# Privacy in Database Publishing

Alin Deutsch[⋆] and Yannis Papakonstantinou[⋆⋆]

Department of Computer Science and Engineering
University of California, San Diego
{deutsch,yannis}@cs.ucsd.edu

**Abstract.** We formulate and study a privacy guarantee to data owners, who share information with clients by publishing views of a proprietary database. The owner identifies the sensitive proprietary data using a secret query against the proprietary database. Given an extra view, the privacy guarantee ensures that potential attackers will not learn any information about the secret that could not already be obtained from the existing views. We define "learning" as the modification of the attacker's a-priori probability distribution on the set of possible secrets. We assume arbitrary a-priori distributions (including distributions that correlate the existence of particular tuples) and solve the problem when secret and views are expressed as unions of conjunctive queries with non-equalities, under integrity constraints. We consider guarantees (a) for given view extents (b) for given domain of the secret and (c) independent of the domain and extents.

## 1 Introduction

Database publishing systems export a set of views of a proprietary database. Clients can access proprietary data only by formulating queries against the views. Data owners are subject to two conflicting requirements when designing a publishing system. On one hand, they need to publish appropriate views of the proprietary data to support the various types of interactions with the clients. On the other hand they must protect sensitive proprietary data. The purpose of this work is to provide a privacy guarantee as well as algorithms for checking it.

**The Publishing Setting.** We consider the following setting, which corresponds to the Global-As-View data integration scenario [12, 18]. We are given a proprietary relational database of schema $\mathcal{PR}$, a set of constraints $\Delta$ formulated in terms of $\mathcal{PR}$ and a set of relational views $\bar{V}$ over $\mathcal{PR}$. The public schema $\mathcal{PU}$ is the collection of all view names. The data owner identifies the sensitive proprietary data using a *secret query* $S$ against $\mathcal{PR}$. Note that no client can ask such a query, as the system only accepts queries against $\mathcal{PU}$. Instead, the attacking client (from now on referred to as *attacker* or *client*) can try to formulate a series of legal queries against $\mathcal{PU}$ (which the system is bound to answer) and combine their results locally to obtain information on the secret answer to $S$, where the notion of "obtaining information" on the secret will be refined shortly. The data owner wants to defend against such attacks.

[⋆] Supported by NSF/CAREER 0347968. [⋆⋆] Supported by NSF/ITR 0313384.

**A Relativized Privacy Guarantee.** We formulate and study a guarantee pertaining to the effect of adding new views in addition to the ones that are already posted. More specifically, we assume that the owner considers the publishing of a new view $N$. While the owner accepts the partial disclosure of the secret by the views $\bar{V}$, he is willing to add $N$ only if it does not disclose any additional information. We view "disclosure" in its strongest, information-theoretic sense: we model the attacker's a priori beliefs about the secret by an assignment of probabilities to the possible secrets and guarantee that, regardless of the a priori beliefs/probabilities of the attacker, knowledge of the extent of view $N$ does not lead to a revision of the a priori beliefs/probabilities, even if the attacker has unbounded computational resources. We first illustrate the key intuitions behind the proposed guarantee with examples.

Example 1 shows that in the common case when the owner cannot make assumptions on what the attacker already knows, the guarantee has to be quantified over all a-priori probability assignments to secrets assumed by the attacker

*Example 1.* Consider the proprietary relational schema

$$\mathcal{PR} = \{RS(reviewer, subcom)\ SP(subcom, paper)\ RP(reviewer, paper)\}$$

where $RS$ associates reviewers with the program subcommittee they belong to, $SP$ associates each paper to the subcommittee it was assigned to, and $RP$ associates reviewers with the papers they reviewed.

The database satisfies the set of constraints $\Delta = \{C_1, C_2, C_3\}$:

$$\begin{array}{l} C_1 : RP[reviewer] \subseteq RS[reviewer] \\ C_2 : RP[paper] \subseteq SP[paper] \\ C_3 : \forall r \forall p\ RP(r,p) \rightarrow \exists c\ RS(r,c) \wedge SP(c,p) \end{array}$$

where $C_1$ states that there are no paper reviewers besides those listed in subcommittees and $C_2$ states that every reviewed paper belongs to a subcommittee, and $C_3$ states that papers submitted to subcommittee $c$ can only be reviewed by reviewers associated to $c$.

| RS | reviewer | subcom |
|---|---|---|
| | $r_1$ | $c_1$ |
| | $r_2$ | $c_1$ |
| | $r_3$ | $c_2$ |
| | $r_4$ | $c_2$ |

| SP | subcom | paper |
|---|---|---|
| | $c_1$ | $p_1$ |
| | $c_1$ | $p_2$ |
| | $c_2$ | $p_3$ |
| | $c_2$ | $p_4$ |

| RP | reviewer | paper |
|---|---|---|
| | $r_1$ | $p_1$ |
| | $r_2$ | $p_2$ |
| | $r_3$ | $p_3$ |
| | $r_4$ | $p_4$ |

**Fig. 1.** Instance $I$ for Example 1

The example instance $I$ appears in Figure 1. Let the public data be described by the schema $\mathcal{PU} = \{V_R, V_S\}$ where the views $V_R, V_S$ expose respectively the set of reviewers and subcommittees:

$$V_R(r) \leftarrow RS(r,s) \qquad V_S(s) \leftarrow RS(r,s)$$

with extents $\{r_1, r_2, r_3, r_4\}$ and $\{c_1, c_2\}$ respectively, when evaluated on $I$. We investigate the privacy breaches associated with posting the additional views

$$V_{RS}(r, s) \leftarrow RS(r, s) \qquad V_{SP}(s, p) \leftarrow SP(s, p).$$

Of course we want to prevent outsiders from obtaining information about who reviewed a given paper, i.e., from changing their a-priori belief on the likelihood of each fact of the form "a given reviewer reviewed $p_1$". Let's say we want to hide who reviewed paper $p_1$.[1] This can be stated precisely as the following secret query against the proprietary schema:

$$S(r) \leftarrow RP(r, p_1).$$

In the absence of knowledge besides $V_R$ and $V_S$, any subset of $V_R$'s extent could have reviewed paper $p_1$. The set of possible secrets therefore contains among others the candidates $s_1 = \{r_1, r_3\}$ $s_2 = \{r_1, r_2\}$, $s_3 = \{r_1, r_2, r_4\}$, etc.

Let's assume that the attacker's domain knowledge (e.g., his assumptions on who is likely to bid and who has declared conflict-of-interest) prompts him to assign a non-zero probability $\texttt{prob}_1$ to $s_1$. If the owner now publishes the extent of $V_{RS}$, the attacker realizes (using constraint $C_3$) that $p_1$ must have been reviewed by somebody who serves on committee $c_1$, unlike $r_3$. The attacker thus adjusts $\texttt{prob}_1$ and $\texttt{prob}_3$ to 0, distributing their value among the probabilities of the remaining possible secrets (such as $s_2$). In other words, the remaining possible secrets are more likely after seeing the extent of the new views. This adjustment is due to learning something about the secret, namely that it cannot contain $r_3$ or $r_4$.

