Privacy Preservation Using Multi-Context Systems and Default Logic

Jürgen Dix[1]*, Wolfgang Faber[2]**, and V.S. Subrahmanian[3]

[1] Department of Informatics
Clausthal University of Technology
38678 Clausthal, Germany
dix@tu-clausthal.de
[2] Department of Mathematics
University of Calabria
87030 Rende (CS), Italy
wf@wfaber.com
[3] Department of Computer Science
University of Maryland
College Park, MD 20742
vs@cs.umd.edu

Abstract. Preserving the privacy of sensitive data is one of the major challenges the information society has to face. Traditional approaches focused on infrastructures for identifying data which is to be kept private and for managing access rights to these data. However, although these efforts are useful, they do not address an important aspect: While the sensitive data itself can be protected nicely using these mechanisms, related data, which is deemed insensitive per se, may be used to *infer* sensitive data. This inference can be achieved by combining insensitive data or by exploiting specific background knowledge of the domain of discourse. In this paper, we present a general formalization of this problem and two particular instantiations of it. The first supports query answering by means of multi-context systems and hybrid knowledge bases, while the second allows for query answering by using default logic.

1 Introduction

With the advent of the Internet and easy access to huge amounts of data, keeping sensitive data private has become a priority for distributed information systems. An example area in which privacy is at stake are medical information systems.

Most databases have privacy mechanisms which are comparatively simple. Often, this boils down to keeping certain columns of the database hidden from certain types

* The first author acknowledges that this work was partly funded by the NTH School for IT Ecosystems (NTH (Niedersächsische Technische Hochschule) is a joint university consisting of Technische Universität Braunschweig, Technische Universität Clausthal, and Leibniz Universität Hannover).

** Supported by M.I.U.R. within the PRIN project LoDeN. Some of the authors of this paper were funded in part by ARO grants W911NF0910206 and W911NF0910525 and W911NF1110344.

of users. There are many approaches dealing with formalisms for this kind of authorization problem, and we refer to [24] and references in that work, in which aspects of the authorization problem in non-monotonic knowledge bases are discussed. What we are interested in, however, is a somewhat different issue: *Can users infer private information by only asking queries that do not involve such information and then making "common sense" inferences from the answers?*

In this paper, we generalize an earlier definition of the *Privacy Preservation Problem* [10]. This definition imposed several restrictions on the underlying knowledge bases. Most importantly, they had to be first-order theories, because in this way it is easily possible to build a default theory around them. In our new definition, we aim at making as few assumptions about the involved knowledge bases as possible. Essentially we will use the terminology that has been introduced for multi-context systems by Brewka and Eiter in [7]: It only assumes the knowledge bases to have an underlying logic of some sort.

We will show that multi-context systems can be used for implementing a system that computes privacy preserving answers. Essentially, we use contexts and bridge rules that link contexts in order to determine whether an answer violates the privacy requirements of some user. An appealing aspect of this instantiation of the general privacy preservation framework is that efficient systems for reasoning with multi-context systems are beginning to emerge [1], making privacy preserving query answering systems feasible in practice. We will then consider a restriction of the framework, which essentially matches the definitions of [10], and review how default logic can be used in order to obtain a privacy preserving query answering system.

In the following, we first provide a general definition of the privacy preservation problem in Section 2, which allows for heterogeneous knowledge bases. In Section 3 we show how to construct a multi-context system for computing answers for the general privacy preservation problem. In Section 4 we show that the setting of [10] is a special case of the general framework and review how to solve these problems by means of default logic. We conclude with Section 5 and outline future work. This work elaborates on earlier results presented in [10] and [12].

2 Privacy Preservation Problem

In this section, we provide a general formalization of the privacy preservation problem, $\mathrm{P3}$ for short. This development is similar to the one in [5] and some earlier work in [23], with slightly modified terminology. We start with basic concepts for describing knowledge bases, using terminology of [7].

We consider a *logic* L as in [7] to be a triple $(\mathbf{KB}_L, \mathbf{BS}_L, \mathbf{ACC}_L)$ where $\mathbf{KB}_L$ is the set of well-formed knowledge bases of L (each of which is a set as well), $\mathbf{BS}_L$ is the set of possible belief sets, and $\mathbf{ACC}_L$ is a function $\mathbf{KB}_L \rightarrow \mathbf{2}^{\mathbf{BS}_L}$ describing the semantics of each knowledge base. In the following, when mentioning knowledge bases, we do not specify the underlying logic (and drop the subscripts from $\mathbf{KB}$, $\mathbf{BS}$, and $\mathbf{ACC}$): It can just be any logic in the sense just described. Moreover, let the finite set $\mathbf{U}$ contain one user ID for each user in the system under consideration. By abuse of language we use the notation $\mathbf{U} = \{u_1, \ldots, u_{|\mathbf{U}|}\}$.

Definition 1 (main knowledge base $\mathbf{MKB}$). *The* main knowledge base $\mathbf{MKB}$ *is a knowledge base of some logic L.*

The main knowledge base is the one that the users will be querying, and around which the privacy preservation mechanism must be implemented. So the users will query the main knowledge base, and the privacy preservation mechanism might prevent certain answers to be disclosed. This mechanism foresees the availability of a model of each user's knowledge. Thus, at any given instance t in time, each user u has some set of *background knowledge*. This background knowledge may be elicited in many ways: One such source is the set of all information disclosed to the user by the system. For example, a hospital accountant may not be allowed to see patient diagnoses, though she may see billing information about them.

Definition 2 (user model). *The function $\mathbf{BK}$ assigns to each user $u \in \mathbf{U}$ a* background knowledge base *$\mathbf{BK}^t(u)$ for each timepoint t. The function $\mathbf{Priv}$ assigns to each user $u \in \mathbf{U}$ a belief set $\mathbf{Priv}(u)$ that should be kept private.*

Note that the various knowledge bases need not be of the same logic, but for practical reasons one would assume the belief sets to be homogeneous. It should be pointed out that $\mathbf{BK}^t(u)$ will usually not be the user's own knowledge base, but rather a model of the user's knowledge, maintained by the information system. Note that $\mathbf{BK}^t(u)$ varies as t varies. For example, as the database discloses answers to the user u, the background knowledge associated to u may increase. *Throughout most of this paper, we assume that t is fixed and we address the problem of preserving privacy at a given timepoint.* As a consequence, we usually write $\mathbf{BK}(u)$ and drop the superscript t.

