INFORMATIQUE THÉORIQUE ET APPLICATIONS

JOACHIM BISKUP
PRATUL DUBLISH

Objects in relational database schemes with functional, inclusion, and exclusion dependencies

Informatique théorique et applications, tome 27, nº 3 (1993), p. 183-219.
<http://www.numdam.org/item?id=ITA_1993__27_3_183_0>

Informatique théorique et Applications/Theoretical Informatics and Applications
(vol. 27, n° 3, 1993, p. 183 à 219)

OBJECTS IN RELATIONAL DATABASE SCHEMES WITH FUNCTIONAL, INCLUSION, AND EXCLUSION DEPENDENCIES (*) (1)

by Joachim BISKUP (2) and Pratul DUBLISH (2)

Communicated by C. CHOFFRUT

Abstract. – *Objects are unique entities that are capable of independent existence. Objects are formally defined and characterized for relational database schemes with functional, inclusion, and exclusion dependencies. Object normal forms are developed and the decision problem for object normal forms is shown to be undecidable. The concept of an object is generalized to that of a high-order object for recognizing different views of the same set of real-world objects during the view-integration approach to database design. High-order objects are formally defined and characterized for database schemes with functional, inclusion, and exclusion dependencies. The recognition problem for high-order objects is also investigated.*

Résumé. – *Les objets sont des entités uniques qui sont capables d'existence indépendante. Les objets sont définis formellement et caractérisés pour les schémas de bases de données relationnelles avec des dépendances fonctionnelles, d'inclusion et d'exclusion. Les formes normales d'objets sont développées et le problème de la détermination des formes normales d'objets est prouvé indécidable. Le concept d'objet est généralisé à des objets d'ordre supérieur ce qui permet de reconnaître différentes vues du même ensemble d'objets du monde réel pendant la phase d'intégration des vues dans la conception de la base de données. Les objets d'ordre supérieur sont définis formellement et caractérisés pour les schémas de bases de données relationnelles avec des dépendances fonctionnelles, d'inclusion et d'exclusion. Le problème de reconnaissance pour les objets d'ordre supérieur est aussi considéré.*

1. INTRODUCTION

Database design theory aims at formally describing desirable properties of database schemes and at semi-formal methods to achieve such properties. For relational database schemes Boyce-Codd normal form (BCNF) is such

(*) Received April 1991, accepted November 1992.
(1) Supported by Deutsche Forschungsgemeinschaft under grant Bi 311/1-2.
(2) Institut für Informatik, Universität Hildesheim, Marienburger Platz 22, D-3200 Hildesheim, FRG.
email: biskup@informatik.uni-hildesheim.de.

0988-3754/93/03 183 37/$5.70/

a desirable property [15, 19]. Intuitively, BCNF expresses that all valid (non-trivial) functional dependencies are key dependencies. Although BCNF is a statement on the static structure of schemes, its motivation originates from avoiding so-called update anomalies. Several authors already studied how BCNF relates static structure and dynamic behaviour [3, 4, 12, 13, 14, 21]. In particular Biskup [4] argued that in the real world we ascribe two properties to an object:

- It is unique within the universe.
- It can emerge and exist independently of the current environment.

This notion of an object as being a unique entity, within a given universe, that is capable of independent existence was then formalized for a restricted class of schemes, namely relation schemes with functional dependencies. Furthermore given a relation scheme $\langle R, F \rangle$, where R is a sequence of attributes and F is a set of functional dependencies on R, the problem of characterizing exactly when is X, X a subsequence of R, an object was studied. For this purpose the notions of weak and strong independent existence were introduced and used to characterize weak and strong objects in terms of F. Finally weak and strong object normal forms were defined and it was shown that there was a strong connection between these normal forms and the well-known Boyce-Codd normal form.

However, inter-relational semantic constraints were not considered in [4]. Inclusion dependencies [7, 18] and exclusion dependencies [8] represent two important types of inter-relational semantic constraints. Inclusion dependencies can be used to capture the constraint that one set is a subset of another and exclusion dependencies can be used to capture disjointness of two sets. We use inclusion and exclusion dependencies to model inter-relational semantic constraints and formalize the notions of strong and weak objects for relational database schemes with functional, inclusion, and exclusion dependencies. We show that the definitions of objects given in [4] remain valid in the presence of inclusion and exclusion dependencies and characterize weak and strong objects in terms of the given functional, inclusion, and exclusion dependencies. We use these characterizations to develop weak and strong object normal forms for database schemes. We show that, in general, it is undecidable to test whether a given database scheme is in weak (resp. strong) object normal form. However, we give polynomial-time heuristics for these problems and also present a polynomial-time algorithm for checking whether a given database scheme with functional and inclusion dependencies is in weak (resp. strong) object normal form.

