Dynamic Controlled Query Evaluation over DL-Lite Ontologies (Extended Abstract)

Piero Bonatti[1], Gianluca Cima[2], Domenico Lembo[2], Lorenzo Marconi[2], Riccardo Rosati[2], Luigi Sauro[1] and Domenico Fabio Savo[3]

[1]Università di Napoli Federico II
[2]Sapienza Università di Roma
[3]Università degli Studi di Bergamo

Abstract
This extended abstract summarizes our recent work [1] in which we study a *dynamic* Controlled Query Evaluation method over Description Logic ontologies.

Keywords
Description Logics, Data Protection, First-Order Rewritability

Semantic Web technologies are increasingly used to represent and link together different sources of information coming from public organizations as well as private citizens. This information may include sensitive knowledge, e.g., medical records or social network activities, whose disclosure may affect the privacy of individuals if not adequately protected [2, 3].

One goal of confidentiality-preserving data publishing is to prevent the disclosure of sensitive information to unauthorized users while being as cooperative as possible, that is, answering queries honestly whenever this does not harm confidentiality. Specifically, in Controlled Query Evaluation (CQE) [4, 5] the data protection policy is declaratively specified through logical formulas and is enforced by altering query answers through so-called censors, which either refuse to answer some queries or lie when this is needed in order to protect some secrets. In general, there exist multiple, mutually incomparable ways of concealing answers, i.e., mutually incomparable censors. Different works proposed static CQE methods, where a censor is constructed (or approximated) beforehand, establishing once and for all which queries should be answered truthfully [2, 6, 7, 8, 9]. In several cases, such approaches are not fully cooperative, because the secure view of the data is chosen without taking the users' interests into account.

Conversely, following the work of Biskup and Bonatti [10], in this paper we introduce a dynamic CQE (dynCQE) method that progressively decides whether to be truthful or to lie, based on the specific stream of queries. Roughly speaking, the dynamic CQE approach selects, at each step, as many censors as possible, coherently with the previous answers. By doing so, it

DL 2023: 36th International Workshop on Description Logics, September 2–4, 2023, Rhodes, Greece
pab@unina.it (P. Bonatti); cima@diag.uniroma1.it (G. Cima); lembo@diag.uniroma1.it (D. Lembo); marconi@diag.uniroma1.it (L. Marconi); rosati@diag.uniroma1.it (R. Rosati); luigi.sauro@unina.it (L. Sauro); domenicofabio.savo@unibg.it (D. F. Savo)
0000-0003-1436-5660 (P. Bonatti); 0000-0003-1783-5605 (G. Cima); 0000-0002-0628-242X (D. Lembo); 0000-0001-9633-8476 (L. Marconi); 0000-0002-7697-4958 (R. Rosati); 0000-0001-6056-0868 (L. Sauro); 0000-0002-8391-8049 (D. F. Savo)

maximizes the possibility of answering the next query honestly by choosing from the current pool of censors those that allow answering the query truthfully (if any).

We consider Description Logic (DL) ontologies $\mathcal{O} = \mathcal{T} \cup \mathcal{A}$, where $\mathcal{T}$ is a TBox and $\mathcal{A}$ is an ABox. We assume that an ABox is a finite set of *ground atoms*, i.e., assertions of the form $A(a)$ and $P(a, b)$, where A and P are an atomic concept and an atomic role, respectively, occurring in the signature of $\mathcal{T}$, and a and b are constants. In what follows, we denote by $\mathsf{cl}_{\mathcal{T}}(\mathcal{A})$ the set of ground atoms entailed by an ontology $\mathcal{T} \cup \mathcal{A}$, i.e., $\mathsf{cl}_{\mathcal{T}}(\mathcal{A}) = \{\gamma \mid \gamma \text{ is a ground atom and } \mathcal{T} \cup \mathcal{A} \models \gamma\}$. We also consider data protection policies (for short, *policies*) that are finite sets of *denials*, i.e., sentences of the form $q \rightarrow \bot$ such that q is a Boolean conjunctive query (BCQ). As for user queries, we focus on *Boolean Union of Conjunctive Queries (BUCQs)*.

