# Reasoning about Domain Semantics over Relations, Bags, Partial Relations and Partial Bags

Sebastian Link
*Department of Computer Science*
*The University of Auckland*
*Auckland, New Zealand*
*s.link@auckland.ac.nz*

***Abstract*—Quality database schemata must capture both the structure and semantics of the domain of interest. Classes of data dependencies have been studied extensively to model domain semantics. Traditionally, the theory of data dependencies has been limited to relations. In practice, duplicate and partial information are permitted to occur in database instances. These features are supported to make data processing more efficient. We study the implication problem for an expressive class of data dependencies over all data structures that arise from the two features features. These include bags that permit duplicate tuples, partial relations that permit null marker occurrences, and partial bags that permit duplicate tuples and null marker occurrences. The class of data dependencies studied encompasses uniqueness constraints, functional and multivalued dependencies. We establish axiomatizations and sharp upper bounds for the worst-case time complexity of the implication problem.**

***Keywords*-Data models; Database design; Database semantics; Decision problems; Mathematical logic.**

## I. INTRODUCTION

A database system manages a collection of persistent information in a shared, reliable, effective and efficient way. Most commercial database systems are still founded on *the relational model of data* [1]. Data administrators utilize various classes $\mathcal{C}$ of first-order formulae, called *data dependencies*, to restrict the relations in the database to those considered meaningful to the application domain at hand. A central problem in logic, mathematics and computer science is the *implication problem* of such classes $\mathcal{C}$ [2]. In terms of data dependencies the problem is to decide whether for an arbitrarily given set $\Sigma \cup \{\varphi\}$ of data dependencies in $\mathcal{C}$, $\Sigma$ implies $\varphi$, i.e., whether every database instance that satisfies all the elements of $\Sigma$ also satisfies $\varphi$. For databases specifically, solutions to the implication problem are essential for their modeling and design [3], and can advance many data processing tasks such as updates [4], queries [5], security [6], maintenance [7], cleaning [8], integration [9] and exchange [10]. According to [11] the combined class of uniqueness constraints (UCs) and functional dependencies (FDs) captures around two-thirds, and the class of multivalued dependencies (MVDs) around one-quarter of all uni-relational dependencies (those defined over a single relation schema) that arise in practice. In particular, MVDs are frequently exhibited in database applications [12], e.g., after de-normalization or in views [3]. The next example illustrates how instances of the implication problem arise naturally from table definitions in SQL [13], which has been the industry standard for defining and querying data for the last three decades.

*Example 1:* Consider a table definition SUPPLIES with column headers *A(rticle)*, *S(upplier)*, *L(ocation)* and *C(ost)*. The intention is to collect information about suppliers that deliver articles from a location at a certain cost.

```
CREATE TABLE SUPPLIES (
    Article CHAR[20],
    Supplier VARCHAR NOT NULL,
    Location VARCHAR NOT NULL,
    Cost CHAR[8]);
```

Suppose the database management system enforces the following set $\Sigma$ of constraints: The FD $A \to S$ says that for every article there is at most one supplier, the FD $AL \to C$ says that the cost is determined by the article and the location, and the MVD $S \twoheadrightarrow L$ says that the locations are determined by the supplier independently of the articles and costs. Do the following semantically meaningful constraints need to be enforced explicitly, or are they already enforced implicitly by $\Sigma$: i) the UC $u(AL)$, ii) the FD $A \to C$, and iii) the MVD $A \twoheadrightarrow L$?

SQL table definitions permit occurrences of duplicate tuples and occurrences of a null marker in columns declared NULL. While these two features are meant to make data processing more efficient, they do distinguish the 32 billion US dollar market of SQL-based relational database systems from Codd's relational model of data. In this paper, we investigate in detail the impact of these two features on the implication problem of the combined class of UCs, FDs and MVDs. In fact, we use these two features to study the implication problem over the four resulting data structures: relations, bags, partial relations and partial bags. Relations are sets of total tuples. That is, all columns are NOT NULL by default and duplicate tuples are not permitted to occur.

*Example 2:* Suppose we use relations to instantiate the SQL table definition of Example 1. Then the constraints i),

 ISBN: 978-1-61208-185-4 

ii) and iii) are all enforced implicitly by $\Sigma$. In particular, the FDs $A \rightarrow S$ and $AL \rightarrow C$ imply the FD $AL \rightarrow ALCS$. Since duplicate tuples are identified in sets of tuples, the latter FD is equivalent to the uniqueness constraint $u(AL)$.

Bags of tuples are more general than relations, i.e., sets of tuples. In fact, relations are bags where no duplicate tuples can occur, i.e., no two different tuples can occur that have matching values on all attributes.

*Example 3:* Suppose we use bags of tuples to instantiate the SQL table definition of Example 1. Then the semantically meaningful constraints ii) and iii) are still enforced implicitly by $\Sigma$. However, the uniqueness constraint $u(AL)$ is not enforced implicitly by $\Sigma$. Indeed, the bag

| *Article* | *Supplier* | *Location* | *Cost* |
|---|---|---|---|
| *Kiwi* | *Kiwifruitz* | *Tauranga* | *3* |
| *Kiwi* | *Kiwifruitz* | *Tauranga* | *3* |

satisfies $\Sigma$ but violates $u(AL)$.

Partial relations are sets of partial tuples $t$, i.e., $t(A)$ can carry a null marker occurrence on every attribute $A$. Here, we adapt the most general interpretation of a null marker, denoted by `ni`, i.e., the *no information* interpretation [14], [15], [16]. In general it may happen that two different tuples *subsume* one another. That is, there are two tuples $t$ and $t'$ such that for every attribute $A$ it holds that $t'(A) = \texttt{ni}$ or $t'(A) = t(A)$. We require partial relations to be *subsumption-free*. This requirement is a natural generalization of relations which are duplicate-free, and is in line with previous research [14], [15], [16]. Furthermore, in SQL one can define any attribute $A$ as `NOT NULL`. That is, for every partial tuple $t$ it must hold that $t(A) \neq \texttt{ni}$. We say that the set of attributes declared `NOT NULL` forms the null-free subschema of the underlying schema.

*Example 4:* Suppose we use partial relations to instantiate the SQL table definition of Example 1. The UC $u(AL)$ is implied by $\Sigma$ since duplicate tuples are not allowed in partial relations. For the semantically meaningful constraints ii) and iii) it depends on the null-free subschema whether they are enforced implicitly by $\Sigma$. If the null-free subschema is $\{S, L\}$, then both ii) and iii) are implied by $\Sigma$. However, if it is $\{A, L, C\}$, then the following partial relation

| *Article* | *Supplier* | *Location* | *Cost* |
|---|---|---|---|
| *Kiwi* | `ni` | *Tauranga* | *3* |
| *Kiwi* | `ni` | *Gisborne* | *4* |

satisfies $\Sigma$, but violates the FD $A \rightarrow C$ and MVD $A \twoheadrightarrow L$.

