mathcal{I}(n) = (\Delta^T, \mathcal{I}(n))$, for each $n \in \mathbb{Z}$. Thus, $\Delta^T \neq \emptyset$ is the common domain of all $\mathcal{I}(n)$. $a_i^{\mathcal{I}(n)} \in \Delta^T$, $A_i^{\mathcal{I}(n)} \subseteq \Delta^T$ and $P_i^{\mathcal{I}(n)} \subseteq \Delta^T \times \Delta^T$. We assume that $a_i^{\mathcal{I}(n)} = a_i^{\mathcal{I}(0)}$, for all $n \in \mathbb{Z}$. To simplify presentation, we adopt the unique name assumption, that is, $a_i^{\mathcal{I}(n)} \neq a_j^{\mathcal{I}(n)}$ for $i \neq j$ (although the obtained results hold without it as the language has no number restrictions). The temporal constructs are interpreted in $\mathcal{I}$ as follows, where $n \in \mathbb{Z}$:

\[
\begin{align*}
\langle \Diamond_p C \rangle^{\mathcal{I}(n)} &= \{ x \mid x \in \mathcal{I}(n)^\mathcal{I}(m), \text{ for some } m < n \}, \\
\langle \Diamond_p C \rangle^{\mathcal{I}(n)} &= \{ x \mid x \in \mathcal{I}(n)^\mathcal{I}(m), \text{ for some } m > n \}, \\
\langle \Diamond_p S \rangle^{\mathcal{I}(n)} &= \{ (x, y) \mid (x, y) \in \mathcal{I}(n)^\mathcal{I}(m), \text{ for some } m < n \}, \\
\langle \Diamond_p S \rangle^{\mathcal{I}(n)} &= \{ (x, y) \mid (x, y) \in \mathcal{I}(n)^\mathcal{I}(m), \text{ for some } m > n \}.
\end{align*}
\]
The satisfaction relation $\models$ is defined as usual. If all inclusions in $\mathcal{T}$ and atoms in $\mathcal{A}$ are satisfied in $\mathcal{I}$, we call $\mathcal{I}$ a model of $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ and write $\mathcal{I} \models \mathcal{K}$.

A conjunctive query (CQ) is a (two-sorted) first-order formula $q(x, s) = \exists y, t. \varphi(x, y, s, t)$, where $\varphi(x, y, s, t)$ is a conjunction of atoms of the form $A_i(\xi, \tau)$, $P_1(\xi, \zeta, \tau)$, $(\tau = \sigma)$ and $(\tau < \sigma)$, with $\xi, \zeta$ being individual terms—individual constants or variables in $x, y$—and $\tau, \sigma$ temporal terms—temporal constants or variables in $t, s$. In a positive existential query (PEQ) $q$, the formula $\varphi$ can also contain $\vee$. A union of CQs (UCQ) is a disjunction of CQs (so every PEQ is equivalent to an exponentially larger UCQ).

Given a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ and a CQ $q(x, s)$, we call tuples $a \subseteq \text{ind}(\mathcal{A})$ and $n \subseteq \text{tem}(\mathcal{A})$ a certain answer to $q(x, s)$ over $\mathcal{K}$ and write $\mathcal{K} \models q(a, n)$, if $\mathcal{I} \models q(a, n)$ for any model $\mathcal{I}$ of $\mathcal{K}$ (understood as a two-sorted first-order model).

Example 1. Suppose Bob was a lecturer at UCL between times $n_1$ and $n_2$, after which he was appointed professor on a permanent contract. To model this situation, we use individual names, $e_1$ and $e_2$, to represent the two events of Bob’s employment. The ABox will contain $n_1 < n_2$ and the atoms $\text{lect}(bob, e_1, n_1)$, $\text{lect}(bob, e_1, n_2)$, $\text{prof}(bob, e_2, n_2 + 1)$. In the TBox, we make sure that everybody is holding the corresponding post over the duration of the contract, and include other knowledge about the university life:

$$\Diamond_p \text{lect} \sqcap \Diamond_p \text{lect} \sqsubseteq \text{lect}, \quad \Diamond_p \text{prof} \sqsubseteq \text{prof},$$
$$\exists \text{lect} \sqsubseteq \text{Lecturer}, \quad \exists \text{prof} \sqsubseteq \text{Professor},$$
$$\text{Professor} \sqsubseteq \exists \text{supervisesPhD}, \quad \text{Professor} \sqsubseteq \text{Staff},$$
$$\Diamond_p \text{supervisesPhD} \sqcap \Diamond_p \text{supervisesPhD} \sqsubseteq \text{supervisesPhD}.$$ We can now obtain staff who supervised PhDs between times $k_1$ and $k_2$ by posing the following CQ: $\exists y, t ((k_1 < t < k_2) \land \text{Staff}(x, t) \land \text{supervisesPhD}(x, y, t))$.