Notice that if the attacker had known this fact from outside sources, he would have set $\texttt{prob}_1$ to 0 to begin with and hence not learned anything new from the additional view extents. However, if the owner cannot predict the attacker's prior knowledge, he must follow the conservative approach that the views breach privacy if they can be used to revise *some* a priori belief of the attacker. □

The following example illustrates the point that privacy breaches depend on the proprietary database instance.

*Example 2.* In Example 1, the publishing of views $V_{RS}$ and $V_{SP}$ was breaching privacy on instance $I$. In contrast, consider an instance $I'$ obtained from $I$ by replacing all subcommittees with the same value $c_0$. Then publishing $V_{RS}$ and $V_{SP}$ does not change the probability distribution on possible secrets since all values in the extent of $V_R$ remain candidates for reviewers of paper $p_1$. In this case the privacy guarantee holds and the new views can be published. □

Finally, note that integrity constraints can significantly boost the attacker's chances of defeating the privacy guarantee and must therefore be taken into account by the owner. We have seen in Example 1 how integrity constraint $C_3$ could be used

[1] Note that in practice we would allow the owner to specify the secret as a parameterized query, i.e., have in the place of $p_1$ a parameter that stands for "any paper id". In the interest of simplifying the notation we assume that the secret involves a particular constant $p_1$. The generalization is straightforward.

by the attacker to revise his a priori probabilities for the secrets. Note that if the attacker did not know $C_1, C_2, C_3$ to hold, the set of possible secrets would not change after publishing the extra views. For instance, it does not matter what subcommittee a paper is assigned to if outside reviewers can also review it. Example 3 shows a scenario in which integrity constraints that specify cardinality constraints lead to much more dramatic privacy breaches, exposing the secret fully. Our results take into account such constraints.

*Example 3.* Let $I''$ be an instance that coincides with $I$ on the extents of $SP$ and $RS$ and in which $RP$ states that papers $p_1$ and $p_2$ are reviewed by both $r_1$ and $r_2$ and that papers $p_3, p_4$ are reviewed by both $r_3$ and $r_4$. Before seeing $V_{RS}, V_{SP}$ the attacker considers any subset of reviewers as plausible, leading to 15 possible secrets to pick from. If the attacker now sees the extents of $V_{RS}$ and $V_{SP}$ corresponding to $I''$, he must conclude that only subsets of $\{r_1, r_2\}$ are plausible secrets, leading to 3 possibilities: $\{r_1\}, \{r_2\}, \{r_1, r_2\}$. Now assume that the attacker has the additional knowledge that each paper has exactly two reviewers. We express this prior knowledge in the form of integrity constraints stating that each paper has at most two ($C_4$) and at least two reviewers ($C_5$).

$$C_4 : \ \forall p \forall r_1 \forall r_2 \forall r \ RP(r_1, p) \wedge RP(r_2, p) \wedge RP(r, p) \rightarrow r = r_1 \vee r = r_2$$
$$C_5 : \ \forall p \forall c \ SP(c, p) \rightarrow \exists r_1 \exists r_2 \ RP(r_1, p) \wedge RP(r_2, p) \wedge r_1 \neq r_2$$

$C_4$ and $C_5$ further prune the set of possible secrets to only $\{r_1, r_2\}$, the probability of which is necessarily 1. In other words the secret is fully exposed! □

**Contributions.** We formulate a novel privacy guarantee that ensures that, given existing views $\bar{V}$ and integrity constraints $\Delta$, a new view $N$ can be safely published. The guarantee does not assume any particular attack method; instead it checks that regardless of the attacker's a priori belief about the secret and computational resources, posting the extent of $N$ can not lead to a revision of the attacker's belief. The owner specifies the secret by a query $S$ over the proprietary database instance $I$. In that case we say that $N$ is *safe* for $S$ on $I$, denoted $\mathit{safe}_{\bar{V}}^{\Delta}(N, S, I)$. We formulate two versions of the safety guarantee. The first, $\mathit{Gsafe}_{\bar{V}}^{\Delta}(N, S, I)$, assumes that the attacker has domain knowledge about the possible worlds which witness (generate) the secret. Then we formulate a less strict guarantee, $\mathit{Esafe}_{\bar{V}}^{\Delta}(N, S, I)$, which applies when the attacker's domain knowledge pertains to the likelihood of secrets, and he has no opinion which distinguishes among the possible worlds witnessing the same secret.

We solve the problem of deciding both guarantees when $S, N$ and all views in $\bar{V}$ are defined by unions of conjunctive queries with non-equalities (UCQ$^{\neq}$) and the constraints in $\Delta$ are equivalent to containment statements between UCQ$^{\neq}$ queries. These constraints extend classical embedded dependencies [2] with disjunction and non-equality, and they can express the standard key and foreign key constraints, but also cardinality constraints and beyond. All constraints in our motivating examples belong to this class. We consider three levels of strengthening for each guarantee.

1. We show that $\mathit{Esafe}_{\bar{V}}^{\Delta}(N, S, I)$ is decidable in **PSPACE** in the size of the instance $I$ and that $\mathit{Gsafe}_{\bar{V}}^{\Delta}(N, S, I)$ is $\mathbf{\Pi_2^P}$-complete in the size of $I$.

2. We prove that for a fixed domain $\mathcal{D}$ we can check in **PSPACE** in the size of $\mathcal{D}$ that $\mathit{Esafe}^{\Delta}_{\bar{V}}(N, S, I)$ holds for all instances $I$ over $\mathcal{D}$. The analogous problem for *Gsafe* is $\mathbf{\Pi_2^P}$-complete in the size of $\mathcal{D}$.
3. For both kinds of safety, we show undecidability of checking safety on all instances $I$ (regardless of their domain).

Our techniques shed additional light on the relationship between privacy and information integration. In particular, in the process of establishing our undecidability results, we expose an interesting connection with a problem from information integration, namely lossless answering of queries using views [4].

## 2 Two Formal Privacy Guarantees

**Possible worlds and plausible secrets.** Let $I$ be a proprietary database instance satisfying $\Delta$. Denote with $E$ the corresponding $\mathcal{PU}$-instance, which associates to each table $V \in \bar{V}$ the extent $V(I)$ (in short $\bar{V}(I) = E$). Given $E$, there is a set of $\mathcal{PR}$-instances $w$ over an infinite domain, that satisfy the constraints $\Delta$ (denoted $w \models \Delta$) and on which the views yield $E$ ($\bar{V}(w) = E$). These instances are known as *possible worlds* in the literature (see [11] and references therein). Denote their set with

$$\mathit{Worlds}^{\Delta}_{\bar{V}}(E) = \{w \mid w \models \Delta \wedge \bar{V}(w) = E\}.$$

Clearly, $I \in \mathit{Worlds}^{\Delta}_{\bar{V}}(E)$. We call a secret $s$ *plausible* given $E$ if it occurs in a possible world, i.e., there exists $w \in \mathit{Worlds}^{\Delta}_{\bar{V}}(E)$ such that $S(w) = s$. Observe that $S(I)$ is trivially plausible.