A *CQE specification* is a pair $\langle \mathcal{T}, \mathcal{P} \rangle$, where $\mathcal{T}$ is a TBox and $\mathcal{P}$ is a policy such that $\mathcal{T} \cup \mathcal{P}$ is a consistent first-order (FO) theory. A *CQE instance* is a triple $\mathcal{E} = \langle \mathcal{T}, \mathcal{P}, \mathcal{A} \rangle$, where $\langle \mathcal{T}, \mathcal{P} \rangle$ is a CQE specification, and $\mathcal{A}$ is an ABox such that $\mathcal{T} \cup \mathcal{A}$ is consistent. In this paper, we assume that a user is aware of both the TBox and the policy, but not the ABox.

Censors specify which consequences of an ontology can be disclosed without violating the policy. The following definition is adapted from [9, Definition 1].

Definition 1 (Censor)**.** *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a CQE instance. A* censor *for $\mathcal{E}$ is an ABox $\mathcal{C} \subseteq cl_{\mathcal{T}}(\mathcal{A})$ such that $\mathcal{T} \cup \mathcal{P} \cup \mathcal{C}$ is consistent.*

Given a CQE instance $\mathcal{E}$ and a censor $\mathcal{C}$ for $\mathcal{E}$, we say that $\mathcal{C}$ is *optimal* if there exists no censor $\mathcal{C}'$ for $\mathcal{E}$ such that $\mathcal{C} \subset \mathcal{C}'$. We denote by *OptCens*$(\mathcal{E})$ the set of all the optimal censors for $\mathcal{E}$. We observe that a censor for a CQE instance $\mathcal{E}$ always exists,[1] and thus *OptCens*$(\mathcal{E}) \neq \emptyset$. Given a BUCQ q, we denote by *OptCens*$(\mathcal{E}, q)$ the set of optimal censors that, together with $\mathcal{T}$, entail q:

$$\mathit{OptCens}(\mathcal{E}, q) = \{\mathcal{C} \in \mathit{OptCens}(\mathcal{E}) \mid \mathcal{T} \cup \mathcal{C} \models q\}$$

We are interested in dynamic CQE systems, which a user interacts with by evaluating one query after another. The following notion of *state* captures the history of such queries.

Definition 2 (State)**.** *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{P}, \mathcal{A} \rangle$ be a CQE instance. A* state of $\mathcal{E}$ *is a pair $\mathcal{S} = \langle \mathcal{E}, \mathcal{Q} \rangle$, where $\mathcal{Q} = \langle q_1, \ldots, q_n \rangle$ (with $n \geq 0$) is a sequence of BUCQs.*

The evaluation of each query provides the user with new information. The query answering semantics adopted by the CQE system must ensure that, even by collecting such information, it is impossible for the user to discover data protected by the policy. At the same time, we aim to get a semantics that returns the longest possible sequence of honest answers before lying (the so-called *"longest honeymoon" approach* [10]). Below we formalize our idea of dynamic CQE (dynCQE), i.e., a CQE that takes into account a state $\langle \mathcal{E}, \mathcal{Q} \rangle$. In what follows, given a CQE instance $\mathcal{E}$, a sequence $\mathcal{Q}_n = \langle q_1, \ldots, q_n \rangle$ of BUCQs, and any integer $i \in [0, n]$, we denote with $\mathcal{Q}_i$ the sequence $\langle q_1, \ldots, q_i \rangle$ and with $\mathcal{S}_i$ the state $\langle \mathcal{E}, \mathcal{Q}_i \rangle$ of $\mathcal{E}$, with the convention that $\mathcal{Q}_0$ is the empty sequence $\langle\rangle$.

Definition 3 (Dynamic CQE – dynCQE)**.** *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{P}, \mathcal{A} \rangle$ be a CQE instance, and let $\mathcal{Q}_n = \langle q_1, \ldots, q_n \rangle$ (with $n \geq 0$) be a sequence of BUCQs. The set StCens$(\mathcal{S}_n)$ of censors of $\mathcal{S}_n$ is inductively defined as follows:*

[1] Trivially, the empty set is a censor for any CQE instance $\mathcal{E}$.

- *StCens*$(\mathcal{S}_0) = $ *OptCens*$(\mathcal{E})$;
- *StCens*$(\mathcal{S}_{i+1}) = \begin{cases} \mathit{StCens}(\mathcal{S}_i) \text{ if } \mathit{StCens}(\mathcal{S}_i) \cap \mathit{OptCens}(\mathcal{E}, q_{i+1}) = \emptyset, \\ \mathit{StCens}(\mathcal{S}_i) \cap \mathit{OptCens}(\mathcal{E}, q_{i+1}) \text{ otherwise,} \end{cases}$
  for every $0 \leq i \leq n-1$.