Finally, partial bags are bags of partial tuples. In particular, partial bags may contain two tuples that subsume one another. This includes the special case of duplicate tuples.

*Example 5:* Suppose we use partial bags to instantiate our SQL table definition. Then the situation is similar to Example 4, but the UC $u(AL)$ is not implied by $\Sigma$.

**Contributions.** We establish finite axiomatizations for the combined class of uniqueness constraints, functional and multivalued dependencies over bags and partial bags. In particular, the presence of duplicate (partial) tuples makes it necessary to include the class of uniqueness constraints into the combined class. That is, in the presence of duplicate tuples, uniqueness constraints are no longer covered by functional dependencies - in contrast to (partial) relations. Our main proof arguments for the case of (partial) bags uses a reduction to the case of (partial) relations, respectively. The benefit of these reductions is to pinpoint exactly which new inference rules are required to gain completeness in each of the cases. Our proof techniques also enable us to establish sharp upper bounds on the worst-case time complexity of the associated decision problems. In particular, the bounds match the currently best known bound for the special case of relations known from the literature. Our findings establish a complete picture of how reasoning about domain semantics in different data structures can be automated effectively and efficiently. Our most general case addresses partial bags, which are used to instantiate SQL table definitions in practice. Our findings close the gap between existing database theory and database practice. The class of data dependencies studied is treated in most introductory textbooks on databases; unfortunately for the case of relations only. Our results provide therefore new insight for students and researchers on the impact of popular data structures on the reasoning about domain semantics. Finally, note that more expressive classes of data dependencies, such as join dependencies, are not finitely axiomatizable [17].

**Organization.** We briefly summarize related work in Section II. The general data model of partial bag schemata is introduced in Section III. The known special cases from the literature are reviewed in Section IV. In Section V we establish axiomatizations for the general case of partial bag schemata. Our proof argument enables us to establish a sharp upper bound on the worst-case time complexity in Section VI. We comment on the applicability of our theories in practice in Section VII. We conclude in Section VIII where we also comment on future work.

## II. Related work

Data dependencies can capture the semantics of the domain of interest in the target database. Therefore, data dependencies are essential to database design, and the maintenance of the database during its lifetime, and all major data processing tasks, cf. [3], [18].

In the relational model, a UC $u(X)$ over relation schema $R$ is satisfied by a relation if and only if the relation satisfies the FD $X \rightarrow R$. Hence, in this context it suffices to study the class of FDs and MVDs. Beeri, Fagin and Howard established the first axiomatization for FDs and MVDs [19], [20], [21]. The associated implication problem can be decided in time almost-linear in the input [22].

 ISBN: 978-1-61208-185-4 

One of the most important extensions of the relational model [1] is partial information [23]. This is mainly due to the high demand for the correct handling of such information in real-world applications. While there are several possible interpretations of a null marker, many of the previous work on data dependencies is based on Zaniolo's *no information* interpretation [14], [15], [24], [16]. Atzeni and Morfuni established an axiomatization for the class of FDs over partial relations [14]. In particular, they did not permit subsumption between partial tuples and did not consider MVDs. Köhler and Link investigated UCs and FDs over bags, but considered neither null markers nor MVDs [25]. Finally, Hartmann and Link established an axiomatization for the class of FDs and MVDs over partial relations [26].

## III. THE SQL DATA MODEL

In this section we introduce the general SQL data model, which is based on partial bags. We utilize this general model to define the remaining three cases of relations, partial relations and bags as important special cases.

### A. Structures and data structures

Let $\mathfrak{A} = \{A_1, A_2, \ldots\}$ be a (countably) infinite set of distinct symbols, called *attributes*. A *partial bag schema* is a pair $\mathcal{S} = (S, \mathit{nfs}(S))$ consisting of a finite non-empty subset $S$ of $\mathfrak{A}$, and a subset $\mathit{nfs}(S) \subseteq S$. Each attribute $A$ is associated with a countably infinite domain $dom(A)$, which represents the possible values that can occur in the column $A$ represents. To encompass partial information every attribute may have a null marker, denoted by $\mathtt{ni} \in dom(A)$. The intention of ni is to mean *no information*. This interpretation can therefore model non-existing as well as existing but unknown information [14], [16], but it cannot distinguish between the two - as is the exact case in SQL.

For attribute sets $X$ and $Y$ we may write $XY$ for the set union $X \cup Y$. If $X = \{A_1, \ldots, A_m\}$, then we may write $A_1 \cdots A_m$ for $X$. In particular, we may write simply $A$ to represent the singleton $\{A\}$. A *partial tuple* over $\mathcal{S}$ is a function $t : S \to \bigcup_{A \in S} dom(A)$ with $t(A) \in dom(A)$ for all $A \in S$, and $t(A) \neq \mathtt{ni}$ for all $A \in \mathit{nfs}(S)$. The null marker occurrence $t(A) = \mathtt{ni}$ associated with an attribute $A$ in a partial tuple $t$ means that no information is available about the attribute $A$ for the partial tuple $t$. For $X \subseteq S$ let $t(X)$ denote the restriction of the partial tuple $t$ over $\mathcal{S}$ to $X$. A partial tuple $t$ is said to be $X$-total, if for all $A \in X$ it holds that $t(A) \neq \mathtt{ni}$. Hence, every partial tuple over $\mathcal{S}$ is $\mathit{nfs}(S)$-total. A *partial bag* over $\mathcal{S}$ is a finite multi-set of partial tuples over $\mathcal{S}$.

A *bag schema* is a partial bag schema $(S, \mathit{nfs}(S))$ where $\mathit{nfs}(S) = S$. Here, we may simply write $S$ instead of $(S, S)$. Consequently, all partial tuples $t$ over a bag schema $S$ are $S$-total partial tuples, i.e., for all $A \in S$ it holds that $t(A) \neq \mathtt{ni}$. In this case, we may also speak of total tuples or just tuples.

For two partial tuples $t$ and $t'$ we say that $t$ *subsumes* $t'$, if for every attribute $A$ it holds that $t'(A) = \mathtt{ni}$ or $t'(A) = t(A)$. A partial relation over $(S, \mathit{nfs}(S))$ is a partial bag that is *subsumption-free*, i.e., there are no two different partial tuples in the partial relation that subsume one another. We call a partial bag schema $(S, \mathit{nfs}(S))$ a *partial relation schema* if we restrict all partial bags over $(S, \mathit{nfs}(S))$ to be partial relations.