Example 1. Consider a small medical knowledge base MedKB containing information about the symptoms and diseases of some patients. Let this knowledge base describe two predicates $\mathsf{symptom}$ and $\mathsf{disease}$ and let the following be its only belief set S_{MedKB}:

$$\begin{array}{lll}
\mathsf{symptom}(john, s_1) & \mathsf{symptom}(jane, s_1) & \mathsf{disease}(jane, aids) \\
\mathsf{symptom}(john, s_2) & \mathsf{symptom}(jane, s_4) & \mathsf{disease}(john, cancer) \\
\mathsf{symptom}(john, s_3) & & \mathsf{disease}(ed, polio)
\end{array}$$

Note that MedKB could very well be just a database. Assume that $john$ and $jane$ are also users of the system and want to keep their diseases private, so $\mathbf{Priv}(john) = \{\mathsf{disease}(john, cancer)\}$, while $\mathbf{Priv}(jane) = \{\mathsf{disease}(jane, aids)\}$. Consider another user $acct$ (an accountant). This person may have the following background knowledge base $\mathbf{BK}(acct)$ in the form of rules (so the underlying logic might be answer set programming).

$$\mathsf{disease}(X, aids) \leftarrow \mathsf{symptom}(X, s_1), \mathsf{symptom}(X, s_4)$$
$$\mathsf{disease}(X, cancer) \leftarrow \mathsf{symptom}(X, s_2), \mathsf{symptom}(X, s_3)$$

We now define the concepts of query and answer to a knowledge base. The precise notation of a query is not so important, only the definition of its answer is. So given a *main knowledge base* $\mathbf{MKB}$ wrt. a *logic* L, we assume there is a set $\mathbf{Q}$ consisting of all queries over $\mathbf{MKB}$.

Definition 3 (query and answer). *We assume that there is a mapping which associates to each* query $Q \in \mathbf{Q}$*, each knowledge base and each semantics in* logic L*, a* belief set *of* L*. This belief set is referred to as the* answer *to* Q *and is denoted by* $\mathbf{Ans}(Q)$.

Users pose a query to the main knowledge base, but the privacy preservation mechanism should allow only answers which do not violate the privacy specifications of users after taking into account, the (presumed) knowledge of the user asking the query.

Definition 4 ((maximal) privacy preserving answer). *A* privacy preserving answer *to a query* Q *over* $\mathbf{MKB}$ *posed by* $u_o \in \mathbf{U}$ *with respect to* $\mathbf{BK}$ *and* $\mathbf{Priv}$ *is* $X \subseteq \mathbf{Ans}(Q)$ *such that for all* $u \in \mathbf{U} \setminus \{u_0\}$ *and for all* $p \in \mathbf{Priv}(u)$*, if* $p \notin \mathbf{ACC}(\mathbf{BK}(u_0))$ *then* $p \notin \mathbf{ACC}(X \cup \mathbf{BK}(u_0))$*. A* maximal privacy preserving answer *is a subset maximal privacy preserving answer.*

Note that here we assume that elements of belief sets can be added to knowledge bases, yielding again a knowledge base of the respective logic. We are now ready to formally define the central problem studied in this paper.

Definition 5 (privacy preservation problem). *A privacy preservation problem* $\mathrm{P3}$ *is a tuple* $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$*. Solutions of this problem are all the (maximal) privacy preserving answers to* Q *posed by* u_0 *over* $\mathbf{MKB}$ *with respect to* $\mathbf{BK}$ *and* $\mathbf{Priv}$.

Example 2. Returning to our MedKB example, posing the query $\mathsf{disease}(john, X)$, we would get as an answer the set $\{\mathsf{disease}(john, cancer)\}$. Likewise, the answer to the query $\mathsf{symptom}(john, X)$ is the set $\{\mathsf{symptom}(john, s_1)$, $\mathsf{symptom}(john, s_2)$, $\mathsf{symptom}(john, s_3)\}$.

We assumed that John and Jane want their diseases kept privately. However, the accountant can violate John's privacy by asking the query $\mathsf{symptom}(john, X)$. The answer that $acct$ would get from the system is $\{\mathsf{symptom}(john, s_1)$, $\mathsf{symptom}(john, s_2)$, $\mathsf{symptom}(john, s_3)\}$. However, recall that the accountant has some background knowledge including the rule

$$\mathsf{disease}(X, cancer) \leftarrow \mathsf{symptom}(X, s_2), \mathsf{symptom}(X, s_3)$$

which, with the answer of the query, would allow $acct$ to infer $\mathsf{disease}(john, cancer)$. Thus the privacy preserving answers to $\mathsf{symptom}(john, X)$ are

$$\begin{aligned}
Ans_1 &= \{\mathsf{symptom}(john, s_1), \mathsf{symptom}(john, s_2)\}\\
Ans_2 &= \{\mathsf{symptom}(john, s_1), \mathsf{symptom}(john, s_3)\}\\
Ans_3 &= \{\mathsf{symptom}(john, s_1)\}\\
Ans_4 &= \{\mathsf{symptom}(john, s_2)\}\\
Ans_5 &= \{\mathsf{symptom}(john, s_3)\}\\
Ans_6 &= \emptyset
\end{aligned}$$

None of these answers allows $acct$ to infer the private knowledge $\mathsf{disease}(john, cancer)$. However, except for the answers Ans_1 and Ans_2, which are maximal, all answers yield less information than could be disclosed without infringing privacy requirements. Any system should also provide only one of these answers to the user, because getting for instance both Ans_1 and Ans_2 would again violate John's privacy requirements.

In a practical system, upon disclosing an answer the system should update the respective user's knowledge model in order to avoid privacy infringements by repeated querying. For example, when the system returns Ans_1 to user $acct$, it should modify $\mathbf{BK}(acct)$ in order to reflect the fact that $acct$ now knows $\mathsf{symptom}(john, s_1)$ and $\mathsf{symptom}(john, s_2)$, such that asking the same query again it is made sure that $\mathsf{symptom}(john, s_3)$ will not be disclosed to $acct$.

A related aspect is how the background knowledge base is determined precisely. We do not want to restrict the formalism by making any assumption on this issue, but in practice this will be a system that is maintained dynamically and will derive from both exogenous knowledge (for instance, information provided by an administrator, which may also be information on what knowledge bases a user is actively using) and endogenous knowledge (as in the example above, answers disclosed by the system to a user in the past).

3 Solving Privacy Preservation Problems Using Multi-Context Systems and Hybrid Knowledge Bases

The definitions in Section 2 were already slightly geared towards multi-context systems. We recall that a multi-context system in the sense of [7] is a tuple $(C_1, \ldots, C_n)$ where for each i ($1 \leq i \leq n$), $C_i = (L_i, kb_i, br_i)$ where L_i is a logic, kb_i is a knowledge base of L_i and br_i is a set of L_i bridge rules over $\{L_1, \ldots, L_n\}$, where an L_i bridge rule over $\{L_1, \ldots, L_n\}$ is a construct

$$s \leftarrow (r_1 : p_1), \ldots, (r_j : p_j), \mathrm{not}\ (r_{j+1} : p_{j+1}), \ldots, \mathrm{not}\ (r_m : p_m)$$

where $1 \leq r_k \leq n$, p_k is an element of a belief set for L_{r_k} and, for each $kb \in \mathbf{KB}_i$, $kb \cup \{s\} \in \mathbf{KB}_i$. Such rules (without negation) were first introduced in [22] and later generalized to include negation (and much more) in [17] and are called "hybrid knowledge bases" based on annotated logic. [4] In the rest of this section, we use multi-construct systems in our syntax simply because a choice has to be made for syntax.