The key idea of OBDA is to reduce answering CQs over KBs to evaluating FO-queries over relational databases. To obtain such a reduction for TQL KBs, we employ a very basic type of temporal databases. With every TQL ABox $\mathcal{A}$, we associate a data instance $[\mathcal{A}]$ that contains all atoms from $\mathcal{A}$ as well as the atoms $(n_1 < n_2)$ such that $n_1, n_2 \in \mathbb{Z}$ with $\text{min tem}(\mathcal{A}) \leq n_1, n_2 \leq \text{max tem}(\mathcal{A})$ and $n_1 < n_2$. Thus, in addition to $\mathcal{A}$, we explicitly include in $[\mathcal{A}]$ the temporal precedence relation over the convex closure of the time points that occur in $\mathcal{A}$. (Note that, in standard temporal databases, the order over timestamps is built-in.) The main result of this paper is the following:

**Theorem 1.** Suppose $q(x, s)$ is a CQ and $\mathcal{T}$ a TQL TBox. Then one can construct a UCQ $q'(x, s)$ such that, for any consistent KB $(\mathcal{T}, \mathcal{A})$ such that $\mathcal{A}$ contains all temporal constants from $q$, any $a \subseteq \text{ind}(\mathcal{A})$ and any $n \subseteq \text{tem}(\mathcal{A})$, we have $(\mathcal{T}, \mathcal{A}) \models q(a, n)$ iff $[\mathcal{A}] \models q'(a, n)$.

Such a UCQ $q'(x, s)$ is called a rewriting for $q$ and $\mathcal{T}$. Note that consistency checking can easily be reduced to CQ-answering. Indeed, let $F$ be a fresh role name. Denote by $\mathcal{T} \perp$ the result of replacing $\perp$ with $F$ in all role inclusions of $\mathcal{T}$.
and with $\exists F$ in all concept inclusions. Clearly, $(\mathcal{T}^\perp, \mathcal{A})$ is consistent for any ABox $\mathcal{A}$, and $(\mathcal{T}, \mathcal{A})$ is inconsistent iff $(\mathcal{T}^\perp, \mathcal{A}) \models q^\perp$, where $q^\perp = \exists x, y, t F(x, y, t)$.

For an ABox $\mathcal{A}$, we denote by $\mathcal{A}^\infty$ the infinite data instance which contains the atoms in $\mathcal{A}$ as well as all $(n_1 < n_2)$ such that $n_1, n_2 \in \mathbb{Z}$ and $n_1 < n_2$. It will be convenient to regard CQs $q(x, s)$ as sets of atoms, so that we can write, e.g., $A(\xi, \tau) \in q$. We say that $q$ is totally ordered if, for any temporal terms $\tau, \tau'$ in $q$, at least one of the constraints $\tau < \tau', \tau = \tau'$ or $\tau' < \tau$ is in $q$ and the set of such constraints is consistent (in the sense that it can be satisfied in $\mathbb{Z}$). Every CQ is equivalent to a union of totally ordered CQs (the empty union is $\bot$).

**Lemma 1.** For every UCQ $q(x, s)$, one can compute a UCQ $q'(x, s)$ such that, for any ABox $\mathcal{A}$ containing all temporal constants from $q$, any $a \subseteq \text{ind}(\mathcal{A})$ and $n \subseteq \text{tem}(\mathcal{A})$, we have $\mathcal{A}^\infty \models q(a, n)$ iff $[\mathcal{A}] \models q'(a, n)$.