Finally, a *relation schema* is a partial relation schema $(S, \mathit{nfs}(S))$ where $\mathit{nfs}(S) = S$. Again, we may simply write $S$ instead of $(S, S)$. Consequently, every partial relation over a relation schema is a relation, i.e., a set of tuples.

*Example 6:* The following database instance over (SUPPLIES, *nfs*(SUPPLIES)), where *nfs*(SUPPLIES) = {*Article, Location, Cost*}, is a partial bag that is not a partial relation.

| *Article* | *Supplier* | *Location* | *Cost* |
|---|---|---|---|
| *Kiwi* | *Kiwifruitz* | *Gisborne* | *4* |
| *Kiwi* | ni | *Gisborne* | *4* |

Indeed, the first tuple subsumes the second tuple.

### B. Semantics

In what follows we define the syntax and semantics of uniqueness constraints, functional and multivalued dependencies in the context of partial bags. We will briefly comment on the restrictions to the special cases of relations, partial relations and bags.

Following the SQL standard a *uniqueness constraint* (UC) over a partial bag schema $\mathcal{S} = (S, \mathit{nfs}(S))$ is an expression $u(X)$ where $X \subseteq S$. A partial bag $\mathfrak{b}$ over $\mathcal{S}$ is said to satisfy the uniqueness constraint $u(X)$ over $\mathcal{S}$ ($\models_{\mathfrak{b}} u(X)$) if and only if for all partial tuples $t, t' \in \mathfrak{b}$ the following holds: if $t \neq t'$ and $t$ and $t'$ are both $X$-total, then there is some $A \in X$ such that $t(A) \neq t'(A)$. Note that the notion of a uniqueness constraint over bag and relation schemata matches the well-known notion of a key.

Functional dependencies are important for the relational [1] and other data models [27], [28], [29], [30], [31]. Generalizing notions by Lien [15], a *functional dependency* (FD) over a partial bag schema $\mathcal{S}$ is a statement $X \to Y$ where $X, Y \subseteq S$. The FD $X \to Y$ over $\mathcal{S}$ is satisfied by a partial bag $\mathfrak{b}$ over $\mathcal{S}$ ($\models_{\mathfrak{b}} X \to Y$) if and only if for all $t, t' \in \mathfrak{b}$ the following holds: if $t$ and $t'$ are both $X$-total and $t(X) = t'(X)$, then $t(Y) = t'(Y)$. We call $X \to Y$ *trivial* whenever $Y \subseteq X$, and non-trivial otherwise. The general FD definition is consistent with the *no information* interpretation [14], [15]. For bag and relation schemata the notion of a functional dependency reduces to that of the standard definition of a functional dependency [18], and so is a sound generalization. Note that any partial relation $\mathfrak{b}$ satisfies the FD $X \to S$ over $\mathcal{S}$ if and only if $\mathfrak{b}$ satisfies the UC $u(X)$. This is invalid for bags and partial bags.

 ISBN: 978-1-61208-185-4

Generalizing notions by Lien [15], a *multivalued dependency* (MVD) over $\mathcal{S}$ is a statement $X \twoheadrightarrow Y$ where $X, Y \subseteq S$. The MVD $X \twoheadrightarrow Y$ over $\mathcal{S}$ is satisfied by a partial bag $\mathfrak{b}$ over $\mathcal{S}$, denoted by $\models_{\mathfrak{b}} X \twoheadrightarrow Y$, if and only if for all $t, t' \in \mathfrak{b}$ the following holds: if $t$ and $t'$ are both $X$-total and $t(X) = t'(X)$, then there is some $\bar{t} \in \mathfrak{b}$ such that $\bar{t}(XY) = t(XY)$ and $\bar{t}(X(S-Y)) = t'(X(S-Y))$. We call $X \twoheadrightarrow Y$ *trivial* whenever $Y \subseteq X$ or $XY = S$, and non-trivial otherwise. This MVD definition is consistent with the *no information* interpretation [15]. For bag and relation schemata the notion of an MVD reduces to that of the standard definition of an MVD [32], and so is a sound generalization.

*Example 7:* Consider the partial bag schema $(\text{SUPPLIES}, \mathit{nfs}(\text{SUPPLIES}))$ where $\mathit{nfs}(\text{SUPPLIES}) = \{\mathit{Supplier}, \mathit{Location}\}$. The partial relation

| *Article* | *Supplier* | *Location* | *Cost* |
|---|---|---|---|
| *Kiwi* | ni | *Tauranga* | *3* |
| *Kiwi* | ni | *Gisborne* | *4* |

satisfies the UC $u(AL)$, the FDs $A \to S$ and $AL \to C$, and the MVD $S \twoheadrightarrow L$. It violates the UC $u(A)$, the FD $A \to C$ and the MVD $A \twoheadrightarrow L$.

### C. Semantic implication and syntactic inference

For a set $\Sigma$ of constraints over some partial bag schema $\mathcal{S}$, we say that a partial bag $\mathfrak{b}$ over $\mathcal{S}$ *satisfies* $\Sigma$ ($\models_{\mathfrak{b}} \Sigma$) if $\mathfrak{b}$ satisfies every $\sigma \in \Sigma$. If for some $\sigma \in \Sigma$, $\mathfrak{b}$ does not satisfy $\sigma$ we say that $\mathfrak{b}$ violates $\sigma$ (and violates $\Sigma$) and write $\not\models_{\mathfrak{b}} \sigma$ ($\not\models_{\mathfrak{b}} \Sigma$). In the general case of partial bags we are interested in the combined class $\mathcal{C}$ of UCs, FDs and MVDs.

Constraints interact with one another. Let $\mathcal{S}$ be a partial bag schema, and let $\Sigma \cup \{\varphi\}$ be a set of UCs, FDs and MVDs over $\mathcal{S}$. We say that $\Sigma$ *implies* $\varphi$ ($\Sigma \models \varphi$) if every partial bag $\mathfrak{b}$ over $\mathcal{S}$ that satisfies $\Sigma$ also satisfies $\varphi$. If $\Sigma$ does not imply $\varphi$ we may also write $\Sigma \not\models \varphi$. For $\Sigma$ we let $\Sigma^* = \{\varphi \mid \Sigma \models \varphi\}$ be the *semantic closure* of $\Sigma$, i.e., the set of all UCs, FDs and MVDs implied by $\Sigma$. In order to determine the implied constraints we use a syntactic approach by applying inference rules. These inference rules have the form

$$\frac{\text{premise}}{\text{conclusion}}\ \text{condition},$$

and inference rules without any premise are called axioms. An inference rule is called *sound*, if whenever the set of constraints in the premise of the rule are satisfied by some partial bag over $\mathcal{S}$ and the constraints satisfy the conditions of the rule, then the partial bag also satisfies the constraint in the conclusion of the rule. We let $\Sigma \vdash_{\mathfrak{S}} \varphi$ denote the *inference* of $\varphi$ from $\Sigma$ by $\mathfrak{S}$. That is, there is some sequence $\gamma = [\sigma_1, \ldots, \sigma_n]$ of constraints such that $\sigma_n = \varphi$ and every $\sigma_i$ is an element of $\Sigma$ or results from an application of an inference rule in $\mathfrak{S}$ to some elements in $\{\sigma_1, \ldots, \sigma_{i-1}\}$.

