Temporal Description Logic for Ontology-Based Data Access

Alessandro Artale

KRDB Centre, University of Bolzano

joint work with

Frank Wolter, Roman Kontchakov, Michael Zakharyaschev

Vladislav Ryzhikov and Alisa Kovtunova
OBDA: Ontology-Based Data Access

Desiderata:

- *Hide* to the user where and how data are stored
- Present to the user a *conceptual view* of the data
- *Query the data sources* through the conceptual model

![Diagram showing data layers and query flow]

**Data Layer**

**Conceptual Layer**

**Ontology**

Query over conceptual layer

KRDB Research School 2014
OBDA: Ontology-Based Data Access

- **ABox** $\mathcal{A}$:

  \begin{align*}
  \text{heartpatient}(peter), & \quad \text{diagnose}(sue, \text{fibrillation}), \quad \text{heartdisease}(\text{fibrillation})
  \end{align*}
OBDA: Ontology-Based Data Access

- **ABox $\mathcal{A}$:**
  
  $\text{heartpatient}(\text{peter}), \quad \text{diagnose}(\text{sue}, \text{fibrillation}), \quad \text{heartdisease}(\text{fibrillation})$

- **Query $q$:**

  $$q(x) = \exists y. (\text{diagnose}(x, y) \land \text{heartdisease}(y))$$

  Answer $q(\mathcal{A}) = \{\text{sue}\}$. 
OBDA: Ontology-Based Data Access

- **ABox \( \mathcal{A} \):**

  \[
  \text{heartpatient}(\text{peter}), \quad \text{diagnose}(\text{sue}, \text{fibrillation}), \quad \text{heartdisease}(\text{fibrillation})
  \]

- **Query \( q \):**

  \[
  q(x) = \exists y. (\text{diagnose}(x, y) \land \text{heartdisease}(y))
  \]

  Answer \( q(\mathcal{A}) = \{ \text{sue} \} \).

- **Ontology/TBox \( \mathcal{T} \):**

  \[
  \text{heartpatient} \sqsubseteq \exists \text{diagnose}. \text{heartdisease}
  \]
OBDA: Ontology-Based Data Access

- **ABox** $\mathcal{A}$:
  
  \[
  \text{heartpatient}(\text{peter}), \quad \text{diagnose}(\text{sue}, \text{fibrillation}), \quad \text{heartdisease}(\text{fibrillation})
  \]

- **Query** $q$:
  
  \[
  q(x) = \exists y. (\text{diagnose}(x, y) \land \text{heartdisease}(y))
  \]
  
  Answer $q(\mathcal{A}) = \{\text{sue}\}$.

- **Ontology/TBox** $\mathcal{T}$:
  
  \[
  \text{heartpatient} \sqsubseteq \exists\text{diagnose}.\text{heartdisease}
  \]
OBDA: Ontology-Based Data Access

- **ABox** $\mathcal{A}$:
  
  heartpatient(peter),  diagnose(sue, fibrillation),  heartdisease(fibrillation)

- **Query** $q$:
  
  \[ q(x) = \exists y. (\text{diagnose}(x, y) \land \text{heartdisease}(y)) \]
  
  Answer $q(\mathcal{A}) = \{\text{sue}\}$.

- **Ontology/TBox** $\mathcal{T}$:
  
  heartpatient $\sqsubseteq \exists \text{diagnose.heartdisease}$

- **Certain Answers**:
  
  \[ \text{cert}_{\mathcal{T}, \mathcal{A}}(q) = \{a \mid \mathcal{T} \cup \mathcal{A} \models q(a)\} \]
  
  In this case
  
  \[ \text{cert}_{\mathcal{T}, \mathcal{A}}(q) = \{\text{sue, peter}\}. \]
OBDA for Temporal Data

In applications, data are often time-dependent: employment contracts end, children are born, aircrafts arrive.
OBDA for Temporal Data

In applications, data are often time-dependent: employment contracts end, children are born, aircrafts arrive.

