Privacy Protection in Database Linking — A Logical Viewpoint *

Tsan-sheng Hsu Churn-Jung Liau Da-Wei Wang
Institute of Information Science, Academia Sinica
Nankang 11529, Taipei, Taiwan
E-mail: {tshsu,liaucj,wdw}@iis.sinica.edu.tw

Abstract

In this paper, we present a logical model for the privacy protection problem in the database linking context. Assume there is a large amount of data records in the data center. Each record has some public attributes, the values of which are known to the public and some confidential attributes, the values of which are to be protected. Users may obtain data from the data center by submitting queries to form database linkage. When a data table is released, the data manager must make sure that the receiver will not know the confidential data of any particular individual by linking the released data and information he had before receiving the data.

To solve the problem, we propose a simple epistemic logic to model the user's knowledge. In the model, the concept of safety is rigorously defined and an effective approach is given to test the safety of the released data. It is shown that some generalization operations can be applied to the original data to make it less precise and the release of the generalized data may prevent the violation of privacy. Two kinds of generalization operations are considered. The level-based one is more restrictive. However, a bottom-up search method can be used to find the maximally informative data satisfying the safety requirement. The set-based one, on the other hand, is more flexible. However, this approach would require a search through the whole space and its computational complexity is much higher, though graph theory is used to help simplify the search procedure. As a result, heuristic methods may be needed to improve the efficiency.
Key words: Privacy, Data table, Epistemic logic.

1 Introduction

With the rapid development of computer and communication technology, it has become much easier to store massive amounts of data in a central location and to spread it to end users via the Internet. Some data may be valuable information sources for scientists, analysts, or policy and decision makers. However, there may be a great danger of privacy invasion if they are accessed without restriction. As in [3], "in the past, most individuals lacked the time and resources to conduct surveillance required to invade an individual's privacy, as well as the means to disseminate the information uncovered, so privacy violations were restricted to those who did, mainly the government and the press". However, the development of Internet technology has changed the situation radically. Nowadays, any individual Internet user can easily spread a piece of information worldwide within seconds. In such a situation, the revelation of private information to unauthorized users, even though not intentionally, may cause a serious invasion of human rights.

There are many technical problems to be addressed for privacy protection. Though the authorization and authentication mechanisms can prevent illegal access of databases to a large extent, it is also the responsibility of the data center to ensure that users cannot infer privacy information from legally received data. This is traditionally called the inference control problem in database security [5, 12, 32]. While logical tools have been applied to the inference control problem, we are interested in the logical model for a more specialized problem, called the database linkage problem. Roughly speaking, the problem is how to prevent users[1] to know the private information of an individual[2] by linking some public or easy-to-know database with the data they receive legally from the data center.

*The preliminary version of the paper has appeared as [22].
[1]In this paper, "user" or "users" refers to anyone receiving data and having the potential of breaching the privacy of individuals
[2]An individual refers to a person whose privacy is to be protected.

Though the protection of privacy is very important, the over-restriction of access to the database may render the data useless. Therefore, the main challenge is to achieve a balance between privacy protection and data availability. To achieve this purpose, the database manager must check all possible kinds of knowledge that can be derived from the to-be-disclosed data before they respond to users' queries. Under the restriction of no privacy violation, the more knowledge the data can derive, the more useful it is for the users. According to the checking results, the database managers may decide to refuse the user's query or respond to it with some modification of the data. Therefore, the modeling of the user's knowledge becomes of primary importance for privacy protection work.

In artificial intelligence and distributed computing, epistemic logic has played a major role in modeling knowledge [16]. Hence it is natural to adopt this framework for the current purpose. However, since the knowledge about knowledge (i.e., the so-called introspective knowledge) is not our concern, nested modalities are not needed in the current framework, so the syntax and semantics can be slightly simplified.

The epistemic logic model provides us with a rigorous definition of privacy violation. However, what should we do if it is found that the direct release of a data table may cause an invasion of privacy? One approach is to simply refuse the release of the data, which is definitely too conservative to be acceptable. Therefore, an alternative approach is to modify the data before it is released. A common modification method is to generalize the values of some data cells to a coarser level of precision. This approach, when used appropriately, will make it impossible to identify the private information of any individual but the user can still induce useful general knowledge from the data [39]. For the generalization of data values, we can partition the domain of values according to some levels of precision and try to generalize the data from the finest to the coarsest level until the privacy requirement is met. This kind of operation is called level-based generalization. On the other hand, we can try to merge some values in the data and replace the single value by the merged result. Since the generalized values may be any nonempty subset of the domains, this process is called set-based generalization. Set-based generalization has more flexibility in avoiding privacy violation while keeping valuable information.

In the database linking context, the issue of anonymity has been addressed in [34, 35, 39]. In those works, the main goal of privacy protection is to keep the anonymity of data records, i.e., to prevent the user from knowing which data record belongs to a specific individual. However, in some cases, the user can discover the individual's confidential information without exactly knowing which data record belongs to him. Therefore, we have to keep not only the anonymity but also the confidentiality of the data. Some general models have been proposed for the confidentiality problem[2, 8, 9, 10, 11, 18, 19, 30, 38, 41], however, these models may be too complicated to be applied to our specific database linking context, so our model will be specially tailored for the application context. Also, complementary to the logical approach proposed in this paper is the quantitative aspect of privacy protection. While this is not the main concern of this paper, we would like to point out that some probabilistic or decision-theoretic approaches to the problem have been proposed in [6, 23, 24, 25, 26, 40].

In the rest of this paper, we first formally state the privacy protection problem in the database linking context. In Section 2, a simplified epistemic logic is used for modeling the situation when only level-based generalization is allowed in the modification of the data. The simple framework is then further generalized in Section 3 to handle the case where set-based generalization operations are also allowed. The computational aspects of our models are explored in Section 4. The comparison with related works is given in Section 5. Finally, some further research directions is discussed in Section 6.

1.1 The Privacy Protection Problem

To state the privacy protection problem in a database linking context, we first define the data representation. The most popular data representation is by data table [33]. The data in many application domains, for example medical records, financial transaction records, employee data, and so on, can all be represented as data tables. A formal definition of data table is given in [33].

Definition 1 *A data table*[3] *is a pair* $T = (U, A)$ *such that*

- U *is a nonempty finite set, called the universe,*
- A *is a nonempty finite set of primitive attributes, and*
- *every primitive attribute* $a \in A$ *is a total function* $a : U \to V_a$*, where* V_a *is the set of values of* a*, called the domain of* a*.*

[3]Also called knowledge representation system, information system, or attribute-value system.

Note that a relational database consists of a number of data tables and a data table may contain more records than those for U. However, we are only concerned with the privacy protection problem in one interactive session between the user and the data center. In such interactive session, the user asks the data center to release the data about a given set of individuals U, so, without loss of generality, we can consider the sub-table consisting only of records of those individuals.

The attributes of a data table can be divided into three sets. The first one consists of the *key attributes*, which can be used to identify to whom a data record belongs. Therefore, they are always masked off in response to a query. Since the key attributes uniquely determine the individuals, we can assume that they are associated with elements in the universe U and omit them henceforth. Second, we have a set of *public attributes*, the values of which are known to the public. For example, in [39] it is pointed out that some attributes like birth-date, gender, ethnicity, etc. are included in some public databases such as census data or voter registration lists. These attributes, if not appropriately generalized, may be used to re-identify an individual's record in a medical data table, and this will cause a privacy violation. The last kind of attributes are the *confidential ones*, the values of which we have to protect. It is often the case that there is an asymmetry between the values of a confidential attribute. For example, if the attribute is an HIV test result, then the revelation of a '+' value may cause a serious invasion of privacy, whereas it does not matter to know that an individual has a '−' value.

Example 1 Let us present the data table in figure 1 as the running example of the paper.

ID	Name	Date of Birth	ZIP	Height	Income	Health Status
A112148172	`Alice`	24/09/56	24126	160	400K	1
B234132167	`Bob`	06/09/56	24129	160	300K	1
F332442132	`Carl`	23/03/56	10427	160	300K	0
D563754135	`Daniel`	18/03/56	10431	165	100K	2
U435100132	`Edward`	20/04/55	26015	170	400K	2
H283514632	`Franz`	18/04/55	26032	170	300K	1
F673812907	`Greg`	12/10/52	26617	175	100K	0
K739823410	`Henry`	25/10/52	26628	175	400K	0

Figure 1: A data table in a data center

The key attributes of the table are "ID" and "Name", so by our convention, the set U of individuals is defined as

$$U = \{(A112148172, \texttt{Alice}), (B234132167, \texttt{Bob}), \cdots\} = \{u_1, \cdots, u_8\}.$$

The public attributes are "Date of Birth", "ZIP", and "Height", whereas the confidential ones are "Income" and "Health Status". The values of "Health Status" denote "normal"(0), "a little ill"(1), and "seriously ill"(2) respectively.

■

To handle the above three kinds of attributes separately, a data table can be reorganized as a data matrix and a mapping from the universe to the rows of the matrix. A data matrix $\mathbf{T}$ is an $n \times m$ matrix $[t_{ij}]_{n\times m}$ such that for each $1 \le j \le m$, $t_{ij} \in V_j$, where V_j is the domain of attribute j. Sometimes, we also write $\mathbf{T}_{ij}$ for the element t_{ij}. The data matrix has n records and each record has m attributes. We denote the row vectors of $\mathbf{T}$ by $\mathbf{t}^1, \ldots, \mathbf{t}^n$. Let $m = m_1 + m_2$, where attributes $1, \ldots, m_1$ are the public ones and $m_1 + 1, \ldots, m$ the confidential ones, then $p(\mathbf{T})$ and $c(\mathbf{T})$ denotes the sub-matrix consisting respectively of the first m_1 columns and the last m_2 columns of $\mathbf{T}$. Analogously, each row vector $\mathbf{t}$ can also be cut into two parts, $p(\mathbf{t})$ and $c(\mathbf{t})$. For a row vector $\mathbf{t}$, $\mathbf{t}_j$ denotes its jth element, so $\mathbf{t}^i_j = \mathbf{T}_{ij} = t_{ij}$ is the (i, j) element of $\mathbf{T}$ and $p(\mathbf{t})_j$ is the jth element of $p(\mathbf{t})$, etc.

The information that a data center has is a triple $(U, \mathbf{T}, \iota)$, where U is a set of individuals (the universe), $\mathbf{T}$ is a data matrix as defined above, and $\iota : U \to \{\mathbf{t}^1, \ldots, \mathbf{t}^n\}$ is an identification mapping which assigns a data record (i.e., a row in the data matrix) to each individual. The information the user has is another triple $(U, p(\mathbf{T}), \iota_p)$, where U and $p(\mathbf{T})$ are defined as above and $\iota_p : U \to \{p(\mathbf{t}^1), \ldots, p(\mathbf{t}^n)\}$ is the identification mapping known by the user. Since $\iota(u)$ and $\iota_p(u)$ are row vectors, we can also write $\iota(u)_j$ and $\iota_p(u)_j$ for their jth elements.

We assume the user's information about the public attributes is correct, so $\iota_p(u) = p(\iota(u))$ for any $u \in U$. Henceforth, we will fix the context with $(U, \mathbf{T}, \iota)$ and $(U, p(\mathbf{T}), \iota_p)$. Thus the privacy protection problem is:

How can **T** *be modified and then sent to the user so that the user will not know any individual's confidential information while the modified matrix is kept as informative as possible?*

To solve the problem, we have to answer the following three questions precisely:

1. What kind of operations can be applied to modify the matrix?
2. What does it mean that the user knows an individual's confidential information?
3. What is the meaning of a matrix being more informative than others?

Example 2 To illustrate the notations, let us re-organize the data table in figure 1 into the data matrices in figure 2.

24/09/56	24126	160	400K	1
06/09/56	24129	160	300K	1
23/03/56	10427	160	300K	0
18/03/56	10431	165	100K	2
20/04/55	26015	170	400K	2
18/04/55	26032	170	300K	1
12/10/52	26617	175	100K	0
25/10/52	26628	175	400K	0

(a) The 8×5 data matrix **T**

24/09/56	24126	160
06/09/56	24129	160
23/03/56	10427	160
18/03/56	10431	165
20/04/55	26015	170
18/04/55	26032	170
12/10/52	26617	175
25/10/52	26628	175

(b) $p(\mathbf{T})$

400K	1
300K	1
300K	0
100K	2
400K	2
300K	1
100K	0
400K	0

(c) $c(\mathbf{T})$

Figure 2: Data matrices

According to the notations introduced above, the data center has the information $(U, \mathbf{T}, \iota)$, where

- $U = \{u_1, \cdots, u_8\}$ is given in example 1,
- **T** is the data matrix given in figure 2(a),
- ι is defined such that $\iota(u_i)$ is the ith row of **T**, for example, $\iota(u_1)$=(09/25/56, 24126, 160, 400K, 1), etc.

On the other hand, the user has the information $(U, p(\mathbf{T}), \iota_p)$, where

- $U = \{u_1, \cdots, u_8\}$ is as above,
- $p(\mathbf{T})$ is the public part of the data matrix given in figure 2(b),
- ι_p is defined such that $\iota_p(u_i)$ is the ith row of $p(\mathbf{T})$, for example, $\iota_p(u_1)$=(09/25/56, 24126, 160), etc.

Note that the data matrix **T** alone does not contain any identification information. However, if **T** is released to the user, then the user could reconstruct the identification mapping ι by linking his information $(U, p(\mathbf{T}), \iota_p)$ with **T**. ■

In [39, 34], the notion of *bin size* is proposed to resolve the database linkage problem. A *bin* is defined as an equivalence class according to the public attributes and bin size is its cardinality. It is required that a safe table must satisfy the condition that the size of any bin should be sufficiently large. However, a large bin size may not be sufficient for the protection of privacy if all individuals in a bin have the same confidential attribute value. Hence, from the epistemic perspective, protection by bin size is not adequate. Though, in general, the chance for the user to know the confidential information is smaller if the bin size is larger, it is well-known that controlling bin size alone is not sufficient to stop inference attacks [12]. To properly protect privacy, we must consider some alternative criterion complementary to bin size.

In the following two sections, we will try to address the problems in an epistemic logic framework. However, before proceeding to the formal definitions, let us explain the basic ideas by our running example.

Example 3 A technique of protecting privacy is to release the data matrix in coarser data granularity. This is called generalization. For example, the date of birth may be given only in year and month, only the first two digits of the ZIP code may be given, and the height and income can be a range instead of a precise value. A concrete generalization of data matrix $\mathbf{T}$ in figure 2(a) is given in figure 3.

