Outline

1. The Controlled Query Evaluation approach in Ontologies and Description Logics
   - CQE in Description Logics through GA censors
   - Computational problems

2. Towards tractability 1: Intersecting the censors
   - IGA censors
   - Expressive limitations of IGA censors

3. Towards tractability 2: Adding preferences
   - Globally optimal and Pareto-optimal censors
   - DD and k-DD censors
   - Experimental results

4. Towards tractability 3: Maximally cooperative approach
   - The dynCQE approach
   - Complexity of dynCQE

5. Conclusions
Outline

1. The Controlled Query Evaluation approach in Ontologies and Description Logics
   - CQE in Description Logics through GA censors
   - Computational problems

2. Towards tractability 1: Intersecting the censors
   - IGA censors
   - Expressive limitations of IGA censors

3. Towards tractability 2: Adding preferences
   - Globally optimal and Pareto-optimal censors
   - DD and k-DD censors
   - Experimental results

4. Towards tractability 3: Maximally cooperative approach
   - The dynCQE approach
   - Complexity of dynCQE

5. Conclusions
Scenario: system providing access (query answering service) to a dataset

Problem: enforce a confidentiality-preserving policy
  i.e. some data cannot be disclosed to the system users

Solution 1: modify the dataset in order to enforce the policy
  and do not change the system/query answering service

Solution 2: modify the query answering service in order to enforce the policy
  and do not change the dataset
We study confidentiality-preserving query answering in Description Logics (DLs) in the spirit of **Controlled Query Evaluation (CQE)**. 

CQE is a confidentiality-preserving query answering approach studied:
- in databases [*Sicherman et al., TODS 1983*]
- in Description Logic (DL) ontologies [*Bonatti and Sauro, ISWC 2013*, *Cuenca Grau et al., IJCAI 2015*]

In CQE:
- the **policy** is specified in terms of **logical formulas**
- the enforcement of the policy is formalized through the notion of **censor**
- a censor models the **answers** that the query answering system should provide
- an **optimal censor** maximizes query answers still guaranteeing that the policy is not violated
We study confidentiality-preserving query answering in Description Logics (DLs) in the spirit of \textit{Controlled Query Evaluation (CQE)}. CQE is a confidentiality-preserving query answering approach studied:

- in databases [Sicherman et al., TODS 1983]
- in Description Logic (DL) ontologies [Bonatti and Sauro, ISWC 2013], [Cuenca Grau et al., IJCAI 2015]

In CQE:

- the \textbf{policy} is specified in terms of \textit{logical formulas}
- the enforcement of the policy is formalized through the notion of \textit{censor}
- a censor models the \textbf{answers} that the query answering system should provide
- an \textbf{optimal censor} maximizes query answers still guaranteeing that the policy is not violated
In [IJCAI 2019] we consider a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where:

- $\mathcal{T}$ is a DL TBox, representing intensional knowledge (i.e., the schema)
- $\mathcal{A}$ is a DL ABox, that is a set of facts (i.e., the data instance)
- $\mathcal{P}$ is the policy, i.e., a set of denial assertions of the form $\forall \vec{x}. cq(\vec{x}) \rightarrow \bot$, s.t. $\exists \vec{x}. cq(\vec{x})$ is a Conjunctive Query (CQ)

A user asks queries over the TBox $\mathcal{T}$, but must not get as result data that let her answer a query $\exists \vec{x}. cq(\vec{x})$ occurring in a denial assertions in $\mathcal{P}$.

A censor is a function that modifies query answers to ensure this behaviour!
Ground Atoms (GA) Censors

**Notation:** $\mathit{cl}_T^\mathcal{GA}(\mathcal{A})$ denotes the set of facts (aka GAs) implied by $\mathcal{T} \cup \mathcal{A}$.

A **GA censor** $c$ for a CQE instance $\mathcal{E} = \langle T, \mathcal{A}, \mathcal{P} \rangle$ is a function that returns a set (called the **theory of the censor**) $\mathsf{Th}_c \subseteq \mathit{cl}_{T}^{\mathcal{GA}}(\mathcal{A})$ s.t. $\mathcal{T} \cup \mathsf{Th}_c \not\models \neg \mathcal{P}$.

$c$ is **optimal** if there is no GA censor $c'$ for $\mathcal{E}$ such that $\mathsf{Th}_c \subset \mathsf{Th}_{c'}$.

**Example:**

$\mathcal{T} = \{\text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}$

$\mathcal{A} = \{\text{TakesMedA}(\text{bob}), \text{TakesMedB}(\text{bob}), \text{Diabetic(\text{bob})}, \text{TakesMedA(\text{ann})}, \text{Diabetic(\text{ann})}, \text{TakesMedA(\text{joe})} \}$

$\mathcal{P} = \{\forall x.\text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \forall x.\text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}$

$\mathsf{Secrets} = \{\{\text{TakesMedA}(\text{bob}), \text{Diabetic(\text{bob})}\}, \{\text{TakesMedB}(\text{bob}), \text{Diabetic(\text{bob})}\},$

$\{\text{TakesMedA(\text{ann})}, \text{Diabetic(\text{ann})}\}\}$

$\mathsf{Th}_{c1} = \{\text{TakesMedA}(\text{bob}), \text{TakesMedB}(\text{bob}), \text{TakesMedA(\text{ann})}, \text{TakesMedA(\text{joe})}\}$

$\mathsf{Th}_{c2} = \{\text{TakesMedA}(\text{bob}), \text{TakesMedB}(\text{bob}), \text{Diabetic(\text{ann})}, \text{TakesMedA(\text{joe})}\}$

$\mathsf{Th}_{c3} = \{\text{Diabetic(\text{bob})}, \text{TakesMedA(\text{ann})}, \text{TakesMedA(\text{joe})}\}$

$\mathsf{Th}_{c4} = \{\text{Diabetic(\text{bob})}, \text{Diabetic(\text{ann})}, \text{TakesMedA(\text{joe})}\}$

$c_1, c_2, c_3$, and $c_4$ are the optimal GA censors for $\mathcal{E} = \langle T, \mathcal{A}, \mathcal{P} \rangle$. 
Ground Atoms (GA) Censors

**Notation:** $\text{cl}_G^T(A)$ denotes the set of facts (aka GAs) implied by $T \cup A$.

A **GA censor** $c$ for a CQE instance $E = \langle T, A, P \rangle$ is a function that returns a set (called the **theory of the censor**) $\text{Th}_c \subseteq \text{cl}_G^T(A)$ s.t. $T \cup \text{Th}_c \not\models \neg P$.

$c$ is **optimal** if there is no GA censor $c'$ for $E$ such that $\text{Th}_c \subset \text{Th}_{c'}$.

**Example:**

\[
T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \quad \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \} \\
A = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(bob), \text{TakesMedA}(ann), \text{Diabetic}(ann), \text{TakesMedA}(joe) \} \\
P = \{ \forall x.\text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \quad \forall x.\text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \} \\
\text{Secrets} = \{ \{ \text{TakesMedA}(bob), \text{Diabetic}(bob) \}, \{ \text{TakesMedB}(bob), \text{Diabetic}(bob) \}, \{ \text{TakesMedA}(ann), \text{Diabetic}(ann) \} \} \\
\text{Th}_{c1} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{TakesMedA}(ann), \text{TakesMedA}(joe) \} \\
\text{Th}_{c2} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(ann), \text{TakesMedA}(joe) \} \\
\text{Th}_{c3} = \{ \text{Diabetic}(bob), \text{TakesMedA}(ann), \text{TakesMedA}(joe) \} \\
\text{Th}_{c4} = \{ \text{Diabetic}(bob), \text{Diabetic}(ann), \text{TakesMedA}(joe) \} \\
\]

$c_1, c_2, c_3,$ and $c_4$ are the optimal GA censors for $E = \langle T, A, P \rangle$. 
Ground Atoms (GA) Censors

**Notation:** $\text{cl}^T_{\text{GA}}(A)$ denotes the set of facts (aka GAs) implied by $T \cup A$.

A **GA censor** $c$ for a CQE instance $E = \langle T, A, P \rangle$ is a function that returns a set (called the **theory of the censor**) $\text{Th}_c \subseteq \text{cl}^T_{\text{GA}}(A)$ s.t. $T \cup \text{Th}_c \not\models \neg P$.

$c$ is **optimal** if there is no GA censor $c'$ for $E$ such that $\text{Th}_c \subset \text{Th}_c'$.

**Example:**

$T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \; \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}$

$A = \{ \text{TakesMedA}(bob), \; \text{TakesMedB}(bob), \; \text{Diabetic}(bob), \; \text{TakesMedA}(ann), \; \text{Diabetic}(ann), \; \text{TakesMedA}(joe) \}$

$P = \{ \forall x.\text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \; \forall x.\text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}$

$\text{Secrets} = \{ \{ \text{TakesMedA}(bob), \; \text{Diabetic}(bob) \}, \{ \text{TakesMedB}(bob), \; \text{Diabetic}(bob) \}, \{ \text{TakesMedA}(ann) \}, \; \text{Diabetic}(ann) \}$

$\text{Th}_{c_1} = \{ \text{TakesMedA}(bob), \; \text{TakesMedB}(bob), \; \text{TakesMedA}(ann), \; \text{TakesMedA}(joe) \}$

$\text{Th}_{c_2} = \{ \text{TakesMedA}(bob), \; \text{TakesMedB}(bob), \; \text{Diabetic}(ann), \; \text{TakesMedA}(joe) \}$

$\text{Th}_{c_3} = \{ \text{Diabetic}(bob), \; \text{TakesMedA}(ann), \; \text{TakesMedA}(joe) \}$

$\text{Th}_{c_4} = \{ \text{Diabetic}(bob), \; \text{Diabetic}(ann), \; \text{TakesMedA}(joe) \}$

$c_1, c_2, c_3, \text{ and } c_4 \text{ are the optimal GA censors for } E = \langle T, A, P \rangle$. 