**Example 2.** Let $\mathcal{T} = \{ \diamond_r C \subseteq A, \diamond_r A \subseteq B \}$ and $q(x, s) = B(x, s)$. Then, for any $\mathcal{A}$, $a \in \text{ind}(\mathcal{A})$, $n \in \text{tem}(\mathcal{A})$, we have $(\mathcal{T}, \mathcal{A}) \models q(a, n)$ iff $\mathcal{A}^\infty \models q'(a, n)$, where $q'(x, s) = B(x, s) \lor \exists r \left( (t < s) \land A(x, t) \right) \lor \exists t \left( (t < s) \land (t < r) \land C(x, r) \right)$. Note, however, that $q'$ is not a rewriting for $q$ and $\mathcal{T}$. Take, for example, $\mathcal{A} = \{ C(a, 0) \}$. Then $(\mathcal{T}, \mathcal{A}) \models B(a, 0)$ but $[\mathcal{A}] \not\models q'(a, 0)$. A correct rewriting is obtained by replacing the last disjunct in $q'$ with $\exists r C(x, r)$; it can be computed by applying Lemma 1 to $q'$ and slightly simplifying the result.

In view of Lemma 1, from now on we will only focus on rewritings over $\mathcal{A}^\infty$.

The problem of finding rewritings for CQs and TQL TBoxes can be reduced to the case where the TBoxes only contain inclusions of the following form:

$$B_1 \cap B_2 \subseteq B, \quad \diamond_r B_1 \subseteq B_2, \quad \diamond_r B_1 \subseteq B_2, \quad R_1 \cap R_2 \subseteq R, \quad \diamond_r R_1 \subseteq R, \quad \diamond_r R_1 \subseteq R_2.$$  

We say that such TBoxes are in normal form.

**Theorem 2.** For every TQL TBox $\mathcal{T}$, one can construct in polynomial time a TQL TBox $\mathcal{T}'$ in normal form (possibly containing additional concept and role names) such that $\mathcal{T}' \models \mathcal{T}$ and, for every model $\mathcal{I}$ of $\mathcal{T}$, there exists a model of $\mathcal{T}'$ that coincides with $\mathcal{I}$ on all concept and role names in $\mathcal{T}$.

Suppose now that we have a UCQ rewriting $q'$ for a CQ $q$ and the TBox $\mathcal{T}'$ in Theorem 2. We obtain a rewriting for $q$ and $\mathcal{T}$ simply by removing from $q'$ those CQs that contain symbols occurring in $\mathcal{T}'$ but not in $\mathcal{T}$. From now on, we assume that all TQL TBoxes are in normal form. The set of role names in $\mathcal{T}$ and with their inverses is denoted by $R_{\mathcal{T}}$, while $|\mathcal{T}|$ is the number of concept and role names in $\mathcal{T}$.

We begin the construction of rewritings by considering the case when all concept inclusions are of the form $C \subseteq A_i$, so existential quantification $\exists F$ does not occur in the right-hand side. TQL TBoxes of this form will be called flat. Note that RDFS statements can be expressed by means of flat TBoxes.
3 UCQ Rewriting for Flat TBoxes

Let $K = (T, A)$ be a KB with a flat TBox $T$ (in normal form). Our first aim is to construct a model $C_K$ of $K$, called the canonical model, for which the following theorem holds:

**Theorem 3.** For any consistent $K = T, A$ with flat $T$ and any CQ $q(x, s)$, we have $K \models q(a, n)$ if and only if $C_K \models q(a, n)$, for all tuples $a \subseteq \text{ind}(A)$ and $n \subseteq Z$.

The construction uses a closure operator, $cl$, which applies the rules (ex), (c1)–(c3), (r1)–(r3) below to a set, $S$, of atoms of the form $R(u, v, n), A(u, n), \exists R(u, n)$ or $(n < n')$:

- **(ex)** If $R(u, v, n) \in S$ then add $\exists R(u, n), \exists R^-(v, n)$ to $S$;
- **(c1)** if $(B_1 \cap B_2 \subseteq B) \in T$ and $B_1(u, n), B_2(u, n) \in S$, then add $B(u, n)$ to $S$;
- **(c2)** if $(\lor_p B \subseteq B') \in T$, $B(u, m) \in S$ for some $m < n$ and $n$ occurs in $S$, then add $B'_{\lor}(u, n)$ to $S$;
- **(c3)** if $(\lor_p B \subseteq B') \in T$, $B(u, m) \in S$ for some $m > n$ and $n$ occurs in $S$, then add $B'_{\lor}(u, n)$ to $S$;
- **(r1)** if $(R_1 \cap R_2 \subseteq R) \in T$ and $R_1(u, v, n), R_2(u, v, n) \in S$, then add $R(u, v, n)$ to $S$;
- **(r2)** if $(\lor_p R \subseteq R') \in T$, $R(u, v, m) \in S$ for some $m < n$ and $n$ occurs in $S$, then add $R'_{\lor}(u, v, n)$ to $S$;
- **(r3)** if $(\lor_p R \subseteq R') \in T$, $R(u, v, m) \in S$ for some $m > n$ and $n$ occurs in $S$, then add $R'_{\lor}(u, v, n)$ to $S$.