09/56	24***	160	400K	1
09/56	24***	160	300K	1
03/56	10***	160	300K	0
03/56	10***	165	100K	2
04/55	26***	170	400K	2
04/55	26***	170	300K	1
10/52	26***	175	100K	0
10/52	26***	175	400K	0

Figure 3: A generalized data matrix

From the generalized data matrix, a bin containing u_1 and u_2 has size 2. However, since the health status attribute of both rows in this bin has the value 1, the user receiving the data matrix can infer that both u_1 and u_2 are a little ill though he does not know which of them has income 400K. ■

2 Logical Model for Level-based Generalization

2.1 The Generalization Operations

From Example 3, we know that the main operations for modifying the matrix is to replace the public attribute values by some less precise ones. This kind of generalization operations has been formulated in [34] for achieving the bin size criterion. We review the generalization approach in [34] and see how it can achieve our safety criterion based on epistemic logic. It is defined by partitioning the domain of values according to different granular scales. Let V be a domain of values for some attribute, then a *partition* π of V is a set $\{s_1, s_2, \ldots, s_k\}$ of mutually disjoint subsets of V such that $\cup_{i=1}^{k} s_i = V$. Each s_i is called an equivalence class of the partition. Let π_1 and π_2 be two partitions of V. Then π_1 is a *refinement* of π_2, written as $\pi_1 \preceq \pi_2$, if for $s \in \pi_1$ and $t \in \pi_2$, either $s \subseteq t$ or $s \cap t = \emptyset$. Also let $\pi_1 \prec \pi_2$ denote $\pi_1 \preceq \pi_2$ and $\pi_1 \neq \pi_2$. Given a number $L > 1$, an L-level domain for V is a set of partitions of V, $\Pi_L(V) = \{\pi_1, \ldots, \pi_L\}$, such that $\pi_1 \prec \cdots \prec \pi_L$ and $\pi_1 = V$. For each $1 \leq i \leq L$, the partition π_i is called the *i-th level partition* of $\Pi_L(V)$.

To do level-based generalization, we have to specify the level of generalization for each attribute. Suppose that for each attribute $1 \leq j \leq m_1$, there is an L_j such that the L_j-level domain $\Pi_{L_j}(V_j)$ for V_j is given in advance. Then a *level-based generalization* operation τ is specified by an m_1-tuple of natural numbers $(k_1, k_2, \ldots, k_{m_1})$ such that $1 \leq k_j \leq L_j$ for each j. The result of applying the operation τ to a data matrix $\mathbf{T}$, denoted by $\tau(\mathbf{T})$, is to replace each element t_{ij} by the equivalence class $\tau(t_{ij})$ in the k_j-th level partition of $\Pi_{L_j}(V_j)$ that contains t_{ij}. Analogously, we can also apply the operation τ to a row vector $\mathbf{t}$ and obtain the result $\tau(\mathbf{t})$. Let τ_1 and τ_2 be two level-based generalization operations. Then τ_1 is at least as specific as τ_2, denoted by $\tau_1 \succeq \tau_2$, if for all i, j, $\tau_1(t_{ij}) \subseteq \tau_2(t_{ij})$. Obviously, $\tau_1 \succeq \tau_2$ iff $\tau_1 \leq \tau_2$ pointwisely.

Note that for simplicity, we do not change the ordering of the data records appearing in the generalized data matrix. In other words, the record of individual u_1 appears as the first row of $\tau(\mathbf{T})$, that of u_2 as the second row, etc. However, in practice, we may reshuffle the rows of the generalized data matrix to prevent the user to locate the records of the corresponding individuals by the ordering. The reshuffling operation is part of the generalization operation implicitly.

In the level-based generalization case, all operations allowed to modify the data matrix are m_1-tuples of this kind. This answers the first question posed in the previous section. To answer the remaining two questions, we need a logical language to specify the user's knowledge.

Example 4 There are three public attributes in our running example, so let us denote their domains by V_1, V_2, V_3, i.e.,

$$V_1 = \{dd/mm/yy \mid dd \text{ is a date, } mm \text{ is a month, and } yy \text{ is a year}\}$$
$$V_2 = \{d_1d_2d_3d_4d_5 \mid 0 \le d_i \le 9 (1 \le i \le 5) \text{ is a digit}\}$$
$$V_3 = \{\cdots, 160, 161, \cdots, 174, 175, \cdots\}.$$

Let us set a 4-level domain for V_1. Thus

$$\Pi_4(V_1) = \{\mathsf{D}, \mathsf{M}, \mathsf{Y}, \{V_1\}\},$$

where $\mathsf{D} = \{\{dd/mm/yy\} \mid dd/mm/yy \in V_1\}$, $\mathsf{M} = \{[mm/yy] \mid dd/mm/yy \in V_1 \text{ for some date } dd\}$, and $\mathsf{Y} = \{[yy] \mid dd/mm/yy \in V_1 \text{ for some date and month } dd/mm\}$. The partition D is the finest on V_1. Each equivalence class of D contains exactly one date from V_1. The partitions M and Y are coarser, each equivalence class of which contains dates of one month and dates of one year respectively. Note that $[mm/yy]$ denotes the set of dates in the specific month mm of the year yy, whereas $[yy]$ denotes the set of dates in the specific year yy. The coarsest partition of V_1 is $\{V_1\}$ whose only equivalence class is the whole set V_1.

Analogously, we can set a 6-level domain for V_2 and a 5-level domain for V_3 as follows:

$$\Pi_6(V_2) = \{\mathsf{5}, \mathsf{4}, \mathsf{3}, \mathsf{2}, \mathsf{1}, \mathsf{0}\},$$

where each equivalence class of the partition $\mathsf{i} (0 \le \mathsf{i} \le 5)$ contains the ZIP codes which are the same in the first i digits. We will use the wild character "*" to denote the equivalence classes. For example, 1142* denotes an equivalence class of $\mathsf{4}$ and is exactly the subset of ZIP codes $\{11420, 11421, \cdots, 11429\}$. Note that $\mathsf{0}$ is the coarsest partition of V_2 whose only equivalence class is V_2, also denoted by ***** in our notation.

$$\Pi_5(V_3) = \{I_1, I_5, I_{10}, I_{20}, \{V_3\}\},$$

where

$$I_1 = \{\cdots, \{160\}, \{161\}, \cdots, \{174\}, \{175\}, \cdots\}$$
$$I_5 = \{\cdots, [160, 165), [165, 170), [170, 175), \cdots\}$$
$$I_{10} = \{\cdots, [160, 170), [170, 180), \cdots\}$$
$$I_{20} = \{\cdots, [160, 180), [180, 200), \cdots\}$$

are to partition the domain V_3 into consecutive intervals of length 1, 5, 10, and 20, respectively. The bottom-up levels of the three domains are depicted in figure 4.

Based on the pre-defined generalized domains for the public attributes, the possible level-based generalization operations are in the set $\{(k_1, k_2, k_3) \mid 1 \le k_1 \le 4, 1 \le k_2 \le 6, 1 \le k_3 \le 5\}$. Let us apply one such generalization $\tau = (2, 4, 1)$ to the data matrix $\mathbf{T}$ in figure 2(a), then the date of birth of each data record is replaced by the equivalence class in the second level partition of V_1, ZIP code by the forth level partition of V_2, etc., so we have the result $\tau(\mathbf{T})$ in figure 3. ■

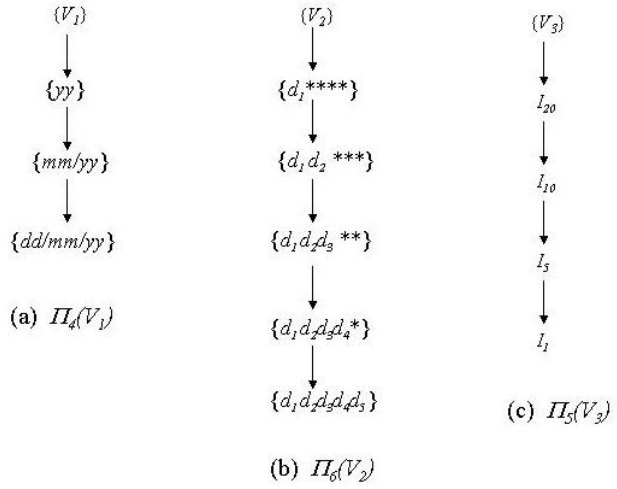

Figure 4: The generalized domains for the public attributes

2.2 The Logical Model

To formally analyze the notion of knowledge, the so-called epistemic logic is commonly used in philosophy and artificial intelligence [21, 16]. Here, we use the following simplified version of the epistemic logic for modeling the user's knowledge:

Definition 2 (Simple epistemic logic) *Let $\mathcal{P}$ be a set of atomic sentences. Then*

1. *the set of objective sentences $\mathcal{L}_0$ is the smallest set containing $\mathcal{P}$ and closed on the Boolean connectives $\neg, \wedge$ and $\vee$,*
2. *the set of epistemic sentences $\mathcal{L}_e = \{K\varphi \mid \varphi \in \mathcal{L}_0\}$, and*
3. *the set of well-formed formulas (wffs) of simple epistemic logic is $\mathcal{L} = \mathcal{L}_0 \cup \mathcal{L}_e$.*

The objective sentences will be used to describe the property of an individual according to his record in the data table, whereas an epistemic sentence $K\varphi$ means that the user knows an individual has property φ. Thus K is a modal operator denoting the knowledge of the user. Other boolean connectives, such as $\rightarrow$ and $\leftrightarrow$, are defined as abbreviations as usual.

The simple epistemic language is formally interpreted on Kripke models[16].

Definition 3

1. *A Kripke model (possible world model) for simple epistemic logic is a triplet $M = \langle W, \equiv, \nu \rangle$, where*
   - *W is a set of possible worlds,*
   - *$\equiv \subseteq W \times W$ is an equivalence relation on W, and*
   - *$\nu : W \times \mathcal{P} \rightarrow \{0, 1\}$ is a truth assignment.*
2. *The satisfaction relation between possible worlds and wffs, $\models_M$, is defined inductively by the following clauses:*
   (a) $w \models_M p$ iff $\nu(w, p) = 1$ if $p \in \mathcal{P}$,
   (b) $w \models_M \neg\varphi$ iff $w \not\models_M \varphi$,
   (c) $w \models_M \varphi \wedge \psi$ iff $w \models_M \varphi$ and $w \models_M \psi$,
   (d) $w \models_M \varphi \vee \psi$ iff $w \models_M \varphi$ or $w \models_M \psi$,

(e) $w \models_M K\varphi$ *iff for all* w' *such that* $w \equiv w'$, $w' \models_M \varphi$.

In epistemic logic, each possible world means a situation about which an agent has knowledge and two possible worlds have the equivalence relation $\equiv$ if the agent cannot distinguish them from each other by his knowledge. Let w, w' be two possible worlds and $w \equiv w'$, then w' is called an epistemic alternative of w. An agent in situation w will think that he may be in situation w' since he cannot distinguish these two situations. If a property φ holds in all epistemic alternatives of a situation, then he can be sure that φ holds, no matters what the real situation is, i.e., he knows φ in this situation.

To describe the information in the data table, we use a special instance of simple epistemic language, which is a variant of the epistemic decision logic(EDL) proposed in [17]. Here, we still call the language EDL. The set of atomic sentences for EDL is $\mathcal{P} = \{(j, \alpha) \mid 1 \leq j \leq m, \emptyset \neq \alpha \subseteq V_j\}$. By somewhat abusing the notation, we also write (j, v) for $(j, \{v\})$ if $v \in V_j$. In the following, the possible worlds for the Kripke models of EDL are just the possible individuals. Thus the intuitive meaning of the sentence (j, α) is that the value of an individual's attribute j is in α.

Let τ be a level-based generalization operation. Then from the released data matrix $\tau(\mathbf{T})$ and the public part of $\mathbf{T}$, the user can construct a Kripke model $M_\tau = (W, \equiv, \nu)$ for the EDL, where

- if U is the set of individuals and $I = \{1, 2 \cdots, n(= |U|)\}$, then

$$W = \{(u, i) \in U \times I \mid \iota_p(u)_j \in \tau(\mathbf{T})_{ij}, \forall 1 \leq j \leq m_1\},$$

  (recall that the notations $\tau(\mathbf{T})_{ij}$ and $\iota_p(u)_j$ denote respectively the (i, j) element of matrix $\tau(\mathbf{T})$ and the jth element of row vector $\iota_p(u)$)

- $(u_1, i_1) \equiv (u_2, i_2)$ iff $u_1 = u_2$,

- $\nu : W \times \mathcal{P} \to \{0, 1\}$ is defined by $\nu((u, i), (j, \alpha)) = 1$ iff $\iota(u)_j \in \alpha$ when j is a public attribute and $\nu((u, i), (j, \alpha)) = 1$ iff $\tau(\mathbf{T})_{ij} \in \alpha$ when j is a confidential attribute.

The model reflects the user's posterior knowledge after he receives the transformed data matrix $\tau(\mathbf{T})$. A possible world (u, i) of the model can be considered as a virtual individual which is formed by concatenating the public part of the real individual u and the confidential part of row i of the released data matrix. Thus the evaluation of an atomic sentence (j, v) in (u, i) accords to the data of u, i.e., $\iota(u)_j$, if j is a public attribute, whereas it is according to the ith record of $\tau(\mathbf{T})$, i.e., $\tau(\mathbf{T})_{ij}$, if j is a confidential attribute.

Because the user only knows the public attribute values of an individual u but not the confidential part, he will consider a generalized record i possible for u if the value of every public attribute of the record is the generalization of the individual's corresponding attribute value. This explains the construction of the set of virtual individuals W. For all $i \in I$ such that $(u, i) \in W$, the user considers (u, i) as a possible individual for u, therefore, the user's epistemic alternative relation $\equiv$ is defined according to the first component of the virtual individuals. Note that $\iota_p(u)$ is a row vector, so the virtual individuals for u and consequently the equivalence relation are determined by all public attributes instead of a single attribute of the individuals.

We are now ready for the main definitions of this section. In the following definition, we will write $u \models_{M_\tau} K\varphi$ instead of $(u, i) \models_{M_\tau} K\varphi$ for any objective sentence φ since $(u, i) \models_{M_\tau} K\varphi$ iff for any j such that $(u, j) \in W$, $(u, j) \models_{M_\tau} K\varphi$.