Ground Atoms (GA) Censors

**Notation:** $\text{cl}_{GA}^T(A)$ denotes the set of facts (aka GAs) implied by $T \cup A$.

A **GA censor** $c$ for a CQE instance $E = \langle T, A, P \rangle$ is a function that returns a set (called the *theory of the censor*) $\text{Th}_c \subseteq \text{cl}_{GA}^T(A)$ s.t. $T \cup \text{Th}_c \not\models \neg P$. $c$ is *optimal* if there is no GA censor $c'$ for $E$ such that $\text{Th}_c \subset \text{Th}_{c'}$.

**Example:**

- $T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \ \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}$
- $A = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{Diabetic}(bob), \ \text{TakesMedA}(ann), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \}$
- $P = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \ \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}$
- $\text{Secrets} = \{ \{ \text{TakesMedA}(bob), \ \text{Diabetic}(bob) \}, \{ \text{TakesMedB}(bob), \ \text{Diabetic}(bob) \}, \{ \text{TakesMedA}(ann)), \ \text{Diabetic}(ann) \} \}$
- $\text{Th}_{c1} = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{TakesMedA}(ann), \ \text{TakesMedA}(joe) \}$
- $\text{Th}_{c2} = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \}$
- $\text{Th}_{c3} = \{ \text{Diabetic}(bob), \ \text{TakesMedA}(ann), \ \text{TakesMedA}(joe) \}$
- $\text{Th}_{c4} = \{ \text{Diabetic}(bob), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \}$

$c_1, c_2, c_3, \text{ and } c_4$ are the optimal GA censors for $E = \langle T, A, P \rangle$. 
Ground Atoms (GA) Censors

**Notation:** $\text{cl}_{GA}^T(A)$ denotes the set of facts (aka GAs) implied by $T \cup A$.

A *GA censor* $c$ for a CQE instance $E = \langle T, A, P \rangle$ is a function that returns a set (called the *theory of the censor*) $\text{Th}_c \subseteq \text{cl}_{GA}^T(A)$ s.t. $T \cup \text{Th}_c \not\models \neg P$.

$c$ is *optimal* if there is no GA censor $c'$ for $E$ such that $\text{Th}_c \subset \text{Th}_c'$.

**Example:**

$T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \; \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}$

$A = \{ \text{TakesMedA}(bob), \; \text{TakesMedB}(bob), \; \text{Diabetic}(bob), \; \text{TakesMedA}(ann), \; \text{Diabetic}(ann), \; \text{TakesMedA}(joe) \}$

$P = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \; \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}$

$\text{Secrets} = \{ \{ \text{TakesMedA}(bob), \; \text{Diabetic}(bob) \}, \; \{ \text{TakesMedB}(bob), \; \text{Diabetic}(bob) \}, \; \{ \text{TakesMedA}(ann)), \; \text{Diabetic}(ann) \}\}$

$\text{Th}_{c_1} = \{ \text{TakesMedA}(bob), \; \text{TakesMedB}(bob), \; \text{TakesMedA}(ann), \; \text{TakesMedA}(joe) \}$

$\text{Th}_{c_2} = \{ \text{TakesMedA}(bob), \; \text{TakesMedB}(bob), \; \text{Diabetic}(ann), \; \text{TakesMedA}(joe) \}$

$\text{Th}_{c_3} = \{ \text{Diabetic}(bob), \; \text{TakesMedA}(ann), \; \text{TakesMedA}(joe) \}$

$\text{Th}_{c_4} = \{ \text{Diabetic}(bob), \; \text{Diabetic}(ann), \; \text{TakesMedA}(joe) \}$

$c_1, c_2, c_3, \text{ and } c_4$ are the optimal GA censors for $E = \langle T, A, P \rangle$. 

Riccardo Rosati
Knowledge-based Confidentiality-preserving Query Answering
WEBIST 2022 (7/39)
Ground Atoms (GA) Censors

Notation: \( \text{cl}^T_{\text{GA}}(A) \) denotes the set of facts (aka GAs) implied by \( T \cup A \).

A \textit{GA censor} \( c \) for a CQE instance \( E = \langle T, A, P \rangle \) is a function that returns a set (called the \textit{theory of the censor}) \( \text{Th}_c \subseteq \text{cl}^T_{\text{GA}}(A) \) s.t. \( T \cup \text{Th}_c \not\models \neg P \).

\( c \) is \textit{optimal} if there is no GA censor \( c' \) for \( E \) such that \( \text{Th}_c \subset \text{Th}_{c'} \).

Example:

\[ T = \{ \text{TakesMedB} \subseteq \exists \text{admissionWard}; \ \text{Diabetic} \subseteq \exists \text{admissionWard} \} \]
\[ A = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{Diabetic}(bob), \ \text{TakesMedA}(ann), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \} \]
\[ P = \{ \forall x. \text{TakesMedA}(x) \wedge \text{Diabetic}(x) \rightarrow \bot; \ \forall x. \text{TakesMedB}(x) \wedge \text{Diabetic}(x) \rightarrow \bot \} \]
\[ \text{Secrets} = \{ \{ \text{TakesMedA}(bob), \ \text{Diabetic}(bob) \}, \{ \text{TakesMedB}(bob), \ \text{Diabetic}(bob) \}, \{ \text{TakesMedA}(ann)), \ \text{Diabetic}(ann) \} \} \]
\[ \text{Th}_{c1} = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{TakesMedA}(ann), \ \text{TakesMedA}(joe) \} \]
\[ \text{Th}_{c2} = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \} \]
\[ \text{Th}_{c3} = \{ \text{Diabetic}(bob), \ \text{TakesMedA}(ann), \ \text{TakesMedA}(joe) \} \]
\[ \text{Th}_{c4} = \{ \text{Diabetic}(bob), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \} \]

\( c_1, c_2, c_3, \text{ and } c_4 \) are the optimal GA censors for \( E = \langle T, A, P \rangle \).
How to deal with multiple GA censors?

1. choose any *arbitrary* GA censor
   - a random choice does not seem to make much sense

2. choose a *predefined* GA censor
   - how? More information would be needed

3. keep *all* the GA censors
   - query answering is done with respect to all the GA censors
   - **skeptical entailment**: only the query answers that are true in all the GA censors are returned
   - studied in [IJCAI 2019]
Given a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ and a BCQ $q$, we are interested in

**Boolean Conjunctive Query (BCQ) entailment under GA censors**

i.e., deciding whether $\mathcal{T} \cup \text{Th}_c \models q$ for *every* optimal GA censor $c$ for $\mathcal{E}$.

**Example (cntd):**

$\mathcal{T} = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \ \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}$

$\mathcal{A} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(bob), \text{TakesMedA}(ann)), \ \text{Diabetic}(ann), \text{TakesMedA}(joe) \}$

$\mathcal{P} = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \ \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}$

$\text{Secrets} = \{ \{ \text{TakesMedA}(bob), \text{Diabetic}(bob) \}, \{ \text{TakesMedB}(bob), \text{Diabetic}(bob) \}, \ \{ \text{TakesMedA}(ann)), \text{Diabetic}(ann)\} \}$

$\text{Th}_{c_1} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{TakesMedA}(ann), \text{TakesMedA}(joe) \}$

$\text{Th}_{c_2} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(ann), \text{TakesMedA}(joe) \}$

$\text{Th}_{c_3} = \{ \text{Diabetic}(bob), \text{TakesMedA}(ann), \text{TakesMedA}(joe) \}$

$\text{Th}_{c_4} = \{ \text{Diabetic}(bob), \text{Diabetic}(ann), \text{TakesMedA}(joe) \}$

$\text{TakesMedA}(joe)$ and $\exists x, y. \text{admissionWard}(x, y)$ are both entailed by $\mathcal{E}$ under GA censors.
Entailment of Boolean Conjunctive Queries (BCQs)

Given a CQE instance $\mathcal{E} = \langle T, A, P \rangle$ and a BCQ $q$, we are interested in Boolean Conjunctive Query (BCQ) entailment under GA censors i.e., deciding whether $T \cup Th_c \models q$ for every optimal GA censor $c$ for $\mathcal{E}$.