**Attacker's knowledge of secret assuming zero views.** We model the attacker's general domain knowledge as a probability distribution $\mathbf{P} : \mathcal{S} \rightarrow [0, 1]$ defined over the set of outcomes $\mathcal{S}$ [17] that consists of all possible instances of the secret which are witnessed by some world that satisfies $\Delta$. As usual, given an event, i.e., a set of outcomes $S \subseteq \mathcal{S}$, we denote by $\mathbf{P}(S)$ the probability $\Sigma_{s \in S}\mathbf{P}(s)$ of the event [17].

Note that we make no assumptions on $\mathbf{P}$, thus allowing for distributions that correlate particular tuples. For example, the distribution may model the knowledge that "reviewers $r_1$ and $r_2$ have the same research background and are likely to review the same papers" or that "a paper is very likely to have exactly three reviews and it is impossible that it has less than two or more than four". This modeling is in contrast to the one used in [15], which assumes independent probability of individual tuples appearing in the secret.

**Induced probability distributions over private database.** The attacker's knowledge of the secret, i.e., the distribution $\mathbf{P}$, induces possible compatible probability distributions $\mathbf{P'} : \mathcal{W} \rightarrow [0, 1]$ over the set $\mathcal{W}$ of instances of the private database which satisfy $\Delta$. Clearly, $\mathit{Worlds}^{\Delta}_{\bar{V}}(E) \subseteq \mathcal{W}$. Note that the attacker is often unaware of the details of those distributions since they may also involve data that are tangential or irrelevant to the secret, i.e., data that the attacker is unaware of or is not interested in. For example, though the attacker of Example 1 only cares about paper $p_1$ and its potential reviewers, the induced probability distribution assigns probabilities to the full set of data pertaining to the conference. Our work considers two assumptions for deducing the compatible probability distributions over the private database instance and produces corresponding results:

1. **General:** The distribution $\mathbf{P}$ induces the set $\mathcal{P}^g$ that consists of all distributions $\mathbf{P}^g$ that are defined on $\mathcal{W}$ and have the property

$$\forall s \in \mathcal{S}: \ \Sigma_{w \in \mathcal{W}, S(w)=s} \mathbf{P}^g(w) = \mathbf{P}(s) \tag{1}$$

   We will see that according to the general assumption, maintaining privacy requires that no possible world $w$ that witnesses a secret instance $s$ (i.e., $S(w) = s$) can be eliminated by the extra view. A less strict requirement, which is compatible with the fact that the attacker may not have an opinion on the non-secret data, is provided by the next assumption.
2. **Equiprobable Witnesses:** The distribution $\mathbf{P}$ induces the unique distribution $\mathbf{P}^e$, called *equiprobable witness*, that is defined on $\mathcal{W}$ and has the property

$$\forall s \in \mathcal{S}, w \in \mathcal{W}: \ S(w) = s \Rightarrow \mathbf{P}^e(w) = \frac{\mathbf{P}(s)}{|\{w' \mid w' \in \mathcal{W}, S(w') = s\}|}$$

   i.e., all witnesses $w$ of a secret $s$ have equal probability. Obviously $\mathbf{P}^e \in \mathcal{P}^g$.

**Belief based on a-priori set of views.** With a slight abuse of notation, in the context of a distribution $\mathbf{P}^g : \mathcal{W} \to [0,1]$ a secret instance $s \in \mathcal{S}$ will also stand for the event $\{w \mid w \in \mathcal{W}, S(w) = s\}$ and $E$ will also stand for the event $\mathit{Worlds}^{\Delta}_{V}(E)$. Then the conditional probability $\mathbf{P}^g(s|E)$ denotes the probability of $s$ being the secret once the view extents $E$ have been observed, but before seeing the extent of the additional view $N$ that the owner considers whether to publish or not. We will call $\mathbf{P}^g(s|E)$ the attacker's *a priori* belief, and according to the conditional probability definition [17] we have

$$\mathbf{P}^g(s|E) = \frac{\sum_{w \in \mathit{Worlds}^{\Delta}_{V}(E), S(w)=s} \mathbf{P}^g(w)}{\sum_{w \in \mathit{Worlds}^{\Delta}_{V}(E)} \mathbf{P}^g(w)} \tag{2}$$

Since $\mathbf{P}^e \in \mathcal{P}^g$, Equation (2) holds also for $\mathbf{P}^e$. Notice that (2) associates probability 0 to implausible secrets. Also, the more possible worlds witness a certain secret candidate $s$, the higher its probability. In particular, if all possible worlds yield the same secret $s$ then $\mathbf{P}^g(s|E) = 1$.

**A-posteriori belief.** Now consider a new view $N$ and let $E'$ be the $\mathcal{PU} \cup \{N\}$-instance which extends $E$ by associating to $N$ the extent $N(I)$. $E'$ is what the attacker would observe after the additional publishing of view $N$. As above, we denote with $\mathit{Worlds}^{\Delta}_{V,N}(E')$ the set of possible worlds of $E'$ and the conditional probability $\mathbf{P}^g(s|E') = \mathbf{P}^g(s|\,\mathit{Worlds}^{\Delta}_{V,N}(E'))$ models the probability of each secret instance once the instance of $N$ is also observed.

**The privacy guarantees.** We propose two privacy guarantees that correspond to the general and the equiprobable witness assumptions. Both guarantees ensure that $N$ can be safely published by checking that, regardless of the attacker's domain knowledge, the a priori and a posteriori beliefs coincide.

**Definition 1 (Instance-dependent View Safety Under Equiprobable Witnesses).** We say that view $N$ is *safe* under equiprobable witnesses for the secret query $S$ on $\mathcal{PR}$-instance $I$ given views $\bar{V}$ and constraints $\Delta$ iff
for each probability distribution $\mathbf{P}$ on the candidate secrets and for each $s$ we have

$$\mathbf{P}^e(s|E) = \mathbf{P}^e(s|E')$$

where $E = \bar{V}(I)$ and $E' = (\bar{V}, N)(I)$. We denote this property as $\mathit{Esafe}^{\Delta}_{\bar{V}}(N, S, I)$.