For each BUCQ q_i *occurring in* $\mathcal{Q}_n$*, we say that* q_i is entailed by $\mathcal{S}_n$*, denoted by* $\mathcal{S}_n \models q_i$*, if* $\mathcal{T} \cup \mathcal{C} \models q_i$ *for every* $\mathcal{C} \in$ *StCens*$(\mathcal{S}_n)$*. We denote by* $EntQ(\mathcal{S}_n)$ *the set of queries of* $\mathcal{Q}_n$ *entailed by* $\mathcal{S}_n$*, i.e.,* $EntQ(\mathcal{S}_n) = \{q \in \mathcal{Q}_n \mid \mathcal{S}_n \models q\}$.

One can see that, for any $i = 1, \ldots, n$, the set of censors of a state $\mathcal{S}_i$ is always non-empty and consists of a subset of the set of censors of its predecessor state $\mathcal{S}_{i-1}$, i.e., *StCens*$(\mathcal{S}_{i-1}) \supseteq$ *StCens*$(\mathcal{S}_i) \supset \emptyset$. This also implies that $EntQ(\mathcal{S}_{i-1}) \subseteq EntQ(\mathcal{S}_i)$ holds for any $i = 1, \ldots, n$.

Informally speaking, each set *StCens*$(\mathcal{S}_i)$ (with $1 \leq i \leq n$) in the above definition progressively selects the optimal censors of $\mathcal{E}$ that agree with $EntQ(\mathcal{S}_i)$. If none of the surviving optimal censors in *StCens*$(\mathcal{S}_i)$ entails (together with $\mathcal{T}$) a query q_{i+1}, then $\mathcal{S}_{i+1} \not\models q_{i+1}$, so we have that *StCens*$(\mathcal{S}_{i+1}) = $ *StCens*$(\mathcal{S}_i)$. Conversely, if at least one of the censors in *StCens*$(\mathcal{S}_i)$, together with the TBox, entails q_{i+1}, then, according to dynCQE, we have a positive answer, and *StCens*$(\mathcal{S}_{i+1})$ keeps only the censors in *StCens*$(\mathcal{S}_i)$ that agree with such answer.

Example 1. *Some pharmaceutical products may reveal with high accuracy which kind of disease is affecting a person. For instance, drugs that contain phenytoin, or that are classified as anti-seizure medications, indicate some form of epilepsy.*

Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{P}, \mathcal{A} \rangle$ *be a CQE instance, where:*

$\mathcal{T} = \{\mathsf{Abc} \sqsubseteq \mathsf{Antiseizure}\}$;
$\mathcal{P} = \{\exists x, y(\mathsf{buy}(x, y) \wedge \mathsf{Antiseizure}(y)) \rightarrow \bot,$
$\quad \exists x, y(\mathsf{buy}(x, y) \wedge \mathsf{contain}(y, \mathsf{phenytoin})) \rightarrow \bot\}$;
$\mathcal{A} = \{\mathsf{buy}(\mathsf{john}, \mathsf{m}_a), \mathsf{Abc}(\mathsf{m}_a), \mathsf{buy}(\mathsf{alice}, \mathsf{m}_b), \mathsf{contain}(\mathsf{m}_b, \mathsf{phenytoin})\}$.

In words, the TBox states that Abc *is an anti-seizure medication, while the policy conceals the presence of patients suffering from epilepsy.*

Let us start by considering the empty sequence of BUCQs. By definition, we have that *StCens*$(\langle \mathcal{E}, \langle\rangle \rangle)$ *coincides with the set of optimal censors for* $\mathcal{E}$*:*

- $\mathcal{C}_1 = \{\mathsf{buy}(\mathsf{john}, \mathsf{m}_a), \mathsf{buy}(\mathsf{alice}, \mathsf{m}_b)\}$;
- $\mathcal{C}_2 = \{\mathsf{buy}(\mathsf{john}, \mathsf{m}_a), \mathsf{contain}(\mathsf{m}_b, \mathsf{phenytoin})\}$;
- $\mathcal{C}_3 = \{\mathsf{Abc}(\mathsf{m}_a), \mathsf{Antiseizure}(\mathsf{m}_a), \mathsf{buy}(\mathsf{alice}, \mathsf{m}_b)\}$;
- $\mathcal{C}_4 = \{\mathsf{Abc}(\mathsf{m}_a), \mathsf{Antiseizure}(\mathsf{m}_a), \mathsf{contain}(\mathsf{m}_b, \mathsf{phenytoin})\}$.