Example (cntd):

$T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \ \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}$

$A = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{Diabetic}(bob), \ \text{TakesMedA}(ann)), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \}$

$P = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \ \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}$

Secrets$= \{ \{ \text{TakesMedA}(bob), \ \text{Diabetic}(bob) \}, \{ \text{TakesMedB}(bob), \ \text{Diabetic}(bob) \}, \{ \text{TakesMedA}(ann)), \ \text{Diabetic}(ann)\} \}$

$Th_{c1} = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{TakesMedA}(ann), \ \text{TakesMedA}(joe) \}$

$Th_{c2} = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \}$

$Th_{c3} = \{ \text{Diabetic}(bob), \ \text{TakesMedA}(ann), \ \text{TakesMedA}(joe) \}$

$Th_{c4} = \{ \text{Diabetic}(bob), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \}$

$\text{TakesMedA}(joe)$ and $\exists x, y. \text{admissionWard}(x, y)$ are both entailed by $\mathcal{E}$ under GA censors
Entailment of Boolean Conjunctive Queries (BCQs)

Major computational problem:

Skeptical entailment is **coNP-hard** in data complexity (i.e., w.r.t. the size of the ABox only) even for ontology languages/Description Logics of low expressiveness

e.g.:
- the Description Logics of the **DL-Lite** family (and thus **OWL2 QL**)
- the Description Logics of the **EL** family (and thus **OWL2 EL**)

Major computational problem:

Skeptical entailment is **coNP-hard** in data complexity (i.e., w.r.t. the size of the ABox only) even for ontology languages/Description Logics of **low expressiveness**

e.g.:
- the Description Logics of the **DL-Lite** family (and thus **OWL2 QL**)
- the Description Logics of the **EL** family (and thus **OWL2 EL**)

Entailment of Boolean Conjunctive Queries (BCQs)
Outline

1. The Controlled Query Evaluation approach in Ontologies and Description Logics
   - CQE in Description Logics through GA censors
   - Computational problems

2. Towards tractability 1: Intersecting the censors
   - IGA censors
   - Expressive limitations of IGA censors

3. Towards tractability 2: Adding preferences
   - Globally optimal and Pareto-optimal censors
   - DD and k-DD censors
   - Experimental results

4. Towards tractability 3: Maximally cooperative approach
   - The dynCQE approach
   - Complexity of dynCQE

5. Conclusions
Towards the identification of a practical approach, in [ISWC 2020] we introduced the notion of Intersection GA (IGA) censor

An **IGA censor** $\text{cens}_{\text{IGA}}$ for a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ is a function that returns the *intersection* $\text{Th}_{\text{cens}_{\text{IGA}}}$ of the theories of all optimal GA censors for $\mathcal{E}$.

**BCQ entailment under IGA censors**: deciding if $\mathcal{T} \cup \text{Th}_{\text{cens}_{\text{IGA}}} \models q$ for a BCQ $q$
From GA censors to IGA censor

Example (cont’d):
\[
\begin{align*}
\mathcal{T} &= \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \ \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \} \\
\mathcal{A} &= \{ \text{TakesMedA}(\text{bob}), \text{TakesMedB}(\text{bob}), \text{Diabetic}(\text{bob}), \text{TakesMedA}(\text{ann}), \\
& \quad \quad \text{Diabetic}(\text{ann}), \text{TakesMedA}(\text{joe}) \} \\
\mathcal{P} &= \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \ \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \} \\
\text{Th}_{c1} &= \{ \text{TakesMedA}(\text{bob}), \text{TakesMedB}(\text{bob}), \text{TakesMedA}(\text{ann}), \text{TakesMedA}(\text{joe}) \} \\
\text{Th}_{c2} &= \{ \text{TakesMedA}(\text{bob}), \text{TakesMedB}(\text{bob}), \text{Diabetic}(\text{ann}), \text{TakesMedA}(\text{joe}) \} \\
\text{Th}_{c3} &= \{ \text{Diabetic}(\text{bob}), \text{TakesMedA}(\text{ann}), \text{TakesMedA}(\text{joe}) \} \\
\text{Th}_{c4} &= \{ \text{Diabetic}(\text{bob}), \text{Diabetic}(\text{ann}), \text{TakesMedA}(\text{joe}) \} \\
\text{Th}_{\text{censIGA}} &= \{ \text{TakesMedA}(\text{joe}) \}.
\end{align*}
\]

\text{TakesMedA}(\text{joe}) \text{ is entailed by } \mathcal{E} \text{ under IGA censors but } \exists x, y. \text{admissionWard}(x, y) \text{ is not.}

BCQ Entailment is \textit{first-order (FO) rewritable}, and thus in \textit{AC}^0 \text{ in data complexity for } \textit{DL-Lite}_R \text{ and } \textit{OWL2QL} \text{[Cima et al.,ISWC 2020].}
From GA censors to IGA censor

Example (cont’d):

\[ T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard} ; \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \} \]

\[ A = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(bob), \text{TakesMedA}(ann) , \text{Diabetic}(ann), \text{TakesMedA}(joe) \} \]

\[ P = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot ; \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \} \]

\[ \text{Th}_{c1} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{TakesMedA}(ann), \text{TakesMedA}(joe) \} \]

\[ \text{Th}_{c2} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(ann), \text{TakesMedA}(joe) \} \]

\[ \text{Th}_{c3} = \{ \text{Diabetic}(bob), \text{TakesMedA}(ann), \text{TakesMedA}(joe) \} \]

\[ \text{Th}_{c4} = \{ \text{Diabetic}(bob), \text{Diabetic}(ann), \text{TakesMedA}(joe) \} \]

\[ \text{Th}_{cens_{IGA}} = \{ \text{TakesMedA}(joe) \}. \]

\[ \text{TakesMedA}(joe) \text{ is entailed by } E \text{ under IGA censors but } \exists x, y. \text{admissionWard}(x, y) \text{ is not.} \]

BCQ Entailment is first-order (FO) rewritable, and thus in AC^0 in data complexity for DL-Lite_R and OWL2QL [Cima et al., ISWC 2020].
Expressive limitations of IGA censors

From GA censors to IGA censor

Example (cont’d):

\[ T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \} \]

\[ A = \{ \text{TakesMedA(bob)}, \text{TakesMedB(bob)}, \text{Diabetic(bob), TakesMedA(ann)}), \]
\[ \text{Diabetic(ann), TakesMedA(joe)} \} \]

\[ P = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \} \]

\[ \text{Th}_{c1} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{TakesMedA}(ann), \text{TakesMedA}(joe) \} \]

\[ \text{Th}_{c2} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(ann), \text{TakesMedA}(joe) \} \]

\[ \text{Th}_{c3} = \{ \text{Diabetic}(bob), \text{TakesMedA}(ann), \text{TakesMedA}(joe) \} \]

\[ \text{Th}_{c4} = \{ \text{Diabetic}(bob), \text{Diabetic}(ann), \text{TakesMedA}(joe) \} \]

\[ \text{Th}_{\text{cens}_{IGA}} = \{ \text{TakesMedA}(joe) \}. \]

TakesMedA(joe) is entailed by \( E \) under IGA censors but \( \exists x, y. \text{admissionWard}(x, y) \) is not.

BCQ Entailment is first-order (FO) rewritable, and thus in \( AC^0 \) in data complexity for \( DL-Lite_R \) and OWL2QL [Cima et al., ISWC 2020].
What does \textbf{first-order rewritable} mean?

for every BCQ $Q$, we can build a first-order (e.g. SQL) query $Q'$ such that the evaluation of $Q'$ on the \textbf{initial knowledge base} is true iff the query $Q$ is true in the IGA censor

Consequences:

- no need to modify the knowledge base (ABox)
- no need to modify the query answering system
IGA censor: properties

- good computational properties
- deterministic, unique solution
- but non-optimal in terms of disclosed information
- however, "quasi-optimal" ([IJCAI 2020])
1. The Controlled Query Evaluation approach in Ontologies and Description Logics
   - CQE in Description Logics through GA censors
   - Computational problems

2. Towards tractability 1: Intersecting the censors
   - IGA censors
   - Expressive limitations of IGA censors

3. Towards tractability 2: Adding preferences
   - Globally optimal and Pareto-optimal censors
   - DD and k-DD censors
   - Experimental results

4. Towards tractability 3: Maximally cooperative approach
   - The dynCQE approach
   - Complexity of dynCQE

5. Conclusions
Limitations of the framework seen so far

Two main limitations arise:

- **the policy allows only for CQs** in denial assertions, thus ruling out practically relevant formulas, e.g.,
  \[
  \forall x, y. \text{Diabetic}(x) \land \text{admissionWard}(x, y) \land y \neq \text{'orthopedics'} \rightarrow \bot
  \]

- **censors** studied so far **do not take into account** possibly available **preferences** about the way in which secret information has to be censored

In [ISWC 2021], we have enriched the framework and considered:

1. CQs using **comparison predicates** (i.e., ≠, ≥, ≤, …) in policy assertions
2. priorities between ontology predicates, specifying (intentionally) **preferences in disclosing facts** involved in a secret
Limitations of the framework seen so far

Two main limitations arise:

- **the policy allows only for CQs** in denial assertions, thus ruling out practically relevant formulas, e.g.,
  \[ \forall x, y. \text{Diabetic}(x) \land \text{admissionWard}(x, y) \land y \neq 'orthopedics' \rightarrow \bot \]

- **censors** studied so far **do not take into account** possibly available **preferences** about the way in which secret information has to be censored

In *ISWC 2021*, we have enriched the framework and considered:

1. **CQs using comparison predicates** (i.e., \(\neq, \geq, \leq, \ldots\)) in policy assertions
2. **priorities between ontology predicates**, specifying (intentionally) **preferences in disclosing facts** involved in a secret
A prioritized CQE instance $\mathcal{E}_{\succ}$ is a tuple $\langle T, A, P, \succ \rangle$ such that $\langle T, A, P \rangle$ is a CQE instance and $\succ$ is an acyclic priority relation over ontology predicates.

If $P_1 \succ P_2$, e.g., TakesMedA $\succ$ Diabetic, in case a secret involves facts over $P_1$ and $P_2$, we prefer to disclose those over $P_1$ and hide those over $P_2$.

Desiderata:

1. new notion of censor taking in the due account the preference relation and that in case $\succ = \emptyset$ coincides with the notion of GA censor.
2. definition allowing for practical CQE over prioritized ontologies, i.e., for which BCQ entailment is FO rewritable for the DLs of the DL-Lite family.
3. experimental validation: using preferences may increase the amount of data disclosed to users (still preserving confidential information).
Prioritized CQE framework

- A prioritized CQE instance $\mathcal{E}_{\succ}$ is a tuple $\langle T, A, P, \succ \rangle$ such that $\langle T, A, P \rangle$ is a CQE instance and $\succ$ is an \textit{acyclic priority relation over ontology predicates}.