- Temporal data (temporal ABoxes) $\mathcal{A}$ are finite sets of pairs consisting of facts and their validity time:

  atrisk(peter, 2013),  diagnose(sue, fibrillation, 1982),  heartdisease(fibrillation)
OBDA for Temporal Data

In applications, data are often time-dependent: employment contracts end, children are born, aircrafts arrive.

- Temporal data (temporal ABoxes) \( \mathcal{A} \) are finite sets of pairs consisting of facts and their validity time:

\[
\text{atrisk}(\text{peter}, 2013), \quad \text{diagnose}(\text{sue}, \text{fibrillation}, 1982), \quad \text{heartdisease}(\text{fibrillation})
\]

- To support querying temporal data, the ontology \( \mathcal{T} \) should model temporal conceptual knowledge as well:

\[
\diamond_p \exists \text{diagnose.heartdisease} \sqsubseteq \text{atrisk}
\]

\[
\forall x, t(((\exists t' < t) \exists y.\text{diagnose}(x, y, t') \land \text{heartdisease}(y, t')) \rightarrow \text{atrisk}(x, t))
\]
OBDA for Temporal Data

In applications, data are often time-dependent: employment contracts end, children are born, aircrafts arrive.

- Temporal data (temporal ABoxes) \( \mathcal{A} \) are finite sets of pairs consisting of facts and their validity time:
  
  \[
  \text{atrisk}(\text{peter}, 2013), \quad \text{diagnose}(\text{sue}, \text{fibrillation}, 1982), \quad \text{heartdisease}(\text{fibrillation})
  \]

- To support querying temporal data, the ontology \( \mathcal{T} \) should model temporal conceptual knowledge as well:
  
  \[
  \diamond P \exists \text{diagnose.heartdisease} \sqsubseteq \text{atrisk}
  \]

  \[
  \forall x, t (((\exists t' < t) \exists y. \text{diagnose}(x, y, t') \land \text{heartdisease}(y, t')) \rightarrow \text{atrisk}(x, t))
  \]

- For \( q = \text{atrisk}(x, 2013) \) we obtain
  
  \[
  \text{cert}_{\mathcal{T}, \mathcal{A}}(q) = \{ \text{peter}, \text{sue} \} \]
Our Aims

• Cover validity time (no transaction time): ABox assertions of the form

\[ A(c, n), \quad P(c, d, n) \]

More succinct intervals \( A(c, [n, m]) \) not yet considered.
Our Aims

• Cover **validity time** (no transaction time): ABox assertions of the form

\[ A(c, n), \quad P(c, d, n) \]

More succinct intervals \( A(c, [n, m]) \) not yet considered.

• Ontology language **temporal extension of OWL 2 QL** (OWL standard for OBDA). Axioms time-independent, but model time-dependent classes and properties. E.g.,

\[ \Diamond_P \text{givesbirth} \sqsubseteq \text{mother} \]
Our Aims

- Cover **validity time** (no transaction time): ABox assertions of the form

  \[ A(c, n), \quad P(c, d, n) \]

  More succinct intervals \( A(c, [n, m]) \) not yet considered.

- Ontology language **temporal extension of OWL 2 QL** (OWL standard for OBDA). Axioms time-independent, but model time-dependent classes and properties. E.g.,

  \[ \Diamond_P \text{givesbirth} \sqsubseteq \text{mother} \]

- Queries at least **two sorted conjunctive queries** with variables for individuals and timepoints, and expressions \( t < t', A(x, t), P(x, y, t) \).
Our Aims

- Cover **validity time** (no transaction time): ABox assertions of the form
  \[ A(c, n), \quad P(c, d, n) \]

  More succinct intervals \( A(c, [n, m]) \) not yet considered.

- Ontology language **temporal extension of OWL 2 QL** (OWL standard for OBDA). Axioms time-independent, but model time-dependent classes and properties. E.g.,
  \[ \Diamond_P \text{givesbirth} \sqsubseteq \text{mother} \]

- Queries at least **two sorted conjunctive queries** with variables for individuals and timepoints, and expressions \( t < t', A(x, t), P(x, y, t) \).