We set

$$cl^0(S) = S, \quad cl^{i+1}(S) = cl(cl^i(S)), \quad cl^{\infty}(S) = \bigcup_{i \geq 0} cl^i(S).$$

Note first that $K = (T, A)$ is inconsistent if and only if $\bot \in cl^\infty(A^\sharp)$. If $K$ is consistent, we define the canonical model $C_K$ of $K$ by taking $\Delta C_K = \text{ind}(A), a \in A^{\Delta C_K(n)}$ iff $A(a, n) \in cl^\infty(A^\sharp)$, and $(a, b) \in P^{\Delta C_K(n)}$ iff $P(a, b, n) \in cl^\infty(A^\sharp)$, for $n \in Z$. (As $T$ is flat, atoms of the form $\exists R(u, n)$ can only be added by (ex).) This gives us Theorem 3. The following lemma shows that to construct $C_K$ we actually need only a bounded number of applications of $cl$ that does not depend on $A$:

**Lemma 2.** Let $T$ be a flat TBox and $n_T = (4 \cdot |T|)^4$. Then $cl^\infty(A^\sharp) = cl^{n_T}(A^\sharp)$, for any ABox $A$.

We now use Lemma 2 to construct a rewriting for any flat TBox $T$ and CQ $q(x, s)$. For a concept $C$ and a role $S$, denote by $C^T$ and $S^T$ their standard FO-translations: for example, $(\lor_p A)^T(\xi, \tau) = \exists ((\tau < t) \land A(\xi, t))$ and $\exists R^T(\xi, \tau) = \exists y R(\xi, y, \tau)$. Now, given a PEQ $\varphi$, we set $\varphi^{0\downarrow} = \varphi$ and define, inductively, $\varphi^{(n+1)\downarrow}$ as the result of replacing every

- $A(\xi, \tau)$ with $A(\xi, \tau) \lor V_{(C \subseteq A) \in T} (C^T(\xi, \tau))^{n\downarrow}$,
- $P(\xi, \tau)$ with $P(\xi, \tau) \lor V_{(S \subseteq P) \in T} (S^T(\xi, \tau))^{n\downarrow}$.
Finally, we set: \( \text{ext}^T_q(x,s) = (q(x,s))^{\nu+1} \). Clearly, \( \text{ext}^T_q(x,s) \) is a PEQ, and so can be equivalently transformed into a UCQ. By Theorem 3, Lemma 2 and Lemma 1, we obtain a rewriting for \( q \) and \( T \):

**Theorem 4.** Let \( T \) be a flat TBox and \( q(x,s) \) a CQ. Then, for any consistent KB \((T,A)\), any \( a \subseteq \text{ind}(A) \) and \( n \subseteq \mathbb{Z} \), \((T,A) \models q(a,n)\) iff \( A^2 \models \text{ext}^T_q(a,n) \).

### 4 Canonical Models for Arbitrary TBoxes

Canonical models for consistent KBs \( K = (T,A) \) with not necessarily flat TBoxes \( T \) (in normal form) can be constructed starting from \( A^2 \) and using the rules given in the previous section together with the following one:

\( (\leadsto) \) if \( \exists R(u,n) \in S \) and \( R(u,v,n) \notin S \) for any \( v \), then add \( R(u,v,n) \) to \( S \), for some fresh individual name \( v \); in this case we write \( u \leadsto^n_R v \).

Denote by \( \text{cl} \) the closure operator under the resulting 8 rules. Again, \( K \) is inconsistent iff \( \bot \in \text{cl}^n(K(A^2)) \). If \( K \) is consistent, we define the **canonical model** \( C_K \) for \( K \) by the set \( \text{cl}^n(K(A^2)) \) in the same way as in Section 3 but taking the domain \( D^C_K \) to contain all the individual names in \( \text{cl}^n(K(A^2)) \).