- If $P_1 \succ P_2$, e.g., TakesMedA $\succ$ Diabetic, in case a secret involves facts over $P_1$ and $P_2$, we prefer to disclose those over $P_1$ and hide those over $P_2$.

Desiderata:

- new notion of censor taking in the due account the preference relation and that in case $\succ = \emptyset$ \textit{coincides with the notion of GA censor}.

- definition allowing for \textit{practical CQE} over prioritized ontologies, i.e., for which BCQ entailment is \textit{FO rewritable} for the DLs of the \textit{DL-Lite} family.

- experimental validation: using \textit{preferences may increase the amount of data disclosed} to users (still preserving confidential information).
Globally-optimal and Pareto-optimal censors

- We initially adapted to our framework the well-known Pareto and Global optimality notions introduced by [Staworko et al., AMAI 2012] for consistent query answering (CQA) over databases, and then adopted in [Bienvenu and Bourgaux, KR 2020] for CQA over prioritized DL ontologies.

- We first defined Pareto (P) and Global (G) censors, and then their approximated version based on intersection, i.e., Intersection P (IP) and Intersection G (IG) censors.

- For $\succ = \emptyset$, the sets of Pareto, Global and optimal GA censors coincide, and the IP, IG and IGA censor coincide.

- Data Complexity of BCQ entailment for DL-Lite:
  - under P and IP censors is coNP-hard
  - under G and IG censor is $\Pi^p_2$-hard

  (follow from [Bienvenu and Bourgaux, KR 2020])
Relationship between censors
(and data complexity of BCQ entailment for DL-Lite)

- $G$ censors are $\Pi^p_2$-hard
- $P$ censors are coNP-hard
- $IG$ censors are $\Pi^p_2$-hard
- $IP$ censors are coNP-hard
- $GA$ censors are coNP-hard
- $IGA$ censors are in $AC^0$

An arrow from $X$ to $Y$ indicates that $X$ is a **sound approximation** of $Y$, i.e., what is entailed under $X$ is entailed also under $Y$.
Looking for a practical notion of preference-based censor

An arrow from $X$ to $Y$ indicates that $X$ is a sound approximation of $Y$, i.e., what is entailed under $X$ is entailed also under $Y$. 

Riccardo Rosati
Knowledge-based Confidentiality-preserving Query Answering
WEBIST 2022
DD-censors

Given \( E \succ \langle T, A, P, \succ \rangle \) we define \( DD_0(E \succ) = DC_0(E \succ) = \emptyset \), and

\[
DD_{i+1}(E \succ) = \{ \alpha \in \text{cl}_{GA}(A) \mid \text{for each secret } S \text{ s.t. } \alpha \in S \text{ there is } \beta \in S \text{ s.t. } \\
\beta \neq \alpha \text{ and either } \alpha \succ \beta \text{ or } \beta \in DC_i(E \succ) \}
\]

\[
DC_{i+1}(E \succ) = \{ \alpha \in \text{cl}_{GA}(A) \mid \text{there is a secret } S \text{ s.t. } S \setminus DD_i(E \succ) = \{ \alpha \} \}
\]

A Deterministically Disclosed (DD) censor for \( E \succ \) returns the least fix point \( DD(E \succ) \) for \( DD_i(E \succ) \), which always exists and it is unique.

Example (cntd)

\[
T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}
\]

\[
A = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(bob), \text{TakesMedA}(ann)), \text{Diabetic}(ann), \text{TakesMedA}(joe) \}
\]

\[
P = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}
\]

\[
\text{Secrets} = \{ \{ \text{TakesMedA}(bob), \text{Diabetic}(bob) \}, \{ \text{TakesMedB}(bob), \text{Diabetic}(bob) \}, \{ \text{TakesMedA}(ann)), \text{Diabetic}(ann) \} \}
\]

\[
\text{Priorities: \text{TakesMedB} \succ \text{Diabetic}}
\]

\[
DD_1(E \succ) = \{ \text{TakesMedA}(joe), \text{TakesMedB}(bob) \} \quad DC_1(E \succ) = \emptyset
\]

\[
DD_2(E \succ) = DD_1(E \succ) \quad DC_2(E \succ) = \{ \text{Diabetic}(bob) \}
\]

\[
DD_3(E \succ) = DD_2(E \succ) \cup \{ \text{TakesMedA}(bob) \} \quad DC_3(E \succ) = DC_2(E \succ) \quad DC_3(E \succ) = DC_2(E \succ)
\]

\[
DD_4(E \succ) = DD_3(E \succ) \quad DC_4(E \succ) = DC_3(E \succ)
\]
**DD-censors**

Given $\mathcal{E}_\succ = \langle \mathcal{T}, \mathcal{A}, \mathcal{P}, \succ \rangle$ we define $DD_0(\mathcal{E}_\succ) = DC_0(\mathcal{E}_\succ) = \emptyset$, and

$$DD_{i+1}(\mathcal{E}_\succ) = \{ \alpha \in cl^{T\mathcal{A}}(\mathcal{A}) \mid \text{for each secret } S \text{ s.t. } \alpha \in S \text{ there is } \beta \in S \text{ s.t. } \beta \neq \alpha \text{ and either } \alpha \succ \beta \text{ or } \beta \in DC_i(\mathcal{E}_\succ) \}$$

$$DC_{i+1}(\mathcal{E}_\succ) = \{ \alpha \in cl^{T\mathcal{A}}(\mathcal{A}) \mid \text{there is a secret } S \text{ s.t. } S \setminus DD_i(\mathcal{E}_\succ) = \{ \alpha \} \}$$

A **Deterministically Disclosed (DD)** censor for $\mathcal{E}_\succ$ returns the **least fix point** $DD(\mathcal{E}_\succ)$ for $DD_i(\mathcal{E}_\succ)$, which always exists and it is unique.

**Example (cntd)**

- $\mathcal{T} = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \ \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}$
- $\mathcal{A} = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{Diabetic}(bob), \ \text{TakesMedA}(ann)), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \}$
- $\mathcal{P} = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \ \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}$
- $\text{Secrets} = \{ \{ \text{TakesMedA}(bob), \ \text{Diabetic}(bob) \}, \ { \text{TakesMedB}(bob), \ \text{Diabetic}(bob) \}, \ { \text{TakesMedA}(ann)), \ \text{Diabetic}(ann) \}$
- **Priorities:** TakesMedB $\succ$ Diabetic

<table>
<thead>
<tr>
<th>$DD_1(\mathcal{E}_\succ)$</th>
<th>$DD_2(\mathcal{E}_\succ)$</th>
<th>$DD_3(\mathcal{E}_\succ)$</th>
<th>$DD_4(\mathcal{E}_\succ)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${ \text{TakesMedA}(joe), \ \text{TakesMedB}(bob) }$</td>
<td>$DD_1(\mathcal{E}_\succ)$</td>
<td>$DD_2(\mathcal{E}_\succ)$</td>
<td>$DD_3(\mathcal{E}_\succ)$</td>
</tr>
<tr>
<td>$DC_1(\mathcal{E}_\succ) = \emptyset$</td>
<td>$DC_2(\mathcal{E}_\succ) = { \text{Diabetic}(bob) }$</td>
<td>$DC_3(\mathcal{E}<em>\succ) = DC_2(\mathcal{E}</em>\succ)$</td>
<td>$DC_4(\mathcal{E}<em>\succ) = DC_3(\mathcal{E}</em>\succ)$</td>
</tr>
</tbody>
</table>
DD-censors

Given $\mathcal{E}_\succ = \langle \mathcal{T}, \mathcal{A}, \mathcal{P}, \succ \rangle$ we define $DD_0(\mathcal{E}_\succ) = DC_0(\mathcal{E}_\succ) = \emptyset$, and

\[
DD_{i+1}(\mathcal{E}_\succ) = \{ \alpha \in \operatorname{cl}_{\mathcal{T}\mathcal{A}}(\mathcal{A}) \mid \text{for each secret } S \text{ s.t. } \alpha \in S \text{ there is } \beta \in S \text{ s.t. } \\
\beta \neq \alpha \text{ and either } \alpha \succ \beta \text{ or } \beta \in DC_i(\mathcal{E}_\succ) \}
\]

\[
DC_{i+1}(\mathcal{E}_\succ) = \{ \alpha \in \operatorname{cl}_{\mathcal{T}\mathcal{A}}(\mathcal{A}) \mid \text{there is a secret } S \text{ s.t. } S \setminus DD_i(\mathcal{E}_\succ) = \{ \alpha \} \}
\]

A Deterministically Disclosed (DD) censor for $\mathcal{E}_\succ$ returns the least fix point $DD(\mathcal{E}_\succ)$ for $DD_i(\mathcal{E}_\succ)$, which always exists and it is unique.