The semantics of a multi-context system is defined by means of *equilibria*. A *belief state* for a multi-context system $(C_1, \ldots, C_n)$ is $S = (S_1, \ldots, S_n)$, where $S_i \in \mathbf{BS}_i$ for $1 \leq i \leq n$. An L_i bridge rule of the form above is applicable in S iff, for $1 \leq k \leq j$, $p_k \in S_{r_k}$ holds and, for $j < k \leq m$, $p_k \notin S_{r_k}$ holds. Let $app(br, S)$ denote the set of all bridge rules in br which are applicable in a belief state S. A belief state $S = (S_1, \ldots, S_n)$ is an equilibrium of a multi-context system $(C_1, \ldots, C_n)$ iff for all $1 \leq i \leq n$, $S_i \in \mathbf{ACC}_i(kb_i \cup \{hd(r) \mid r \in app(br_i, S)\})$, where $hd(r)$ is the head of a bridge rule r, viz. s in the bridge rule schema given above.

[4] In [22], the main difference in this syntax was that the $r_i : p_i$'s were instead written $p_i : r_i$. [17] extended this to include not just "annotated" rules in the body, but also many other constructs including references to non-logical data structures and software instead of just logics L_i. However, [7] allows non-atomic constructs in rule bodies. Thus, [7] may be viewed as a generalization of [22] but not of [17].

Given a $\mathrm{P3}$ $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u)$, with $\mathbf{U} = \{u_1, \ldots, u_{|\mathbf{U}|}\}$, in order to identify privacy preserving answers, we build a multi-context system $M_{\mathrm{P3}} = (C_1, C_2, C_3, C_4, \ldots, C_{|\mathbf{U}|+3})$, where $C_1 = (L_{\mathbf{MKB}}, \mathbf{MKB}, \emptyset)$, $C_2 = (L_{\mathbf{MKB}}, \emptyset, br_2)$, $C_3 = (L_{\mathbf{MKB}}, \emptyset, br_3)$, $C_4 = (L_{\mathbf{BK}(u_1)}, \mathbf{BK}(u_1), br_4)$ $\ldots$, $C_{|\mathbf{U}|+3} = (L_{\mathbf{BK}(u_{|\mathbf{U}|})}, \mathbf{BK}(u_{|\mathbf{U}|}), br_{|\mathbf{U}|+3})$. Here L_{kb} is the logic of the knowledge base kb. The meaning is that C_1 provides just the belief sets for $\mathbf{MKB}$ (no bridge rules), C_2 and C_3 are used to identify those belief sets which constitute privacy preserving answers, while $C_4, \ldots, C_{|\mathbf{U}|+3}$ represent the user information, that is, the background knowledge base of the querying user and the privacy requirements of the other users. The important part are the bridge rules, which we will describe next. In many cases, we will create one rule for each symbol that can occur in some belief set of $\mathbf{Ans}(Q)$, so for convenience let $\mathcal{D} = \{p \mid p \in B, B \in \mathbf{Ans}(Q)\}$.

The set br_2 contains one bridge rule $p \leftarrow (1 : p), \mathrm{not}\ (3 : p)$ for each $p \in \mathcal{D}$. Symmetrically, br_3 contains one bridge rule $p \leftarrow (1 : p), \mathrm{not}\ (2 : p)$ for each $p \in \mathcal{D}$. The intuition is that the belief sets of C_2 will be subsets of the belief set of C_1 in any equilibrium, and hence potential privacy preserving answers. C_3 exists only for technical reasons.

For i such that $u_{i-2} = u$, thus for the context C_i of the querying user, we add one bridge rule $p \leftarrow (2 : p)$ for each $p \in \mathcal{D}$. This means that in any equilibrium, the belief set for i will contain all consequences of the privacy preserving answer with respect to u's knowledge base.

For each i where $3 < i \leq |\mathbf{U}|+3$ such that $u_{i-2} \neq u$, thus for contexts representing non-querying users, br_i contains one bridge rule $p_1 \leftarrow (j : p_1), \ldots, (j : p_l), \mathrm{not}\ (i : p_1)$ for $u_j = u$ and $\{p_1, \ldots, p_l\} \in \mathbf{Priv}(u_{i-2})$. The idea is that no belief state can be an equilibrium, in which the querying user derives information which u_{i-2} wants to keep private.

Note that the tuple $(S_1, S_2, S_3, S_4, \ldots, S_{|\mathbf{U}|+3})$ was constructed in such a way, that S_2 represents the potential privacy preserving answers. The following proposition shows that our construction does indeed reflect this.

Proposition 1. *Given a* $\mathrm{P3}$ $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u)$*, each equilibrium belief state* $(S_1, S_2, S_3, S_4, \ldots, S_{|\mathbf{U}|+3})$ *for* M_{P3} *is such that* S_2 *is a privacy preserving answer to* $\mathrm{P3}$*. Also, each privacy preserving answer* S *to* $\mathrm{P3}$ *is the second component of an equilibrium for* M_{P3}*.*

Example 3. In the example examined above, consider the $\mathrm{P3}$ $(\mathsf{MedKB}, \{john, jane, acct\}, \mathbf{BK}, \mathbf{Priv}, \mathsf{symptom}(john, X), acct)$. Note that we did not define background knowledge bases for users $john$ and $jane$, but their nature is not important for the example, just assume that they exist. We also have not defined any privacy statement for $acct$, but also this is not important for our example and we will assume that it is empty, that is, $acct$ does not require anything to be kept private. We construct a multi-context system $(C_1, C_2, C_3, C_4, C_5, C_6)$ where $C_1 = (L_{\mathsf{MedKB}}, \mathsf{MedKB}, \emptyset)$, $C_2 = (L_{\mathsf{MedKB}}, \emptyset, br_2)$ with bridge rules br_2 being

$$\mathsf{symptom}(john, s_1) \leftarrow (1 : \mathsf{symptom}(john, s_1)), \mathrm{not}\ (3 : \mathsf{symptom}(john, s_1))$$
$$\mathsf{symptom}(john, s_2) \leftarrow (1 : \mathsf{symptom}(john, s_2)), \mathrm{not}\ (3 : \mathsf{symptom}(john, s_2))$$
$$\mathsf{symptom}(john, s_3) \leftarrow (1 : \mathsf{symptom}(john, s_3)), \mathrm{not}\ (3 : \mathsf{symptom}(john, s_3))$$