**Definition 2 (Instance-dependent View Safety Under General Induced Probabilities).** We say that view $N$ is *safe* under general induced probabilities for the secret query $S$ on $\mathcal{PR}$-instance $I$ given views $\bar{V}$ and constraints $\Delta$ iff
for each probability distribution $\mathbf{P}$ on the candidate secrets, for each $s$, and for each $\mathbf{P}^g \in \mathcal{P}^g$ we have

$$\mathbf{P}^g(s|E) = \mathbf{P}^g(s|E')$$

where $E = \bar{V}(I)$ and $E' = (\bar{V}, N)(I)$. We denote this property as $\mathit{Gsafe}^{\Delta}_{\bar{V}}(N, S, I)$.

**Safety over classes of instances.** As shown in Example 2, the satisfaction of the privacy guarantee depends on the proprietary database $I$. The owner is thus faced with the following dilemma. Checking the guarantee on a given instance $I$ avoids being overly conservative and rejecting the publishing of many extra views because they breach privacy on another instance $I'$. On the other hand, this means re-checking the privacy guarantee upon each update to $I$. Alternatively, we consider the following two levels of strengthening the safety guarantees from Definitions 1 and 2 to take into account classes of instances.

$$\mathit{Esafe}^{\Delta}_{\bar{V}}(N, S, \mathcal{D}) := \forall I \in \mathit{Inst}(\mathcal{D}) : \ \mathit{Esafe}^{\Delta}_{\bar{V}}(N, S, I) \tag{3}$$
$$\mathit{Esafe}^{\Delta}_{\bar{V}}(N, S) := \forall I : \ \mathit{Esafe}^{\Delta}_{\bar{V}}(N, S, I) \tag{4}$$
$$\mathit{Gsafe}^{\Delta}_{\bar{V}}(N, S, \mathcal{D}) := \forall I \in \mathit{Inst}(\mathcal{D}) : \ \mathit{Gsafe}^{\Delta}_{\bar{V}}(N, S, I) \tag{5}$$
$$\mathit{Gsafe}^{\Delta}_{\bar{V}}(N, S) := \forall I : \ \mathit{Gsafe}^{\Delta}_{\bar{V}}(N, S, I) \tag{6}$$

(3) and (5) extend safety to a (finite) set $\mathit{Inst}(\mathcal{D})$ of $\mathcal{PR}$-instances over some given, finite domain $\mathcal{D}$ (useful when modeling dictionary attacks), while (4) and (6) extend safety to all $\mathcal{PR}$-instances.

**Dictionary Attacks.** It is often appropriate to assume that the attacker already knows the domain of the secret and hence is able to launch *dictionary attacks*, i.e., attacks that consist of potentially large numbers of queries that involve constants that have not been retrieved from the database; instead the attacker already knows those constants from his "dictionary knowledge". A typical example is an insurance database, in which we may want to assume that the list of potential patients and the list of diseases are publicly known (from the employee lists of the participating companies and a medical encyclopedia) but the data owner wants to hide the association between patients and diseases. When dictionary attacks are of concern, we model the dictionary knowledge of the attacker by including among the published views *dictionary views* which publish projections of the secret on those attributes whose domain is considered to be known to the attacker. Notice that in our running example dictionary views arise naturally and need not be added: $V_R$ is already one.

## 3 Preliminaries: Queries and Constraints

**Queries.** A *term* is a variable or constant. By $\bar{x}$ we denote a finite sequence of terms $x_1, \ldots, x_k$. The language of conjunctive queries with non-equalities (CQ$^{\neq}$) consists of expressions of the form $Q(\bar{z}) \leftarrow \ell_1(\bar{x}_1), \ldots, \ell_n(\bar{x}_n)$ where each $\ell_i(\bar{x}_i)$ in the rule *body* is a *literal*, i.e., an atom $R(\bar{x})$, an equality $x_i = x_j$ or an inequality $x_i \neq x_j$. Given $Q \in \mathrm{CQ}^{\neq}$, we define head$(Q)$ and body$(Q)$ to give the parts to the left and to the right of the arrow, respectively. A union of conjunctive queries with non-equalities (UCQ$^{\neq}$) is an expression of the form $Q = \bigvee_{i=1}^{n} Q_i$ where $Q_i \in \mathrm{CQ}^{\neq}$ for each $1 \leq i \leq n$. We have $Q(\mathcal{D}) = \bigcup_i Q_i(\mathcal{D})$, where $Q(\mathcal{D})$ denotes the result of query $Q$ on database $\mathcal{D}$. All queries and views in the motivating examples belong to UCQ$^{\neq}$.

**Constraints.** For a given query language $\mathcal{L}$, we consider the corresponding constraint language

$$\mathrm{IC}(\mathcal{L}) := \{\forall \bar{x}(U \rightarrow V) : U, V \in \mathcal{L}\}$$

where $\bar{x}$ is the set of free variables in both $U$ and $V$. These kinds of constraints express the containment of the queries $U$ in $V$ and are known as *embedded dependencies* when $\mathcal{L} = \mathrm{CQ}$ (conjunctive queries). Given a set of constraints $\Sigma \subseteq \mathrm{IC(CQ)}$, there is a well known procedure for extending a query $Q \in \mathrm{CQ}$ to another query $Q'$ by an iterative procedure known as the *chase*. However, the constraints in Example 1 belong to the more expressive language IC(UCQ$^{\neq}$) (see also the cardinality constraints in Example 3). In [8, 5], we extended the chase to $Q \in \mathrm{UCQ}^{\neq}$ and $\Sigma \subseteq \mathrm{IC}(\mathrm{UCQ}^{\neq})$. The extension is repeated in the full version of this paper [6]. We only give an example here, which illustrates that the chase produces unions of conjunctive queries with non-equalities (or, equivalently, queries whose body is in disjunctive normal form).

*Example 4.* Consider the query body $T(x, y)$ and the constraint $\sigma := \forall x \forall y T(x, y) \rightarrow (\exists z\ R(x, z)) \vee (x \neq y)$. [2] A chase step of $T(x, y)$ with $\sigma$ yields the following query body in disjunctive normal form: $T(x, y) \wedge R(x, z) \vee T(x, y) \wedge x \neq y$. □

It is well-known that checking termination of the chase is undecidable even for the constraint language IC(CQ). In the full paper [6], we repeat a sufficient condition for termination introduced in [8], namely the property of a set of constraints having *stratified witnesses*. This condition is the most general termination condition we are aware of, and it is efficiently checkable (in PTIME in the size of the constraint set). Essentially, it ensures that only a finite number of new variables (such as $z$ in Example 4) can be introduced into the chase result, which therefore must be finite.