$$\frac{}{XY \to Y}\ \text{(reflexivity, } \mathcal{R}_{\text{F}}) \qquad \frac{X \to Y}{XU \to YV}\ V \subseteq U\ \text{(FD augmentation, } \mathcal{A}_{\text{F}})$$

$$\frac{X \to Y \qquad Y \to Z}{X \to Z}\ \text{(transitivity, } \mathcal{T}'_{\text{F}})$$

$$\frac{X \twoheadrightarrow Y}{X \twoheadrightarrow S - Y}\ (S\text{-complementation, } \mathcal{C}^S_{\text{M}}) \qquad \frac{X \twoheadrightarrow Y}{XU \twoheadrightarrow YV}\ V \subseteq U\ \text{(MVD augmentation, } \mathcal{A}_{\text{M}})$$

$$\frac{X \twoheadrightarrow Y \qquad Y \to Z}{X \twoheadrightarrow Z - Y}\ \text{(pseudo-transitivity, } \mathcal{T}'_{\text{M}})$$

$$\frac{X \to Y}{X \twoheadrightarrow Y}\ \text{(MVD implication, } \mathcal{I}_{\text{FM}}) \qquad \frac{X \twoheadrightarrow Y \qquad Y \to Z}{X \to Z - Y}\ \text{(mixed pseudo-transitivity, } \mathcal{T}'_{\text{FM}})$$

Table I
AXIOMATIZATION $\mathfrak{R}$ OVER RELATION SCHEMA $S$

For a finite set $\Sigma$, let $\Sigma^+_{\mathfrak{S}} = \{\varphi \mid \Sigma \vdash_{\mathfrak{S}} \varphi\}$ be its *syntactic closure* under inferences by $\mathfrak{S}$. A set $\mathfrak{S}$ of inference rules is said to be *sound* (*complete*) for the implication of UCs, FDs and MVDs if for every partial bag schema $\mathcal{S}$ and for every set $\Sigma$ of UCs, FDs and MVDs over $\mathcal{S}$ we have $\Sigma^+_{\mathfrak{S}} \subseteq \Sigma^*$ ($\Sigma^* \subseteq \Sigma^+_{\mathfrak{S}}$). The (finite) set $\mathfrak{S}$ is said to be a (finite) *axiomatization* for the implication of UCs, FDs and MVDs if $\mathfrak{S}$ is both sound and complete.

## IV. AXIOMATIZATIONS FOR RELATIONS AND PARTIAL RELATIONS

In this section we briefly review a well-known axiomatization for the class of FDs and MVDs over relations. We then review a recent axiomatization for the same class of data dependencies over partial relations.

### A. Relations

Beeri, Fagin, and Howard [19] established the first axiomatization for the class of FDs and MVDs over relations. The axiomatization $\mathfrak{R}$ of Table I is based on the (mixed) pseudo-transitivity rules by Zaniolo [21].

The following example demonstrates the use of the inference rules to infer some data dependencies implied over relations. Firstly, it highlights how in the absence of partial data, the transitivity rules can be applied soundly. Secondly, it highlights how in the absence of duplicate tuples, FDs can be used to infer uniqueness constraints.

*Example 8:* Consider the relation schema SUPPLIES and the set $\Sigma$ containing the FDs $A \to S$, $AL \to C$ and the MVD $S \twoheadrightarrow L$.

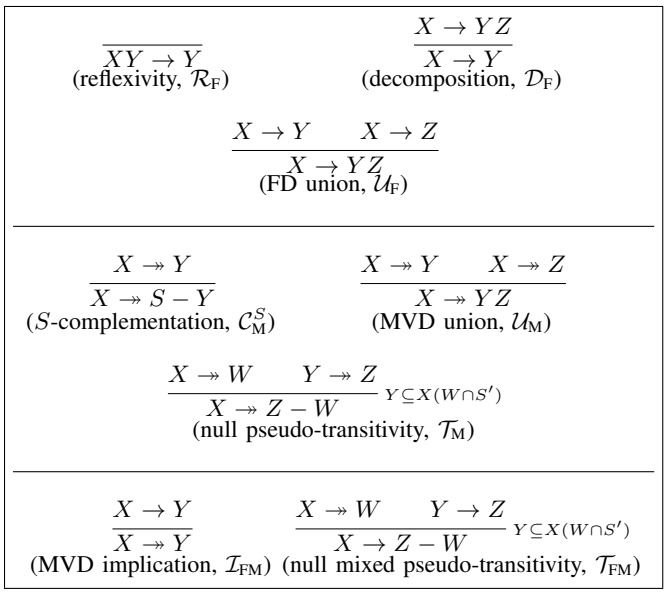


Table II
AXIOMATIZATION $\mathfrak{p}\mathfrak{R}$ OVER PARTIAL RELATION SCHEMA $(S, S')$

An application of the *MVD implication rule* $\mathcal{I}_{\mathrm{FM}}$ to $A \to S$ results in the MVD $A \twoheadrightarrow S$. We can apply the *pseudo-transitivity rule* $\mathcal{T}'_{\mathrm{M}}$ to the MVDs $A \twoheadrightarrow S$ and $S \twoheadrightarrow L$ to infer the MVD $A \twoheadrightarrow L$.

An application of the *MVD augmentation rule* $\mathcal{A}_{\mathrm{M}}$ to $A \twoheadrightarrow L$ yields the MVD $A \twoheadrightarrow AL$. An application of the *mixed pseudo-transitivity rule* $\mathcal{T}'_{\mathrm{FM}}$ to the MVD $A \twoheadrightarrow AL$ and the FD $AL \to C$ results in the FD $A \to C$.

An application of the *FD augmentation rule* $\mathcal{A}_{\mathrm{F}}$ to $A \to S$ yields the FD $AC \to SC$, and an application of the same rule to the FD $AL \to C$ yields the FD $AL \to AC$. An application of the *transitivity rule* $\mathcal{T}'_{\mathrm{F}}$ to the FDs $AL \to AC$ and $AC \to SC$ results in the FD $AL \to SC$. A final application of the *FD augmentation rule* $\mathcal{A}_{\mathrm{F}}$ to $AL \to SC$ yields the FD $AL \to ACLS$. Note that over a relation schema $R$ the FD $X \to R$ is satisfied by the same relations as the UC $u(AL)$. Thus, the last inference yields the UC $u(AL)$.