then $C_3 = (L_{\mathsf{MedKB}}, \emptyset, br_3)$ with bridge rules br_3 being

$$\begin{aligned}
&\mathsf{symptom}(john, s_1) \leftarrow (1 : \mathsf{symptom}(john, s_1)), \text{not } (2 : \mathsf{symptom}(john, s_1))\\
&\mathsf{symptom}(john, s_2) \leftarrow (1 : \mathsf{symptom}(john, s_2)), \text{not } (2 : \mathsf{symptom}(john, s_2))\\
&\mathsf{symptom}(john, s_3) \leftarrow (1 : \mathsf{symptom}(john, s_3)), \text{not } (2 : \mathsf{symptom}(john, s_3))
\end{aligned}$$

then $C_4 = (L_{\mathbf{BK}(john)}, \mathbf{BK}(john), br_4)$ with bridge rules br_4 being

$$\mathsf{disease}(john, cancer) \leftarrow (6 : \mathsf{disease}(john, cancer)), \text{not } (4 : \mathsf{disease}(john, cancer))$$

then $C_5 = (L_{\mathbf{BK}(jane)}, \mathbf{BK}(jane), br_5)$ with bridge rules br_5 being

$$\mathsf{disease}(jane, aids) \leftarrow (6 : \mathsf{disease}(jane, aids)), \text{not } (5 : \mathsf{disease}(jane, aids)$$

and finally $C_6 = (L_{\mathbf{BK}(acct)}, \mathbf{BK}(acct), br_6)$ with bridge rules br_6 being

$$\begin{aligned}
&\mathsf{symptom}(john, s_1) \leftarrow (2 : \mathsf{symptom}(john, s_1))\\
&\mathsf{symptom}(john, s_2) \leftarrow (2 : \mathsf{symptom}(john, s_2))\\
&\mathsf{symptom}(john, s_3) \leftarrow (2 : \mathsf{symptom}(john, s_3))
\end{aligned}$$

M_{P3} has six equilibria

$$\begin{aligned}
E_1 &= (S_{\mathsf{MedKB}}, Ans_1, \mathbf{Ans}(\mathsf{symptom}(john, X)) \setminus Ans_1, Ans_1, \emptyset, \emptyset)\\
E_2 &= (S_{\mathsf{MedKB}}, Ans_2, \mathbf{Ans}(\mathsf{symptom}(john, X)) \setminus Ans_2, Ans_2, \emptyset, \emptyset)\\
E_3 &= (S_{\mathsf{MedKB}}, Ans_3, \mathbf{Ans}(\mathsf{symptom}(john, X)) \setminus Ans_3, Ans_3, \emptyset, \emptyset)\\
E_4 &= (S_{\mathsf{MedKB}}, Ans_4, \mathbf{Ans}(\mathsf{symptom}(john, X)) \setminus Ans_4, Ans_4, \emptyset, \emptyset)\\
E_5 &= (S_{\mathsf{MedKB}}, Ans_5, \mathbf{Ans}(\mathsf{symptom}(john, X)) \setminus Ans_5, Ans_5, \emptyset, \emptyset)\\
E_6 &= (S_{\mathsf{MedKB}}, Ans_6, \mathbf{Ans}(\mathsf{symptom}(john, X)) \setminus Ans_6, Ans_6, \emptyset, \emptyset)
\end{aligned}$$

where S_{MedKB} is as in Example 1 and the second belief set of each E_i is exactly the respective Ans_i of Example 2 and the third belief set is the complement of Ans_i with respect to $\mathbf{Ans}(\mathsf{symptom}(john, X)) = \{\mathsf{symptom}(john, s_1), \mathsf{symptom}(john, s_2), \mathsf{symptom}(john, s_3)\}$.

We would like to point out that in this construction the original knowledge bases are not changed, we only create contexts and bridge rules. All of the background knowledge bases could be multi-context systems themselves; for instance, if the user model for *acct* foresees that *acct* is aware of SNOMED and PEPID, then *acct*'s background knowledge base could be a multi-context system comprising these two medical knowledge bases.

In order to obtain maximal privacy preserving answers using the described construction, the simplest way is to postprocess all privacy preserving answers. More involved solutions would have to interfere with the underlying multi-context system reasoner, for instance by dynamically changing the multi-context system. It is not clear to us at the moment whether it is possible to modify the construction such that the equilibria of the obtained multi-context system correspond directly to the maximal privacy preserving answers.

We note that the "equilibria" in multi-context systems are similar to the non-monotonic constructs in the prior "hybrid knowledge bases" work of [17] based on annotated logic and so the results of this section also show that hybrid knowledge bases can be used to encode privacy constructs. Hybrid knowledge bases were extensively implemented and used to build a very large number of applications on top of real databases [3].

4 Solving First-Order Privacy Preservation Problems Using Default Logic

In this section, we show that the formalism in [10], called *first-order privacy preservation problem*, is an instance of the formalism defined in Section 2.

4.1 First-Order Privacy Preservation Problems

For defining a first-order language (without equality), we assume the existence of some finite set of constant symbols, function symbols and predicate symbols. As usual, a term is inductively defined as follows: (i) Each constant is a term, (ii) Each variable is a term, and (iii) if f is an n-ary predicate symbol and $t_1, \ldots, t_n$ are terms, then $f(t_1, \ldots, t_n)$ is a term. A *ground term* is any term that contains no variable symbols. Similarly, if p is an n-ary predicate symbol and $t_1, \ldots, t_n$ are terms, then $p(t_1, \ldots, t_n)$ is an atom. A *ground atom* is any atom that contains no variable symbols. A well formed formula (wff) is inductively defined as follows. (i) Every atom is a wff, (ii) If F, G are wffs then so are $(F \wedge G), (F \vee G)$ and $\neg F$. The semantics is given as usual (all formulae are considered to be universally quantified, so we do not need to introduce quantifiers).

Definition 6 (first-order privacy preservation problem). *A first-order privacy preservation problem* $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$ *is a* P3 *in which* $\mathbf{MKB}$ *is a set of ground atoms (also called logic database), each* $\mathbf{BK}^t(u)$ *is a set of wffs, and* $\mathbf{Priv}(u)$ *is also a set of wffs, represented by its set of models.*

Now, given a first-order privacy preservation problem $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$, we define a translation trans, which produces a default logic theory $\Delta = (D, W)$ such that there is a bijective correspondence between the solutions to the privacy preservation problem and the extensions of the default theory (restricted to the query) returned by the translation [8]. The consequence of this translation is that standard (and well studied) methods to evaluate default logic theories may be used to preserve privacy effectively, efficiently, and elegantly.

We refer to standard textbooks (e.g. [18, 6]) for an introduction to default theories. We denote defaults as usual by $\frac{a \ : \ b}{c}$ *if* a *holds and it is consistent to assume* b *then conclude* c. Most of our defaults are supernormal, i.e. of the form $\frac{: \ f}{f}$. A default theory is a pair $\Delta = (D, W)$ where the first component consists of the whole set of defaults and the second is a set of formulae (the classical theory).