- Every such query should be **SQL/FO-rewritable** (with linear-order < available).

KRDB Research School 2014
The Ontology Language: TQL

TQL contains **OWL 2 QL**, where OWL 2 QL ontologies consist of inclusions

\[ B_1 \cap B_2 \sqsubseteq \bot, \quad B_1 \sqsubseteq B_2, \quad R_1 \sqsubseteq R_2 \]

with

\[ R_i ::= \bot \mid P \mid P^-, \]

\[ B_i ::= A \mid \exists R_i, \]
The Ontology Language: TQL

TQL contains **OWL 2 QL**, where OWL 2 QL ontologies consist of inclusions

\[ B_1 \sqcap B_2 \sqsubseteq \bot, \quad B_1 \sqsubseteq B_2, \quad R_1 \sqsubseteq R_2 \]

with

\[ R_i ::= \bot \mid P \mid P^-, \]

\[ B_i ::= A \mid \exists R_i, \]

and should be “maximal” FO-rewritable with:

- rigid concept and roles;
- persistent in the future concepts and roles;
- instantaneous concepts and roles;
- convex concepts and roles.
- etc.
Syntax: OWL 2 QL extended by $\Diamond_F$ and $\Diamond_P$

TQL ontologies/TBox consist of inclusions

\[ C \sqsubseteq B, \quad S \sqsubseteq R \]

where

\[ R ::= \bot \mid P \mid P^-, \]

\[ B ::= \bot \mid A \mid \exists R, \]

KRDB Research School 2014
Syntax: OWL 2 QL extended by $\Diamond_F$ and $\Diamond_P$

TQL ontologies/TBox consist of inclusions

$$C \sqsubseteq B, \quad S \sqsubseteq R$$

where

$$R ::= \bot \mid P \mid P^-,$$

$$B ::= \bot \mid A \mid \exists R,$$

and $C$ and $S$ are defined by:

$$C ::= B \mid C_1 \sqcap C_2 \mid \Diamond_P C \mid \Diamond_F C,$$

$$S ::= R \mid S_1 \sqcap S_2 \mid \Diamond_P S \mid \Diamond_F S,$$
Syntax: OWL 2 QL extended by $\Diamond_F$ and $\Diamond_P$

TQL ontologies/TBox consist of inclusions

$$C \sqsubseteq B, \quad S \sqsubseteq R$$

where

$$R ::= \bot \mid P \mid P^-, \quad B ::= \bot \mid A \mid \exists R,$$

and $C$ and $S$ are defined by:

$$C ::= B \mid C_1 \sqcap C_2 \mid \Diamond_P C \mid \Diamond_F C,$$

$$S ::= R \mid S_1 \sqcap S_2 \mid \Diamond_P S \mid \Diamond_F S,$$

Thus TQL has a Horn-like TBox with temporal operators only on the left-hand side.
TQL: Expressivity

TQL can express the following temporal constraints:

• person is rigid: $\diamond_F \diamond_P person \sqsubseteq person$;

• mother is persistent: $\diamond_P mother \sqsubseteq mother$;

• givesbirth is instantaneous: givesbirth $\sqcap \diamond_P givesbirth \sqsubseteq \bot$;

• employed is convex: $\diamond_P employed \sqcap \diamond_F employed \sqsubseteq employed$. 
Semantics

Temporal interpretations $\mathcal{I}$ are given by $(\mathbb{Z}, <)$ (time points) and standard (atemporal) interpretations $\mathcal{I}(n) = (\Delta^\mathcal{I}, \cdot^\mathcal{I}(n))$, for each $n \in \mathbb{Z}$. We assume constant domain and rigid interpretation of individuals. Thus, interpretations look as follows:
Temporal interpretations $\mathcal{I}$ are given by $(\mathbb{Z}, <)$ (time points) and standard (atemporal) interpretations $\mathcal{I}(n) = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}(n)})$, for each $n \in \mathbb{Z}$. We assume constant domain and rigid interpretation of individuals. Thus, interpretations look as follows:

$$(\Diamond_P C)^{\mathcal{I}(n)} = \{ x \mid x \in C^{\mathcal{I}(m)}, \text{ for some } m < n \},$$

$$(\Diamond_F C)^{\mathcal{I}(n)} = \{ x \mid x \in C^{\mathcal{I}(m)}, \text{ for some } m > n \}.$$
Consider again