### B. Partial relations

Over partial relations, Hartmann and Link recently established the axiomatization $\mathfrak{p}\mathfrak{R}$ for the class of FDs and MVDs [26]. As Example 4 shows, the choice of a null-free subschema has an impact on the data dependencies implied. In particular, the presence of partial data requires that the applicability of the (mixed) pseudo-transitivity rules are suitably restricted.

The next example illustrates applications of the null (mixed) pseudo-transitivity rules to infer implied data dependencies. Note how changes in the null-free subschema influence the applicability of these rules, cf. Example 4.

*Example 9:* Consider the partial relation schema $(\text{SUPPLIES}, \{SL\})$ and the set $\Sigma$ containing the FDs $A \to S$, $AL \to C$ and the MVD $S \twoheadrightarrow L$.

An application of the *MVD implication rule* $\mathcal{I}_{\mathrm{FM}}$ to $A \to S$ results in the MVD $A \twoheadrightarrow S$. We can apply the *null pseudo-transitivity rule* $\mathcal{T}'_{\mathrm{M}}$ to the MVDs $A \twoheadrightarrow S$ and $S \twoheadrightarrow L$ to infer the MVD $A \twoheadrightarrow L$. Note that $S \in \mathit{nfs}(\text{SUPPLIES})$.

An application of the *reflexivity axiom* $\mathcal{R}_{\mathrm{F}}$ followed by an application of the *MVD implication rule* $\mathcal{I}_{\mathrm{FM}}$ results in the MVD $A \twoheadrightarrow A$. An application of the *MVD union rule* $\mathcal{U}_{\mathrm{M}}$ to $A \twoheadrightarrow A$ and $A \twoheadrightarrow L$ results in the MVD $A \twoheadrightarrow AL$. An application of the *null mixed pseudo-transitivity rule* $\mathcal{T}_{\mathrm{FM}}$ to the MVD $A \twoheadrightarrow AL$ and the FD $AL \to C$ results in the FD $A \to C$. Again, note here that $L \in \mathit{nfs}(\text{SUPPLIES})$.

## V. AXIOMATIZATIONS FOR BAGS AND PARTIAL BAGS

In this section we establish the first main results of this article, i.e., finite axiomatizations for the combined class of UCs, FDs, and MVDs over bags and over partial bags. We prove the general case of partial bags in detail.

### A. Partial bags

Let $\mathfrak{p}\mathfrak{B}$ denote the set of inference rules in Table IV. We first establish the soundness of the rules in $\mathfrak{p}\mathfrak{B}$.

*Lemma 1:* The set $\mathfrak{p}\mathfrak{B}$ of inference rules is sound.

*Proof:* The soundness of the rules in $\mathfrak{p}\mathfrak{R}$ has been established in previous work [26]. It remains to show the soundness of the *FD implication rule* $\mathcal{I}_{\mathrm{UF}}$ and the *null pullback rule* $\mathcal{P}_{\mathrm{UF}}$.

For the soundness of the *FD implication rule* $\mathcal{I}_{\mathrm{UF}}$ assume there is some partial bag $\mathfrak{b}$ that violates the FD $X \to Y$. Then there must be two different tuples $t, t' \in \mathfrak{b}$ that are $X - total$ and $t(X) = t'(X)$. Consequently, $\mathfrak{b}$ also violates the UC $u(X)$.

For the soundness of the *null pullback rule* $\mathcal{P}_{\mathrm{UF}}$ assume there is some partial bag $\mathfrak{b}$ over $\mathcal{S} = (S, S')$ that violates the uniqueness constraint $u(X)$. Then there must be two different tuples $t, t' \in \mathfrak{b}$ that are $X - total$ and $t(X) = t'(X)$. If $\mathfrak{b}$ violates the FD $X \to Y$, then we are done. Otherwise, it follows that $t(Y) = t'(Y)$. If $Y \not\subseteq XS'$, then we are done. Otherwise, it follows that $t$ and $t'$ are both $Y$-total. Consequently, $\mathfrak{b}$ violates the UC $u(Y)$. ■

For the completeness of $\mathfrak{p}\mathfrak{B}$ we use the result that the set $\mathfrak{p}\mathfrak{R}$ forms an axiomatization for FDs and MVDs over partial relations [26]. In fact, the completeness of $\mathfrak{p}\mathfrak{B}$ follows from that of $\mathfrak{p}\mathfrak{R}$ and the following lemma. For a set $\Sigma$ of UCs, FDs, and MVDs over partial bag schema $(S, \mathit{nfs}(S))$ let $\Sigma[\text{FM}] = \{X \to S \mid u(X) \in \Sigma\} \cup \{X \to Y \mid X \to Y \in \Sigma\} \cup \{X \twoheadrightarrow Y \mid X \twoheadrightarrow Y \in \Sigma\}$.

*Lemma 2:* Let $\Sigma$ be a set of UCs, FDs and MVDs over the partial bag schema $(S, S')$. Then the following hold:

1) $\Sigma \models X \to Y$ if and only if $\Sigma[\text{FM}] \models X \to Y$,
2) $\Sigma \models X \twoheadrightarrow Y$ if and only if $\Sigma[\text{FM}] \models X \twoheadrightarrow Y$,

| $XS'$ | $S - XS'$ |
|---|---|
| $0 \cdots 0$ | ni $\cdots$ ni |
| $0 \cdots 0$ | ni $\cdots$ ni |

Table III
THE PARTIAL BAG $\mathfrak{b}_X$

3) $\Sigma \models u(X)$ if and only if $\Sigma$[FM] $\models X \to S$ and there is some $u(Z) \in \Sigma$ such that $Z \subseteq XS'$.

*Proof:* The *if* directions of 1) and 2) follows straight from the soundness of the *FD implication rule* $\mathcal{I}_{\text{UF}}$. For the *only if* direction of 2), for example, assume that there is a partial bag $\mathfrak{b}$ over $\mathcal{S}$ such that $\models_{\mathfrak{b}} \Sigma$[FM] and $\not\models_{\mathfrak{b}} X \twoheadrightarrow Y$. Then it follows from a result in [26] that there are two tuples $t, t' \in \mathfrak{b}$ such that $\models_{\{t,t'\}} \Sigma$[FM] and $\not\models_{\{t,t'\}} X \twoheadrightarrow Y$. Consequently, $t(X) = t'(X)$, and $t, t'$ are $X$-total, and for some $A \in S - X$, $t(A) \neq t'(A)$. Suppose there is some $u(Z) \in \Sigma$ such that $\not\models_{\{t,t'\}} u(Z)$. Then $t, t'$ are $Z$-total and $t(Z) = t'(Z)$. However, $\models_{\{t,t'\}} Z \to S$ and thus $t(S) = t'(S)$, a contradiction. Consequently, $\models_{\{t,t'\}} \Sigma$ and $\not\models_{\{t,t'\}} X \twoheadrightarrow Y$. The remaining direction of 1) is similar.