Definition 4

1. *The user's knowledge about the individual* u *after the level-based generalization* τ *is* $IK_\tau : U \to 2^{\mathcal{L}_0}$ *such that*

$$IK_\tau(u) = \{\varphi \mid u \models_{M_\tau} K\varphi\}.$$

2. *The user's general knowledge after the level-based generalization* τ *is the subset of* $\mathcal{L}_0$

$$GK_\tau = \bigcap_{u \in U} IK_\tau(u) = \{\varphi \mid \forall u \in U, u \models_{M_\tau} K\varphi\}.$$

3. *The confidential data is a function* $CON : U \to 2^{\mathcal{L}_0}$. *We further assume* $CON(u)$ *is finite for each* $u \in U$.

According to the semantics of EDL, $u \models_{M_\tau} K\varphi$ means that upon the receipt of the generalized data matrix $\tau(\mathbf{T})$, all individuals which are not distinguishable from u by the user has the property φ, so the user actually knows that φ is true of u even though he cannot re-identify which data record in $\tau(\mathbf{T})$ belongs to u. This explains why the set $IK_\tau(u)$ is the user's knowledge about u.

Among the elements of $IK_\tau(u)$, some may be about the general properties holding for all individuals. This is the so-called general knowledge. What makes the released data matrix useful to the user should be the general knowledge. For example, the general knowledge may describe some regularities in the whole population of individuals. The more general knowledge a data matrix can reveal, the more useful it is, so one goal of releasing data is to achieve maximal general knowledge. However, the goal may be in conflict with that of privacy protection. Everyone has something which he does not want others to know. The set $CON(u)$ stipulates such things for the individual u. Note that the requirement of privacy may not be uniform for different individuals. For u_1, φ may be sensitive, so $\varphi \in CON(u_1)$. However, it is completely possible that u_2 does not care if someone knows he has property φ at all, so $\varphi \notin CON(u_2)$. Also, the sentences in $CON(u)$ are in a very general form, so it may involve the combination of more than one confidential attribute.

Definition 5

1. *A level-based generalization τ is* safe *for u if $IK_\tau(u) \cap CON(u) = \emptyset$.*
2. *A level-based generalization τ is* safe *if it is safe for all $u \in U$.*
3. *Let τ_1 and τ_2 be two level-based generalization operations. Then τ_1 is at least as* informative *as τ_2, denoted by $\tau_1 \sqsupseteq \tau_2$, if $GK_{\tau_2} \subseteq GK_{\tau_1}$.*

In the light of the discussion regarding definition 4, a level-based generalization τ is *safe* for u if what the user may know about u does not contain any sensitive data of u after releasing the generalized data matrix $\tau(\mathbf{T})$ and the level-based generalization is safe if it is so for all individuals. With regard to the usefulness of the released data, τ_1 is more *informative* than τ_2 if τ_1 can induce more general knowledge than τ_2. Thus our goal is to find maximally informative generalizations respecting safety constraints.

Example 5 Continuing our running example, let $\tau = (2, 4, 1)$ as defined in example 4 be applied to obtain the generalized data matrix in figure 3. We first specify the EDL language for the example. The atomic sentences contain among others something like $(1, */*/57)$ which is true for some individual if his date of birth is in 1957, $(2, 114**)$ which is true if the first three digits of the individual's ZIP code are 114, $(3, (158, 167])$ which is true if the individual's height is between 158cm and 167cm, $(4, > 200K)$ which is true if the individual's income is more than 200K, and $(5, 0)$ which means that the individual has a normal health status.

Then we construct the model $M_\tau = (W, \equiv, \nu)$ for the generalization τ. Recall that the set of individuals U is $\{u_1, \cdots, u_8\}$ in the example. By inspecting the first row of the original data matrix, it can be seen that $\iota_p(u_1) = (24/09/56, 24126, 160)$, so $\iota_p(u_1)_j \in \tau(\mathbf{T})_{1j}$ and $\iota_p(u_1)_j \in \tau(\mathbf{T})_{2j}$ for $1 \leq j \leq 3$. Thus $(u_1, 1)$ and $(u_1, 2)$ are two virtual individuals in W. In the analogous way, we can find all virtual individuals and define

$$W = \{(u_i, i) \mid 1 \leq i \leq 8\} \cup \{(u_1, 2), (u_2, 1), (u_5, 6), (u_6, 5), (u_7, 8), (u_8, 7)\},$$

whereas $\equiv$ and ν are defined directly. For example, $\nu((u_5, 6), (1, */*/55)) = 1$ since attribute 1(i.e., the birth date) is public and according to the original data matrix $\iota(u_5) = (20/04/55, 26015, 170, 400K, 2)$ in which the birth date, 20/05/55 is indeed a date in the year 1955. On the other hand, $\nu((u_5, 6), (4, > 300K)) = 0$ since attribute 4 is confidential and $\tau(\mathbf{T})_{64} = 300K$ which is not more than 300K. The model can be illustrated by the table in figure 5. In the figure, the virtual individuals in different equivalence classes of $\equiv$ is separated into different sub-tables.

Now, let us consider the general knowledge which can be induced from the released generalized data matrix. One formula representing such general knowledge is

$$((1, */*/55) \vee (1, */*/56)) \wedge (2, 2****) \rightarrow (4, > 200K),$$

which means that any individuals born in 1956 or 1955 and living in the area with the first digit of the ZIP code being 2 have an income of more than 200K. This kind of knowledge may be useful for the marketing researchers of insurance companies. The sentence is true because the possible worlds satisfying $(1, */*/55) \vee (1, */*/56)) \wedge (2, 2*$

$(u_1, 1)$	24/09/56	24126	160	400K	1
$(u_1, 2)$	24/09/56	24126	160	300K	1
$(u_2, 1)$	06/09/56	24129	160	400K	1
$(u_2, 2)$	06/09/56	24129	160	300K	1
$(u_3, 3)$	23/03/56	10427	160	300K	0
$(u_4, 4)$	18/03/56	10431	165	100K	2
$(u_5, 5)$	20/04/55	26015	170	400K	2
$(u_5, 6)$	20/04/55	26015	170	300K	1
$(u_6, 5)$	18/04/55	26032	170	400K	2
$(u_6, 6)$	18/04/55	26032	170	300K	1
$(u_7, 7)$	12/10/52	26617	175	100K	0
$(u_7, 8)$	12/10/52	26617	175	400K	0
$(u_8, 7)$	25/10/52	26628	175	100K	0
$(u_8, 8)$	25/10/52	26628	175	400K	0

Figure 5: The possible worlds of model M_τ

$***$) are $(u_1, 1), (u_1, 2), (u_2, 2), (u_2, 1), (u_5, 5), (u_5, 6), (u_6, 5)$ and $(u_6, 6)$ whose second components are all records with income value more than 200K.

Next, let us consider the safety problem due to the release of the data. If $CON(u_1) = \{(4, > 300K) \wedge (5, \{1, 2\})\}$, that is, u_1 is sensitive to the fact that his income is more than 300K and that he is a little or seriously ill, while nothing else is sensitive to him, then the release of the data is safe for u_1 since in some epistemic alternative of $(u_1, 1)$, i.e., $(u_1, 2)$, $(4, > 300K) \wedge (5, \{1, 2\})$ is not satisfied. On the other hand, if serious illness alone is sensitive to all individuals, then the release of the data is unsafe for u_4 since the only epistemic alternative of $(u_4, 4)$ is itself and $(5, 2)$ is satisfied by it. ■

The advantages of the logical approach is made clear by the example. The main advantage is its flexibility in the representation of confidential information. Not only can we prevent the re-identification of the individuals, but we can also protect their confidential information in arbitrary logical forms. The logical formulas for representing the confidential information may contain a single atomic sentence or the combination of several compound sentences involving different attributes. Furthermore, different individuals can impose different safety constraints based on their personal preferences.

Proposition 1 *Let τ_1 and τ_2 be two level-based generalization operations. Then $\tau_1 \succeq \tau_2$ implies $\tau_1 \sqsupseteq \tau_2$.*

Proof: Let $M_1 = (W_1, \equiv_1, \nu_1)$ and $M_2 = (W_2, \equiv_2, \nu_2)$ be the models corresponding to operations τ_1 and τ_2 respectively. Recall that $\tau_1 \succeq \tau_2$ means that τ_2 generalizes the data matrix to a higher (coarser) level than τ_1, so $\iota_p(u)_j \in \tau_1(\mathbf{T})_{ij}$ implies $\iota_p(u)_j \in \tau_2(\mathbf{T})_{ij}$ for any individual $u \in U$, $1 \leq j \leq m_1$ and $1 \leq i \leq n$. This results in $W_1 \subseteq W_2$. Furthermore, by the construction, ν_1 and ν_2 agrees on the truth assignments to elements of W_1, so $w \models_{M_1} \varphi$ iff $w \models_{M_2} \varphi$ if φ is an objective sentence and $w \in W_1$. By the definition, it is clear that $GK_{\tau_i} = \{\varphi \in \mathcal{L}_0 \mid \forall w \in W_i, w \models_{M_i} \varphi\}$, so $GK_{\tau_2} \subseteq GK_{\tau_1}$ and the result follows immediately. ■

Since the privacy protection problem is to find a safe and maximally informative (i.e., the $\sqsupseteq$-maximal) level-based generalization operation, we can solve the problem by a bottom-up search among the finite set of all possible level-based generalizations. The search algorithm is discussed in more detail in Section 4.

3 Logical Model for Set-Based Generalization

3.1 The Generalization Operations

We have seen the definition of the privacy protection problem in a database linking context and our effective solution to it when level-based generalization operations are applied. However, the level-based approach may lack sufficient flexibility due to the following reasons:

First, in the level-based approach, a level is set up for one attribute of all individuals, so once the level is given, all records in the data table must be generalized to the same level. This is sometimes restrictive and may cause over-generalization. Since the safety requirements of different individuals may be different, we do not have to generalize their values to the same level.

Second, the generalized domain for the attribute values must be a partition of the basic domain in the level-based approach. It is sometimes not easy to achieve a natural partition of the domain. When a domain contains nominal instead of numerical values, it may not be clear which partition of the domain should be taken as its generalized domains.

Third, in the level-based approach, only generalization of public attributes is allowed. However, it may also be desirable to generalize the confidential attributes in some cases. An extreme example is when all individuals in the data table have the same sensitive values. In such a case, generalization of confidential attributes may be inevitable.

Finally, in [34], a complementary approach to generalization for achieving the bin size criterion, called suppression, has also been proposed. The main idea of suppression is to remove some outliers so that generalization to lower levels is sufficient for the purpose of safety. We would also like to propose an approach which can incorporate the suppression operations. These reasons motivate us to propose the set-based generalization approach.

Example 6 We have seen in example 5 that the generalization $\tau = (2, 4, 1)$ is not safe for u_4 if serious illness is sensitive information for him. To solve the problem, the data should be further generalized. Now, if we take an alternative generalization $\tau' = (2, 4, 3)$, then the generalized matrix $\tau'(\mathbf{T})$ is the one given in figure 6. It is clear

09/56	24***	[160,170)	400K	1
09/56	24***	[160,170)	300K	1
03/56	10***	[160,170)	300K	0
03/56	10***	[160,170)	100K	2
04/55	26***	[170,180)	400K	2
04/55	26***	[170,180)	300K	1
10/52	26***	[170,180)	100K	0
10/52	26***	[170,180)	400K	0

Figure 6: A generalized data matrix $\tau'(\mathbf{T})$

that the safety for u_4 is restored if his only concern is the serious illness. However, is the generalization of the height of other individuals, except u_3 and u_4, necessary? In fact, it can be seen that the generalization of other individuals does not cause any new elements to be added into W, whereas only that of u_3 and u_4 adds $(u_3, 4)$ and $(u_4, 3)$ to the set of possible worlds. Furthermore, since both u_7 and u_8 have a normal health status, the generalization of their birth dates and ZIP codes is also unnecessary. Thus the alternative generalized matrix in figure 7 should be sufficient to protect the information regarding serious illness of the individuals.

24/09/56	24126	160	400K	1
06/09/56	24129	160	300K	1
03/56	104**	[160,170)	300K	0
03/56	104**	[160,170)	100K	2
04/55	260**	170	400K	2
04/55	260**	170	300K	1
12/10/52	26617	175	100K	0
25/10/52	26628	175	400K	0

Figure 7: An alternative generalized data matrix

It is obvious that the alternative generalized data matrix cannot be obtained by level-based generalization. ■

Example 7 Let us consider an attribute "blood type" whose domain is $\{A, B, O, AB\}$, then $\{\{A, B\}, \{O, AB\}\}$ and $\{\{A, O\}, \{B, AB\}\}$ are two different ways to partition the domain. However, there is no obvious evidence, theoretical or experiential, to show that one partition is preferred to the other. In fact, it is intuitively reasonable that any subset of the domain can serve a generalized value of the blood type attribute, so its generalized domain should be the lattice shown in figure 8. A characteristic of such generalized domain is that partial overlap may

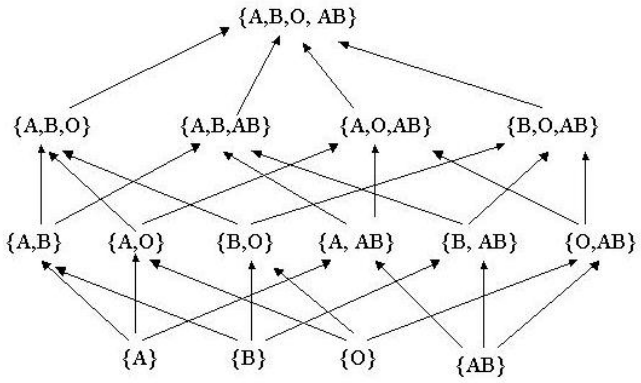

Figure 8: The generalized domains for blood type attribute

exist in different generalized values, so if the blood type of an individual is A, then it can be generalized to $\{A, O\}$, $\{A, B\}$, or $\{A, AB\}$ at the second level and $\{A, B, O\}$, $\{A, B, AB\}$, or $\{A, O, AB\}$ at the third level. ■

In the following definition, we assume a set of generalized values $\mathcal{Z}_j \subseteq 2^{V_j}$ is given for each $1 \leq j \leq m$.