Let $q_1 = \mathsf{buy}(\mathsf{john}, \mathsf{m}_a)$ *be the first query. The censors* $\mathcal{C}_1$ *and* $\mathcal{C}_2$ *agree with answering true to this query. All the censors that disagree with such an answer are then removed, obtaining* *StCens*$(\langle \mathcal{E}, \langle q_1 \rangle \rangle) = $ *StCens*$(\langle \mathcal{E}, \langle\rangle \rangle) \cap$ *OptCens*$(\mathcal{E}, q_1) = \{\mathcal{C}_1, \mathcal{C}_2\}$*. Then, let* $q_2 = \mathsf{Abc}(\mathsf{m}_a)$ *be a new query in the sequence. Since neither* $\mathcal{T} \cup \mathcal{C}_1$ *nor* $\mathcal{T} \cup \mathcal{C}_2$ *entail* q_2*, then* *StCens*$(\langle \mathcal{E}, \langle q_1, q_2 \rangle \rangle) =$ *StCens*$(\langle \mathcal{E}, \langle q_1 \rangle \rangle)$*. Now, consider adding* $q_3 = \exists x \mathsf{buy}(x, \mathsf{m}_b)$ *to the sequence. Since* $\mathcal{T} \cup \mathcal{C}_1 \models q_3$ *while* $\mathcal{T} \cup \mathcal{C}_2 \not\models q_3$*, we have* *StCens*$(\mathcal{S}) = \{\mathcal{C}_1\}$*, where* $\mathcal{S} = \langle \mathcal{E}, \mathcal{Q} \rangle$ *with* $\mathcal{Q} = \langle q_1, q_2, q_3 \rangle$*. Clearly,* $\mathcal{S} \models q_1$ *and* $\mathcal{S} \models q_3$*, but* $\mathcal{S} \not\models q_2$. □

Note that the stream of queries is processed greedily, answering the truth as long as some of the censors in *StCens*$(\mathcal{S}_n)$ allow to do it. In fact, we will show that dynCQE satisfies the *maximally cooperative property*, which implies and strengthens the longest honeymoon approach.

Definition 4 (Cooperativity)**.** *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{P}, \mathcal{A} \rangle$ be a CQE instance, $\mathcal{Q} = \langle q_1, \ldots, q_n \rangle$ (with $n \geq 0$) a sequence of BUCQs, and $\mathcal{C}$ and $\mathcal{C}'$ two censors for $\mathcal{E}$. We say that $\mathcal{C}$ is* more cooperative *than $\mathcal{C}'$ with respect to $\mathcal{Q}$ if there exists a non-negative integer $m < n$ such that*

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

We also say that $\mathcal{C}$ is maximally cooperative *with respect to $\mathcal{Q}$ if there does not exist any censor $\mathcal{C}''$ for $\mathcal{E}$ that is more cooperative than $\mathcal{C}$.*

We are now ready to prove that, for each state $\mathcal{S} = \langle \mathcal{E}, \mathcal{Q} \rangle$ of a CQE instance, the set *StCens*$(\mathcal{S})$ coincides with the set of all censors that are maximally cooperative with respect to $\mathcal{Q}$.

Theorem 1. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{P}, \mathcal{A} \rangle$ be a CQE instance, and $\mathcal{Q} = \langle q_1, \ldots, q_n \rangle$ be a sequence of BUCQs. A censor $\mathcal{C}$ for $\mathcal{E}$ is maximally cooperative with respect to $\mathcal{Q}$ iff $\mathcal{C} \in$ StCens$(\langle \mathcal{E}, \mathcal{Q} \rangle)$.*

The next technical results focus on *DL-Lite*$_\mathcal{R}$ CQE specifications and instances, i.e., when TBox and ABox are expressed in *DL-Lite*$_\mathcal{R}$ [11], the logical underpinning of OWL 2 QL [12].

We first show that the behavior of dynCQE can not be simulated by *static* CQE, for which algorithms are already known [8, 13, 9], through data-independent modifications of the intensional components of the framework.

Theorem 2. *There exists a DL-Lite$_\mathcal{R}$ CQE specification $\langle \mathcal{T}, \mathcal{P} \rangle$ and a BUCQ q such that there exists no DL-Lite$_\mathcal{R}$ CQE specification $\langle \mathcal{T}', \mathcal{P}' \rangle$ such that, for every ABox $\mathcal{A}$, OptCens$(\langle \mathcal{T}', \mathcal{P}', \mathcal{A} \rangle) =$ StCens$(\mathcal{S})$, where $\mathcal{S} = \langle \langle \mathcal{T}, \mathcal{P}, \mathcal{A} \rangle, \langle q \rangle \rangle$.*

On the other hand, we show that dynCQE query processing is *first-order rewritable* by providing a tailored query rewriting algorithm. This implies that the associated decision problem is in AC^0 in data complexity [14] (like the evaluation of FO sentences, i.e., SQL queries).