**Theorem 5.** For every consistent \( K = (T,A) \) and every CQ \( q(x,s) \), we have \( K \models q(a,n) \) iff \( C_K \models q(a,n) \), for any tuples \( a \subseteq \text{ind}(A) \) and \( n \subseteq \mathbb{Z} \).

**Example 3.** Let \( K = (T,A) \) with \( A = \{A(a,0)\} \) and

\[
T = \{ A \sqsubseteq \exists R, R, R \sqsubseteq Q, \exists Q, S, \Diamond P, \Diamond S \subseteq S' \}.
\]

A fragment of the model \( C_K \) is shown in the picture below:

\[
\begin{array}{c}
u_2 \quad \cdots \quad \cdots \quad \cdots \\
u_1 \quad \cdots \\
u \quad \cdots \quad \cdots \\
u \quad \cdots \\
0 \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad 1 \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad 2
\end{array}
\]

We say that the individuals \( a \in \text{ind}(A) \) are of depth 0 in \( C_K \); now, if \( u \) is of depth \( d \) in \( C_K \) and \( u \leadsto_R^n v \), for some \( n \in \mathbb{Z} \) and \( R \), then \( v \) is of depth \( d + 1 \) in \( C_K \). Thus, both \( u_1 \) and \( u_2 \) in Example 3 are of depth 2 and \( v \) is of depth 1. The restriction of \( C_K \), treated as a set of atoms, to the individual names of depth \( \leq d \) is denoted by \( C^d_K \). Note that this set is not necessarily closed under the rule \( (\leadsto) \).

In the remainder of this section, we describe the structure of \( C_K \), which is required for the rewriting in the next section. We split \( C_K \) into two parts: one consists of the elements in \( \text{ind}(A) \), while the other contains the fresh individuals introduced by \( (\leadsto) \). As this rule always uses fresh individuals, to understand the structure of the latter part it is enough to consider KBs of the form \( K_{T,R} = (T \cup \{A \sqsubseteq \exists R\}, \{A(a,0)\}) \) with fresh \( A \). We begin by analysing the behaviour of the atoms \( R'(a,u,n) \) entailed by \( R(a,u,0) \), where \( a \leadsto_R^0 u \).
Lemma 3. Let \( a \sim_R^n u \) in \( C_{K,T,R} \). If either \( m < n < 0 \) or \( 0 < n < m \), then \( R'(a,u,n) \in C_{K,T,R} \) implies \( R'(a,u,m) \in C_{K,T,R} \); moreover, if \( n < m = -|R_T| \) or \( |R_T| = m < n \), then \( R'(a,u,m) \in C_{K,T,R} \).

The atoms \( R'(a,u,n) \) entailed by \( R(a,u,0) \) in \( C_{K,T,R} \), via (r1)-(r3), also have an impact, via (ex), on the atoms of the form \( B(a,n) \) and \( B(u,n) \) in \( C_{K,T,R} \). Thus, in Example 3, \( R(a,v,0) \) entails \( \exists Q(a,n) \), for \( n > 0 \). To analyse the behaviour of such atoms, it is helpful to assume that \( T \) is in concept normal form (CoNF) in the following sense: for every role \( R \in R_T \), the TBox \( T \) contains

\[
\exists R \subseteq A^0_R \cup \exists R \subseteq A^{-1}_R \cup \forall \exists A^{-m}_R \subseteq A^{-m-1}_R \cup \forall \exists A^m_R \subseteq A^m_R, \quad \text{for } 0 \leq m \leq |R_T| \text{ and some concepts } A^i_R, \quad A^m_R \subseteq \exists R', \quad \text{for all } m \leq |R_T| \text{ and } R'(a,v,m) \in C_{K,T,R}.
\]

(In Example 3, \( C_K \) will contain the atoms \( A^1_R(a,n) \) and \( A^2_R(a,n+1) \), for \( n \geq 1 \).) By Lemma 3, if \( T \) is in CoNF, then we can compute the atoms \( B(a,n) \) and \( B(u,n) \) in \( C_{K,T,R} \) without using the rules (r1)-(r3). Lemma 3 also implies that we can add the inclusions above (with fresh \( A^i_R \)) to \( T \) if required, thereby obtaining a conservative extension of \( T \); so from now on we always assume \( T \) to be in CoNF. These observations enable the proof of the following two lemmas. The first one characterises the atoms \( B(u,n) \) in \( C_{K,T,R} \):