Example (cntd)

\[
\mathcal{T} = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}
\]

\[
\mathcal{A} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(bob), \text{TakesMedA}(ann)), \text{Diabetic}(ann), \text{TakesMedA}(joe) \}
\]

\[
\mathcal{P} = \{ \forall x.\text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \forall x.\text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}
\]

Secrets = \{ \{ \text{TakesMedA}(bob), \text{Diabetic}(bob) \}, \{ \text{TakesMedB}(bob), \text{Diabetic}(bob) \}, \{ \text{TakesMedA}(ann)), \text{Diabetic}(ann) \} \}

Priorities: TakesMedB $\succ$ Diabetic

\[
DD_1(\mathcal{E}_\succ) = \{ \text{TakesMedA}(joe), \text{TakesMedB}(bob) \} \quad DC_1(\mathcal{E}_\succ) = \emptyset
\]

\[
DD_2(\mathcal{E}_\succ) = DD_1(\mathcal{E}_\succ) 
DD_3(\mathcal{E}_\succ) = DD_2(\mathcal{E}_\succ) \cup \{ \text{TakesMedA}(bob) \}
DD_4(\mathcal{E}_\succ) = DD_3(\mathcal{E}_\succ)
\]

\[
DC_2(\mathcal{E}_\succ) = \{ \text{Diabetic}(bob) \}
DC_3(\mathcal{E}_\succ) = DC_2(\mathcal{E}_\succ)
DC_4(\mathcal{E}_\succ) = DC_3(\mathcal{E}_\succ)
\]
Given $\mathcal{E}_\succ = \langle \mathcal{T}, \mathcal{A}, \mathcal{P}, \succ \rangle$ we define $DD_0(\mathcal{E}_\succ) = DC_0(\mathcal{E}_\succ) = \emptyset$, and

$$DD_{i+1}(\mathcal{E}_\succ) = \{\alpha \in \text{cl}_{GA}(\mathcal{A}) \mid \text{for each secret } S \text{ s.t. } \alpha \in S \text{ there is } \beta \in S \text{ s.t. } \beta \neq \alpha \text{ and either } \alpha \succ \beta \text{ or } \beta \in DC_i(\mathcal{E}_\succ)\}$$

$$DC_{i+1}(\mathcal{E}_\succ) = \{\alpha \in \text{cl}_{GA}(\mathcal{A}) \mid \text{there is a secret } S \text{ s.t. } S \setminus DD_i(\mathcal{E}_\succ) = \{\alpha\}\}$$

A Deterministically Disclosed (DD) censor for $\mathcal{E}_\succ$ returns the least fix point $DD(\mathcal{E}_\succ)$ for $DD_i(\mathcal{E}_\succ)$, which always exists and it is unique.

**Example (cntd)**

\[\mathcal{T} = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \quad \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}\]

\[\mathcal{A} = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(bob), \text{TakesMedA}(ann)), \text{Diabetic}(ann), \text{TakesMedA}(joe) \}\]

\[\mathcal{P} = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \quad \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}\]

**Secrets** = \{\{ \text{TakesMedA}(bob), \text{Diabetic}(bob) \}, \{ \text{TakesMedB}(bob), \text{Diabetic}(bob) \}, \{ \text{TakesMedA}(ann)), \text{Diabetic}(ann) \}\}

**Priorities**: TakesMedB $\succ$ Diabetic

\[DD_1(\mathcal{E}_\succ) = \{\text{TakesMedA}(joe), \text{TakesMedB}(bob)\} \quad DC_1(\mathcal{E}_\succ) = \emptyset\]
\[DD_2(\mathcal{E}_\succ) = DD_1(\mathcal{E}_\succ) \quad DC_2(\mathcal{E}_\succ) = \{\text{Diabetic}(bob)\}\]
\[DD_3(\mathcal{E}_\succ) = DD_2(\mathcal{E}_\succ) \cup \{\text{TakesMedA}(bob)\} \quad DC_3(\mathcal{E}_\succ) = DC_2(\mathcal{E}_\succ)\]
\[DD_4(\mathcal{E}_\succ) = DD_3(\mathcal{E}_\succ) \quad DC_4(\mathcal{E}_\succ) = DC_3(\mathcal{E}_\succ)\]
Given $\mathcal{E}_\succ = \langle T, A, P, \succ \rangle$ we define $DD_0(\mathcal{E}_\succ) = DC_0(\mathcal{E}_\succ) = \emptyset$, and

$$DD_{i+1}(\mathcal{E}_\succ) = \{ \alpha \in \text{cl}_T^A(A) | \text{ for each secret } S \text{ s.t. } \alpha \in S \text{ there is } \beta \in S \text{ s.t. } \beta \neq \alpha \text{ and either } \alpha \succ \beta \text{ or } \beta \in DC_i(\mathcal{E}_\succ) \}$$

$$DC_{i+1}(\mathcal{E}_\succ) = \{ \alpha \in \text{cl}_T^A(A) | \text{ there is a secret } S \text{ s.t. } S \setminus DD_i(\mathcal{E}_\succ) = \{ \alpha \} \}$$

A Deterministically Disclosed (DD) censor for $\mathcal{E}_\succ$ returns the least fix point $DD(\mathcal{E}_\succ)$ for $DD_i(\mathcal{E}_\succ)$, which always exists and it is unique.

Example (cntd)

$T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \ \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \}$

$A = \{ \text{TakesMedA}(bob), \ \text{TakesMedB}(bob), \ \text{Diabetic}(bob), \ \text{TakesMedA}(ann)), \ \text{Diabetic}(ann), \ \text{TakesMedA}(joe) \}$

$P = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \ \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \}$

$\text{Secrets}= \{ \{ \text{TakesMedA}(bob), \ \text{Diabetic}(bob) \}, \{ \text{TakesMedB}(bob), \ \text{Diabetic}(bob) \}, \{ \text{TakesMedA}(ann)), \ \text{Diabetic}(ann) \} \}$

$\text{Priorities}: \ \text{TakesMedB} \succ \text{Diabetic}$

$DD_1(\mathcal{E}_\succ) = \{ \text{TakesMedA}(joe), \ \text{TakesMedB}(bob) \}$

$DD_2(\mathcal{E}_\succ) = DD_1(\mathcal{E}_\succ)$

$DD_3(\mathcal{E}_\succ) = DD_2(\mathcal{E}_\succ) \cup \{ \text{TakesMedA}(bob) \}$

$DD_4(\mathcal{E}_\succ) = DD_3(\mathcal{E}_\succ)$

$DC_1(\mathcal{E}_\succ) = \emptyset$

$DC_2(\mathcal{E}_\succ) = \{ \text{Diabetic}(bob) \}$

$DC_3(\mathcal{E}_\succ) = DC_2(\mathcal{E}_\succ)$

$DC_4(\mathcal{E}_\succ) = DC_3(\mathcal{E}_\succ)$
DD-censors

Given $\mathcal{E}_\succ = \langle T, A, P, \succ \rangle$ we define $DD_0(\mathcal{E}_\succ) = DC_0(\mathcal{E}_\succ) = \emptyset$, and

\[
DD_{i+1}(\mathcal{E}_\succ) = \{\alpha \in \text{cl}_{GA}(T) \mid \text{for each secret } S \text{ s.t. } \alpha \in S \text{ there is } \beta \in S \text{ s.t. } \beta \neq \alpha \text{ and either } \alpha \succ \beta \text{ or } \beta \in DC_i(\mathcal{E}_\succ)\}
\]

\[
DC_{i+1}(\mathcal{E}_\succ) = \{\alpha \in \text{cl}_{GA}(T) \mid \text{there is a secret } S \text{ s.t. } S \setminus DD_i(\mathcal{E}_\succ) = \{\alpha\}\}
\]

A **Deterministically Disclosed (DD)** censor for $\mathcal{E}_\succ$ returns the least fix point $DD(\mathcal{E}_\succ)$ for $DD_i(\mathcal{E}_\succ)$, which always exists and it is unique.

**Example (cntd)**

\[T = \{\text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \quad \text{Diabetic} \sqsubseteq \exists \text{admissionWard}\}\]

\[A = \{\text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(bob), \text{TakesMedA}(ann)), \text{Diabetic}(ann), \text{TakesMedA}(joe)\}\]

\[P = \{\forall x.\text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \quad \forall x.\text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot\}\]

**Secrets** = \{\{\text{TakesMedA}(bob), \text{Diabetic}(bob)\}, \{\text{TakesMedB}(bob), \text{Diabetic}(bob)\}, \{\text{TakesMedA}(ann)), \text{Diabetic}(ann)\}\}

**Priorities:** TakesMedB $\succ$ Diabetic

\[
DD_1(\mathcal{E}_\succ) = \{\text{TakesMedA}(joe), \text{TakesMedB}(bob)\} \quad DC_1(\mathcal{E}_\succ) = \emptyset
\]

\[
DD_2(\mathcal{E}_\succ) = DD_1(\mathcal{E}_\succ) \quad DC_2(\mathcal{E}_\succ) = \{\text{Diabetic}(bob)\}
\]

\[
DD_3(\mathcal{E}_\succ) = DD_2(\mathcal{E}_\succ) \cup \{\text{TakesMedA}(bob)\} \quad DC_3(\mathcal{E}_\succ) = DC_2(\mathcal{E}_\succ)
\]

\[
DD_4(\mathcal{E}_\succ) = DD_3(\mathcal{E}_\succ) \quad DC_4(\mathcal{E}_\succ) = DC_3(\mathcal{E}_\succ)
\]
from DD-censors to \( k\)-DD-censors

**BCQ entailment under DD censors**, i.e., deciding whether \( T \cup DD(\mathcal{E}_\succ) \models q \) for a BCQ \( q \), is **PTIME-hard** in data complexity for **DL-Lite**

By fixing a \( k \), we get \( DD_k(\mathcal{E}_\succ) \), which we call \( k\)-DD censor for \( \mathcal{E}_\succ \).