It remains to show 3). Suppose that $\Sigma$[FM] $\models X \to S$ and there is some $u(Z) \in \Sigma$ such that $Z \subseteq XS'$. Then it follows that $X \to Z$ is implied by $\Sigma$[FM] due to the soundness of the *decomposition rule* $\mathcal{D}_{\text{F}}$. The soundness of the *null pullback rule* $\mathcal{P}_{\text{UF}}$ allows us to derive the fact that $u(X)$ is implied by $\Sigma$. It remains to show the *only if* direction of 3).

From $\Sigma \models u(X)$ we conclude $\Sigma \models X \to S$ by soundness of the *FD implication rule* $\mathcal{I}_{\text{UF}}$. According to 1) it follows that $\Sigma$[FM] $\models X \to S$. It remains to show that there is some $u(Z) \in \Sigma$ such that $Z \subseteq XS'$. Assume to the contrary that for all $u(Z) \in \Sigma$ we have $Z \not\subseteq XS'$. Under this assumption we will derive the contradiction that $\Sigma \not\models u(X)$ by constructing a two-tuple partial bag $\mathfrak{b}_X$ that satisfies $\Sigma$ and violates $u(X)$. Indeed, $\mathfrak{b}_X$ is the bag in Table III. Clearly, it satisfies $\Sigma$[FM]. Moreover, under our current assumption it is true that for every $u(Z) \in \Sigma$ we have $Z \cap (S - XS') \neq \emptyset$. Thus, $\models_{\mathfrak{b}_X} u(Z)$ for all $u(Z) \in \Sigma$. It also follows that $\not\models_{\mathfrak{b}_X} u(X)$. Hence, $\Sigma \not\models u(X)$, a contradiction. Consequently, our assumption must have been wrong and there is some $u(Z) \in \Sigma$ such that $Z \subseteq XS'$. ■

*Theorem 1:* The set $\mathfrak{p}\mathfrak{B}$ of inference rules forms a finite axiomatization for the implication of UCs, FDs and MVDs over partial bags.

*Proof:* The soundness of $\mathfrak{p}\mathfrak{B}$ was shown in Lemma 1. We establish the completeness of $\mathfrak{p}\mathfrak{B}$ by showing that for an arbitrary partial bag schema $\mathcal{S} = (S, \mathit{nfs}(S))$, and an arbitrary set $\Sigma \cup \{\varphi\}$ of UCs, FDs and MVDs over $\mathcal{S}$ the following holds: if $\Sigma \models \varphi$, then $\Sigma \vdash_{\mathfrak{p}\mathfrak{B}} \varphi$. We consider two cases. In case (1) $\varphi$ denotes the FD $X \to Y$ or the MVD $X \twoheadrightarrow Y$. Then we know by Lemma 2 that $\Sigma$[FM] $\models \varphi$ holds. From the completeness of $\mathfrak{p}\mathfrak{R}$ for the implication of FDs and MVDs over partial relations we conclude that $\Sigma$[FM] $\vdash_{\mathfrak{B}} \varphi$. Since $\mathfrak{p}\mathfrak{R} \subseteq \mathfrak{p}\mathfrak{B}$ holds we know that $\Sigma$[FM] $\vdash_{\mathfrak{p}\mathfrak{B}} \varphi$ holds, too. The *FD implication rule* $\mathcal{I}_{\text{UF}}$ shows for all $\sigma \in \Sigma$[FM] that $\Sigma \vdash_{\mathfrak{p}\mathfrak{B}} \sigma$ holds. Consequently, we have $\Sigma \vdash_{\mathfrak{p}\mathfrak{B}} \varphi$. This concludes case (1). In case (2) $\varphi$ denotes the UC $u(X)$. From $\Sigma \models u(X)$ we conclude by Lemma 2 that there is some $u(Z) \in \Sigma$ such that $Z \subseteq XS'$ holds. We also conclude from $\Sigma \models u(X)$ that $\Sigma \models X \to Z$ holds by soundness of the *FD implication rule* $\mathcal{I}_{\text{UF}}$. From case (1) it follows that $\Sigma \vdash_{\mathfrak{p}\mathfrak{B}} X \to Z$ holds. A final application of the *null pullback rule* $\mathcal{P}_{\text{UF}}$ shows that $\Sigma \vdash_{\mathfrak{p}\mathfrak{B}} \varphi$ holds. ■

$$\frac{}{XY \to Y}\ (\text{reflexivity, } \mathcal{R}_{\text{F}}) \qquad \frac{X \to YZ}{X \to Y}\ (\text{decomposition, } \mathcal{D}_{\text{F}})$$

$$\frac{X \to Y \qquad X \to Z}{X \to YZ}\ (\text{FD union, } \mathcal{U}_{\text{F}})$$

$$\frac{X \twoheadrightarrow Y}{X \twoheadrightarrow S - Y}\ (S\text{-complementation, } \mathcal{C}^S_{\text{M}}) \qquad \frac{X \twoheadrightarrow Y \qquad X \twoheadrightarrow Z}{X \twoheadrightarrow YZ}\ (\text{MVD union, } \mathcal{U}_{\text{M}})$$

$$\frac{X \twoheadrightarrow W \qquad Y \twoheadrightarrow Z}{X \twoheadrightarrow Z - W}\ Y \subseteq X(W \cap S')\ (\text{null pseudo-transitivity, } \mathcal{T}_{\text{M}})$$

$$\frac{u(X)}{X \to Y}\ (\text{FD implication, } \mathcal{I}_{\text{UF}}) \qquad \frac{X \to Y \quad u(Y)}{u(X)}\ Y \subseteq XS'\ (\text{null pullback, } \mathcal{P}_{\text{UF}})$$

$$\frac{X \to Y}{X \twoheadrightarrow Y}\ (\text{MVD implication, } \mathcal{I}_{\text{FM}}) \qquad \frac{X \twoheadrightarrow W \qquad Y \to Z}{X \to Z - W}\ Y \subseteq X(W \cap S')\ (\text{null mixed pseudo-transitivity, } \mathcal{T}_{\text{FM}})$$

Table IV
AXIOMATIZATION $\mathfrak{p}\mathfrak{B}$ OVER PARTIAL BAG SCHEMA $(S, S')$