- $A$: 
  
  \begin{align*}
  \text{atrisk}(peter, 2013), & \quad \text{diagnose}(sue, \text{fibrillation}, 1982), \\
  \text{heartdisease}(\text{fibrillation})
  \end{align*}

- $T$: 
  
  $\Diamond_p \exists \text{diagnose. heartdisease} \sqsubseteq \text{atrisk}$
Temporal SQL/FO-Rewritability

Consider again

- $\mathcal{A}$:
  
  $\text{atrisk}(\text{peter}, 2013), \text{diagnose}(\text{sue}, \text{fibrillation}, 1982), \text{heartdisease}(\text{fibrillation})$

- $\mathcal{T}$:
  
  $\Diamond_P \exists \text{diagnose.heartdisease} \sqsubseteq \text{atrisk}$

- Then $q = \text{atrisk}(x, 2013)$ can be rewritten into
  
  $q_T = \text{atrisk}(x, 2013) \lor \exists t' < 2013. \exists y. \text{diagnose}(x, y, t') \land \text{heartdisease}(y, t')$

and

$(\mathcal{T}, \mathcal{A}) \models q(a, 2013) \text{ iff } \mathcal{A} \models q_T(a, 2013)$
Temporal Datalog\(\exists\) Formulation

Let

\[ B = A \mid \exists R \]

TBoxes consist of “datalog” rules of the form

\[ B(x, t) \leftarrow \text{Body}(x, \bar{t}) \]

where \(\text{Body}(x, \bar{t})\) is a conjunction of atoms of the form \(B'(x, t')\) and \(t' < t''\) and

\[ P(x, y, t) \leftarrow \text{Body}(x, y, \bar{t}) \]

where \(\text{Body}(x, y, \bar{t})\) is a conjunction of atoms of the form \(B'(x, y, t')\) and \(t' < t''\).

Note: Link between rules for unary and binary predicates only via \(\exists R\).
Main Result

Queries are two-sorted conjunctive queries (CQs):

$$\exists \bar{y} \bar{t} \varphi(\bar{x}, \bar{y}, \bar{s}, \bar{t})$$

where the atoms are of the form

$$A(x, t), \quad P(x, y, t), \quad (t_1 = t_2), \quad (t_1 < t_2)$$
Main Result

Queries are two-sorted conjunctive queries (CQs):

$$\exists \bar{y} \bar{t} \varphi(\bar{x}, \bar{y}, \bar{s}, \bar{t})$$

conjunction of atoms

where the atoms are of the form

$$A(x, t), \quad P(x, y, t), \quad (t_1 = t_2), \quad (t_1 < t_2)$$

Theorem. Let $q(\bar{x}, \bar{t})$, be a CQ and $\mathcal{T}$ a TQL ontology. Then one can construct a disjunction of CQs $q_\mathcal{T}(\bar{x}, \bar{t})$ such that, for any $\mathcal{A}$, any $\bar{a} \subseteq \text{ind}(\mathcal{A})$, and any $\bar{n} \subseteq \text{tem}(\mathcal{A})$, we have

$$(\mathcal{T}, \mathcal{A}) \models q(\bar{a}, \bar{n}) \quad \text{iff} \quad \mathcal{A} \models q_\mathcal{T}(\bar{a}, \bar{n})$$
Extensions not tractable and not FO-rewritable

• Mixing concepts and roles: $\exists R.A \sqsubseteq A$ not FO-rewritable.
Extensions not tractable and not FO-rewritable

- Mixing concepts and roles: $\exists R.A \sqsubseteq A$ not FO-rewritable.
  