Theorem 3. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{P}, \mathcal{A} \rangle$ be a DL-Lite$_\mathcal{R}$ CQE instance, $\mathcal{Q} = \langle q_1, \ldots, q_n \rangle$ (with $n \geq 0$) be a sequence of BUCQs, and $q \in \mathcal{Q}$. The problem of deciding whether $q \in EntQ(\mathcal{S})$, where $\mathcal{S} = \langle \mathcal{E}, \mathcal{Q} \rangle$, is AC^0 with respect to the size of $\mathcal{A}$.*

The present work can be extended in several interesting directions. First, while the presented results indicate the possibility of a query rewriting approach to dynamic CQE, more work is still needed to define a practical query answering technique. Moreover, we are currently working on extending our dynamic CQE approach also to non-Boolean UCQs.

Acknowledgments

This work was partially supported by: projects FAIR (PE0000013) and SERICS (PE00000014) under the MUR National Recovery and Resilience Plan funded by the European Union - NextGenerationEU; Glaciation project funded from the European Union's HE research and innovation programme (grant agreement No 101070141); ANTHEM project funded by the National Plan for NRRP Complementary Investments (CUP: B53C22006700001).

References

[1] P. A. Bonatti, G. Cima, D. Lembo, L. Marconi, R. Rosati, L. Sauro, D. F. Savo, Controlled query evaluation in OWL 2 QL: A "longest honeymoon" approach, in: Proc. of ISWC 2022, volume 13489 of *LNCS*, 2022, pp. 428–444.

[2] P. A. Bonatti, L. Sauro, A confidentiality model for ontologies, in: Proc. of ISWC 2013, volume 8218 of *LNCS*, Springer, 2013, pp. 17–32.

[3] B. Cuenca Grau, E. V. Kostylev, Logical foundations of linked data anonymisation, J. Artif. Intell. Res. 64 (2019) 253–314.

[4] J. Biskup, For unknown secrecies refusal is better than lying, Data and Knowledge Engineering 33 (2000) 1–23.

[5] J. Biskup, P. A. Bonatti, Controlled query evaluation for enforcing confidentiality in complete information systems, Int. J. of Information Security 3 (2004) 14–27.

[6] B. Cuenca Grau, E. Kharlamov, E. V. Kostylev, D. Zheleznyakov, Controlled query evaluation for datalog and OWL 2 profile ontologies, in: Proc. of IJCAI 2015, 2015, pp. 2883–2889.

[7] D. Lembo, R. Rosati, D. F. Savo, Revisiting controlled query evaluation in description logics, in: Proc. of IJCAI 2019, 2019, pp. 1786–1792.

[8] G. Cima, D. Lembo, R. Rosati, D. F. Savo, Controlled query evaluation in description logics through instance indistinguishability, in: Proc. of IJCAI 2020, 2020, pp. 1791–1797.

[9] G. Cima, D. Lembo, L. Marconi, R. Rosati, D. F. Savo, Controlled query evaluation over prioritized ontologies with expressive data protection policies, in: Proc. of ISWC 2021, volume 12922 of *LNCS*, 2021, pp. 374–391.

[10] J. Biskup, P. A. Bonatti, Controlled query evaluation for known policies by combining lying and refusal, Ann. Math. Artif. Intell. 40 (2004) 37–62.

[11] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family, J. of Automated Reasoning 39 (2007) 385–429.

[12] B. Motik, A. Fokoue, I. Horrocks, Z. Wu, C. Lutz, B. Cuenca Grau, OWL Web Ontology Language Profiles, W3C Recommendation, W3C, 2009. Available at http://www.w3.org/TR/owl-profiles/.

[13] G. Cima, D. Lembo, L. Marconi, R. Rosati, D. F. Savo, Controlled query evaluation in ontology-based data access, in: Proc. of ISWC 2020, volume 12506 of *LNCS*, 2020, pp. 128–146.

[14] M. Y. Vardi, The complexity of relational query languages, in: Proc. of STOC 1982, 1982, pp. 137–146.