Definition 7 (trans). *Let* $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$ *be a first-order privacy preservation problem. The* translation*, trans*$(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$ *of a privacy preservation problem into default logic is the default logic theory* $\Delta = (D, W)$ *where:*

$$\begin{aligned}
W &= \mathbf{BK}(u_0).\\
D &= \{\frac{: \ f}{f} \mid f \in \mathbf{MKB}\} \bigcup\\
&\quad \{\frac{p \ :}{\neg p} \mid (\exists u \in \mathbf{U} - \{u_0\})\ p \in \mathbf{Priv}(u) \textit{ and } \mathbf{BK}(u_0) \not\models p\}.
\end{aligned}$$

We now present an example to show how the result of transforming the privacy preservation problem into default logic looks like.

Example 4. Let us return to the case of the accountant. Assume that MedKB of Example 1 is a logic database, that the "rules" of $\mathbf{BK}$ and $\mathbf{Priv}$ in Example 1 are wffs, making the problem in the example a first-order privacy preservation problem. In this case, W consists of the following two rules (which need to be written slightly differently so as to comply with the wff's as defined in the beginning of this section):

$$\mathsf{symptom}(X, s_1)\,\&\,\mathsf{symptom}(X, s_4) \rightarrow \mathsf{disease}(X, aids)$$
$$\mathsf{symptom}(X, s_2)\,\&\,\mathsf{symptom}(X, s_3) \rightarrow \mathsf{disease}(X, cancer).$$

In addition, D consists of the following defaults:

$$\frac{:\ \mathsf{symptom}(john{,}s_1)}{\mathsf{symptom}(john{,}s_1)} \qquad \frac{:\ \mathsf{symptom}(john{,}s_2)}{\mathsf{symptom}(john{,}s_2)} \qquad \frac{:\ \mathsf{symptom}(john{,}s_3)}{\mathsf{symptom}(john{,}s_3)}$$

$$\frac{:\ \mathsf{symptom}(jane{,}s_1)}{\mathsf{symptom}(jane{,}s_1)} \qquad \frac{:\ \mathsf{symptom}(jane{,}s_4)}{\mathsf{symptom}(jane{,}s_4)}$$

$$\frac{:\ \mathsf{disease}(ed{,}polio)}{\mathsf{disease}(ed{,}polio)} \qquad \frac{:\ \mathsf{disease}(jane{,}aids)}{\mathsf{disease}(jane{,}aids)} \qquad \frac{:\ \mathsf{disease}(john{,}cancer)}{\mathsf{disease}(john{,}cancer)}$$

$$\frac{\mathsf{disease}(jane{,}aids)\ :}{\neg\mathsf{disease}(jane{,}aids)} \qquad \frac{\mathsf{disease}(john{,}cancer)\ :}{\neg\mathsf{disease}(john{,}cancer)}$$

Note that we are assuming here that Ed has not marked his disease as being a private fact.

Our translation uses linear space. The time complexity of the translation depends on the complexity of checking entailment. For example, assuming a finite number of constants in our language (reasonable) and assuming that all rules in $\mathbf{BK}$ are definite clauses (i.e. clauses with exactly one atom), then the translation is implementable in polynomial time. But if $\mathbf{BK}$ consists of arbitrary first order formulas, then the translation can take exponential time.

We remind the reader of some basic terminology associated with default theories. Given a default $d = \frac{\alpha:\beta}{\gamma}$, we use the notation $pre(d)$ to denote α, $j(d)$ to denote β and $c(d)$ to denote γ. In addition, given any default theory $\Delta = (D, W)$, we may associate with Δ, a mapping Γ_Δ which maps sets of wffs to sets of wffs. $\Gamma_\Delta(Y) = \mathsf{CN}(W \cup \{pre(d) \rightarrow c(d) \mid j(d) \text{ is consistent with } Y\})$. As usual, the function $\mathsf{CN}(X)$ denotes the set of all first order logical consequences of X. A set Y of wffs is an *extension* of Δ iff $Y = \Gamma_\Delta(Y)$.

We are now ready to present a key result linking the privacy preservation problem to default logic extensions. Suppose we consider any privacy preservation problem. The privacy preserving answers to that privacy preservation problem are in a one-one correspondence with the consistent extensions of the translation (restricted to the query) of the privacy preservation problem into default logic (using the translation trans shown in Definition 7).

Theorem 1. *Suppose that Q is an atom and that $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$ is a first-order privacy preservation problem and $\mathsf{trans}(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0) = \Delta = (D, W)$. Then: X is a solution to the above privacy preservation problem iff there is a consistent extension E of $\Delta = (D, W)$ such that $X = \{A\theta \mid A\theta \in E \cap \mathbf{MKB}\}$.*

In order to prove Theorem 1, we first formulate a useful abstract lemma.

Lemma 1. *Let W, MKB and P be consistent sets of formulae s.t. $W \cup MKB$ is consistent as well. Let $D_P = \{\frac{p\,:}{\neg p} \;:\; p \in P\}$ and $D_{MKB} = \{\frac{:\,f}{f} \;:\; f \in MKB\}$.*

Then the consistent extensions of the theory $(D_P \cup D_{MKB}, W)$ are the sets $Cn(W \cup \{f :\; f \in F\})$ where F is a subset of MKB that is maximal wrt. set inclusion (i.e. there is no larger set F' such that $W \cup \{f :\; f \in F'\} \not\models p$ for all $p \in P$).

Proof. Clearly the sets $Cn(W \cup \{f :\; f \in F\})$ where F is a maximal subset of MKB are extensions of the default theory: The defaults in D_P do not apply and we are left with a supernormal default theory (the result follows from well-known characterizations in default logic, see eg. [9, 18]).

Conversely, let E be a consistent extension. Then no default in D_P applies. Because extensions are grounded and we are dealing with a supernormal theory, E must have the form $Cn(W \cup \{f :\; f \in F\})$ for a subset F of MKB. Because E is maximal (no other extension can contain E), the set $Cn(W \cup \{f :\; f \in F\})$ is maximal in the sense defined in the lemma. □

Now we are able to prove Theorem 1:

Proof. The proof of Theorem 1 is an application of Lemma 1. Suppose X is a solution to $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$ and let $\mathsf{trans}(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0) = \Delta = (D, W)$. Then we let $F := X$, $W := \mathbf{BK}(u_0)$ and $P := \{p :\; (\exists u \in \mathbf{U} - \{u_0\})\; p \in \mathbf{Priv}(u)$ and $\mathbf{BK}(u_0) \not\models p\}$ and apply our lemma. The set $Cn(W \cup \{f :\; f \in F\})$ is an extension (it is maximal because of (3) and (2) in the definition of a privacy preserving answer).