Definition 6

1. *A primitive operation on a data matrix* $\mathbf{T}$ *is a triplet* (i, j, α)*, where* $1 \leq i \leq n, 1 \leq j \leq m$*, and* $t_{ij} \in \alpha \in \mathcal{Z}_j$*.*
2. *A set-based generalization operation is of the form* $\cap_{i \in I, j \in J}(i, j, \alpha_{ij})$ *for some* $I \subseteq \{1, \ldots, n\}$ *and* $J \subseteq \{1, \ldots, m\}$*.*
3. *A row deletion operation* $Rdel(i)$ *is an abbreviation of* $\bigcap_{1 \leq j \leq m}(i, j, V_j)$*.*
4. *A column deletion operation* $Cdel(j)$ *is an abbreviation of* $\bigcap_{1 \leq i \leq n}(i, j, V_j)$*.*

The application of a primitive operation (i, j, α) to $\mathbf{T}$ results in the replacement of t_{ij} by α and a *set-based generalization operation* is the simultaneous application of some primitive operations. The set-based generalization operations are more flexible than the level-based ones since, without predefined levels of generalization, all subsets of V_j (except the void one) can serve as the generalized values of attribute j. Unlike the level-based case, not all records are generalized at the same level in set-based generalization. Therefore, for outlying data, we may need a coarser level of generalization, whereas for centric data we can use a finer level of values. In the set-based case, both public and confidential attributes are allowed to be generalized. Furthermore, a row deletion operation is exactly the same as the suppression operation introduced in [34].

Example 8 The set-based generalization operation to generate the alternative data matrix in figure 7 is the conjunction of the following primitive operations:

$$(3, 1, 03/56)\ \ (3, 2, 104 * *)\ \ (3, 3, [160, 170))$$

$$(4, 1, 03/56)\ \ (4, 2, 104 * *)\ \ (4, 3, [160, 170))$$

$$(5, 1, 04/55)\ \ (5, 2, 260 * *)\ \ (6, 1, 04/55)\ \ (6, 2, 260 * *).$$

■

3.2 The Logical Model

In the logical model for level-based generalization, the truth assignment function ν is two-valued. The main reason is that the generalization of confidential attributes is not allowed. Since the truth assignment of an atomic sentence in a possible world (u, i) is determined by the public part of the individual u in the original matrix and the confidential part of the data record i in the generalized matrix which both have exact values, the truth value of a wff in a possible world is either true or false. However, once the generalization of confidential attributes is also allowed, it is possible that the confidential part of a data record i in the generalized matrix has an imprecise value. This will result in the failure of dichotomy on the evaluation of wffs according to the confidential part of the record.

There are at least two approaches to resolve the problem. The first one is to consider all possible values in a generalized value and construct a set of records with precise values for each record with imprecise generalized values. The set is called the realization of the generalized record. In this way, each virtual individual is formed by concatenating the public part of a real individual and the confidential part of an element in the realization of some generalized record. Since each element in the realization is an exact record, the truth assignment in such virtual individual is still two-valued. However, since for each generalized record there may be a large number of elements in its realization, this approach could result in the combinatorial explosion of the number of possible virtual individuals. Even worse, when a continuous domain is considered, the set of possible virtual individuals will be infinite. This means that we have to construct an infinite model for testing the safety of releasing the data if we insist on the two-valued semantics for the possible worlds.

The alternative method we adopt in this paper is to give up the two-valued semantics and allow a third truth value for the wffs which are possibly, but not necessarily, true according to the generalized values. In this way, each virtual individual is still formed by concatenating the public part of a real individual and the confidential part of a generalized record. The number of possible virtual individuals is not more than n^2 if n is the number of real individuals, though not all wffs can be evaluated as "true" or "false" in such a virtual individual due to the generalized values in its confidential part. Therefore, we have to include partial possible worlds with three-valued truth assignments in the semantics of simple epistemic logic. A partial possible world is one in which not all atomic sentences can be completely known to be true or false[27].

Definition 7

1. *A partial Kripke model (possible world model) for simple epistemic logic is a triplet $M = \langle W, \equiv, \nu \rangle$, where*
   - *W is a set of (partial) possible worlds,*
   - *$\equiv$ is an equivalence relation on W, and*
   - *$\nu : W \times \mathcal{P} \to \{0, 1, *\}$ is a three-valued truth assignment.*
2. *For each $w \in W$, let $\nu(w)$ be the restriction of ν to w. Then a completion of $\nu(w)$ is a classical logic interpretation $\mu : \mathcal{P} \to \{0, 1\}$ such that $\mu(p) = \nu(w, p)$ if $\nu(w, p) \neq *$ for any $p \in \mathcal{P}$. We write $\mu \sqsupseteq \nu(w)$ if μ is a completion of $\nu(w)$.*
3. *The satisfaction relation between classical interpretations and objective sentences is standard and denoted by $\mu \models \varphi$.*
4. *Let $\mathcal{C}$ be a class of classical logic interpretations, then the satisfaction relation between partial possible worlds and wffs with respect to $\mathcal{C}$ is defined by the following clauses for $\varphi \in \mathcal{L}_0$:*
   (a) $w \models^{\mathcal{C}}_M \varphi$ iff for all $\mu \sqsupseteq \nu(w)$ and $\mu \in \mathcal{C}$, $\mu \models \varphi$, and
   (b) $w \models^{\mathcal{C}}_M K\varphi$ iff for all w' such that $w \equiv w'$, $w' \models^{\mathcal{C}}_M \varphi$.

In a partial possible world, each atomic sentence is assigned one of the three truth values $1, 0$, or $*$ which respectively means that the sentence is true, false, or unknown in the world. Conceptually, a partial possible world can be seen as the abbreviation of a set of two-valued possible worlds, so we do not evaluate the objective sentences in a truth-functional way as in [27]. Instead, the evaluation is like a super-valuation introduced in [42]. For example, even though p is unknown in a world, it is still known that $p \vee \neg p$ is true in that world.

We also have to parameterize the satisfaction relation by a class of classical interpretations $\mathcal{C}$ because in the EDL for describing information in data tables, not all completions of a partial interpretation are reasonable. The interpretations in $\mathcal{C}$ are also called *reasonable* interpretations.

The same EDL language as in the level-based case is again used here. Class $\mathcal{C}$ of reasonable interpretations consists of those μ satisfying the following: For any $1 \leq j \leq m$, there exists exactly one $v \in V_j$ such that

1. $\mu((j, v)) = 1$, and
2. $\mu((j, \alpha)) = 1$ iff $v \in \alpha$.

Given a set-based generalization τ, the user can now construct a partial Kripke model $M_\tau = (W, \equiv, \nu)$ in the following way: First, to find the set of possible virtual individuals W, a binary relation $R_\tau \subseteq U \times I$ must be established:

$$R_\tau = \{(u, i) \in U \times I \mid \iota_p(u)_j \in \tau(\mathbf{T})_{ij}, \forall 1 \leq j \leq m_1\}$$

The relation R_τ is determined by the compatibility between the actual public attribute values of the individuals and their generalization appearing in the generalized matrix. However, since each individual must have some corresponding records in the data matrix and vice versa, some links between individuals and records are impossible due to other competitive links. To remove the impossible links, we can use an algorithm for testing the existence of perfect matching[1]. A *perfect matching* for $R \subseteq U \times I$ is a bijection (1-1 onto mapping) $f : U \to I$ such that for all $u \in U$ and $i \in I$, if $f(u) = i$, then $(u, i) \in R$. Let us define the reduced relation $R_\tau^\downarrow$ as the subset which keeps all $(u, i) \in R_\tau$ such that there is a perfect matching for the relation $R_\tau \cap ((U - \{u\}) \times (I - \{i\}))$. In other words, (u, i) is kept in the reduced relation iff there exists a perfect matching f such that $f(u) = i$.

The reduced relation $R_\tau^\downarrow$ can be computed from an $O(\sqrt{n}\,|R_\tau|^2)$-time algorithm because the best available time bound for the existence test of perfect matching is $O(\sqrt{n}\,|R_\tau|)$[14] and we have to do the test for each $(u, i) \in R_\tau$. This algorithm is more efficient than that proposed in [22][4].

The set of possible worlds W in M_τ is then defined as $R_\tau^\downarrow$ and the epistemic alternative relation $\equiv$ is defined as in the level-based case. As for the truth assignment function $\nu : W \times \mathcal{P} \to \{0, 1, *\}$, it is defined in the following two cases:

1. if j is a confidential attribute, then

$$\nu((u, i), (j, \alpha)) = \begin{cases} 1, & \text{if } \tau(t_{ij}) \subseteq \alpha; \\ 0, & \text{if } \tau(t_{ij}) \cap \alpha = \emptyset; \\ *, & \text{otherwise.} \end{cases}$$

2. if j is a public attribute, then

$$\nu((u, i), (j, \alpha)) = \begin{cases} 1, & \text{if } \iota(u)_j \in \alpha; \\ 0, & \text{if } \iota(u)_j \notin \alpha. \end{cases}$$

Let us now write $(u, i) \models^{\mathcal{C}}_{M_\tau} K\varphi$ as $u \models_{M_\tau} K\varphi$, then definitions 4 and 5 can be used without modification in the set-based case, and we can effectively test whether a set-based generalization operation is safe.

Example 9 Let us extend the set-based generalization operation in example 8 to the conjunction of the following primitive operations:

$$(3, 1, 03/56)\ \ (3, 2, 104 * *)\ \ (3, 3, [160, 170))$$
$$(4, 1, 03/56)\ \ (4, 2, 104 * *)\ \ (4, 3, [160, 170))$$
$$(5, 1, 04/55)\ \ (5, 2, 260 * *)\ \ (6, 1, 04/55)\ \ (6, 2, 260 * *)$$
$$(i, 4, (200K, 400K])\ \ i = 1, 2, 5, 6$$

and again denote the new set-based generalization operation by τ. The resultant data matrix by applying τ to our running example is presented in figure 9.

By the construction of M_τ, we first compute the binary relation

$$\{R_\tau = \{(u_i, i) \mid 1 \leq i \leq 8\} \cup \{(u_3, 4), (u_4, 3), (u_5, 6), (u_6, 5)\}.$$

It is easy to verify that each $(u, i) \in R_\tau$ can appear in a perfect matching of R_τ, so $R_\tau = R_\tau^\downarrow$ in this case. Thus $W = R_\tau$ is shown in figure 9.

[4]We would like to thank an anonymous referee for suggesting the more efficient algorithm.

24/09/56	24126	160	(200K,400K]	1
06/09/56	24129	160	(200K,400K]	1
03/56	104**	[160,170)	300K	0
03/56	104**	[160,170)	100K	2
04/55	260**	170	(200K,400K]	2
04/55	260**	170	(200K,400K]	1
12/10/52	26617	175	100K	0
25/10/52	26628	175	400K	0

Figure 9: A new set-based generalized data matrix $\tau(\mathbf{T})$

$(u_1, 1)$	24/09/56	24126	160	(200K,400K]	1
$(u_2, 2)$	06/09/56	24129	160	(200K,400K]	1
$(u_3, 3)$	23/03/56	10427	160	300K	0
$(u_3, 4)$	23/03/56	10427	160	100K	2
$(u_4, 3)$	18/03/56	10431	165	300K	0
$(u_4, 4)$	18/03/56	10431	165	100K	2
$(u_5, 5)$	20/04/55	26015	170	(200K,400K]	2
$(u_5, 6)$	20/04/55	26015	170	(200K,400K]	1
$(u_6, 5)$	18/04/55	26032	170	(200K,400K]	2
$(u_6, 6)$	18/04/55	26032	170	(200K,400K]	1
$(u_7, 7)$	12/10/52	26617	175	100K	0
$(u_8, 8)$	25/10/52	26628	175	400K	0

Figure 10: The set of possible worlds for M_τ

If all individuals consider only the information $(4, \geq 300K) \wedge (5, \{1, 2\})$ (high income and abnormal health status) or $(5, 2)$ (serious illness alone)as sensitive, i.e., $CON(u) = \{(4, \geq 300K) \wedge (5, \{1, 2\}), (5, 2)\}$, then it can be seen that the generalization is safe for all individuals since $u \not\models_{M_\tau} K\varphi$ for any u and $\varphi \in CON(u)$. Note that the privacy of u_1 and u_2 is protected by generalizing their confidential attribute "income" instead of the public attributes. ■

As in the case of level-based generalizations, we can compare the specificity of two set-based generalization operations. A set-based generalization operation τ_1 is said to be at least as specific as τ_2, denoted still by $\tau_1 \succeq \tau_2$, if for each primitive operation (i, j, α) in τ_1, there exists a primitive operation (i, j, β) in τ_2 such that $\alpha \subseteq \beta$. This means that τ_2 generalizes the entries of the data matrix to coarser values than τ_1 does. By an argument analogous to that for proposition 1, we can prove the following proposition:

Proposition 2 *Let τ_1 and τ_2 be two set-based generalization operations, then $\tau_1 \succeq \tau_2$ implies $\tau_1 \sqsupseteq \tau_2$, where $\tau_1 \sqsupseteq \tau_2$ means that $GK_{\tau_2} \subseteq GK_{\tau_1}$ as in the level-based case.*

4 Computational Aspects

While this paper is mainly concerned with the logical aspects of the privacy protection problem in the data linking context, we also briefly discuss its computational aspects in this section.

4.1 Computational problems for level-based generalization

In level-based generalization, the privacy protection problem is to find a safe and maximally informative generalization operation for a data table. However, for the sake of flexibility, we will design a search algorithm to find all safe

and maximally informative generalization operations for a given data table. This is a bottom-up search algorithm. We start from the most specific generalization operation $(1, 1, \cdots, 1)$ and test its safety according to definition 5. If this operation is safe, then we can stop without further generalization. Otherwise, we have to climb up the level tree according to the partial order $\succeq$ between generalization operations. Each new generalization operation must be tested for its safety. If it is safe, then all generalization operations above it can be pruned since we are to find maximally informative ones. This will substantially reduce the number of generalization operations which must be visited. The search algorithm is presented in figure 11. In the algorithm, the function GET-FROM-QUEUE takes a queue as its argument and returns its first element, whereas the procedure PUT-INTO-QUEUE takes a queue and a level-based generalization as its inputs and modifies the queue by adding the level-based generalization to its end. These operations are standard and can be found in textbooks for algorithms.