Lemma 4. Let \( a \sim_R^n u \) in \( C_{K,T,R} \). If either \( m < n < 0 \) or \( 0 < n < m \), then \( B(u,n) \in C_{K,T,R} \) implies \( B(u,m) \in C_{K,T,R} \); moreover, if \( n < m = -|R_T| \) or \( |R_T| = m < n \), then \( B(u,m) \in C_{K,T,R} \) if \( B(u,n) \in C_{K,T,R} \).

The second lemma characterises the ABox part of \( C_K \) and is a straightforward generalisation of Lemma 2:

Lemma 5. For any KB \( K = (T,A) \) and any atom \( \alpha \) of the form \( A(a,n) \), \( \exists R(a,n) \) or \( R(a,b,n) \), where \( a,b \in \text{ind}(A) \) and \( n \in \mathbb{Z} \), we have \( \alpha \in C_K \) iff \( \alpha \in \mathbb{C}^n_T(A^\subseteq) \).

An obvious extension of the rewriting of Theorem 4 provides, for every CQ \( q(x,s) \), a UCQ \( \text{ext}_T^T(x,s) \) of the appropriate length such that

\[
C_K \models q(a,n) \iff A^\subseteq \models \text{ext}_T^T(a,n), \quad \text{for all } a \subseteq \text{ind}(A), n \subseteq \mathbb{Z}. \quad (2)
\]

In particular, for every basic concept of the form \( \exists R \), we have \( \exists R(a,n) \in C_K \) iff \( A^\subseteq \models \text{ext}^T_R(a,n) \), for all \( a \subseteq \text{ind}(A) \) and \( n \in \mathbb{Z} \).

We now use the obtained results to show that one can find all answers to a CQ \( q \) over a TQL KB \( K \) by only considering a fragment of \( C_K \) whose size is polynomial in \( |T| \) and \( |q| \). This property is called the polynomial witness
property \cite{11}. Denote by $C^{d,\ell}_K$, for $d, \ell \geq 0$, the restriction of $C^d_K$ to the moments of time in the interval $[\min\text{tem}(A) - \ell, \max\text{tem}(A) + \ell]$.

Let $q(x, s)$ be a CQ. Tuples $a \subseteq \text{ind}(A)$ and $n \subseteq \text{tem}(A)$ give a certain answer to $q(x, s)$ over $K = (\mathcal{T}, A)$ iff there exists a homomorphism $h$ from $q$ to $C_K$, which maps individual (temporal) terms of $q$ to individual (respectively, temporal) terms of $C_K$ in such a way that the following conditions hold: $h(x) = a$ and $h(s) = n$; $h(b) = b$ and $h(m) = m$, for any individual and temporal constants $b$ and $m$; and $h(q) \subseteq C_K$, where $h(q)$ is the set of atoms obtained by replacing every term in $q$ with its $h$-image, e.g., $P(\xi, \zeta, \tau)$ with $P(h(\xi), h(\zeta), h(\tau))$ and $(\tau_1 < \tau_2)$ with $h(\tau_1) < h(\tau_2)$. Now, using the monotonicity lemmas for the temporal dimension and the fact that the atoms of depth $> |R_T|$ in the canonical models duplicate atoms of smaller depth, we obtain

**Theorem 6.** There are polynomials $f_1$ and $f_2$ such that, for any consistent TQL KB $K = (\mathcal{T}, A)$, any CQ $q(x, s)$ and any $a \subseteq \text{ind}(A)$ and $n \subseteq \text{tem}(A)$, we have $K \models q(a, n)$ iff there is a homomorphism $h$: $q \rightarrow C_K$ such that $h(q) \subseteq C^{d,\ell}_K$, where $d = f_1(|\mathcal{T}|, |q|)$ and $\ell = f_2(|\mathcal{T}|, |q|)$.