### B. Bags

The set $\mathfrak{B}$ of inference rules in Table V forms a finite axiomatization for the combined class of UCs, FDs, and MVDs over bag schemata. The proofs necessary to establish this axiomatization are similar to those we have just described in detail for the case of partial bags. Indeed, the main argument utilizes the fact that $\mathfrak{R}$ forms an axiomatization for FDs and MVDs over relations, and the restriction of Lemma 2 to bag schemata. We omit the details. Finally, we would like to emphasize the uniformity in generalizing the axiomatization from partial relations to partial bags, and relations to bags. It suffices to add the *FD implication rule* $\mathcal{I}_{\text{UF}}$ and, in case of bags, the *pullback rule* $\mathcal{P}'_{\text{UF}}$, and, in case of partial bags the *null pullback rule* $\mathcal{P}_{\text{UF}}$, respectively.

$$\frac{}{XY \to Y}\ (\text{reflexivity, } \mathcal{R}_{\text{F}}) \qquad \frac{X \to Y}{XU \to YV}\ V \subseteq U\ (\text{FD augmentation, } \mathcal{A}_{\text{F}})$$

$$\frac{X \to Y \qquad Y \to Z}{X \to Z}\ (\text{transitivity, } \mathcal{T}'_{\text{F}})$$

---

$$\frac{X \twoheadrightarrow Y}{X \twoheadrightarrow S - Y}\ (S\text{-complementation, } \mathcal{C}^S_{\text{M}}) \qquad \frac{X \twoheadrightarrow Y}{XU \twoheadrightarrow YV}\ V \subseteq U\ (\text{MVD augmentation, } \mathcal{A}_{\text{M}})$$

$$\frac{X \twoheadrightarrow Y \qquad Y \twoheadrightarrow Z}{X \twoheadrightarrow Z - Y}\ (\text{pseudo-transitivity, } \mathcal{T}'_{\text{M}})$$

---

$$\frac{u(X)}{X \to Y}\ (\text{FD implication, } \mathcal{I}_{\text{UF}}) \qquad \frac{X \to Y \quad u(Y)}{u(X)}\ (\text{pullback, } \mathcal{P}'_{\text{UF}})$$

$$\frac{X \to Y}{X \twoheadrightarrow Y}\ (\text{MVD implication, } \mathcal{I}_{\text{FM}}) \qquad \frac{X \twoheadrightarrow Y \qquad Y \to Z}{X \to Z - Y}\ (\text{mixed pseudo-transitivity, } \mathcal{T}'_{\text{FM}})$$

Table V
AXIOMATIZATION $\mathfrak{B}$ OVER BAG SCHEMA $S$

## VI. SHARP UPPER BOUNDS FOR THE TIME COMPLEXITY

Lemma 2 also establishes an algorithmic characterization of the associated implication problems. In fact, it suffices to compute the *attribute set closure* $X^*_{\Sigma[\text{FM}]} := \{A \in S \mid \Sigma[\text{FM}] \vdash_{\mathfrak{pR}} X \to A\}$ and the *dependency basis* $\mathit{DepB}_{\Sigma[\text{FM}]}(X)$ of $X$ with respect to $\Sigma[\text{FM}]$ [26]. In particular, $\mathit{DepB}_{\Sigma[\text{FM}]}(X)$ is the set of atoms for the Boolean algebra $(\mathit{Dep(X)}, \subseteq, \cup, \cap, (\cdot)^{\mathcal{C}}_S, \emptyset, S)$ where $\mathit{Dep(X)} = \{Y \subseteq S \mid \Sigma[\text{FM}] \vdash_{\mathfrak{pR}} X \twoheadrightarrow Y\}$. The size $||\varphi||$ of $\varphi$ is the total number of attributes occurring in $\varphi$, and the size $||\Sigma||$ of $\Sigma$ is the sum of $||\sigma||$ over all elements $\sigma \in \Sigma$. For a set $\Sigma$ of FDs and MVDs let $k_\Sigma$ denote the number of MVDs in $\Sigma$, $p_\Sigma$ denote the number of sets in the dependency basis $\mathit{DepB}_\Sigma(X)$ of $X$ with respect to $\Sigma$, $\bar{p}_\Sigma$ denote the number of sets in $\mathit{DepB}_\Sigma(X)$ that have non-empty intersection with the right-hand side of $\varphi$, and $\Sigma[XS']$ denote the set of FDs and MVDs in $\Sigma$ where the left-hand side is a subset of $XS'$. The following result follows from Lemma 2 and the upper time bound established by Galil for relational databases [22].

*Theorem 2:* Let $\varphi$ denote either the UC $u(X)$, the FD $X \to Y$, or the MVD $X \twoheadrightarrow Y$ over the partial bag schema $\mathcal{S} = (S, S')$. The problem whether $\varphi$ is implied by a set $\Sigma$ of UCs, FDs and MVDs over $\mathcal{S}$ can be decided in $\mathcal{O}(||\Sigma|| + \min\{k_{\Sigma[\text{FM}][XS']}, \log \bar{p}_{\Sigma[\text{FM}][XS']}\} \times ||\Sigma[\text{FM}][XS']||)$ time.

The bound from Theorem 2 becomes $\mathcal{O}(||\Sigma|| + \min\{k_{\Sigma[\text{FM}]}, \log \bar{p}_{\Sigma[\text{FM}]}\} \times ||\Sigma[\text{FM}]||)$ time for bag schemata.

## VII. APPLICABILITY OF THEORY IN PRACTICE

As mentioned in the introduction data dependencies are essential in the design and maintenance of databases, and useful for many data processing tasks. In database practice, e.g., in SQL database systems, partial and duplicate information are allowed to occur. Partial information allows users of the database system to enter information into the database that is incomplete. The permission of duplicate information is motivated by the costs of duplicate identification and removal. The applicability of the theory of data dependencies in practice has been limited since partial and duplicate information has been largely ignored in theory. The current paper provides a summary of solutions to the implication problem for the expressive class of uniqueness constraints, functional and multivalued dependencies over all data structures arising from the permission or prohibition of partial and duplicate information. While the permission of both features covers the general case of SQL databases, many data engineers decide to specify at least a primary key on their schemata, i.e., a set of attributes that are all specified `NOT NULL`, and enforce uniqueness of tuples on their projection. Thus, duplicate tuples are not allowed to occur in database instances, which results in the usefulness of our theory over partial relations. On the other hand, data engineers may want to prohibit the occurrence of partial information. This can be done by specifying all attributes of the schema as `NOT NULL`. In this case, our theory over bags is useful. Whenever the engineers decide to prohibit partial and duplicate information by the methods above, the theory over relations can be applied.