As for the function SAFETY, it takes a level-based generalization τ, a data matrix $\mathbf{T}$, the identification mapping of the data center, and the confidential data function CON as its arguments and returns an 1 if τ is safe with respect to the data matrix $\mathbf{T}$ according to the confidential requirement specified by CON, otherwise, it returns a 0. The function SAFETY is presented in figure 12. To explain the function, recall that $M_\tau = (W, \equiv, \nu)$ can be constructed from the input arguments of SAFETY. Let us define an equivalence relation $\approx$ on I such that $i_1 \approx i_2$ iff the i_1-th and i_2-th rows of $\tau(\mathbf{T})$ are identical in the values of public attributes. Then it can be seen that $i_1 \approx i_2$ iff there exists $u \in U$ such that $(u, i_1) \equiv (u, i_2)$. Note that $(u, i_1) \equiv (u, i_2)$ implies that both (u, i_1) and (u, i_2) are in W. Furthermore, $(u, i) \in W$ iff $i \in [i_1]_\approx$ where $\iota(u) = \mathbf{t}^{i_1}$ and $[i_1]_\approx$ denotes the $\approx$-equivalence class containing i_1. Therefore, by sorting the matrix $\tau(\mathbf{T})$ according to its public attributes, we can partition I into $\approx$-equivalence classes and assign to each individual u a corresponding equivalence class $G[u]$. Then we use a Boolean variable SF and two Boolean arrays US and KN respectively indexed by U and $U \times \mathcal{L}_0$ to compute the output. Here SF means the safety of the generalization τ and it is initialized to 1, whereas $US[u] = 1$ means that τ is unsafe for u, so the final safety is computed by the repeat conjunction of SF with $\neg US[u]$ for all $u \in U$. The array KN is to denote the individual knowledge, so $KN(u, \varphi) = 1$ means that $\varphi \in IK_\tau(u)$ and $KN(u, \varphi)$ is computed by repeat conjunction of its initial value 1 with $(u, i) \models_{M_\tau} \varphi$ for all $i \in G[u]$. Furthermore, τ is unsafe for u if for some $\varphi \in CON[u]$, $KN(u, \varphi) = 1$, so $US(u)$ is computed by repeat disjunction of its initial value 0 with $KN(u, \varphi)$ for all $\varphi \in CON[u]$.

The complexity of the algorithm SAFETY can be analyzed as follows: First, the sorting step 1 needs $O(n \log n)$ time by standard algorithms and the assignment step 2 can be done in $O(n)$ time. Let us assume that the evaluation $(u, i) \models_{M_\tau} \varphi$ can be done in constant-bounded time, then the total execution time of step 4 is

$$\sum_{u \in U} |CON[u]| \cdot |G[u]|.$$

Assume the size of each $CON[u]$ is bounded above by a constant C, then the total execution time of step 4 is at most

$$C \cdot \sum_{u \in U} |G[u]|,$$

which is in $O(n^2)$ time since $|G[u]| \leq n$ for all $u \in U$. The $O(n^2)$ bound is quite loose since $|G[u]|$ may be far less than n. Furthermore, in the special case where all individuals have the same set of confidential data or, at least, for all $u_1, u_2 \in U$, $G[u_1] = G[u_2]$ implies $CON[u_1] = CON[u_2]$, step 4(b) has to be executed only once for each individual corresponding to a different $\approx$-equivalence class, so the computation time of step 4 is reduced to $O(n)$. Therefore, the total time complexity of the safety test procedure is $O(n^2)$ in general and $O(n \log n)$ in the special case.

4.2 Computational problems for set-based generalization

The main computational problems for set-based generalization in privacy protection are still the search of maximally informative safe generalizations and the test of safety for a given generalization. The search problem in this case is far more complex than the one in the level-based case, so we defer its consideration to the next subsection and concentrate on the safety test procedure in this subsection.

The safety test procedure for set-based generalization is similar to the one presented in figure 12. However, there are some subtle differences. For a set-based generalization τ, the possible worlds (or possible individuals) are constructed from a binary relation R_τ introduced in section 3.2, so each individual u does not naturally correspond

Procedure SEARCH

Input:

1. a data matrix $\mathbf{T}_{n\times m}$ such that $m = m_1 + m_2$ and each $1 \leq i \leq m_1$ is a public attribute;
2. the identification mapping of data center, ι;
3. m_1 integers $L_1, \cdots, L_{m_1}$;
4. a confidential data array $CON[u \in U]$

Output: all safe and maximally informative level-based generalization operations.

Begin

1. $S \leftarrow \{(k_1, \cdots, k_{m_1}) \mid \forall 1 \leq i \leq m_1, 1 \leq k_i \leq L_i\}$;
2. Initialize a Boolean array $F[\tau] := 0$ for all $\tau \in S$;
3. Initialize a queue of level-based generalization operations $Q \leftarrow \{(1, 1, \cdots, 1)\}$;
4. While $Q \neq \emptyset$ do
   (a) $\tau \leftarrow$GET-FROM-QUEUE(Q);
   (b) $F[\tau] \leftarrow 1$;
   (c) If SAFETY($\tau, \mathbf{T}, \iota, CON$)
   then
   begin
   Output(τ);
   $F[\tau'] \leftarrow 1$ for all τ' such that $\tau \succeq \tau'$;
   end
   else for $i = 1$ to m_1 do
   begin
   $\tau' \leftarrow (k_1, \cdots, k_i + 1, \cdots, k_{m_1})$ if $\tau = (k_1, \cdots, k_i, \cdots, k_{m_1})$ and $k_i + 1 \leq L_i$;
   if $F[\tau'] = 0$, then **begin** PUT-INTO-QUEUE(Q, τ'); $F[\tau'] \leftarrow 1$ **end**
   end

End

Figure 11: The search algorithm for level-based generalization

Function SAFETY

Input Arguments:

1. a level based generalization τ;
2. a data matrix $\mathbf{T}_{n\times m}$ such that $m = m_1+m_2$ and each $1 \leq i \leq m_1$ is public attribute;
3. the identification mapping of data center, ι;
4. a confidential data array $CON[u \in U]$.

Output: 1 if τ is safe and 0 otherwise.

Begin

1. Sort the data matrix $\tau(\mathbf{T})$ according to values of all public attributes;
2. For each $u \in U$ do if $\iota(u) = \mathbf{t}^i$ then $G[u] \leftarrow [i]_{\approx}$;
3. Initialize Boolean $SF \leftarrow 1$;
4. For each $u \in U$ do
   (a) $US[u] \leftarrow 0$;
   (b) For each $\varphi \in CON[u]$ do
   $KN(u,\varphi) \leftarrow 1$;
   For each $i \in G[u]$ do $KN(u,\varphi) \leftarrow KN(u,\varphi) \wedge ((u,i) \models_{M_\tau} \varphi)$;
   $US(u) \leftarrow US(u) \vee KN(u,\varphi)$
   (c) $SF \leftarrow SF \wedge \neg US[u]$;

End

Figure 12: The safety test function for level-based generalization

to a $\approx$-equivalence class. This means that the sorting of $\tau(\mathbf{T})$ may help only in the initial construction of R_τ. It was shown in section 3.2 that there is an $O(\sqrt{n}\cdot|R_\tau|^2)$-time algorithm to compute the reduced relation $R_\tau^\downarrow$ from R_τ. By the construction of the reduced relation $R_\tau^\downarrow$, we can associate to each individual u a subset $G[u] = \{i \mid (u,i) \in R_\tau^\downarrow\}$. Therefore, steps 1 and 2 of the SAFETY algorithm must be modified to reflect the considerations. Steps 3 and 4 of the SAFETY algorithm for set-based generalization are then the same as those presented in figure 12 except that the evaluation $(u,i) \models_{M_\tau} \varphi$ in step 4(b) must be replaced by $(u,i) \models^{\mathcal{C}}_{M_\tau} \varphi$ in order to take the parameter $\mathcal{C}$ into account. If the size of each $CON[u]$ is still bounded above by a constant C, this kind of evaluations in step 4(b) will be executed at most $\sum_{u\in U} G[u]$ times. Since $\sum_{u\in U} G[u]$ is exactly the cardinality of W, the total execution time of the SAFETY procedure for the set-based case is $O(n^2 + \sqrt{n}\cdot|R_\tau|^2 + |W|)$ where n^2 time is needed for the construction of R_τ.

In the level-based case, the evaluation time of $(u,i) \models_{M_\tau} \varphi$ is assumed to be a constant since evaluation can be done via ordinary truth tables of Boolean functions. However, in the set-based case, the evaluation may be far more complex, so a preprocessing procedure is necessary. The preprocessing steps transform each wff $\varphi \in CON[u]$ into a normal form as follows:

1. Convert φ into clausal form, i.e., a conjunction of clauses, where each clause is a disjunction of literals and each literal is an atomic formula or its negation. This step is standard and can be found in logic textbooks.

2. Replace each literal $\neg(j,\alpha)$ occurring in the resultant clauses of step 1 by $(j, V_j - \alpha)$ so that all literals are positive.

3. For each resultant clause of step 2, if both (j,α_1) and (j,α_2) appear as its disjuncts, replace them by $(j,\alpha_1 \cup \alpha_2)$. Repeat this step until for every attribute j, an atomic formula of the form (j,α) occurs at most

once in each clause.

Assume φ is in normal form, then $(u,i) \models^{\mathcal{C}}_{M_\tau} \varphi$ iff for each clause of φ, there is an atomic disjunct (j,α) in the clause such that $\nu((u,i),(j,\alpha)) = 1$. Therefore, the evaluation time is linear to the size of the normal form of the formula.

4.3 On the search efficiency of set-based generalization

Though we have an effective approach to test the safety of a set-based generalization operation, the problem of how to find the maximally informative ones become more difficult in the set-based case. The degree of difficulty depends on the set of allowable generalized values $\mathcal{Z}_j$. So the choice of an appropriate $\mathcal{Z}_j$ for each attribute j is a crucial factor for the efficiency of the search problem. The most flexible way of choosing the $\mathcal{Z}_j$ is to let the data manager specify the sets since he knows the most about the data. Thus it is possible for the data manager to find the most appropriate sets of generalized values according to the nature of the data tables. However, it is often the case that this practice is considered too burdensome for the data manager. Therefore it may be desirable to have some default choices.

The most straightforward choice is $\mathcal{Z}_j = 2^{V_j} - \{\emptyset\}$. In this case, all subsets of V_j can serve the generalized values of the attribute j. So the data manager does not have to make any specification. However, one drawback of this choice is that it easily leads to combinatorial explosion of the search space. In particular, if for some attribute j, the domain of values V_j is huge or infinite, the search through all subsets of V_j will become impossible. Therefore, some heuristic methods may have to be employed to improve the search efficiency. For example, if the domain of values is endowed with a distance metric function $d : V_j \times V_j \to \mathcal{R}$ which satisfies some general criteria such as symmetry($d(x,y) = d(y,x)$), reflexivity($d(x,x) = 0$), and triangle inequality($d(x,y) + d(y,z) \geq d(x,z)$), then for any real number $r \in \mathcal{R}$ we can define the r-neighborhood of a value $v \in V_j$, as $N(v,r) = \{x \in V_j \mid d(v,x) \leq r\}$. The set can be seen as the r-radius sphere centering around v. Let $\mathcal{N}_r = \{N(v,r) \mid v \in V_j\}$ denote the sets of all r-radius spheres. The data manager can specify a number r_0 as the radius of searching. The search starts with the generalized values in $\bigcup_{S \in \mathcal{N}_{r_0}} 2^S$ and continues with increasing r until the appropriate generalization satisfying the safety criteria is found. Each generalized value in $\bigcup_{S \in \mathcal{N}_{r_0}} 2^S$ is a subset of an r_0-radius sphere centering around some $v \in V_j$, so the search will start from generalized values which can be totally included in an r_0-radius sphere of the metric space V_j and the radius of the search will increase gradually. Note that each element $S \in \mathcal{N}_{r_0}$ is a sphere of radius r_0, so it can be seen as a cluster of values. Consequently, each generalized value drawn from $\bigcup_{S \in \mathcal{N}_{r_0}} 2^S$ is a subset composed of the original values which are close enough according to the distance metric.

Though the search method described in the previous paragraph is more systematic than the exhaustive one, it does not prune the search space too much. In the worst case, we have to search through the whole space if we allow any subsets of V_j as the generalized values. However, not all subsets of V_j are natural generalized values according to the semantics of the data. For example, if the attribute is height, then it does not make much sense to consider the union of two separated intervals $[150, 160] \cup [170, 180]$ as a natural generalization of 175. Thus the notion of neighborhood can be used to provide a set of natural generalized values. In this case, we set $\mathcal{Z}_j = \bigcup_{r \geq 0} \mathcal{N}_r$.

In practice, there are two main types of attributes in data tables. The first is the nominal type and the second the numerical. The nominal attributes seldom have a natural metric in their domains. For example, the blood type of a patient has the domain $\{A, B, AB, O\}$ which does not possess a natural distance metric. Fortunately, they have a smaller domain in general, so we can take $\mathcal{Z}_j = 2^{V_j} - \{\emptyset\}$ if j is a nominal attribute. On the other hand, it is easy to define a metric in the domains of the numerical attributes, though in general the domains are large. Thus we can take $\mathcal{Z}_j = \bigcup_{r \geq 0} \mathcal{N}_r$ if j is a numerical attribute. Sometimes, even though a natural metric exists for the domain of an attribute, not all r-radius neighborhoods in $\mathcal{N}_r$ provide natural generalized values for the attribute. For example, if the birth date of a patient is "December 25, 2000", then it is natural to generalize the data to "December, 2000" or "2000". However, from the human aspect, it makes little sense to generalize the date to "the period from December 25, 2000 to January 15, 2001" even though it covers the same number of days as in "December, 2000". For this kind of attributes, we can just set $\mathcal{Z}_j$ as the L_j-level domain $\Pi_{L_j}(V_j)$ introduced in section 2.1.

Finally, if we relax the requirement of maximally informativeness, a kind of goal-directed search can be employed to improve the efficiency. Given an unsafe set-based generalization τ, the basic idea is to merge two or more individuals for whom τ is unsafe by replacing their attribute values with a common generalized value or to merge

an individual for whom τ is unsafe with some individuals for whom τ is safe. The main problem is then how to choose the individuals to be merged. This will be further considered in the future implementation of our approach.

4.4 Implementation

The logical approach taken in this paper has been implemented as part of the privacy protection system Cellsecu 2.0. As depicted in Figure 13 we envision that Cellsecu 2.0 can serve as a gate-keeper such that users can freely query the data center. All the answers approved by Cellsecu 2.0 preserve data confidentiality according to rigorously defined criteria. In addition to the logical criteria described in this paper, some quantitative criteria have also been implemented and reported in [6, 7].

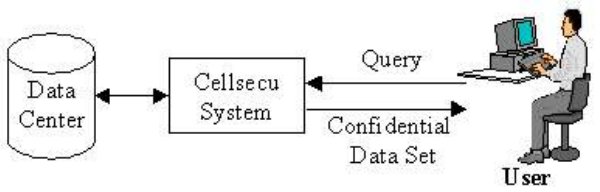

Figure 13: Cellsecu 2.0 system as a gate-keeper.

The overall architecture of system Cellsecu 2.0 is shown in Figure 14. When a query is submitted to the data center, the data center first produces the original query results by issuing queries to the corresponding databases. The filter process then removes all key attributes to form the filtered query-set. The confidentiality test module then tests the safety condition. If the filtered query-set does not meet the safety requirements, it will be processed by the "generalization" module to reduce the specificity of data to produce a confidential query-set. The audit center records the user identifier as well as the result of the safety test. The admin/configure tool allows the data manager to set the sensitivity of each attribute, to partition the attributes into three sets, to decide the safety conditions, and to set the generalization parameters. Some graphical user interfaces (GUI) of using the system is shown in the appendix.