Conversely let a consistent extension E of $\mathsf{trans}(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$ be given and consider $X := \{Q\theta \mid Q\theta \in E \cap \mathbf{MKB}\}$. Our lemma implies that X is a subset of MKB that is maximal. Therefore X is also a privacy preserving answer (if there were a larger X' satisfying (2) in the definition of pp answer, then E would not be maximal and thus not be an extension). □

The preceding theorem applies to *atomic* queries. A straightforward extension of the above proof gives us the following corollary, which applies to arbitrary queries.

Corollary 1. *Suppose that $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$ is a privacy preservation problem and that $\mathsf{trans}(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0) = (D, W)$. Then: X is a solution to the above privacy preservation problem iff there is a consistent extension E of (D, W) such that $X = \{Q\theta \mid Q\theta \in E \cap \mathbf{MKB}\}$.*

In order to illustrate this theorem, we revisit the example privacy preservation problem and its default logic translation that we presented earlier.

Example 5. Let us return to the MedKB example. Consider the privacy preservation problem of Example 1 and the default logic translation shown in Example 4. As seen in Example 1, there are two privacy preserving answers to this problem. They are:

$$Ans1 = \{\mathsf{symptom}(john, s_1), \mathsf{symptom}(john, s_2)\}$$
$$Ans2 = \{\mathsf{symptom}(john, s_1), \mathsf{symptom}(john, s_3)\}$$

The default logic translation of this privacy preservation problem shown in Example 4 has exactly four consistent extensions $E_1, \ldots, E_4$.

$$\begin{aligned} E_1 &= \mathsf{CN}(W \cup \{\mathsf{symptom}(john, s_1), \mathsf{symptom}(john, s_2),\\ &\qquad \mathsf{symptom}(jane, s_1), \mathsf{disease}(ed, polio)\})\\ E_2 &= \mathsf{CN}(W \cup \{\mathsf{symptom}(john, s_1), \mathsf{symptom}(john, s_3),\\ &\qquad \mathsf{symptom}(jane, s_1), \mathsf{disease}(ed, polio)\})\\ E_3 &= \mathsf{CN}(W \cup \{\mathsf{symptom}(john, s_1), \mathsf{symptom}(john, s_2),\\ &\qquad \mathsf{symptom}(jane, s_4), \mathsf{disease}(ed, polio)\})\\ E_4 &= \mathsf{CN}(W \cup \{\mathsf{symptom}(john, s_1), \mathsf{symptom}(john, s_3),\\ &\qquad \mathsf{symptom}(jane, s_4), \mathsf{disease}(ed, polio)\}) \end{aligned}$$

However, if we are only interested in answers to the query $\mathsf{symptom}(john, X)$ in the above extensions, then the extensions E_1, E_4 only contain $\{\mathsf{symptom}(john, s_1),$ $\mathsf{symptom}(john, s_2)\}$ while E_2, E_3 only contain $\{\mathsf{symptom}(john, s_1), \mathsf{symptom}(john, s_3)\}$. These restrictions of the extensions are in a one-one correspondence with the privacy preserving answers to the query posed by the accountant.

4.2 Complexity of First-Order Privacy Preservation Problems

Computing a privacy-preserving answer typically involves *guessing* a subset of answers, and subsequently checking it with respect to privacy preservation and maximality. Intuitively, this computational task has a correspondence to common non-monotonic reasoning tasks, because the maximality condition for privacy-preserving answers has its counterpart the minimality conditions in non-monotonic semantics, while guessing a model candidate and checking it on a set of formulae is even more closely related.

It therefore does not come as a surprise that a non-monotonic logic neatly represents the privacy preservation problem. Concerning the complexity analysis, we can indeed leverage the translation trans to use well-known results concerning the complexity of default logic in order to prove membership of various subclasses of first-order privacy preservation problems.

As already shown in [19], default reasoning involving function symbols is undecidable. Note that computing maximal privacy preserving answers involves checking $\mathbf{BK}(u_0) \not\models p$, which is clearly undecidable for arbitrary first-order formulae. We will therefore focus on decidable fragments. In particular, we will assume in our analysis below that problems are restricted to those for which deciding $\mathbf{BK} \not\models p$, $p \in \mathbf{Priv}$ is feasible in polynomial time. We will focus on theories in a Datalog setting, the data complexity (we consider only $\mathbf{MKB}$, i.e. the knowledge base, as input, $\mathbf{BK}$ and $\mathbf{Priv}$ are fixed) of which corresponds to propositional default theories.

Then, membership can be seen by virtue of trans and the form of formulae in $\mathbf{BK}$ and $\mathbf{Priv}$. In particular, brave reasoning for non-disjunctive default theories is NP-complete (see e.g. [16, 20] for such classes), while brave reasoning for arbitrary default theories is Σ_2^P-complete, see [14] and [21].

We thus consider first-order privacy preservation problems with the following restrictions:

1. We vary $\mathbf{BK}(u)$ to be an arbitrary theory (without syntactic retrictions), a non-disjunctive theory (as in [16, 20]), and a set of facts (a theory containing only ground atoms).
2. We vary $\mathbf{Priv}(u)$ to be a set of arbitrary formulas, a non-disjunctive theory, and a set of facts.

Table 1 summarizes our results on the complexity of privacy preservation in the Datalog case.

Priv/BK	Facts	Non-disjunctive	Arbitrary
Facts	P	P	Σ_2^P
Non-disjunctive	NP	NP	Σ_2^P
Arbitrary	Σ_2^P	Σ_2^P	Σ_2^P

Table 1. Data Complexity of First-Order Privacy Preservation Problems

Theorem 2. *The data complexity for first-order privacy preservation problems without function symbols under various syntactic restrictions are as reported in Table 1. Completeness holds for* NP *and* Σ_2^P *results.*

Next, we will prove some of the hardness results.

Corollary 2. *First-order privacy preservation problems with* $\mathbf{BK}$ *containing non-disjunctive rules and* $\mathbf{Priv}$ *made of facts is hard for* NP.

Proof. We show NP-hardness by a reduction from 3SAT to a first-order privacy preservation problem in which $\mathbf{BK}()$ contains only rules with negation on $\mathbf{MKB}$ predicates and in which $\mathbf{Priv}$ contains only one fact: Given a CNF $\phi = \bigwedge_{i=1}^{n} L_{i,1} \vee L_{i,2} \vee L_{i,3}$, we create a P3 with $\mathbf{MKB} = \{c_i \mid c_i \text{ is an atom in } \phi\} \cup \{q\}$, two users u_0, u_1, $\mathbf{BK}(u_0) = \{L'_{i,1} \wedge L'_{i,2} \wedge L'_{i,3} \rightarrow unsat\}$, where $(\neg x)' = x$ and $x' = \neg x$. Finally, $\mathbf{Priv}(u_1) = \{unsat\}$, and $Q = q$. It is not hard to see that q is an answer iff ϕ is satisfiable: If q is an answer, then a truth assignment can be obtained from the subset $X \subseteq \mathbf{MKB}$ in which exactly the c_i in X are interpreted as true.