Give a positive integer \( k \), **BCQ entailment under \( k\)-DD censors**, i.e., deciding whether \( T \cup DD_k(\mathcal{E}_\succ) \models q \) for a BCQ \( q \), is **FO rewritable** and thus in **AC\(^0\)** in data complexity for **DL-Lite\(_A\)**

**Example (cntd)**

\[
T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \ \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \} \\
A = \{ \text{TakesMedA}(\text{bob}), \ \text{TakesMedB}(\text{bob}), \ \text{Diabetic}(\text{bob}), \ \text{TakesMedA}(\text{ann}), \ \text{Diabetic}(\text{ann}), \ \text{TakesMedA}(\text{joe}) \} \\
\mathcal{P} = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \ \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \} \\
\text{Priorities: } \text{TakesMedB} \succ \text{Diabetic}
\]

for \( k = 3 \) we have \( DD_k(\mathcal{E}_\succ) = \{ \text{TakesMedA}(\text{joe}), \ \text{TakesMedB}(\text{bob}), \ \text{TakesMedA}(\text{bob}) \} \)

Given the CQ \( q(x) : \neg \text{TakesMedA}(x) \) and \( k = 3 \), \( q(\vec{x}) \) is reformulated in the FO query \( q'(x) : \neg \text{TakesMedA}(x) \land (\neg \text{Diabetic}(x) \lor \text{TakesMedB}(x)) \). The evaluation of \( q' \) over \( A \) returns \{bob, joe\}, which is the set of answers under \( k\)-DD-censor for \( k = 3 \) (in this specific case this holds for any \( k \geq 3 \)).
from DD-censors to $k$-DD-censors

**BCQ entailment under DD censors**, i.e., deciding whether $\mathcal{T} \cup DD(\mathcal{E}_{\succ}) \models q$ for a BCQ $q$, is **PTIME-hard** in data complexity for **DL-Lite**

By fixing a $k$, we get $DD_k(\mathcal{E}_{\succ})$, which we call $k$-DD censor for $\mathcal{E}_{\succ}$.

Give a positive integer $k$, **BCQ entailment under $k$-DD censors**, i.e., deciding whether $\mathcal{T} \cup DD_k(\mathcal{E}_{\succ}) \models q$ for a BCQ $q$, is **FO rewritable and thus in $AC^0$** in data complexity for **DL-Lite$_A$**

**Example (cntd)**

$\mathcal{T} = \{\text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \text{Diabetic} \sqsubseteq \exists \text{admissionWard}\}$

$\mathcal{A} = \{\text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(bob), \text{TakesMedA}(ann)), \\
\text{Diabetic}(ann), \text{TakesMedA}(joe)\}$

$\mathcal{P} = \{\forall x.\text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \forall x.\text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot\}$

**Priorities:** TakesMedB$\succ$Diabetic

for $k = 3$ we have $DD_k(\mathcal{E}_{\succ}) = \{\text{TakesMedA}(joe), \text{TakesMedB}(bob), \text{TakesMedA}(bob)\}$

Given the CQ $q(x) : \neg \text{TakesMedA}(x)$ and $k = 3$, $q(\vec{x})$ is reformulated in the FO query

$q'(x) : \neg \text{TakesMedA}(x) \land (\neg \text{Diabetic}(x) \lor \text{TakesMedB}(x))$. The evaluation of $q'$ over $\mathcal{A}$ returns $\{bob, joe\}$, which is the set of answers under $k$-DD-censor for $k = 3$ (in this specific case this holds for any $k \geq 3$).
from DD-censors to \( k \)-DD-censors

**BCQ entailment under DD censors**, i.e., deciding whether \( T \cup DD(\mathcal{E}_{\succ}) \models q \) for a BCQ \( q \), is **PTIME-hard** in data complexity for **DL-Lite**

By fixing a \( k \), we get \( DD_k(\mathcal{E}_{\succ}) \), which we call \( k \)-DD censor for \( \mathcal{E}_{\succ} \).

Give a positive integer \( k \), **BCQ entailment under \( k \)-DD censors**, i.e., deciding whether \( T \cup DD_k(\mathcal{E}_{\succ}) \models q \) for a BCQ \( q \), is **FO rewritable** and thus in **AC^0** in data complexity for **DL-Lite_A**

**Example (cntd)**

\( T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \} \)

\( A = \{ \text{TakesMedA}(bob), \text{TakesMedB}(bob), \text{Diabetic}(bob), \text{TakesMedA}(ann)), \text{Diabetic}(ann), \text{TakesMedA}(joe) \} \)

\( P = \{ \forall x. \text{TakesMedA}(x) \land \text{Diabetic}(x) \to \bot; \forall x. \text{TakesMedB}(x) \land \text{Diabetic}(x) \to \bot \} \)

**Priorities:** TakesMedB\( \succ \)Diabetic

for \( k = 3 \) we have \( DD_k(\mathcal{E}_{\succ}) = \{ \text{TakesMedA}(joe), \text{TakesMedB}(bob), \text{TakesMedA}(bob) \} \)

Given the CQ \( q(x) : \neg \text{TakesMedA}(x) \) and \( k = 3 \), \( q(\vec{x}) \) is reformulated in the FO query \( q'(x) : \neg \text{TakesMedA}(x) \land (\neg \text{Diabetic}(x) \lor \text{TakesMedB}(x)) \). The evaluation of \( q' \) over \( A \) returns \{bob, joe\}, which is the set of answers under \( k \)-DD-censor for \( k = 3 \) (in this specific case this holds for any \( k \geq 3 \)).
BCQ entailment under DD censors, i.e., deciding whether \( T \cup DD(\mathcal{E}_{\succ}) \models q \) for a BCQ \( q \), is \textbf{PTIME-hard} in data complexity for \textit{DL-Lite}.

By fixing a \( k \), we get \( DD_k(\mathcal{E}_{\succ}) \), which we call \( k \)-DD censor for \( \mathcal{E}_{\succ} \).

Give a positive integer \( k \), BCQ entailment under \( k \)-DD censors, i.e., deciding whether \( T \cup DD_k(\mathcal{E}_{\succ}) \models q \) for a BCQ \( q \), is \textbf{FO} rewritable and thus in \( AC^0 \) in data complexity for \textit{DL-Lite}_A.

\textbf{Example (cntd)}

\( T = \{ \text{TakesMedB} \sqsubseteq \exists \text{admissionWard}; \ \text{Diabetic} \sqsubseteq \exists \text{admissionWard} \} \)

\( A = \{ \text{TakesMedA}(\text{bob}), \ \text{TakesMedB}(\text{bob}), \ \text{Diabetic}(\text{bob}), \ \text{TakesMedA}(\text{ann})), \ \text{Diabetic}(\text{ann}), \ \text{TakesMedA}(\text{joe}) \} \)

\( P = \{ \forall x.\text{TakesMedA}(x) \land \text{Diabetic}(x) \rightarrow \bot; \ \forall x.\text{TakesMedB}(x) \land \text{Diabetic}(x) \rightarrow \bot \} \)

\textbf{Priorities:} \text{TakesMedB } \succ \text{Diabetic}

for \( k = 3 \) we have \( DD_k(\mathcal{E}_{\succ}) = \{ \text{TakesMedA}(\text{joe}), \ \text{TakesMedB}(\text{bob}), \ \text{TakesMedA}(\text{bob}) \} \)

Given the CQ \( q(x) : \neg \text{TakesMedA}(x) \) and \( k = 3 \), \( q(\vec{x}) \) is reformulated in the FO query \( q'(x) : \neg \text{TakesMedA}(x) \land \neg \text{Diabetic}(x) \lor \text{TakesMedB}(x) \). The evaluation of \( q' \) over \( A \) returns \{\text{bob, joe}\}, which is the set of answers under \( k \)-DD-censor for \( k = 3 \) (in this specific case this holds for any \( k \geq 3 \)).
An arrow from $X$ to $Y$ indicates that $X$ is a sound approximation of $Y$, i.e., what is entailed under $X$ is entailed also under $Y$. 
Experiments

- We implemented our technique and tested it over the NPD benchmark [Lanti et al., EDBT 2015] (approximated to DL-Lite$_A$).

- We specified a policy $\mathcal{P}$ with 6 denials, a priority relation $\succ$ specifying 6 priorities, and selected 9 queries from the benchmark.

- We executed each query in six settings: the first with empty policy and priority relation ($\emptyset$, $\emptyset$), the second with policy $\mathcal{P}$ and no priorities ($\mathcal{P}$, $\emptyset$), the others with policy $\mathcal{P}$, priorities in $\succ$, and $k = 1, 3, 5, 7$.

- For $k \leq 3$, time is only slightly affected, and for 6 out of 9 queries we get already the same answers we obtain for $k > 3$. For $k > 3$, the gain in the number of answers is limited, and for 4 queries performances decrease.