The platform and environments for the system implementation are as follows:

1. Hardware: PIII-733 with 768MB SDRAM.
2. OS: Chinese Microsoft Windows 2000 SP2.
3. WWW Server: Apache 1.3.19 with Apache Jserv version 1.1.2.
4. Database Server: Microsoft SQL Server 2000.
5. Adm/Configure Tool: written in Java language and compiled in jdk1.3.0-02
6. Database Filter: a Java servlet program compiled in jdk1.3.0-02.
7. The Java Native Interface is adopted to link the Windows DLL files written in Microsoft C++ Version 6.0.

Some experiments have been carried out in the prototype system and the execution time for finding a maximally informative generalization meeting the level-based safety criteria (and some other quantitative criteria) is shown in table 1. The data for the experiments is mainly from the national health insurance research database provided by National Health Research Institute(NHRI) of Taiwan. Each data record contains five public attributes as follows:

1. APPL-DATE: reporting date of the medical record,
2. ID-BIRTHDAY: birth date of the patient,
3. T-AMT: the cost of the medical treatment,
4. PRSN-ID: identification number of the physician,

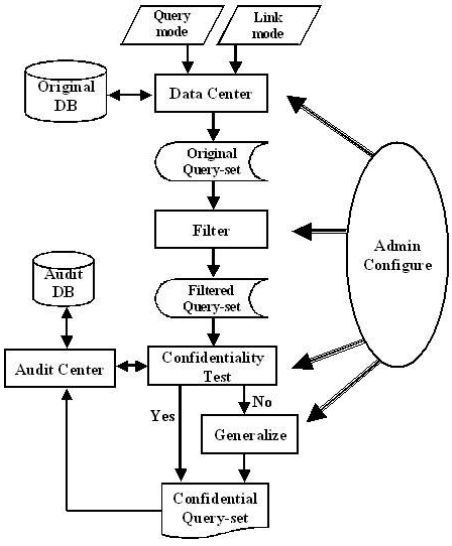

Figure 14: System architecture of Cellsecu 2.0

5. ID-SEX: gender of the patient.

Each confidential attribute is the diagnosed disease of the patient denoted by the 9th version of the international code of diseases(ACODE-ICD9).

The table shows the average execution time of running the program 5 times for different data sizes. The performance of the system is actually better than the theoretical worst-case analysis results because some fine-tuned techniques have been taken into account in the implementation of the system.

Table 1: The experimental results of Cellsecu 2.0

Data size (about)	Time (Sec)
200	1.15
2000	4.3
20000	47.022
200000	1101.94

5 Related Works

A model of information based on possible worlds has been proposed by Sutherland in [38]. In his model, an *information function* f is a mapping from the set of possible worlds to a domain of values. Let W be the set of possible worlds and f be an information function, then f can induce an equivalence relation $\equiv_f$ on W such that $w_1 \equiv_f w_2$ iff $f(w_1) = f(w_2)$. Let W/f, called the quotient set of f, denote the set of $\equiv_f$-equivalence classes. For any two information functions f_1 and f_2, it is said that information does not flow from f_1 to f_2 if for each equivalence class $E \in W/f_1$, $f_2(E) = f_2(W)$. In other words, information does not flow from f_1 to f_2 if observing the value of the f_1 attribute of a possible world does not exclude any possible values of its f_2 attribute. When f_1 denotes the set of public attributes and f_2 denotes the set of confidential attributes, the security condition of Sutherland's model is to ensure that information does not flow from f_1 to f_2.

In some sense, our framework can be roughly seen as an instance of Sutherland's model. However, there is also a subtle difference. To highlight the difference, let us instantiate Sutherland's model to our framework. In the instantiation, U, the set of individuals in our logical framework, corresponds to the set of possible worlds in

his model. For a level-based or set-based generalization operation τ, an information function f_τ assigns to each individual u the generalized values of his public attributes after the operation τ is applied and for a sensitive formula φ, another $\{0,1\}$-valued information function f_φ assigns to each individual u the value 1 iff φ is satisfied with respect to the generalized data record for u. By our definition, the generalization operation τ is safe if for each $E \in U/f_\tau$, $f_\varphi(E) \neq \{1\}$ for any sensitive formula φ, whereas by Sutherland's model, information does not flow from f_τ to f_φ if for each $E \in U/f_\tau$, $f_\varphi(E) = \{0,1\} = f_\varphi(U)$. Therefore, in our logical safety criterion, it is only required that some of the possible generalized records corresponding to the individual u do not satisfy φ if φ denotes sensitive information for u. However, in Sutherland's model, information does flow from f_τ to f_φ even if none of the possible generalized records corresponding to the individual u satisfies φ, i.e., in the case of $f_\varphi(E) = \{0\}$. Indeed, in such case, the observation of the public attributes may reveal some kind of information, i.e., the individual u does not satisfy φ. However, it does not matter since this is not the sensitive information we try to protect.

The instantiation above transforms our logical framework into a Sutherland's model. Conversely, given a Sutherland's model $S = (W, f_1, f_2)$, we can transform it into an epistemic logic framework. First, an epistemic logic language with atomic sentences of the form (f_2, α) is defined, where α is a nonempty subset of the domain of values for f_2, i.e., $\alpha \subseteq f_2(W)$. Second, a (two-valued) possible model $M_S = (W, \equiv_{f_1}, \nu)$ can be induced from Sutherland's model by letting $\nu(w, (f_2, \alpha)) = 1$ iff $f_2(w) \in \alpha$. It can then be shown that information flows from f_1 to f_2 iff there exists some *proper* subset $\alpha \subset f_2(W)$ such that the epistemic wff $K(f_2, \alpha)$ is satisfiable in M_S. Therefore, the security condition of Sutherland's model corresponds to our safety condition when the set of confidential data is the same for all possible worlds and is equal to

$$\{(f_2, \alpha) \mid \alpha \text{ is a maximal proper subset of } f_2(W)\}.$$

In this sense, our logical framework is more general than Sutherland's model since we allow objective wffs of arbitrary forms in the confidential data set and the confidential data for different possible worlds (or individuals) may be different.

An early attempt for the application of epistemic logic to the analysis of security has been made in [18]. The logical formalism is called security logic (SL) and includes branching time temporal operators, epistemic operators, and deontic operators. For the purpose of comparison, we will only present the epistemic and deontic fragment. Essentially, this is an extension of the multi-agent epistemic logic[16] with deontic operators. Let Sb be a set of subjects, then the wffs of SL is the smallest set containing the set of propositional symbols and being closed under the Boolean connectives $\neg$ and $\wedge$ and the following rule:

$$\text{if } \varphi \text{ is a wff, so are } K_i\varphi, P_i\varphi, \text{ and } O_i\varphi \text{ for each subject } i \in Sb.$$

The intuitive meaning of $K_i\varphi$ is "the subject i knows φ", so it corresponds with $K\varphi$ in our models, though in our logic the epistemic operator is not indexed by the subjects. On the other hand, $P_i\varphi$ and $O_i\varphi$ respectively means that subject i is permitted and obligated to know φ. Thus $P_i\varphi$ is true for the wff φ if it is not confidential in our model. As for $O_i\varphi$, it is useful for specifying the integrity policies, so no such components exist in our logic.

The intuition of these modal operators can be interpreted in a standard Kripke semantic framework. A Kripke model for SL is a 5-tuple $M = (W, (\mathcal{K}_i)_{i \in Sb}, \pi, \Phi_i, \Theta_i)$, where

- W is a set of possible worlds,
- $\mathcal{K}_i$ is an equivalence relation over W for each subject i,
- π is a truth valuation which assigns to each propositional symbol a subset of W, and
- Φ_i and Θ_i are subsets of wffs satisfying the following three constraints:
  - C1: $\Theta_i \subseteq \Phi_i$
  - C2: $\top \in \Phi_i$ and $\top \in \Theta_i$, where $\top$ denotes any propositional tautology
  - C3: Φ_i is closed under deduction and conjunction, i.e.,
    - if $\varphi \rightarrow \psi \in \Phi_i$ and $\varphi \in \Phi_i$, then $\psi \in \Phi_i$ and
    - if $\varphi \in \Phi_i$ and $\psi \in \Phi_i$, then $\varphi \wedge \psi \in \Phi_i$

The satisfaction of a wff with respect to a model M and a possible world w is defined inductively as follows:

1. $w \models_M p$ iff $w \in \pi(p)$ for any propositional symbol p,
2. $w \models_M \neg\varphi$ and $w \models_M \varphi \wedge \psi$ are defined classically,
3. for each subject i, $w \models_M K_i\varphi$ iff for every w' such that $(w, w') \in \mathcal{K}_i$, $w' \models_M \varphi$
4. for each subject i, $w \models_M P_i\varphi$ iff $\varphi \in \Phi_i$
5. for each subject i, $w \models_M O_i\varphi$ iff $w \models \varphi$ and $\varphi \in \Theta_i$

A security policy is then specified by the set of permitted knowledge Φ_i and it is said that the system satisfies the security requirement if $K_i\varphi \rightarrow P_i\varphi$ is true in the model of the system. For the security requirement to be satisfied, the constraints C2 and C3 are not only reasonable but also desirable due to the logical omniscience property of epistemic operators[16]. For example, according to the epistemic logic axioms, $K_i(\varphi \wedge \psi)$ is derivable from $K_i\varphi$ and $K_i\psi$, so if i is permitted to know φ and ψ but not $\varphi \wedge \psi$, then the security requirement will be violated when subject j knows φ and ψ separately. This seems counterintuitive since i knows (explicitly) only what he is permitted to know, i.e., φ and ψ.

However, imposing the constraint C3 on Φ_i has some drastic effects. While it is not unusual that some subject is permitted to know a certain φ and its negation $\neg\varphi$, it would be permitted to know everything if we put both φ and $\neg\varphi$ into Φ_i due to constraint C3. The SL proposed in [18] is articulated further in [19] and the problem is solved by weakening the constraint C3 to a set of closure properties on Φ_i and Θ_i. Among them is the following:

$$\text{if } \varphi \wedge \psi \text{ is in } \Phi_i, \text{ then so are } \varphi \text{ and } \psi \text{ separately,}$$

but not the converse. Thus it is now possible that Φ_i includes both φ and $\neg\varphi$ but not $\varphi \wedge \neg\varphi$. To conform with the logical omniscience property of epistemic operators, the semantics of $P_i\varphi$ has to be modified. In [19], $w \models_M P_i\varphi$ is defined by some clauses according to the form of φ and those relevant to our purpose are the following:

$w \models_M P_i\varphi$ iff $w \models_M \varphi$ and

$\varphi \in \Phi_i$, or
φ is $\psi \wedge \gamma$ and $w \models_M P_i\psi$ and $w \models_M P_i\gamma$,or
.....

One direct consequence of the definition is the validity of the schema $P_i\varphi \rightarrow \varphi$. This means that even though i is permitted to know any wff in Φ_i *potentially*, only those true in the current world can in fact be permitted.

The presentation above reveals some differences between SL and our logic. Roughly, the set Φ_i corresponds to the complement of our confidential information $\cup_{u \in U} CON(u)$ if we consider only one particular subject i. Thus we adopt a prohibition-based approach whereas SL employs a permission-based one. In our model, $\cup_{u \in U} CON(u)$ contains only wffs which are explicitly forbidden to be known. However, due to the logical omniscience property of the normal epistemic operators, we can guarantee that the implicitly prohibited information is also protected by our safety criteria. For example, if $\varphi \in CON(u)$, then, even though neither ψ nor $\psi \rightarrow \varphi$ is explicitly put in $CON(u)$ for some wff ψ, it is still forbidden for both ψ and $\psi \rightarrow \varphi$ to be known because, from $K\psi$ and $K(\psi \rightarrow \varphi)$, we can infer $K\varphi$ which means a violation of the security policy. Therefore we do not bother to put ψ or $\psi \rightarrow \varphi$ into the set $CON(u)$. Furthermore, since no deontic operators are included in our language, we can simplify the matter without regarding the semantic connection of the deontic formulas and the closure properties of the confidential information set. Therefore, we simply choose not to impose any restriction on the confidential function CON.

Another essential difference between SL and our logic is that SL is more general in its application scope, while our logic is specially tailored for the database linking problem. Since we allow the generalization of confidential attributes in the set-based case, three-valued truth assignments are also adopted in our semantics. On the other hand, we do not consider the knowledge about knowledge, so nested epistemic modalities are not allowed in our logic.

In [2, 9], a logic of security (LS) is proposed for the analysis of security of multilevel computer systems. The common methodology of SL, LS, and our models is the epistemic logic framework, though we consider a more specific application context. The systems to be analyzed by the LS are dynamic systems with multiple subjects

where each subject is permitted to know different levels of confidential information according to the roles played by him. Ignoring the details of the systems and the temporal aspect of the logic, LS can also be viewed as an extension of the multi-agent (or more precisely, multi-role) epistemic logic. Let Sb still be a set of subjects and Ro be a set of roles such that $Sb \subseteq Ro$, then the modalities of LS contain K_i for each $i \in Ro$ and R_j for each subject $j \in Sb$. The wffs of LS is the smallest set containing the set of propositional symbols and being closed under the Boolean connectives and the following rule:

$$\text{if } \varphi \text{ is a wff, so are } K_i\varphi \text{ and } R_j\varphi.$$

The intuitive meaning of $R_j\varphi$ is the same as that of $P_j\varphi$ in SL, though it is interpreted in a different way. The semantics of LS is based on the systems to be analyzed[9], however, we can reformulate it by the ordinary Kripke semantics. A Kripke model for LS is a quadruple $M = (W, (\mathcal{K}_i)_{i \in Ro}, \pi, \mathcal{R})$, where

- W and π are defined as in SL,
- $\mathcal{K}_i$ is an equivalence relation over W for each role i,
- $\mathcal{R} : Sb \times W \to 2^{Ro}$ is a right function for the subjects

The possible worlds are called traces in the systems of [9] and the equivalence relation $\mathcal{K}_i$ is based on the observation of a subject playing the role i. The set $\mathcal{R}(j, w)$ stipulates the roles that j is permitted to play in the world w.