As $X \cup \mathbf{BK}(()u_0) \not\models unsat$, no conjunct in ϕ evaluates to false under this assignment, which therefore satisfies ϕ. Conversely, if ϕ is satisfiable, each cardinality maximal satisfying truth assignment induces an $X \subseteq \mathbf{MKB}$, such that $X \cup \mathbf{BK}(()u_0) \not\models unsat$. □

Corollary 3. *First-order privacy preservation problems with empty* $\mathbf{BK}$ *and arbitrary* $\mathbf{Priv}$ *are hard for* Σ_2^P*.*

Proof. We show Σ_2^P-hardness by a reduction from a $QBF_{2,\exists}$ to a P3 in which $\mathbf{BK}$ is empty and $\mathbf{Priv}$ contains arbitrary formulae. Consider $\psi = \exists x_1 \cdots \exists x_n \forall y_1 \cdots \forall y_m \phi$, where ϕ is a propositional formula. We create a P3 with $\mathbf{MKB} = \{x_1, \ldots, x_n\} \cup \{q\}$, two users u_0, u_1, $\mathbf{Priv}(u_1) = \{\neg\phi\}$, and $Q = q$. An answer X induces a valuation ν of the existentially quantified variables. Then, no extension ν' of ν to the universally quantified variables can exist such that ϕ is false, hence ψ is valid. Conversely, if ψ is valid, each cardinality maximal satisfying truth assignment for $x_1, \ldots, x_n$ induces an answer. □

This proof can easily be adapted so that $\mathbf{BK}(u_0)$ contains the arbitrary formula $(\neg\phi) \rightarrow unsat$ and $\mathbf{Priv}(u_1)$ contains only $unsat$.

All complexity results above refer to propositional theories or data complexity. In our setting this means that only $\mathbf{MKB}$ is considered as input, while especially $\mathbf{BK}$ and $\mathbf{Priv}$ are considered to be fixed. For considering program complexity, i.e. the knowledge base $\mathbf{MKB}$ is fixed but $\mathbf{BK}$ and $\mathbf{Priv}$ are considered as inputs, we can adapt the data complexity results by using techniques from [15]. Due to space constraints, we do not present proofs. It is obvious that allowing programs (not just facts) as input increases the complexity problem. This is shown in Table 2. Allowing function symbols would make all problems undecidable.

Theorem 3. *The program complexity for problems without function symbols under various syntactic restrictions are as reported in the Table 2.*

Priv/BK	Facts	Non-disj.	Arbitrary
Facts	EXPTIME	EXPTIME	NEXPTIME$^{\mathrm{NP}}$
Non-disj.	NEXPTIME	NEXPTIME	NEXPTIME$^{\mathrm{NP}}$
Arbitrary	NEXPTIME$^{\mathrm{NP}}$	NEXPTIME$^{\mathrm{NP}}$	NEXPTIME$^{\mathrm{NP}}$

Table 2. Program Complexity of Privacy Preservation Problems

4.3 Algorithm for First-Order Privacy Preservation Problems

We now describe an algorithm that leverages our translation to default logic. First and foremost, we recall the important observation of [2] that Reiter's Γ_Δ operator is anti-monotonic - hence, the operator Γ_Δ^2 that applies Γ_Δ is monotonic. As a consequence, Γ_Δ^2 has both a least fixpoint and a greatest fixpoint, denoted $\mathsf{lfp}(\Gamma_\Delta^2)$ and $\mathsf{gfp}(\Gamma_\Delta^2)$ respectively.

Theorem 4 ([2]). *Recall the following properties:*

1. *If* $Y_1 \subseteq Y_2$ *then* $\Gamma_\Delta(Y_2) \subseteq \Gamma_\Delta(Y_1)$.

2. *Γ^2_Δ has a least and a greatest fixpoint, denoted respectively as $\mathsf{lfp}(\Gamma^2_\Delta)$ and $\mathsf{gfp}(\Gamma^2_\Delta)$.*
3. *$\Gamma_\Delta(\mathsf{lfp}(\Gamma^2_\Delta)) = \mathsf{gfp}(\Gamma^2_\Delta)$.*

An immediate consequence of the above theorem is that one can compute extensions of default theories by first computing $\mathsf{lfp}(\Gamma^2_\Delta)$ and $\mathsf{gfp}(\Gamma^2_\Delta)$. Anything in $\mathsf{lfp}(\Gamma^2_\Delta)$ is true in all extensions, while anything not in $\mathsf{gfp}(\Gamma^2_\Delta)$ is false in all extensions. We can therefore start by computing both $\mathsf{lfp}(\Gamma^2_\Delta)$ and $\mathsf{gfp}(\Gamma^2_\Delta)$. If $\mathsf{lfp}(\Gamma^2_\Delta)$ is not an extension, we nondeterministically add things in $\mathsf{gfp}(\Gamma^2_\Delta)$ to the default theory and iteratively compute the least fixpoint of Γ^2_Δ w.r.t. the modified theory. This algorithm for arbitrary default theories gives rise to the specialization for computing answers depicted in Figure 1.

```
P3Alg($\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0$)
$\Delta = \mathsf{trans}(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0) = (D, W)$;
$Todo = \mathbf{MKB} \cap (\mathsf{gfp}(\Gamma^2_\Delta) \setminus \mathsf{lfp}(\Gamma^2_\Delta))$;
if $\mathsf{lfp}(\Gamma^2_\Delta) = \Gamma_\Delta(\mathsf{lfp}(\Gamma^2_\Delta))$ then
    $done = true$;
while $Todo \neq \emptyset \wedge \neg done$ do
    Nondeterministically select an $a \in Todo$;
    Let $\Delta = (D, W \cup \{a\})$;
    if $\mathsf{lfp}(\Gamma^2_\Delta) = \Gamma_\Delta(\mathsf{lfp}(\Gamma^2_\Delta))$ then
        $done = true$;
    else
        $Todo = Todo \setminus \{a\}$;
% end-while
return $\mathbf{MKB} \cap \mathsf{lfp}(\Gamma^2_\Delta)$;
```

Fig. 1. Algorithm computing privacy preserving answers.

The algorithm proceeds as follows: First the problem is translated to a default theory using trans. Subsequently, the least and greatest fixpoint of Γ^2_Δ are computed. Anything which is in the greatest, but not in the least fixpoint can or cannot be true in some extension, so we store it in $Todo$ to nondeterministically assume its truth.

The crucial point here is that we restrict these nondeterministic choices to $\mathbf{MKB}$, which can dramatically decrease the search space. Then we enter the nondeterministic phase of the algorithm, in which a truth assignment for $Todo$ is generated until a fixpoint (i.e., an extension) is reached, if at all. As a final step, a projection of the extension onto $\mathbf{MKB}$ is generated. The following proposition states that the above algorithm is always guaranteed to return the correct answer.