## 5 UCQ Rewriting

We now define a rewriting for any given CQ and TQL TBox. Suppose $q(x, s)$ is a CQ and $T$ a TQL TBox (in CoNF). Without loss of generality we assume $q$ to be totally ordered. By a sub-query of $q$ we understand any subset $q' \subseteq q$ containing all temporal constraints $(\tau < \tau')$ and $(\tau = \tau')$ that occur in $q$. In the rewriting for $q$ and $T$ given below, we consider all possible splittings of $q$ into two sub-queries (sharing the same temporal terms). One is to be mapped to the ABox part of the canonical model $C_{(\mathcal{T}, A)}$, and so we can rewrite it using (2). The other sub-query is to be mapped to the non-ABox part of $C_{(\mathcal{T}, A)}$ and requires a different rewriting.

For every $R \in R_T$, we construct the set $C^{d,\ell}_{K, T, R}$, where $d$ and $\ell$ are provided by Theorem 6. Let $h$ be a map from a sub-query $q_h \subseteq q$ to $C^{d,\ell}_{K, T, R}$ such that $h(q_h) \subseteq C^{d,\ell}_{K, T, R}$. Denote by $\lambda_h$ the set of individual terms $\xi$ in $q_h$ with $h(\xi) = a$, and let $\gamma_h$ be the remaining set of individual terms in $q_h$. We call $h$ a witness for $R$ if

- $\lambda_h$ contains at most one individual constant;
- every term in $\gamma_h$ is an existentially quantified variable in $q$;
- $q_h$ contains all atoms in $q$ with a variable from $\gamma_h$.

Let $h$ be a witness for $R$. Denote by $\sim$ the union of all $\sim^R$ in $C^{d,\ell}_{K, T, R}$. Clearly, $\sim$ is a tree order on the individuals in $C^{d,\ell}_{K, T, R}$, with root $a$. Let $T_h$ be its minimal sub-tree containing $a$ and the $h$-images of all the individual terms in $q_h$. For each $v \in T_h \setminus \{a\}$, we take the (unique) moment $\xi(v)$ with $u \sim^{R(v)} v$, for some $u$ and $R$, and set $\xi(a) = 0$. For $A(y, \tau) \in q_h$, we say that $h(y)$ realises $A(y, \tau)$. For any $P(\xi, \zeta, \tau) \in q_h$, there are $u, u' \in T_h$ with $u \sim u'$ and $\{u, u'\} = \{h(\xi), h(\zeta)\}$; we say that $u'$ realises $P(\xi, \zeta, \tau)$. Let $r$ be a list of fresh temporal variables $r_u$, for
Consider the following formula, whose free variables are $r_a$ and the temporal variables of $q_h$:

$$t_h = \exists r ( \bigwedge_{u \sim v} \delta^{v}(u) \rightarrow g(u, r_u) \land \bigwedge_{u \text{ realises } \alpha(x, \tau)} \delta^{h}(\tau) \rightarrow g(u, \tau)),$$

where the formulas $\delta^t(t, s)$ say that $t$ is at least $n$ moments before $s$: that is, $\delta^0(t, s)$ is $(t = s)$ and $\delta^n(t, s)$ is

$$\begin{align*}
\exists s_1, \ldots, s_{n-1} (t < s_1 < \cdots < s_{n-1} < s), & \quad \text{if } n > 0, \\
\exists s_1, \ldots, s_{|n|-1} (t > s_1 > \cdots > s_{|n|-1} > s), & \quad \text{if } n < 0.
\end{align*}$$

Take a fresh variable $x_h$ and associate with $h$ the formula

$$w_h = \exists r_a \exists x_h [\text{ext}^T_R(x_h, r_a) \land \bigwedge_{h(\xi) = a} \{\xi = x_h\} \land t_h].$$

To give the intuition behind $w_h$, suppose that $C_{(\mathcal{T}, \mathcal{A})} \models g w_h$, for some assignment $g$. Then $g$ maps all terms in $X_h$ to $g(x_h) \in \text{ind}(\mathcal{A})$ such that $\exists R(g(x_h), g(r)) \in C_{(\mathcal{T}, \mathcal{A})}$, so $(g(x_h), g(r))$ is the root of a substructure of $C_{(\mathcal{T}, \mathcal{A})}$ isomorphic to $C_{\mathcal{K}_{\mathcal{T}, r}}$ in which the variables from $Y_h$ can be mapped according to $h$. For temporal terms, the formula $t_h$ cannot specify the values prescribed by $h$: without $\neg$ in UCQs, we can only say that $\tau$ is at least (not exactly) $n$ moments before $\tau'$. However, by Lemmas 3 and 4, this is still enough to ensure that $g$ and $h$ give a homomorphism from $q_h$ to $C_{(\mathcal{T}, \mathcal{A})}$.