The satisfaction of a wff of LS with respect to a model M and a possible world w, is defined inductively as follows:

1. the atoms and Boolean connectives are defined as in SL
2. for each role i, $w \models K_i\varphi$ iff for every w', $(w, w') \in \mathcal{K}_i$ implies $w' \models \varphi$
3. for each subject j, $w \models R_j\varphi$ iff for some $i \in \mathcal{R}(j, w)$, $w \models K_i\varphi$

Besides the differences between the application contexts, the main difference between LS and our models is the definition of the confidential information set. However, unlike the case of SL, there are no potential permission sets of knowledge Φ_i in the semantics of LS which can naturally correspond with the complement of the confidential information set in our logic.

Obviously, the role-based interpretation of $R_j\varphi$ somewhat alleviates the effect of C3 since it is now possible that both $R_j\varphi$ and $R_j\psi$ hold in a possible world without the derivation of $R_j(\varphi \wedge \psi)$. This only occurs when j can play two roles where one knows φ and the other knows ψ.

In [35, 34], another safety criterion for privacy protection in database linking has been proposed. The criterion is called k-anonymity, where k is a natural number. The k-anonymity criterion is a requirement that every combination of values of quasi-identifiers can be indistinctly matched to at least k individuals, where each quasi-identifier is roughly a set of public attributes in our models. In these works, two techniques are provided for enforcing the k-anonymity. One is called generalization and is essentially equivalent to our level-based generalization operation. The other is called suppression and can be achieved by our row deletion operation given in definition 6. A sufficient condition for achieving the k-anonymity is given in [35] which can be exactly formulated as a constraint on the size of equivalence classes for $\equiv$ in our (partial) Kripke models. More specifically, let $M_\tau = (W, \equiv, \nu)$ be a partial Kripke model constructed from a set-based generalization τ, then it achieves k-anonymity if $|[(u, i)]_\equiv| \geq k$ for any $u \in U$, where $[(u, i)]_\equiv$ is the equivalence class of the binary relation $\equiv$ containing (u, i). From the formulation, it can be seen that k-anonymity guarantees the non-uniqueness of the identification of each record in the released data table when $k \geq 2$. However, our definition also emphasizes the non-uniformity of the information in the records which are indistinguishable by their quasi-identifiers. Theoretically, it is easy to find examples where k-anonymity is satisfied but the confidential information of some individuals is revealed if non-uniformity is not satisfied. On the other hand, the larger the k, the less likely the same confidential attribute values appear in all the k records. From this viewpoint, k-anonymity in fact provides a semi-quantitative criterion for safety, whereas our definition is purely qualitative. Thus our definition should be complementary to the k-anonymity criterion for the privacy protection problem in database linking.

The properties of anonymity may also be required in other applications, such as web browsing, secure communication, electronic commerce, etc. Recently, Syverson and Stubblebine have provided an epistemic logic formalization

for various notions of anonymity in [41]. In their logic, the epistemic operators $\Box_P$ and $\Diamond_P$ are used for each principal P, where $\Box_P\varphi$ means that principal P knows φ and $\Diamond_P\varphi$ is defined as $\neg\Box_P\neg\varphi$ as usual. Furthermore, in their formalism, all condenda (i.e. things to be hidden) are of the form $\varphi(P)$. That is, they are formulas in which only a single principal name occurs freely. Thus the k-anonymity property corresponds to "$(\geq k)$-anonymous property" in [41] and is expressed as

$$\Diamond_I\varphi(P_0) \wedge \Diamond_I\varphi(P_1) \wedge \cdots \wedge \Diamond_I\varphi(P_{k-1})$$

where I denotes an intruder. On the other hand, our safety criterion conceptually corresponds to their "possible anonymity property" which is expressed as

$$\Diamond_I\varphi(P) \wedge \Diamond_I\neg\varphi(P).$$

Since it is assumed that $\Box_I(\varphi(P) \wedge \varphi(Q) \rightarrow P = Q)$ holds for all condenda $\varphi(P)$, the $(\geq k)$-anonymous property implies possible anonymity according to epistemic reasoning. However, the assumption does not hold in our application setting since a wff φ may be true of more than one individual in a data table.

6 Concluding Remarks

In this paper, we have presented a logical model for privacy protection in the database linking context. To avoid privacy violation, a requested data table generated by database linkage may have to be modified before it can be released to the user. A simple epistemic logic and one of its special instances, called epistemic decision logic, are employed to model the user's knowledge. According to the model, a safety criterion of the data is defined and can be tested effectively. Moreover, we also discuss the approach of finding the maximally informative generalizations satisfying the safety requirement.

The safety criterion defined in this paper can be seen as a generalization of bin size [39, 34]. According to the bin size criterion, the individuals having a particular public attribute value must be non-unique. Here we require not only non-uniqueness but also non-uniformity conditions. That is, the individuals having a common public attribute value must have different confidential attribute values. This criterion is purely qualitative. We can further consider some quantitative criteria of safety. This will depend on the distribution of the individuals on the different values of confidential attributes. In this regard, we may have to extend the epistemic logic with some probabilistic reasoning mechanism [15, 20]. To do this, our logical language must be extended with the probabilistic epistemic operators K^r for each $r \in (0, 1]$ and the semantics of the epistemic sentence $K^r\varphi$ is defined by

$$(u, i) \models_{M_\tau} K^r\varphi \text{ iff } \frac{|\{w \in [(u, i)]_\equiv : w \models_{M_\tau} \varphi\}|}{|[(u, i)]_\equiv|} \geq r.$$

The user's knowledge about individual u will be ranked according to the precision. For example,

$$IK_\tau(u, r) = \{\varphi \mid (u, i) \models_{M_\tau} K^r\varphi\}$$

is the user's r-precise knowledge about individual u after he receives the generalized data matrix $\tau(\mathbf{T})$. The general approach to achieve security of computer systems by taking both epistemic and probabilistic aspects into account has also been explored in previous literature[40, 25, 24, 26].

From the quantitative aspect, we must also consider the amount of information the user obtained after receiving the data table. After receiving the data table, it is possible that even though the user cannot know any individual's private information with certainty, he may learn the probability distribution of the confidential attribute values among a group of individuals and this may result in privacy invasion. To solve the problem, some information measures associated with data tables may aid the data manager in deciding how dangerous the release of the data table is to the privacy protection policy. These measures include Shannon's entropy [36], the Kolmogorov complexity [29], uncertainty-based information measures [28], the posterior probing cost of the user [4], etc. Some frameworks along this direction have been proposed recently[6, 23].

In this paper, we have measured the usefulness or quality of the released data by the amount of general knowledge which can be induced from it. Other quantitative measures based on statistics, such as mean and variance, are also possible[13]. In particular, the works on achieving both the goals of protecting personal privacy and of discovering useful patterns from data have received much attention recently[37]. This is also a worthwhile direction for further extension of our framework.

Another direction is to consider the protection of privacy for not only a single individual but also a group of individuals. In this paper, $u \models K\varphi$ means that the user knows u has the property φ. However, for a group of individuals G, the user may know there are some individuals in G with the property φ even though he cannot identify which one. In such a case, there may be some invasion of privacy for the group though the invasion is not towards any particular individual in this group.

The modification operations allowed in this paper are to replace precise values by imprecise ones, i.e., a subset of values. A further generalization is to allow the use of fuzzy values. In this regard, we can replace a precise value by a fuzzy set containing that value[43]. Then the set of possible worlds W and the truth assignment ν in the construction of models will be fuzzified and our logic will become a $[0, 1]$-valued logic.

The protection of privacy is not only a technological but also a legal problem. In the model above, the user's prior knowledge is assumed to be known to the data center. This is due to the query posed by the user. However, how can we be sure that users always tell the truth to the data center? How about the situation where users actually know more than what they tell the data center? Technically, we can keep a user's profiles to prevent a malicious user from obtaining confidential information through a series of queries. Nevertheless, some regulations or contracts may also be mandatory to enforce that the user should provide the necessary information in his query. This means that we should also model the deontic aspect of the problem as well as the epistemic one described in this paper [31]. Some works regarding the application of deontic logic to the specification of security policies have been done in [8, 10, 11], which may provide some insights to the extension of our work along the legal aspect.

References

[1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Inc., 1993.

[2] P. Bieber and F. Cuppens. "A definition of secure dependencies using the logic of security". In *Proc. of the 4th IEEE Computer Security Foundations Workshop*, pages 2–11, 1991.

[3] L.J. Camp. *Trust and Risk in Internet Commerce*. The MIT Press, 2000.

[4] Y.C. Chiang. Protecting privacy in public database (in Chinese). Master's thesis, Graduate Institute of Information Management, National Taiwan University, 2000.

[5] Y.C. Chiang, T.-s. Hsu, S. Kuo, and D.W. Wang. Preserving confidentially when sharing medical data. In *Proceedings of Asia Pacific Medical Informatics Conference*, 2000.

[6] Y.T. Chiang, Y.C. Chiang, T.-s. Hsu, C.J. Liau, and D.W. Wang. How much privacy? - a system to safe guard personal privacy while releasing database. In *Proceedings of the 3rd International Conference on Rough Sets and Current Trends in Computing*, LNCS 2475, pages 226–233. Springer-Verlag, 2002.

[7] Y.T. Chiang, Y.C. Chiang, T. s. Hsu, C.J. Liau, and D.W. Wang. The implementation of the software system cellsecu for privacy protection(in Chinese). In *Proceedings of the 2nd Conference on Information Technology and Applications in Outlying Inslands*, pages 341–353, 2002.

[8] L. Cholvy and F. Cuppens. "Analysing consistency of security policies". In *Proc. of the IEEE Symposium on Security and Privacy*, pages 103–112, 1997.

[9] F. Cuppens. "A logical formalization of secrecy". In *Proc. of the 6th IEEE Computer Security Foundations Workshop*, pages 53–62, 1993.

[10] F. Cuppens and R. Demolombe. "A deontic logic for reasoning about confidentiality". In M.A. Brown and J. Carmo, editors, *Deontic logic, agency, and normative systems: ΔEON'96, Third International Workshop on Deontic Logic in Computer Science*, pages 66–79, 1996.

[11] F. Cuppens and R. Demolombe. "A modal logical framework for security policies". In Z. Ras and A. Skowron, editors, *Proc. of the 10th International Symposium on Methodologies for Intelligent Systems*, LNAI 1325, pages 579–589. Springer-Verlag, 1997.

[12] D.E.R. Denning. *Cryptography and Data Security*. Addison-Wesley Publishing Company, 1982.

[13] J. Domingo-Ferrer. Advances in inference control in statistical databases: An overview. In *Inference Control in Statistical Databases: From Theory to Practice*, LNCS 2316, pages 1–7. Springer-Verlag, 2002.

[14] S. Even and R.E. Tarjan. "Network flow and testing graph connectivity". *SIAM Journal on Computing*, 4:507–518, 1975.

[15] R. Fagin and J. Halpern. "Reasoning about knowledge and probability". *Journal of the ACM*, 41(2):340–367, 1994.

[16] R. Fagin, J.Y. Halpern, Y. Moses, and M.Y. Vardi. *Reasoning about Knowledge*. MIT Press, 1996.

[17] T.F. Fan, C.J. Liau, and Y.Y. Yao. "On modal decision logics". In T.Y. Lin and C.J. Liau, editors, *Proceedings of the PAKDD'02 Workshop on the Foundation of Data Mining*, pages 21–26, 2002.

[18] J. Glasgow, G. MacEwen, and P. Panangaden. "A logic for reasoning about security". In *Proc. of the 3rd IEEE Computer Security Foundations Workshop*, pages 2–13, 1990.

[19] J. Glasgow, G. MacEwen, and P. Panangaden. "A logic for reasoning about security". *ACM Transactions on Computer Systems*, 10(3):226–264, 1992.

[20] J. Halpern. "A logical approach to reasoning about uncertainty: a tutorial". In X. Arrazola, K. Korta, and F.J. Pelletier, editors, *Discourse, Interaction, and Communication*, pages 141–155. Kluwer Academic Publishers, 1998.

[21] J. Hintikka. *Knowledge and Belief*. Cornell University Press, 1962.

[22] T.-s. Hsu, C.J. Liau, and D.W. Wang. A logical model for privacy protection. In *Proceedings of the 4th International Conference on Information Security*, LNCS 2200, pages 110–124. Springer-Verlag, 2001.

[23] T.-s. Hsu, C.J. Liau, D.W. Wang, and Jeremy K.P. Chen. Quantifying privacy leakage through answering database queries. In *Proceedings of the 5th International Conference on Information Security*, LNCS 2433, pages 162–175. Springer-Verlag, 2002.

[24] J.W. Gray III and P.F. Syverson. A logical approach to multilevel security of probabilistic systems. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 164–176, 1992.

[25] J.W. Gray III and P.F. Syverson. "Epistemology of information flow in the multilevel security of probabilistic systems". NRL Memo Report 5540-95-7733, Naval Research Laboratory, Washington, 1995.

[26] J.W. Gray III and P.F. Syverson. "A logical approach to multilevel security of probabilistic systems". *Distributed Computing*, 11(2):73–90, 1998.

[27] J. Jaspars and E. Thijsse. "Fundamentals of partial modal logic". In P. Doherty, editor, *Partiality, Modality, and Nonmonotonicity*, pages 111–141. CSLI Publications, 1996.

[28] G.J. Klir and M.J. Wierman. *Uncertainty-Based Information : Elements of Generalized Information Theory*. Physica-Verlag, 1998.

[29] M. Li and P. Vitanyi. *An introduction to Kolmogorov Complexity and its Applications*. Springer-Verlag, 1993.

[30] G. MacEwen, X.J. Chen, and S. Kinght. "An action-based logic of causality, knowledge, permission and obligation". Technical Report 1997-405, Department of Computing and Information Science, Queen's University, Canada, 1997.

[31] J.-J. Ch. Meyer and R.J. Wieringa, editors. *Deontic Logic in Computer Science: Normative System Specification*. John Wiley & Sons Ltd., 1993.

[32] M. Morgenstern. "Controlling logical inference in multilevel database systems". In *Proc. of the IEEE Symposium on Security and Privacy*, pages 245–255, 1988.

[33] Z. Pawlak. *Rough Sets–Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, 1991.

[34] P. Samarati. "Protecting respondents' identities in microdata release". *IEEE Transactions on Knowledge and Data Engineering*, 13(6):1010–1027, 2001.

[35] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report SRI-CSL-98-04, Computer Science Laboratory, SRI International, 1998.

[36] C.E. Shannon. "The mathematical theory of communication". *The Bell System Technical Journal*, 27(3&4):379–423,623–656, 1948.

[37] R. Srikant. Privacy preserving data mining: challenges and opportunities. In *Proceedings of the 6th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, LNCS 2336, page 13. Springer-Verlag, 2002.