**Theorem 1 ([8]).** *If $\Delta \subseteq \mathrm{IC}(\mathrm{UCQ}^{\neq})$ has stratified witnesses, then the chase with $\Delta$ of any $Q \in \mathrm{UCQ}^{\neq}$ terminates. It yields a result $\bigvee_{i=1}^{n} Q_i$ where each $Q_i \in \mathrm{CQ}^{\neq}$ has size polynomial in the size of $Q$ and $n$ is exponential in the size of $Q$.*

**In this paper, we assume that all queries belong to UCQ$^{\neq}$ and that all constraints belong to IC(UCQ$^{\neq}$).**

[2] $\sigma$ belongs to IC(UCQ$^{\neq}$) as it can be restated as the containment of $Q_1(x, y) \leftarrow T(x, y)$ in $Q_2(x, y) \leftarrow T(x, y) \wedge R(x, z) \vee T(x, y) \wedge x \neq y$.

## 4 General Induced Probability

**Privacy on Given Instance or Domain.** The main difficulty we need to overcome when checking $\mathit{Gsafe}^{\Delta}_{\bar{V}}(N, S, I)$ is the fact that the guarantee is universally quantified over infinitely many probability distributions $\mathbf{P}$ on the secrets and over infinitely many induced probability distributions $\mathbf{P}^g$ on the possible worlds. The following result solves this problem partially, showing that we can ignore probability distributions altogether, reducing the problem to comparing possible worlds only. Recall that $E'$ is $E$ extended with the new materialized view $N$.

**Lemma 1.** *$\mathit{Gsafe}^{\Delta}_{\bar{V}}(N, S, I)$ holds if and only if $\mathit{Worlds}^{\Delta}_{\bar{V}}(E) = \mathit{Worlds}^{\Delta}_{\bar{V},N}(E')$.*

What is left to do is to compute the sets of possible worlds, $\mathit{Worlds}^{\Delta}_{\bar{V}}(E)$ and $\mathit{Worlds}^{\Delta}_{\bar{V},N}(E')$. The problem here is that these sets have potentially infinite cardinality. In the remainder of this section, we solve this problem as follows. First, we show that the infinite set of possible worlds is finitely representable by a set of *templates*, denoted $\mathit{TWorlds}^{\Delta}_{\bar{V}}(E)$. Then we show how do adapt Lemma 1 to compare only $\mathit{TWorlds}^{\Delta}_{\bar{V}}(E)$ and $\mathit{TWorlds}^{\Delta}_{\bar{V},N}(E')$ (Theorem 3 below). Finally, we show how to compute $\mathit{TWorlds}^{\Delta}_{\bar{V}}(E)$.

**Possible world templates.** It was shown in [11] that for conjunctive query views and in the absence of constraints, the infinite set of possible worlds is finitely representable by a set of *templates*. We extend this result to UCQ$^{\neq}$ views and in the presence of constraints. Let $\mathcal{D}$ be a set of constants and $\mathcal{V}$ a set of variables. A database over $\mathcal{D}$ associates to each relation in its schema a set of tuples of constants from $\mathcal{D}$. A database template over $\mathcal{D}$ and $\mathcal{V}$ associates to each relation a set of tuples of constants and variables from $\mathcal{D} \cup \mathcal{V}$ [11]. The notion of evaluating a UCQ$^{\neq}$ query over a database template extends in the obvious way. Given the views $\bar{V}$ of extent $E$, a *possible world template* is a database template $T$ such that $\bar{V}(T) = E$.

*Example 5.* Consider a proprietary database of schema $R(A, B, C)$ and domain $\mathcal{D}$. Also consider the view $V(A, C) \leftarrow R(A, B, C)$ of extent $E = \{(a_1, c_1), (a_2, c_2)\}$. Then $T_1 = \{R(a_1, x_1, c_1), R(a_2, x_2, c_2)\}$ and $T_2 = \{R(a_1, x_3, c_1), R(a_2, x_3, c_2)\}$ are possible world templates since $V(T_1) = V(T_2) = E$. $\mathit{Worlds}_{V}(E)$ is represented by $\{T_1, T_2\}$ in the following sense: for any possible world $W \in \mathit{Worlds}_{V}(E)$, there is an injective homomorphic embedding from $T$ into $W$. In particular, if we instantiate $x_1, x_2, x_3$ with constants from $\mathcal{D}$ in all possible ways (but never $x_1$ and $x_2$ with the same constant, as $T_2$ takes care of that case), we are sure to obtain only possible worlds (infinitely many if $\mathcal{D}$ is infinite). Notice that in general we need more than one template to represent the possible worlds. If in the above example we also exported the view $V'(B) \leftarrow R(A, B, C)$ of extent $\{b_1, b_2\}$ then the possible worlds would be given by the templates $T_1 = \{R(a_1, b_1, c_1), R(a_1, b_2, c_2)\}$, $T_2 = \{R(a_1, b_2, c_1), R(a_1, b_1, c_2)\}$, $T_3 = \{R(a_1, b_1, c_1), R(a_1, b_1, c_2)\}$, $T_4 = \{R(a_1, b_2, c_1), R(a_1, b_2, c_2)\}$, which happen to be full-fledged databases as they mention no variables.

**Definition 3 (Reduced Universal Set of Possible World Templates).** We say that a set $\mathcal{T}$ of possible world templates is *universal* for a view extent $E$ if for any possible world $W$ of $E$, there is an injective homomorphic embedding $h$ from some $T \in \mathcal{T}$ into $W$, i.e. the images under $h$ of distinct variables from $T$ are distinct. $\mathcal{T}$ is *reduced* if (i) for each $T_1, T_2 \in \mathcal{T}$ with $T_1 \neq T_2$ there is no injective homomorphic embedding from $T_1$ into $T_2$ and (ii) for each $T \in \mathcal{T}$ there is no injective homomorphism from $T$ into a proper subset of $T$'s tuples.

Given the integrity constraints $\Delta$ and the published views $\bar{V}$ of extent $E$, there may be several universal sets of possible world templates, but only a single reduced one:

**Theorem 2.** *The reduced universal set of possible world templates is unique up to isomorphism. We denote this set with* $\mathit{TWorlds}^{\Delta}_{\bar{V}}(E)$.

It turns out that instead of comparing sets of possible worlds, we can compare their reduced universal sets of templates:

**Theorem 3.** $\mathit{Gsafe}^{\Delta}_{\bar{V}}(N, S, I)$ *holds if and only if* $\mathit{TWorlds}^{\Delta}_{\bar{V}}(E) = \mathit{TWorlds}^{\Delta}_{\bar{V},N}(E')$.

We next provide an algorithm for finding $\mathit{TWorlds}^{\Delta}_{\bar{V}}(E)$. The algorithm is based on capturing the view definitions with a set of constraints $\Sigma_V$ and *chasing* the extent $E$ with $\Sigma_V$ as well as the integrity constraints in $\Delta$. All these constraints belong to IC(UCQ$^{\neq}$) and are described below.