**Example 4.** Let $\mathcal{T}$ be the same as in Example 3 and let

$$q(x, t) = \exists y, z, t' ((t < t') \land Q(x, y, t) \land S'(y, z, t')).$$

The map $h = \{x \mapsto a, y \mapsto u, z \mapsto u_1, t \mapsto 1, t' \mapsto 2\}$ is a witness for $R$, with $q_h = q$ and $w_h$ is the following formula

$$\exists r_a \exists x_h \{\text{ext}^T_R(x_h, r_a) \land (x_h = x) \land \exists r_v \exists r_u (\delta^0(r_a, r_v) \land \delta^1(r_v, r_u) \land \delta^1(r_v, t) \land \delta^1(r_u, t')))\}.$$
6 Non-Rewritability

In this section, we show that the language TQL is nearly optimal as far as rewritability of CQs and ontologies is concerned.

We note first, that the syntax of TQL allows concept inclusions and role inclusions; ‘mixed’ axioms such as the datalog rule \( A(x, t) \land R(x, y, t) \rightarrow B(x, t) \) are not expressible. The reason is that mixed rules often lead to non-rewritability, as is well known from \( \mathcal{EL} \). For example, for \( T = \{ A(y, t) \land R(x, y, t) \rightarrow A(x, t) \} \), there is no FO-query \( q(x, t) \) such that \( (T, A) \models A(a, n) \) iff \( A^2 \models q(a, n) \) since such a query has to express that at time-point \( t \) there is an \( R \)-path from \( x \) to some \( y \) with \( A(y, t) \).

Second, it would seem to be natural to extend TQL with the temporal next/previous-time operators as concept or role constructs. However, again this would lead to non-rewritability: any FO-rewriting for \( A(x, t) \) and the TBox \( \{ \circ_f A \subseteq B, \circ_f B \subseteq A \} \) has to express that there exists \( n \geq 0 \) such that \( A(x, t - 2n) \) or \( B(x, t - (2n + 1)) \), which is impossible [16].

Another natural extension would be inclusions of the form \( A \sqsubseteq 3F B \). (Note that inclusions of the form \( A \sqsubseteq \exists R.B \) are expressible in OWL 2 QL.) But again such an extension would ruin rewritability. The reason is that temporal precedence \( < \) is a total order, and so one can construct an ABox \( A \) and a UCQ \( q(x) = q_1 \lor q_2 \) such that \( (T, A) \models q(a) \) but \( (T, A) \not\models q_i(a), i = 1, 2 \), for \( T = \{ A \sqsubseteq \diamond_f B \} \). Indeed, we can take \( A = \{ A(a, 0), C(a, 1) \} \) and

\[
q_1(x) = \exists t (C(x, t) \land B(x, t)), \\
q_2(x) = \exists t, t' ((t < t') \land C(x, t) \land B(x, t')).
\]

In fact, by reduction of 2+2-SAT [21], we prove the following:

**Theorem 8.** Answering CQs over the TBox \( \{ A \sqsubseteq \diamond_f B \} \) is coNP-hard for data complexity.

7 Conclusion

In this paper, we have proved UCQ rewritability for conjunctive queries and TQL ontologies over data instances with validity time. Our focus was solely on the existence of rewritings, and we did not consider efficiency issues such as finding shortest rewritings, using temporal intervals in the data representation or mappings between temporal databases and ontologies. We only note here that these issues are of practical importance and will be addressed in future work. It would also be of interest to investigate the possibilities to increase the expressive power of both ontology and query language. For example, we believe that the extension of TQL with the next/previous time operators, which can only occur in TBox axioms not involved in cycles, will still enjoy rewritability. We can also increase the expressivity of conjunctive queries by allowing the arithmetic operations \( + \) and \( \times \) over temporal terms, which would make the CQ \( A(x, t) \) and the TBox \( \{ \circ_f A \subseteq B, \circ_f B \subseteq A \} \) rewritable in the extended language.
References