<table>
<thead>
<tr>
<th>Setting</th>
<th>$q_3$</th>
<th>$q_4$</th>
<th>$q_5$</th>
<th>$q_9$</th>
<th>$q_{12}$</th>
<th>$q_{13}$</th>
<th>$q_{14}$</th>
<th>$q_{18}$</th>
<th>$q_{44}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#</td>
<td>time</td>
<td>#</td>
<td>time</td>
<td>#</td>
<td>time</td>
<td>#</td>
<td>time</td>
<td>#</td>
</tr>
<tr>
<td>$\emptyset, \emptyset$</td>
<td>910</td>
<td>207</td>
<td>1558</td>
<td>168</td>
<td>17254</td>
<td>585</td>
<td>1566</td>
<td>320</td>
<td>96671</td>
</tr>
<tr>
<td>$\mathcal{P}, \emptyset$</td>
<td>910</td>
<td>278</td>
<td>252</td>
<td>295</td>
<td>14797</td>
<td>825</td>
<td>416</td>
<td>331</td>
<td>13028</td>
</tr>
<tr>
<td>$\mathcal{P}, \succ, 1$</td>
<td>910</td>
<td>221</td>
<td>252</td>
<td>179</td>
<td>17254</td>
<td>612</td>
<td>416</td>
<td>216</td>
<td>96671</td>
</tr>
<tr>
<td>$\mathcal{P}, \succ, 3$</td>
<td>910</td>
<td>249</td>
<td>521</td>
<td>1445</td>
<td>17254</td>
<td>749</td>
<td>1252</td>
<td>1148</td>
<td>96671</td>
</tr>
<tr>
<td>$\mathcal{P}, \succ, 5$</td>
<td>910</td>
<td>242</td>
<td>566</td>
<td>8942</td>
<td>17254</td>
<td>723</td>
<td>1456</td>
<td>7715</td>
<td>96671</td>
</tr>
<tr>
<td>$\mathcal{P}, \succ, 7$</td>
<td>910</td>
<td>472</td>
<td>—</td>
<td>t.o.</td>
<td>17254</td>
<td>993</td>
<td>—</td>
<td>t.o.</td>
<td>96671</td>
</tr>
</tbody>
</table>
Our experiments show the applicability of our technique in practice.

Our results hold also for general CQs (i.e., non-Boolean ones).

Our FO rewritability result holds for CQs with comparison predicates (provided that the $DL-Lite_A$ TBox satisfies a safeness condition).

But: $k$-DD censor is only a sound approximation of the real (intractable) semantics of preferences.
The Controlled Query Evaluation approach in Ontologies and Description Logics
- CQE in Description Logics through GA censors
- Computational problems

Towards tractability 1: Intersecting the censors
- IGA censors
- Expressive limitations of IGA censors

Towards tractability 2: Adding preferences
- Globally optimal and Pareto-optimal censors
- DD and k-DD censors
- Experimental results

Towards tractability 3: Maximally cooperative approach
- The dynCQE approach
- Complexity of dynCQE

Conclusions
In [ISWC 2022] we have proposed a "dynamic" approach to CQE called **dynCQE**

In dynCQE, the censors are progressively selected according to the user queries.

We prove that this is the only possible approach that satisfies the "**longest-honeymoon**" property (be honest as long as you can).

Such a cooperative behavior:
- takes into account the user’s interests
- allows for revealing more information than all the previous approaches
Example (cont’d)

- $\mathcal{T} = \emptyset$, $\mathcal{A} = \{ A(o), B(o) \}$, $\mathcal{P} = \{ \exists x (A(x) \land B(x)) \rightarrow \bot \}$
- $\text{OptCens}(\mathcal{E}) = \{ \{A(o)\}, \{B(o)\} \}$
- Given $q = A(o)$, we have that:
  - $\mathcal{T} \cup \{A(o)\} \models q$ (honest)
  - $\mathcal{T} \cup \{B(o)\} \not\models q$ (liar)
- No good reason to lie, but we have to “remember” that $q$ was asked!
Example

$P = \{\exists x, y (\text{Buy}(x, y) \land \text{Antiseizure}(y)) \rightarrow \bot, \exists x, y (\text{Buy}(x, y) \land \text{Contains}(y, \text{phenytoin})) \rightarrow \bot\}$

$T = \{\text{DrugA} \sqsubseteq \text{Antiseizure}\}$

$A = \{\text{Buy(ann, d}_1), \text{DrugA(d}_1), \text{Buy(bob, d}_2), \text{Contains(d}_2, \text{phenytoin})\}$
Example

\[ \mathcal{P} = \{ \exists x, y (\text{Buy}(x, y) \land \text{Antiseizure}(y)) \rightarrow \perp, \]
\[ \exists x, y (\text{Buy}(x, y) \land \text{Contains}(y, \text{phenytoin})) \rightarrow \perp \} \]

\[ \mathcal{T} = \{ \text{DrugA} \sqsubseteq \text{Antiseizure} \} \]

\[ \mathcal{A} = \{ \text{Buy}(\text{ann}, d_1), \text{DrugA}(d_1), \text{Buy}(\text{bob}, d_2), \text{Contains}(d_2, \text{phenytoin}) \} \]

\[ \mathcal{C}_1 = \{ \text{Buy}(\text{ann}, d_1), \text{Buy}(\text{bob}, d_2) \} \]

\[ \mathcal{C}_2 = \{ \text{Buy}(\text{ann}, d_1), \text{Contains}(d_2, \text{phenytoin}) \} \]

\[ \mathcal{C}_3 = \{ \text{DrugA}(d_1), \text{Antiseizure}(d_1), \text{Buy}(\text{bob}, d_2) \} \]

\[ \mathcal{C}_4 = \{ \text{DrugA}(d_1), \text{Antiseizure}(d_1), \text{Contains}(d_2, \text{phenytoin}) \} \]
Example

\[ \mathcal{P} = \{ \exists x, y \text{Buy}(x, y) \wedge \text{Antiseizure}(y) \rightarrow \bot, \]
\[ \quad \exists x, y \text{Buy}(x, y) \wedge \text{Contains}(y, \text{phenytoin}) \rightarrow \bot \} \]

\[ \mathcal{T} = \{ \text{DrugA} \sqsubseteq \text{Antiseizure} \} \]

\[ \mathcal{A} = \{ \text{Buy}(\text{ann}, d_1), \text{DrugA}(d_1), \text{Buy}(\text{bob}, d_2), \text{Contains}(d_2, \text{phenytoin}) \} \]

\[ \mathcal{C}_1 = \{ \text{Buy}(\text{ann}, d_1), \text{Buy}(\text{bob}, d_2) \} \]

\[ \mathcal{C}_2 = \{ \text{Buy}(\text{ann}, d_1), \text{Contains}(d_2, \text{phenytoin}) \} \]

\[ \mathcal{C}_3 = \{ \text{DrugA}(d_1), \text{Antiseizure}(d_1), \text{Buy}(\text{bob}, d_2) \} \]

\[ \mathcal{C}_4 = \{ \text{DrugA}(d_1), \text{Antiseizure}(d_1), \text{Contains}(d_2, \text{phenytoin}) \} \]

\[ q_1 = \text{Buy}(\text{ann}, d_1) \]
$P = \{ \exists x, y (\text{Buy}(x, y) \land \text{Antiseizure}(y)) \rightarrow \bot, \\
\exists x, y (\text{Buy}(x, y) \land \text{Contains}(y, \text{phenytoin})) \rightarrow \bot \}$

$T = \{ \text{DrugA} \sqsubseteq \text{Antiseizure} \}$

$A = \{ \text{Buy}(\text{ann}, d_1), \text{DrugA}(d_1), \text{Buy}(\text{bob}, d_2), \text{Contains}(d_2, \text{phenytoin}) \}$

$C_1 = \{ \text{Buy}(\text{ann}, d_1), \text{Buy}(\text{bob}, d_2) \}$

$C_2 = \{ \text{Buy}(\text{ann}, d_1), \text{Contains}(d_2, \text{phenytoin}) \}$

$C_3 = \{ \text{DrugA}(d_1), \text{Antiseizure}(d_1), \text{Buy}(\text{bob}, d_2) \}$

$C_4 = \{ \text{DrugA}(d_1), \text{Antiseizure}(d_1), \text{Contains}(d_2, \text{phenytoin}) \}$

$q_1 = \text{Buy}(\text{ann}, d_1)$

$\implies$ We have both that $T \cup C_1 \models q_1$ and $T \cup C_2 \models q_1$

The system answers $true$
Example

\[\mathcal{P} = \{\exists x, y (\text{Buy}(x, y) \land \text{Antiseizure}(y)) \rightarrow \bot,\]
\[\exists x, y (\text{Buy}(x, y) \land \text{Contains}(y, \text{phenytoin})) \rightarrow \bot\}\]

\[\mathcal{T} = \{\text{DrugA} \sqsubseteq \text{Antiseizure}\}\]

\[\mathcal{A} = \{\text{Buy}(\text{ann}, d_1), \text{DrugA}(d_1), \text{Buy}(\text{bob}, d_2), \text{Contains}(d_2, \text{phenytoin})\}\]

\[\mathcal{C}_1 = \{\text{Buy}(\text{ann}, d_1), \text{Buy}(\text{bob}, d_2)\}\]
\[\mathcal{C}_2 = \{\text{Buy}(\text{ann}, d_1), \text{Contains}(d_2, \text{phenytoin})\}\]
\[\mathcal{C}_3 = \{\text{DrugA}(d_1), \text{Antiseizure}(d_1), \text{Buy}(\text{bob}, d_2)\}\]
\[\mathcal{C}_4 = \{\text{DrugA}(d_1), \text{Antiseizure}(d_1), \text{Contains}(d_2, \text{phenytoin})\}\]

\[q_2 = \text{DrugA}(d_1)\]
Example

$P = \{\exists x, y (\text{Buy}(x, y) \land \text{Antiseizure}(y)) \rightarrow \bot, \exists x, y (\text{Buy}(x, y) \land \text{Contains}(y, \text{phenytoin})) \rightarrow \bot\}$

$T = \{\text{DrugA} \sqsubseteq \text{Antiseizure}\}$

$A = \{\text{Buy(ann, d}_1\text{), DrugA(d}_1\text{), Buy(bob, d}_2\text{), Contains(d}_2\text{, phenytoin)}\}$

$C_1 = \{\text{Buy(ann, d}_1\text{), Buy(bob, d}_2\text{)}\}$

$C_2 = \{\text{Buy(ann, d}_1\text{), Contains(d}_2\text{, phenytoin)}\}$

$C_3 = \{\text{DrugA(d}_1\text{), Antiseizure(d}_1\text{), Buy(bob, d}_2\text{)}\}$

$C_4 = \{\text{DrugA(d}_1\text{), Antiseizure(d}_1\text{), Contains(d}_2\text{, phenytoin)}\}$

$q_2 = \text{DrugA(d}_1\text{)}$

$\implies$ No censor (among the surviving ones) entails $q_2$!