## VIII. CONCLUSION AND FUTURE WORK

Quality database schemata must capture the structure and semantics of their application domain. Database constraints enforce the domain semantics within database systems. They are therefore invaluable for database design and data processing. Surprisingly, the existing theory of database constraints has only addressed the idealized special case where database instances are relations. In practice, e.g., SQL databases, duplicate tuples and partial information are permitted to occur. In this article, a complete theory has been established for reasoning about an expressive class of database constraints over partial relations, bags and partial bags. This closes the gap between theory and practice.

In future work it would be worthwhile to study many application areas of database constraints, including database normalization, query optimization, and data cleaning to name a few. One may also study the implication problem over different data structures, including trees and graphs, and consider other classes of data dependencies, e.g., join and inclusion dependencies. It is still an open problem whether the implication problem of MVDs can be decided in linear time. Finally, the study of Armstrong instances over different data structures may also be rewarding [33], [34], [35].

 ISBN: 978-1-61208-185-4

## REFERENCES


[1] E. F. Codd, "A relational model of data for large shared data banks," *Commun. ACM*, vol. 13, no. 6, pp. 377–387, 1970.

[2] E. Börger, E. Grädel, and Y. Gurevich, *The classical decision problem*. Heidelberg, Germany: Springer, 1997.

[3] S. Abiteboul, R. Hull, and V. Vianu, *Foundations of Databases*. Addison-Wesley, 1995.

[4] M. Arenas and L. Libkin, "An information-theoretic approach to normal forms for relational and XML data," *J. ACM*, vol. 52, no. 2, pp. 246–283, 2005.

[5] A. Deutsch, L. Popa, and V. Tannen, "Query reformulation with constraints," *SIGMOD Record*, vol. 35, no. 1, pp. 65–73, 2006.

[6] J. Biskup, *Security in computing systems*. Heidelberg, Germany: Springer, 2009.

[7] A. Klug and R. Price, "Determining view dependencies using tableaux," *ACM Trans. Database Syst.*, vol. 7, no. 3, pp. 361–380, 1982.

[8] W. Fan, F. Geerts, X. Jia, and A. Kementsietsidis, "Conditional functional dependencies for capturing data inconsistencies," *ACM Trans. Database Syst.*, vol. 33, no. 2, 2008.

[9] A. Cali, D. Calvanese, G. De Giacomo, and M. Lenzerini, "Data integration under integrity constraints," *Inf. Syst.*, vol. 29, no. 2, pp. 147–163, 2004.

[10] R. Fagin, P. Kolaitis, R. Miller, and L. Popa, "Data exchange: semantics and query answering," *Theor. Comput. Sci.*, vol. 336, no. 1, pp. 89–124, 2005.

[11] C. Delobel and M. Adiba, *Relational database systems*. North Holland, 1985.

[12] M. Wu, "The practical need for fourth normal form," in *ACM SIGCSE Conference*, 1992, pp. 19–23.

[13] C. Date and H. Darwen, *A guide to the SQL standard*. Reading, MA, USA: Addison-Wesley Professional, 1997.

[14] P. Atzeni and N. Morfuni, "Functional dependencies and constraints on null values in database relations," *Information and Control*, vol. 70, no. 1, pp. 1–31, 1986.

[15] E. Lien, "On the equivalence of database models," *J. ACM*, vol. 29, no. 2, pp. 333–362, 1982.

[16] C. Zaniolo, "Database relations with null values," *J. Comput. Syst. Sci.*, vol. 28, no. 1, pp. 142–166, 1984.

[17] S. Y. Petrov, "Finite axiomatization of languages for representation of system properties: Axiomatization of dependencies," *Information Sciences*, vol. 47, pp. 339–372, 1989.

[18] B. Thalheim, *Dependencies in relational databases*. Teubner, 1991.

[19] C. Beeri, R. Fagin, and J. H. Howard, "A complete axiomatization for fds and mvds in database relations," in *SIGMOD Conference*. ACM, 1977, pp. 47–61.

[20] S. Hartmann and S. Link, "On a problem of Fagin concerning multivalued dependencies in relational databases," *Theor. Comput. Sci.*, vol. 353, no. 1-3, pp. 53–62, 2006.

[21] C. Zaniolo, "Mixed transitivity for functional and multivalued dependencies in database relations," *Inf. Process. Lett.*, vol. 10, no. 1, pp. 32–34, 1980.

[22] Z. Galil, "An almost linear-time algorithm for computing a dependency basis in a relational database," *J. ACM*, vol. 29, no. 1, pp. 96–102, 1982.

[23] T. Imielinski and W. Lipski Jr., "Incomplete information in relational databases," *J. ACM*, vol. 31, no. 4, pp. 761–791, 1984.

[24] S. Link, "On the implication of multivalued dependencies in partial database relations," *Int. J. Found. Comput. Sci.*, vol. 19, no. 3, pp. 691–715, 2008.

[25] H. Köhler and S. Link, "Armstrong axioms and Boyce-Codd-Heath normal form under bag semantics," *Inf. Process. Lett.*, vol. 110, no. 16, pp. 717–724, 2010.

[26] S. Hartmann and S. Link, "When data dependencies over SQL tables meet the Logics of Paradox and $S$-3," in *Proceedings of the 29th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PoDS)*, 2010, pp. 317–326.

[27] M. Arenas and L. Libkin, "A normal form for XML documents," *ACM Trans. Database Syst.*, vol. 29, no. 1, pp. 195–232, 2004.

[28] S. Hartmann, S. Link, and K.-D. Schewe, "Weak functional dependencies in higher-order data models," in *FoIKS Conference*, 2004, pp. 134–154.

[29] S. Hartmann and S. Link, "Efficient reasoning about a robust XML key fragment," *ACM Trans. Database Syst.*, vol. 34, no. 2, 2009.

[30] ——, "Numerical constraints on XML data," *Inf. Comput.*, vol. 208, no. 5, pp. 521–544, 2010.

[31] M. Vincent, J. Liu, and C. Liu, "Strong FDs and their application to normal forms in XML," *ACM Trans. Database Syst.*, vol. 29, no. 3, pp. 445–462, 2004.

[32] R. Fagin, "Multivalued dependencies and a new normal form for relational databases," *ACM Trans. Database Syst.*, vol. 2, no. 3, pp. 262–278, 1977.

[33] ——, "Armstrong databases," IBM Research Laboratory, San Jose, California, USA, Tech. Rep. RJ3440(40926), 1982.

[34] S. Hartmann, M. Kirchberg, and S. Link, "Design by example for SQL table definitions with functional dependencies," *The VLDB Journal*, doi: 10.1007/ s00778-011-0239-5, 2012.

[35] S. Hartmann, U. Leck, and S. Link, "On Codd families of keys over incomplete relations," *The Computer Journal*, vol. 54, no. 7, pp. 1166–1180, 2011.


 ISBN: 978-1-61208-185-4 