Let $\bar{V} = V_1, \ldots, V_n$. We define $\Sigma_V$ as the following set of constraints:

$$\begin{aligned}\Sigma_V := &\{\forall \bar{x}_i \bar{y}_i (\mathrm{body}(V_i) \rightarrow \mathrm{head}(V_i)) \mid 1 \leq i \leq n\} \\ &\cup \{\forall \bar{x}_i (\mathrm{head}(V_i) \rightarrow \exists \bar{y}_i \mathrm{body}(V_i)) \mid 1 \leq i \leq n\}\end{aligned}$$

where $\bar{x}_i$ are the variables in head$(V_i)$, and where $\bar{y}_i$ are the variables in body$(V_i)$ which do not appear in head$(V_i)$.

For a given extent $E$ of the views, we introduce the following set of constraints $\Sigma_E$. Let $E$ associate to view $V_i$ the set of tuples $\{t_1, \ldots, t_{n_i}\}$. Then define

$$\Sigma_E := \{\forall t \; V_i(t) \rightarrow \bigvee_{j=1}^{n_i} t = t_j \mid 1 \leq i \leq n\}$$

which states that for each $i$, the only tuples in $V_i$ are the ones given by $E$.

Finally, define the following axiom about equality: $\sigma_{\neq} := \forall x \forall y \; \mathit{true} \rightarrow x = y \vee x \neq y$. Also, let the *canonical tableau* of $E$ be the conjunction of all facts in $E$:

$$\mathit{CanT}(E) := \bigwedge_{i=1}^{n} \bigwedge_{j=1}^{n_i} V_i(t_j).$$

Function PWT below returns the desired set of possible world templates.

```
function PWT($E; \bar{V}; \Delta$)
(1) Compute $\Sigma := \Delta \cup \Sigma_V \cup \Sigma_E \cup \{\sigma_{\neq}\}$.
(2) Let $\bigvee_{l=1}^{m} T_l$ be the result of chasing $CanT(E)$ with $\Sigma$.
(3) For each $l$, compute $T'_l := T_l|_{\mathcal{PR}}$ (that is, keep only the $\mathcal{PR}$ literals).
(4) Set $\mathcal{T}_1 := \{T'_l | 1 \leq l \leq m\}$.
(5) Let $\mathcal{T}_2$ be the reduced $\mathcal{T}_1$, obtained by dropping each $T$ from $\mathcal{T}_1$
    for which there is another $T' \in \mathcal{T}_1$ and a homomorphic embeddingfrom $T'$ into $T$.
(6) Return $\mathcal{T}_2$.
```

Since function PWT is based on chasing, it is not a priori clear that it even terminates. Theorem 4 guarantees termination of PWT and implies the finiteness and computability of $\mathit{TWorlds}^{\Delta}_{\bar{V}}(E)$.

**Theorem 4.** *If $\Delta$ has stratified witnesses then:*

1. *Function* PWT *is guaranteed to terminate for any $\bar{V}$ and $E$.*
2. *The result of* PWT *is a template set of cardinality at most exponential in the size of $E$. Each template has size polynomial in the size of $E$.*
3. $\mathrm{PWT}(E; \bar{V}; \Delta) = TWorlds^{\Delta}_{\bar{V}}(E)$.

Theorems 3 and 4 immediately suggest a decision procedure for $Gsafe^{\Delta}_{\bar{V}}(N, S, I)$:

**Corollary 1.** *If $\Delta$ has stratified witnesses, then $Gsafe^{\Delta}_{\bar{V}}(N, S, I)$ holds if and only if* $\mathrm{PWT}(\bar{V}; \Delta; E) = \mathrm{PWT}(\bar{V}, N; \Delta; E')$.

Notice that, by Theorem 4 (2), the naive algorithm which eagerly computes the results of PWT requires exponential space in the size of $I$. However, checking that $\mathrm{PWT}(\bar{V}; \Delta; E) \neq \mathrm{PWT}(\bar{V}, N; \Delta; E')$ is clearly in $\Sigma^p_2$: guess a template $T \in \mathrm{PWT}(\bar{V}; \Delta; E)$ and then ask an NP oracle whether $T \in \mathrm{PWT}(\bar{V}; \Delta; E')$. Hence $Gsafe^{\Delta}_{\bar{V}}(N, S, I)$ is in $\mathbf{\Pi^P_2}$, which turns out to be asymptotically optimal:

**Theorem 5.** *If $\Delta$ has stratified witnesses then*

1. $Gsafe^{\Delta}_{\bar{V}}(N, S, I)$ *is $\mathbf{\Pi^P_2}$-complete in the size of $I$.*
2. $Gsafe^{\Delta}_{\bar{V}}(N, S, \mathcal{D})$ *is $\mathbf{\Pi^P_2}$-complete in the size of $\mathcal{D}$.*

**Unrestricted Privacy.** We next show that the strongest level of *Gsafe*, namely $Gsafe^{\Delta}_{\bar{V}}(N, S) := \forall I \;\; Gsafe^{\Delta}_{\bar{V}}(N, S, I)$ is undecidable. Towards achieving this result, we expose an interesting connection with a problem that has recently received considerable attention in the area of information integration, namely lossless query answering using views.

**Lossless Query Answering.** Given a set of views $\bar{V}$ and a query $Q$ (both formulated against the same schema) in data integration we are interested in answering $Q$ using only the extents $E$ of the views. Typical algorithms proposed in the literature (e.g. [9]) find the *certain* answers to $Q$, defined as $cert_Q(E) := \bigcap_{w \in Worlds_{\bar{V}}(E)} Q(w)$. Notice that regardless of which possible world $I \in Worlds_{\bar{V}}(E)$ actually generated the view extents $E$, we have $cert_Q(E) \subseteq Q(I)$. [4] asks whether for each $I$ and corresponding $E$, we can retrieve the exact answer to $Q(I)$ from $E$, i.e. $\forall I \;\; E = \bar{V}(I) \rightarrow cert_Q(E) = Q(I)$. If so we say that the views $\bar{V}$ can be used to losslessly answer $Q$, denoted $\bar{V} \models Q$. [4] identifies the decidable cases for regular path queries and views over semistructured data. In contrast, in the relational model [7] shows that even in the absence of constraints, if $Q$ and $\bar{V}$ belong to UCQ, the problem is undecidable.

It turns out that the problem $\bar{V} \models Q$ reduces to $Gsafe^{\emptyset}_{\bar{V}}(Q, \mathtt{id})$ where `id` is the identity secret query which returns the entire database. This implies:

**Theorem 6.** *$Gsafe^{\Delta}_{\bar{V}}(N, S)$ is undecidable, even under no constraints ($\Delta = \emptyset$).*

In some scenarios the $Gsafe^{\Delta}_{\bar{V}}(N, S, I)$ guarantee may turn out to be too strong. By Lemma 1, it requires the set of possible worlds not to change, which in turn means that $N(I)$ can be obtained solely from $\bar{V}(I)$. Depending on $I$, only few and non-interesting $N$'s could pass this test. In the next section we relax this guarantee assuming that attackers treat witnesses for a secret as equiprobable.