  $\diamond pA \sqsubseteq A$ is rewritable only because $<$ is transitive.
Extensions not tractable and not FO-rewritable

• Mixing concepts and roles: $\exists R.A \sqsubseteq A$ not FO-rewritable.

  $\Diamond_P A \sqsubseteq A$ is rewritable only because $<$ is transitive.

• NEXT-operators: $\bigcirc_P A \sqsubseteq B$ and $\bigcirc_P B \sqsubseteq A$ can be used to express even distance between time points.
Extensions not tractable and not FO-rewritable

• Mixing concepts and roles: $\exists R. A \sqsubseteq A$ not FO-rewritable.
  $\Diamond_p A \sqsubseteq A$ is rewritable only because $<$ is transitive.

• NEXT-operators: $\bigcirc_p A \sqsubseteq B$ and $\bigcirc_p B \sqsubseteq A$ can be used to express even distance between time points.

• CQ answering for $\{ A \sqsubseteq \Diamond_p B \}$ NP-hard—by reduction of $2 + 2$-SAT.
Extensions with NEXT - $\diamond^F$

The TQL language with \texttt{nextime}, $\diamond^F$, Atomic Concepts and Horn axioms is not in $AC^0$. 
Extensions with NEXT - $\bigcirc^F$

The TQL language with nextime, $\bigcirc^F$, Atomic Concepts and Horn axioms is not in $AC^0$.

Parity problem: Given a binary string output 1 iff the number of 1s is even.

We reduce the Parity problem which is not computable in $AC^0$ (Furst, Saxe and Sipser, 1984) to query answering in TQL TBox with $\bigcirc^F$. 
Extensions with NEXT - $\bigcirc_F$

The TQL language with nextime, $\bigcirc_F$, Atomic Concepts and Horn axioms is not in $\mathcal{AC}^0$.

**Parity problem**: Given a binary string output 1 iff the number of 1s is even.

We reduce the Parity problem which is not computable in $\mathcal{AC}^0$ (Furst, Saxe and Sipser, 1984) to query answering in TQL TBox with $\bigcirc_F$.

**TBox**

$$
\mathcal{T} = \{ C_1 \cap \bigcirc_F C_{\text{even}} \sqsubseteq C_{\text{odd}}, C_1 \cap \bigcirc_F C_{\text{odd}} \sqsubseteq C_{\text{even}} \\
C_0 \cap \bigcirc_F C_{\text{even}} \sqsubseteq C_{\text{even}}, C_0 \cap \bigcirc_F C_{\text{odd}} \sqsubseteq C_{\text{odd}} \} 
$$

**ABox.** Encodes the binary strings and terminates with $C_{\text{even}}(a, n + 1)$. E.g., the binary string $w = 01001$ is encoded as:

$$
\mathcal{A}_w = \{ C_0(a, 0), C_1(a, 1), C_0(a, 2), C_0(a, 3), C_1(a, 4), C_{\text{even}}(a, 5) \} 
$$

$(\mathcal{T}, \mathcal{A}_w) \models C_{\text{even}}(a, 0)$ iff $w$ has an even number of 1’s
Extensions with NEXT and Automata

We can construct a Non-Deterministic Finite Automata (NFA) to compute query answers. E.g., the automaton $A_T$ for the parity TBox starting at $t = n$ is:

$A_T$ accepts $A$ iff $(T, A) \models C_{\text{even}}(a, 0)$.

- **Upper Bound.** The problem whether an automata accepts a word is tractable: it belongs to complexity class $NC^1$ (contained in LogSpace).

- **Future Work.** The automata encoding without roles is obvious: We intend to extend it to languages with roles.
Future Work

- Investigate **efficient** rewritings, implementation.
- Consider **datalog-rewritability**: then NEXT-operator should be ok.
- The TQL languages with $\bigcirc_F$ seems to be still FO-rewritable with arithmetic predicates, e.i., $\text{TQL}_{\text{core}, \bigcirc_F}$ is conjectured to be in $\text{FO}(+, \times)$. 