Proposition 2. *Let $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$ be a first-order privacy preservation problem. Then the algorithm* **P3Alg**$(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$ *returns X iff X is a privacy preserving answer to* $(\mathbf{MKB}, \mathbf{U}, \mathbf{BK}, \mathbf{Priv}, Q, u_0)$.

5 Conclusion and Future Work

We have presented a general definition of the privacy preservation problem, which allows for using knowledge bases of different kinds. Finding privacy preserving answers can then be accomplished by building an appropriate multi-context system and computing one of its belief states. Since systems for solving multi-context systems begin to emerge, for example DMCS [1], this also implies that these privacy preserving answers can be effectively computed.

However, usually one is interested in maximal privacy preserving answers. It is unclear to us whether a similar construction to the one presented in this paper can be used for finding privacy preserving answers which are maximal, by just creating appropriate contexts and bridge rules and without modifying the involved knowledge bases or adding new knowledge bases of particular logics. One possible line of investigation is to examine work on diagnosing inconsistent multi-context systems [11, 4], since in diagnosis tasks there is an implicit minimization criterion, which could be exploited for encoding maximality.

Furthermore, we showed that the formalism subsumes an earlier definition of the privacy preservation problem, for which it is possible to determine maximal privacy preserving answers by a translation into default logic. For this formalism, we conjecture that a similar transformation to first-order theories interpreted using the stable model semantics [13] exists. In future work, we intend to investigate such a transformation in detail.

References

1. Bairakdar, S.E., Dao-Tran, M., Eiter, T., Fink, M., Krennwallner, T.: The DMCS Solver for Distributed Nonmonotonic Multi-Context Systems. In: Janhunen, T., Niemelä, I. (eds.) Proceedings of the 12th European Conference on Logics in Artificial Intelligence (JELIA 2010). Lecture Notes in Computer Science, vol. 6341, pp. 352–355. Springer Verlag (2010)
2. Baral, C., Subrahmanian, V.: Dualities Between Alternative Semantics for Logic Programming and Non-Monotonic Reasoning. Journal of Automated Reasoning 10(3), 399–420 (1993)
3. Benton, J., Subrahmanian, V.: Hybrid Knowledge Bases for Missile Siting Applications. In: IEEE Conference on AI Applications. pp. 141–148 (1993)
4. Bögl, M., Eiter, T., Fink, M., Schüller, P.: The mcs-ie System for Explaining Inconsistency in Multi-Context Systems. In: Janhunen, T., Niemelä, I. (eds.) Proceedings of the 12th European Conference on Logics in Artificial Intelligence (JELIA 2010). Lecture Notes in Computer Science, vol. 6341, pp. 356–359. Springer Verlag (2010)
5. Bonatti, P.A., Kraus, S., Subrahmanian, V.: Foundations of secure deductive databases. IEEE Transactions on Knowledge and Data Engineering 7(3), 406–422 (1995)
6. Brewka, G., Dix, J., Konolige, K.: Nonmonotonic Reasoning: An Overview, CSLI Lecture Notes, vol. 73. CSLI Publications, Stanford, CA (1997)
7. Brewka, G., Eiter, T.: Equilibria in Heterogeneous Nonmonotonic Multi-Context Systems. In: Proceedings of the Twenty-Second National Conference on Artificial Intelligence (AAAI-2007). pp. 385–390. AAAI Press (2007)
8. Cadoli, M., Eiter, T., Gottlob, G.: Default Logic as a Query Language. IEEE Transactions on Knowledge and Data Engineering 9(3), 448–463 (May/June 1997)

9. Dix, J.: Default Theories of Poole-Type and a Method for Constructing Cumulative Versions of Default Logic. In: Neumann, B. (ed.) Proc. of 10th European Conf. on Artificial Intelligence ECAI 92. pp. 289–293. John Wiley & Sons (1992)
10. Dix, J., Faber, W., Subrahmanian, V.: The Relationship between Reasoning about Privacy and Default Logics. In: Sutcliffe, G., Voronkov, A. (eds.) Logic for Programming, Artificial Intelligence, and Reasoning, 12th International Conference, LPAR 2005. Lecture Notes in Computer Science, vol. 3835, pp. 637–650. Springer Verlag (Dec 2005)
11. Eiter, T., Fink, M., Schüller, P., Weinzierl, A.: Finding Explanations of Inconsistency in Multi-Context Systems. In: Lin, F., Sattler, U., Truszczyński, M. (eds.) Proceedings of the Twelfth International Conference on Knowledge Representation and Reasoning (KR 2010). AAAI Press (2010)
12. Faber, W.: Privacy Preservation Using Multi-Context Systems. In: Mileo, A., Fink, M. (eds.) Proceedings of the 2nd International Workshop on Logic-based Interpretation of Context: Modeling and Applications. pp. 45–51 (May 2011)
13. Ferraris, P., Lee, J., Lifschitz, V.: A New Perspective on Stable Models. In: Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07). pp. 372–379 (Jan 2007)
14. Gottlob, G.: Complexity Results for Nonmonotonic Logics. Journal of Logic and Computation 2(3), 397–425 (1992)
15. Gottlob, G., Leone, N., Veith, H.: Succinctness as a Source of Expression Complexity. Annals of Pure and Applied Logic 97(1–3), 231–260 (1999)
16. Kautz, H., Selman, B.: Hard Problems for Simple Default Logics. Artificial Intelligence 49, 243–279 (1991)
17. Lu, J., Nerode, A., Subrahmanian, V.: Hybrid Knowedge Bases. IEEE Transactions on Knowledge and Data Engineering 8(3), 773–785 (1996)
18. Marek, W., Truszczyński, M.: Nonmonotonic Logics; Context-Dependent Reasoning. Springer, Berlin, 1st edn. (1993)
19. Reiter, R.: A Logic for Default Reasoning. Artificial Intelligence 13(1–2), 81–132 (1980)
20. Stillman, J.: It's Not My Default: The Complexity of Membership Problems in Restricted Propositional Default Logic. In: Proceedings AAAI-90. pp. 571–579 (1990)
21. Stillman, J.: The Complexity of Propositional Default Logic. In: Proceedings AAAI-92. pp. 794–799 (1992)
22. Subrahmanian, V.: Amalgamating Knowedge Bases. ACM Transactions on Database Systems 19(2), 291–331 (1994)
23. Winslett, M., Smith, K., Qian, X.: Formal query languages for secure relational databases. ACM Transactions on Database Systems 19(4), 626–662 (1994)
24. Zhao, L., Qian, J., Chang, L., Cai, G.: Using ASP for knowledge management with user authorization. Data & Knowledge Engineering 69(8), 737–762 (2010)