# 5 Equiprobable Witnesses

**Privacy on Given Instance or Domain.** As was the case for the *Gsafe* guarantee, the main difficulty to overcome when checking $\mathit{Esafe}^{\Delta}_{\bar{V}}(N, S, I)$ is the universal quantification over infinitely many probability distributions $\mathbf{P}$ on the candidates for secrets. Again we solve this problem by showing that we can ignore probability distributions entirely. This time however we reduce the problem to *counting* possible worlds and *plausible* secrets. Denote the multiplicity of secret $s$ when $E$ is published as the number of possible worlds on which the secret query evaluates to $s$: $\mathit{mult}_E(s) = |\{w \mid w \in \mathit{Worlds}^{\Delta}_{\bar{V}}(E), S(w) = s\}|$ and $\mathit{mult}_{E'}(s) = |\{w' \mid w' \in \mathit{Worlds}^{\Delta}_{\bar{V},N}(E'), S(w') = s\}|$. Notice that $s$ is plausible for $E$ if and only if $\mathit{mult}_E(s) > 0$.

**Lemma 2.** *$\mathit{Esafe}^{\Delta}_{\bar{V}}(N, S, I)$ holds if and only if*

1. *each plausible secret for $E$ stays plausible for $E'$, and*
2. *all pairs $s_1, s_2$ of secrets that are plausible for $E$ satisfy $\frac{\mathit{mult}_E(s_1)}{\mathit{mult}_E(s_2)} = \frac{\mathit{mult}_{E'}(s_1)}{\mathit{mult}_{E'}(s_2)}$.*

What is left to do is to compute the multiplicities of secrets, which requires computing the sets of possible worlds, $\mathit{Worlds}^{\Delta}_{\bar{V}}(E)$ and $\mathit{Worlds}^{\Delta}_{\bar{V},N}(E')$. We again use the finite representations of these sets $\mathit{TWorlds}^{\Delta}_{\bar{V}}(E)$, respectively $\mathit{TWorlds}^{\Delta}_{\bar{V},N}(E')$ and we show next (Theorem 7) that the privacy guarantee reduces to running the test of Lemma 2 on these template sets. We first introduce a notation for the multiplicity of templates witnessing $s$: $\mathit{Tmult}_E(s) = |\{t \in \mathit{TWorlds}^{\Delta}_{\bar{V}}(E) \mid S(t) = s\}|$ and $\mathit{Tmult}_{E'}(s) = |\{t' \in \mathit{TWorlds}^{\Delta}_{\bar{V},N}(E') \mid S(t') = s\}|$.

**Theorem 7.** *1. Assume that the set of views $\bar{V}$ contains dictionary views for each projection of the secret query $S$. Then every candidate secret $s$ is plausible for $E$ if and only if there exists $T \in \mathit{TWorlds}^{\Delta}_{\bar{V}}(E)$ with $S(T) = s$.*

*2. $\mathit{Esafe}^{\Delta}_{\bar{V}}(N, S, I)$ holds if and only if for every pair of plausible secrets $s_1, s_2$ we have $\frac{\mathit{Tmult}_E(s_1)}{\mathit{Tmult}_E(s_2)} = \frac{\mathit{Tmult}_{E'}(s_1)}{\mathit{Tmult}_{E'}(s_2)}$ .*

Putting together Theorem 7 and Theorem 4, we obtain that algorithm ESAFE below is a decision procedure for $\mathit{Esafe}^{\Delta}_{\bar{V}}(N, S, I)$.

```
algorithm ESAFE ($\bar{V}, \Delta, N, S, I$)
(1) Compute $E := \bar{V}(I)$ and $E' := (\bar{V}, N)(I)$.
(2) Compute $\mathit{TWorlds}^{\Delta}_{\bar{V}}(E) := \mathrm{PWT}(\bar{V}; \Delta; E)$, $\mathit{TWorlds}^{\Delta}_{\bar{V},N}(E') := \mathrm{PWT}(\bar{V}, N; \Delta; E')$.
(3) Compute $\mathit{Secrets}^{\Delta}_{\bar{V}}(E) := \{S(w) \mid w \in \mathit{TWorlds}^{\Delta}_{\bar{V}}(E)\}$.
(4) For each $s_1, s_2 \in \mathit{Secrets}^{\Delta}_{\bar{V}}(E)$ do
        if $\frac{\mathit{Tmult}_E(s_1)}{\mathit{Tmult}_E(s_2)} \neq \frac{\mathit{Tmult}_{E'}(s_1)}{\mathit{Tmult}_{E'}(s_2)}$ then return false.
(5) Return true.
```

Notice that, as presented, algorithm ESAFE needs exponential space in the size of $I$. Indeed, the two calls of function PWT yield results of size exponential in the size of $E$ and $E'$ (therefore exponential in the size of $I$). This presentation was chosen for the sake of simplicity. It turns out that we can do better.

**Theorem 8.** *If $\Delta$ has stratified witnesses then*

1. *$Esafe^{\Delta}_{\bar{V}}(N, S, I)$ is decidable in* **PSPACE** *in the size of $I$.*
2. *$Esafe^{\Delta}_{\bar{V}}(N, S, \mathcal{D})$ is decidable in* **PSPACE** *in the size of $\mathcal{D}$.*

The proof is based on the key idea that we do not need to first list the entire result of PWT, instead enumerating the possible world templates on demand. The technique extends straightforwardly to deciding $Esafe^{\Delta}_{\bar{V}}(N, S, \mathcal{D})$: enumerate in **PSPACE** in the size of $\mathcal{D}$ all instances $I \in Inst(\mathcal{D})$ and check $Esafe^{\Delta}_{\bar{V}}(N, S, I)$ using algorithm ESAFE.

We do not have a matching lower bound for these results. Indeed, we conjecture that the exact complexity lies in the counting complexity class $\mathbf{C_{=}P}$ [19] which is included in **PSPACE**.

**Unrestricted Privacy.** Using a reduction from the problem of lossless query answering, we show that $Esafe^{\Delta}_{\bar{V}}(N, S) := \forall I \; Esafe^{\Delta}_{\bar{V}}(N, S, I)$ is undecidable:

**Theorem 9.** *$Esafe^{\Delta}_{\bar{V}}(N, S)$ is undecidable even under no constraints ($\Delta = \emptyset$).*

## 6 Discussion

The key insight on which our framework for privacy diagnostics is based is the fact that the modeling of the attacker's knowledge should start from possible worlds or at least plausible secrets. The individual tuples in the secret are correlated by appearing together in possible worlds.