The system answers $false$
Example

\[ P = \{ \exists x, y (\text{Buy}(x, y) \land \text{Antiseizure}(y)) \rightarrow \bot, \\
\quad \exists x, y (\text{Buy}(x, y) \land \text{Contains}(y, \text{phenytoin})) \rightarrow \bot \} \]

\[ T = \{ \text{DrugA} \sqsubseteq \text{Antiseizure} \} \]

\[ A = \{ \text{Buy(ann, d}_1), \text{DrugA(d}_1), \text{Buy(bob, d}_2), \text{Contains(d}_2, \text{phenytoin}) \} \]

\[ C_1 = \{ \text{Buy(ann, d}_1), \text{Buy(bob, d}_2) \} \]

\[ C_2 = \{ \text{Buy(ann, d}_1), \text{Contains(d}_2, \text{phenytoin}) \} \]

\[ C_3 = \{ \text{DrugA(d}_1), \text{Antiseizure(d}_1), \text{Buy(bob, d}_2) \} \]

\[ C_4 = \{ \text{DrugA(d}_1), \text{Antiseizure(d}_1), \text{Contains(d}_2, \text{phenytoin}) \} \]

\[ q_3 = \exists x \text{ Contains}(x, \text{phenytoin}) \]
Example

\[ P = \{ \exists x, y (\text{Buy}(x, y) \land \text{Antiseizure}(y)) \rightarrow \bot, \\
\exists x, y (\text{Buy}(x, y) \land \text{Contains}(y, \text{phenytoin})) \rightarrow \bot \} \]

\[ T = \{ \text{DrugA} \sqsubseteq \text{Antiseizure} \} \]

\[ A = \{ \text{Buy}(\text{ann}, d_1), \text{DrugA}(d_1), \text{Buy}(\text{bob}, d_2), \text{Contains}(d_2, \text{phenytoin}) \} \]

\[ C_1 = \{ \text{Buy}(\text{ann}, d_1), \text{Buy}(\text{bob}, d_2) \} \]

\[ C_2 = \{ \text{Buy}(\text{ann}, d_1), \text{Contains}(d_2, \text{phenytoin}) \} \]

\[ C_3 = \{ \text{DrugA}(d_1), \text{Antiseizure}(d_1), \text{Buy}(\text{bob}, d_2) \} \]

\[ C_4 = \{ \text{DrugA}(d_1), \text{Antiseizure}(d_1), \text{Contains}(d_2, \text{phenytoin}) \} \]

\[ q_3 = \exists x \text{Contains}(x, \text{phenytoin}) \]

\[ \implies \text{We have that } C_2 \models q_3 \]

The system answers \textit{true}
Semantics

We keep track of the history of user queries through the notion of state:

**State**

A *state* of a CQE instance $\mathcal{E}$ is a pair $S = \langle \mathcal{E}, Q \rangle$ where $Q = \langle q_1, \ldots, q_n \rangle$ is a sequence of BUCQs.

The set $StCens(S_n)$ of *censors of the state* $S_n = \langle \mathcal{E}, Q_n \rangle$ is recursively defined as follows:

- $StCens(S_0) = OptCens(\mathcal{E})$
- For $1 \leq i \leq n$:
  - $StCens(S_i) = StCens(S_{i-1})$, if $\forall C \in StCens(S_{i-1})$ s.t. $\mathcal{T} \cup C \models q_i$
  - $StCens(S_i) = \{ C \in StCens(S_{i-1}) \mid \mathcal{T} \cup C \models q_i \}$, otherwise

We say that the state $S_n$ *entails* $q_i$ ($S_n \models q_i$) if $\mathcal{T} \cup C \models q_i$ for all the censors $C \in StCens(S_n)$. 

Semantics

We keep track of the history of user queries through the notion of state:

<table>
<thead>
<tr>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>A <em>state</em> of a CQE instance $\mathcal{E}$ is a pair $S = \langle \mathcal{E}, Q \rangle$ where $Q = \langle q_1, \ldots, q_n \rangle$ is a sequence of BUCQs</td>
</tr>
</tbody>
</table>

The set $StCens(S_n)$ of *censors of the state* $S_n = \langle \mathcal{E}, Q_n \rangle$ is recursively defined as follows:

- $StCens(S_0) = OptCens(\mathcal{E})$
- for $1 \leq i \leq n$:
  - $StCens(S_i) = StCens(S_{i-1})$, if $\nexists C \in StCens(S_{i-1})$ s.t. $\mathcal{T} \cup C \models q_i$
  - $StCens(S_i) = \{ C \in StCens(S_{i-1}) | \mathcal{T} \cup C \models q_i \}$, otherwise

We say that the state $S_n$ *entails* $q_i$ ($S_n \models q_i$) if $\mathcal{T} \cup C \models q_i$ for all the censors $C \in StCens(S_n)$
We say that $C$ is **maximally cooperative** w.r.t. $Q$ if there do not exist a censor $C'$ and a number $m$ s.t.:

1. $T \cup C \models q_i \iff T \cup C' \models q_i$ for every $1 \leq i \leq m$, and
2. $T \cup C \not\models q_{m+1}$ and $T \cup C' \models q_{m+1}$.

**Theorem**

Every censor for $\mathcal{E}$ is maximally cooperative w.r.t. $Q$ iff $C \in StCens(\langle \mathcal{E}, Q \rangle)$.
Maximal cooperativity

We say that $C$ is \textit{maximally cooperative} w.r.t. $Q$ if there do not exist a censor $C'$ and a number $m$ s.t.:

1. $T \cup C \models q_i \iff T \cup C' \models q_i$ for every $1 \leq i \leq m$, and
2. $T \cup C \not\models q_{m+1}$ and $T \cup C' \models q_{m+1}$.

**Theorem**

Every censor for $E$ is maximally cooperative w.r.t. $Q$ iff $C \in StCens(\langle E, Q \rangle)$
We study the data complexity of the following decision problem for OWL 2 QL ontologies:

**Query entailment in a state**

Given a state $S = \langle E, Q \rangle$ and a BUCQ $q \in Q$, check whether $S \models q$

We proved that the above problem is **FO-rewritable**, i.e. it is always possible to reformulate $q$ into a FO query $q_r$ such that $S \models q$ iff $A \models q_r$

- No need to materialize censors
- Problem belongs to $\text{AC}^0$ class in data complexity
We study the data complexity of the following decision problem for OWL 2 QL ontologies:

### Query entailment in a state

Given a state $\mathcal{S} = \langle \mathcal{E}, \mathcal{Q} \rangle$ and a BUCQ $q \in \mathcal{Q}$, check whether $\mathcal{S} \models q$

We proved that the above problem is **FO-rewritable**, i.e. it is always possible to reformulate $q$ into a FO query $q_r$ such that $\mathcal{S} \models q$ iff $\mathcal{A} \models q_r$

- No need to materialize censors
- Problem belongs to $\text{AC}^0$ class in data complexity
First-order rewriting example

Example (cont’d)

The query $q_3 = \exists x \text{Contains}(x, \text{phenytoin})$ can be rewritten as:

$q_r = \neg \text{Buy}(\text{ann}, d_1) \land \neg \text{DrugA}(d_1) \land \exists x \text{Contains}(x, \text{phenytoin}) \lor$
\begin{align*}
&\neg (\text{Buy}(\text{ann}, d_1) \land \text{DrugA}(d_1) \land \neg (\top)) \land \exists x \left( \text{Buy}(\text{ann}, d_1) \land \\
&\text{Contains}(x, \text{phenytoin}) \land \neg (d_1 = x) \right) \lor \\
&\neg \text{Buy}(\text{ann}, d_1) \land (\exists x \left( \text{DrugA}(d_1) \land \text{Contains}(x, \text{phenytoin}) \right)) \lor \\
&\exists x \left( \text{Buy}(\text{ann}, d_1) \land \text{DrugA}(d_1) \land \text{Contains}(x, \text{phenytoin}) \right) \land \neg (\top \lor d_1 = x) \right)
\end{align*}

Due to the second disjunct, we have that $q_r$ evaluates to true in $\mathcal{A}$ (indeed we had that $S \models q_3$)
Outline

1. The Controlled Query Evaluation approach in Ontologies and Description Logics
   - CQE in Description Logics through GA censors
   - Computational problems

2. Towards tractability 1: Intersecting the censors
   - IGA censors
   - Expressive limitations of IGA censors

3. Towards tractability 2: Adding preferences
   - Globally optimal and Pareto-optimal censors
   - DD and k-DD censors
   - Experimental results

4. Towards tractability 3: Maximally cooperative approach
   - The dynCQE approach
   - Complexity of dynCQE

5. Conclusions
Main open problems

- Extend the results to **different knowledge bases/databases** (beyond DLs)
- Improve the **user/attacker model**
- Increase the **policy language**
- Increase the **query language**
- Optimized practical **algorithms**
- More powerful notions of **censor** (not only tuple-deletion based)
Our ongoing work

- Extend the dynCQE framework to **non-Boolean UCQs**
- **Implement** dynCQE (as we did for previous frameworks)
- Extend the *policy language* (as done for static CQE)
- Refine the current framework for allowing to **delete specific cells** of a table instead of full tuples