[38] D. Sutherland. A model of information. In *Proceedings of the 9 th National Computer Security Conference*, pages 175–183, 1986.

[39] L. Sweeney. "Guaranteeing anonymity when sharing medical data, the Datafly system". A.I. Working Paper AIWP-WP344, MIT AI Lab., 1997.

[40] P.F. Syverson and J.W. Gray III. The epistemic representation of information flow security in probabilistic systems. In *Proc. of the 8th IEEE Computer Security Foundations Workshop*, pages 152–166, 1995.

[41] P.F. Syverson and S.G. Stubblebine. Group principals and the formalization of anonymity. In *Proc. of the 1999 World Congress on Formal Methods*, LNCS 1708, pages 814–833, 1999.

[42] B.C. van Fraassen. *Formal Semantics and Logic*. Macmillan, New York, 1971.

[43] L.A. Zadeh. "Fuzzy sets". *Information and Control*, 8:338–353, 1965.

A Some GUI's of the Cellsecu 2.0 system

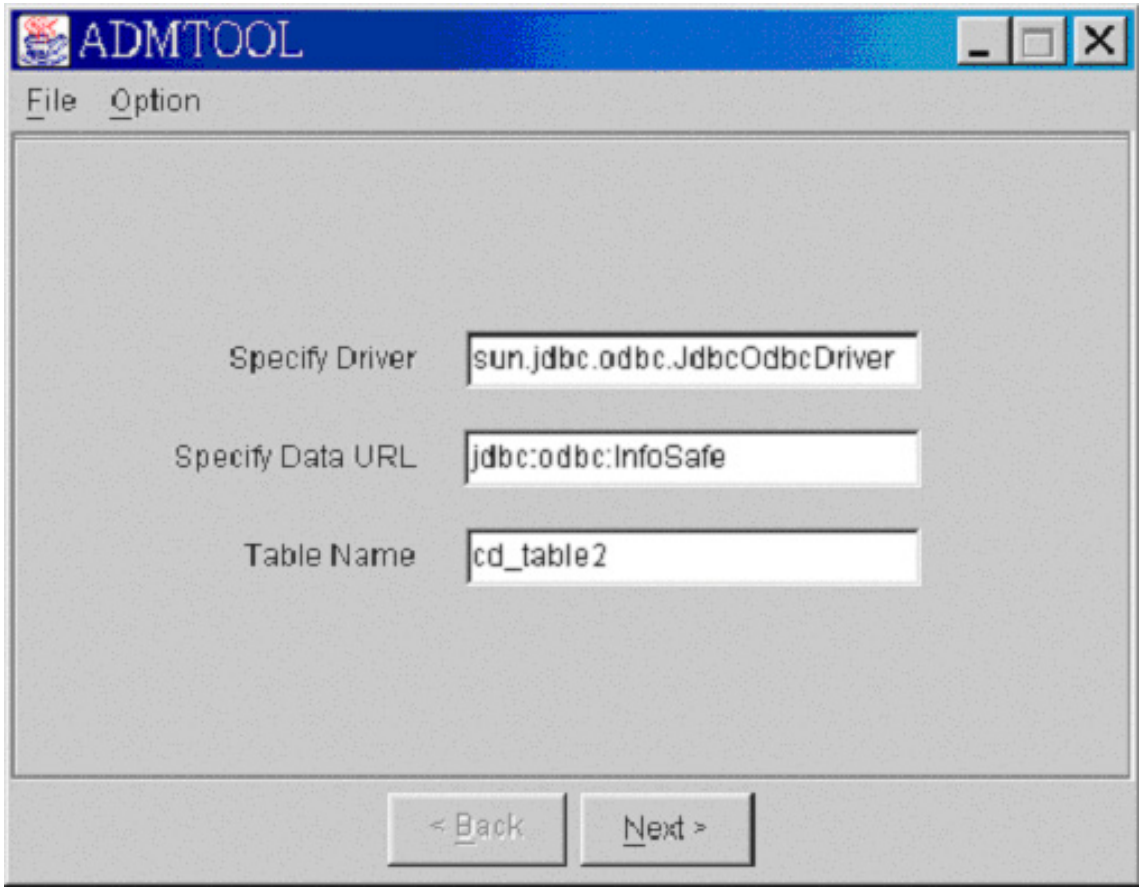

Figure 15: Specify the source database

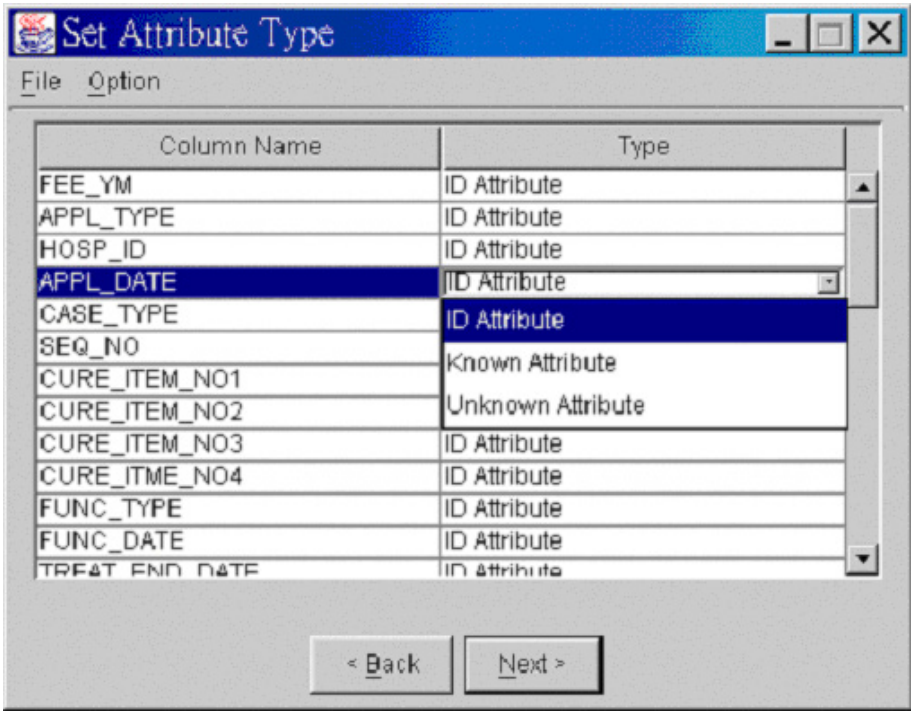

Figure 16: Define the type of attributes

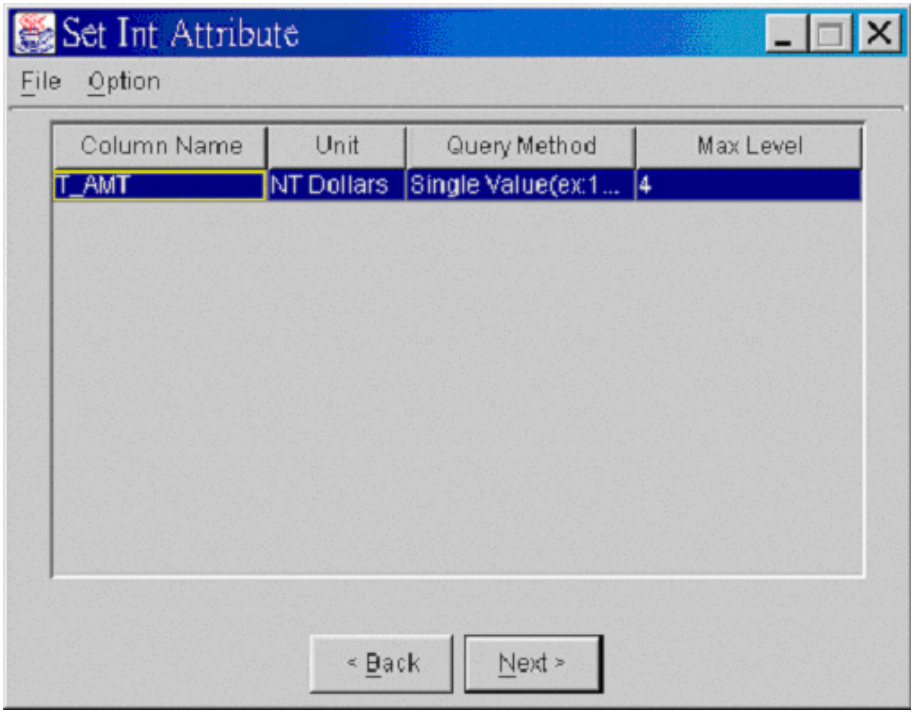

Figure 17: Set up data types of public attributes

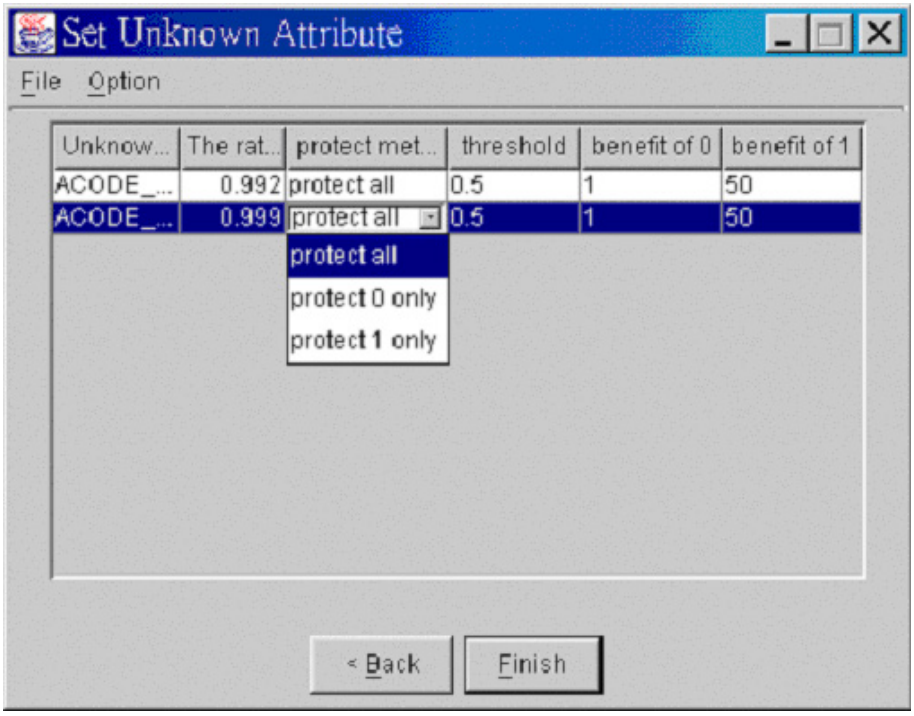

Figure 18: Set up confidential attributes

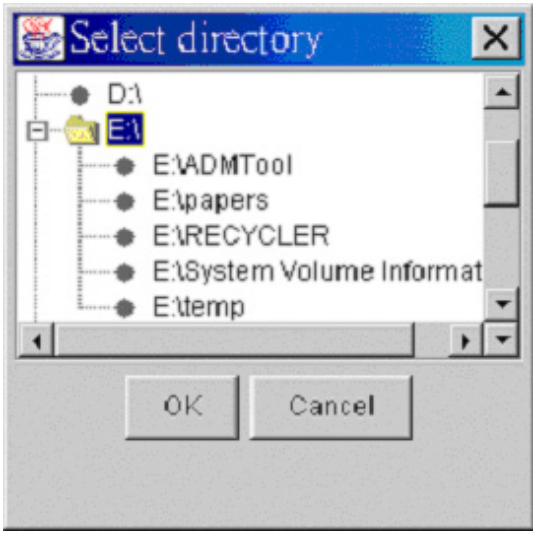

Figure 19: Save files

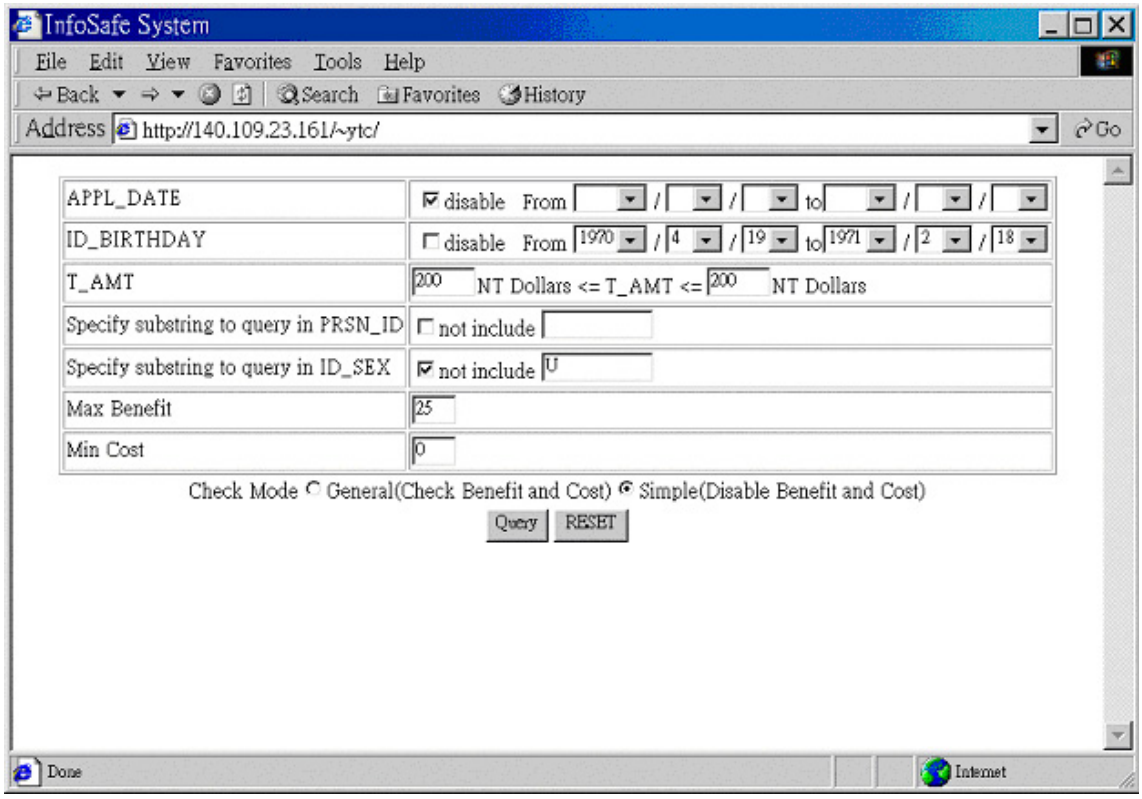

Figure 20: Use the database filter