For a comparison of the two proposed flavors of privacy guarantees, assume that $E$ has 200 possible worlds, on which the secret query evaluates to $s_1$ for 100 worlds and to $s_2$ for the remaining worlds. If after publishing $E'$, only 100 worlds remain, of which none witnesses $s_1$, both guarantees will fail. The same happens if 101 world remain, of which 1 witnesses $s_1$ and the rest $s_2$. However, if a posteriori we are left with 100 secrets of which half witness $s_1$ and half witness $s_2$, *Gsafe* fails while *Esafe* holds. We leave it to the data owner to decide which guarantee is more appropriate for a specific application.

Notice that our framework can easily model and defend against collusion by multiple attackers. Suppose that access control mechanisms allow attacker $a_1$ to see a set of views $\bar{V}_1$ and attacker $a_2$ to access $\bar{V}_2$. Then defending against their collusion requires checking $safe^{\Delta}_{\bar{V}_1, \bar{V}_2}(E)$.

Also observe that since integrity constraints have the same effect as additional views, namely of ruling out possible worlds, the publishing of integrity constraints can also lead to privacy breaches. The publisher can employ the same framework to decide whether the publication of a constraint is safe.

In light of the high complexity bounds we obtained in terms of data complexity, our future work will focus on finding special cases for the view and secret definitions which yield tractability. We are also looking into further relaxations of the privacy guarantees.

## 7 Related Work

Prior work on privacy in databases has focused on implementing access control, i.e. allowing clients to see only those published views which they are authorized to. The techniques are based on cryptographically encoding the data (see [13, 14] and references within). Other techniques involve the authentication of users via credentials, as in the TrustBuilder project (see [20] for a comprehensive list of publications). Our work is orthogonal to work on access control, as it helps data owners design the views such that attackers cannot breach privacy using only *authorized* accesses.

[1] introduces $c$-tables, a compact formalism for finitely representing large (and potentially infinite) sets of possible worlds, and shows $\Pi_2^p$-complete data complexity for checking that the sets of possible worlds represented by two $c$-tables are the same. $c$-tables are not sufficiently expressive to model the set of possible worlds given by a view instance. [11] introduces *database templates* to this end and shows how to compute them using the chase, but does not address the comparison of the sets of possible worlds. Our approach for finding possible world templates coincides with the one in [11] when there are no constraints on the private database and the views are conjunctive queries.

[10] solves the problem of limiting privacy breaches in a scenario in which the aggregation of a set of private client data items is computed at the server. A privacy breach is essentially defined as a significant difference between the a posteriori and the a priori probability distributions. [10] provides not only a diagnostic tool, it also scrambles the data to improve privacy. The model assumes independence among the private values at the clients. Thus, the techniques do not apply directly to our scenario, where the secret tuples are not independent of each other (indeed they are correlated via the possible worlds in which they appear). On the other hand, we do not handle aggregation, which is at the center of the model in [10]. [3] takes aggregation into account and shows that exposing the result of counting queries allows the retrieval of an isomorphic copy of the structure of the database.

[16] takes a dual approach to ours. While we use queries to specify what cannot be disclosed, [16] uses conjunctive query views to specify what may be seen by outsiders. In this setting, conjunctive client queries asked against the proprietary database are answered only if they have a rewriting using the allowable views.

[15] is the closest work in spirit to ours. It pioneers the idea of specifying the secret as a conjunctive query and checking that the new view does not leak information about the secret by modifying the a priori probabilities of possible secrets. The most significant difference stems from the fact that [15] assumes that the tuples in the secret answer are *independent events*. This fails to defend against attackers who take into account correlations between tuples. This restriction is used to derive decidability even for the unrestricted guarantee. [15] lists as open the problem of deciding the guarantee when the independence assumption on secret tuples is lifted. This is the problem we address in this work. Not surprisingly, this problem is harder: the unrestricted guarantee becomes undecidable. Furthermore, we needed to refine the privacy guarantee in order to model whether the attacker knows or does not know anything about the witnesses of the secrets. Other differences are the fact that the guarantee is checked in [15] only for restricted integrity constraints (functional dependencies) and a-priori views (only boolean views). Also, [15] does not address the case when the instance is given, focusing on given domain and unrestricted guarantee only. Extending the results to the instance-based guarantee when no finite domain is given would require generating the set of possible world templates.

# References

1. S. Abiteboul, P. Kanellakis, and G. Grahne. On the representation and querying of sets of possible worlds. *Theoretical Computer Science*, 78:159–187, 1991.
2. Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison-Wesley, 1995.
3. Michal Bielecki and Jan Van den Bussche. Database interrogation using conjunctive queries. In *ICDT*, pages 259–269, 2003.
4. Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Moshe Y. Vardi. Lossless regular views. In *Symposium on Principles of Database Systems (PODS 2002)*, pages 247–258, 2002.
5. Alin Deutsch and Val Tannen. XML Queries and Constraints, Containment and Reformulation. To appear in *Journal of Theoretical Computer Science (TCS)*, 2005.
6. Alin Deutsch and Yannis Papakonstantinou. Privacy in Database Publishing. Technical report, Department of Computer Science and Engineering, UCSD, 2004. Extended version of this paper, available from http://www.db.ucsd.edu.
7. A. Deutsch, L. Sui, and V. Vianu. Queryies determined by views. Manuscript available from http://www.db.ucsd.edu/people/alin/papers/QdV.ps, 2004.
8. Alin Deutsch and Val Tannen. Reformulation of xml queries and constraints. In *ICDT*, 2003.
9. Oliver M. Duschka, Michael R. Genesereth, and Alon Y. Levy. Recursive query plans for data integration. *Journal of Logic Programming*, 43(1):49–73, 2000.
10. A. Evfimievski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *PODS*, 2003.
11. Gösta Grahne and Alberto O. Mendelzon. Tableau techniques for querying information sources through global schemas. In *ICDT*, 1999.
12. Alon Halevy. Logic-based techniques in data integration. In *Logic Based Artificial Intelligence*, 2000.
13. G. Miklau and D. Suciu. Cryptographically enforced conditional access for xml. In *WebDB*, 2002.
14. Gerome Miklau and Dan Suciu. Controlling access to published data using cryptography. In *VLDB*, 2003.
15. Gerome Miklau and Dan Suciu. A formal analysis of information disclosure in data exchange. In *SIGMOD Conference*, 2004.
16. Shariq Rizvi, Alberto O. Mendelzon, S. Sudarshan, and Prasan Roy. Extending query rewriting techniques for fine-grained access control. In *SIGMOD Conference*, 2004.
17. Murray R Spiegel, John J. Schiller, and R. Alu Srinivasan. *Schaum's Outline of Probability and Statistics*. MCGraw-Hill, 2000.
18. Jeffrey D. Ullman. Information integration using logical views. In *Proceedings of the Sixth International Conference on Database Theory*, 1997.
19. K. Wagner. The complexity of combinatorial problems with succinct input representation. *Acta Informatica*, 23:325–356, 1986.
20. Winslett et. al. The TrustBuilder Project. Publications Available from http://drl.cs.uiuc.edu/security/pubs.html.