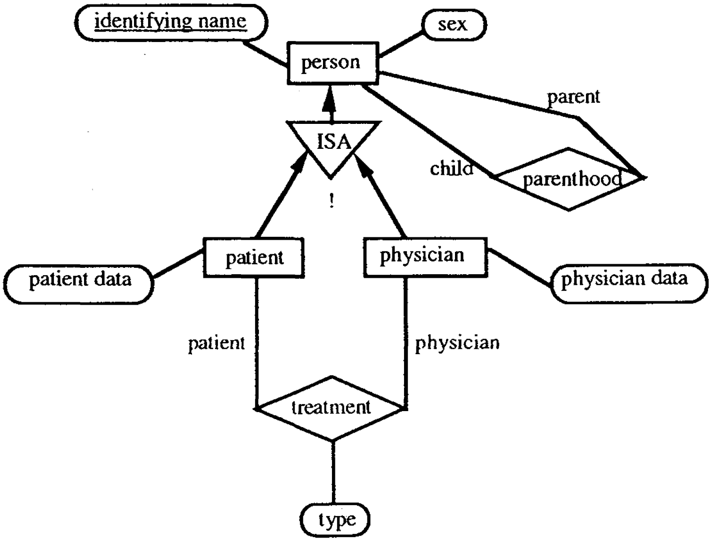
Given a relation scheme $\langle R1, F1 \rangle$, let $X1$ be a subsequence of $R1$. It is possible that, when viewed in isolation, $X1$ may not qualify as an object. However, there may be another relation scheme $\langle R2, F2 \rangle$ such that $\{X1, X2\}$, where $X2$ is a subsequence of $R2$, can be treated as an object. Intuitively, this happens when the existence of $X1$ and $X2$ objects is dependent on each other but $\{X1, X2\}$ is capable of independent existence. Such a scenario may occur during the view-integration approach to database design [2, 5, 8, 11] where $R1$ and $R2$ may represent two different views of the same set of real-world objects. During view-integration, it is desirable that different views of the same set of real-world objects be identified and replaced by a single global view. Thus, it is of interest to study and characterize such objects. We generalize the notion of an object to that of a high-order object and characterize exactly when is $\{X1, \ldots, Xk\}$, where Xi is a subsequence of Ri, $1 \leqq i \leqq k$, a high-order object. In particular, we define the notions of weak, partially-strong and strong independent existence and use them to characterize corresponding high-order objects in terms of the specified functional, inclusion, and exclusion dependencies. We do not attempt to define normal forms for high-order objects since we feel that their presence is undesirable in a well-designed database scheme. We study the recognition problem for high-order objects and show that, in general, this problem is undecidable. However, we present a polynomial-time heuristic for recognizing strong high-order objects and use it to derive a polynomial-time algorithm for recognizing strong high-order objects in database schemes with functional and inclusion dependencies.

2. SOME DEFINITIONS

We give a brief set of definitions assuming some familiarity with relational database theory [15, 18, 19]. Let U be a finite set of attributes. If X and Y are sequences of attributes, we use $X \cap Y$ to indicate the subsequence of X formed by removing those attributes that do not occur in Y, $X \supseteqq Y$ to indicate that Y is a permutation of a subsequence of X and $X - Y$ to indicate the subsequence of X formed by removing all attributes that occur in Y. A *relation scheme* $\langle R, F \rangle$ is composed of a sequence R of attributes from U such that no attribute repeats in R and a set F of *functional dependencies* (*FD s*) $X \rightarrow Y$ where $R \supseteqq X$ and $R \supseteqq Y$. A *relation* r on R is a finite set of R-tuples, *i.e.*, tuples that are defined exactly for the attributes in R. The values of tuples are elements of a countably infinite set of constants. For an R-tuple t, $t[X]$ denotes the subtuple of t defined on X only where $X \subset R$. A relation r on R is an *instance* of $\langle R, F\rangle$ if all *FDs* of F are *valid* in r, *i.e.*, for each *FD* $X \to Y$ in F and for any two tuples $t1$ and $t2$ in r, $t1[X]=t2[X] \Rightarrow t1[Y]=t2[Y]$.

Let $RS=(\langle R_1, F_1\rangle, \ldots, \langle R_n, F_n\rangle)$, where $R_i=R_j \Rightarrow i=j$, be a set of relation schemes. We do not assume any implicit inter-relational constraints through names of attributes. Hence, without loss of generality (*wlg*) we assume that for each $u \in U$ there is atmost one i, $1 \leqq i \leqq n$, such that $R_i \supseteq u$. An *inclusion dependency* (*ID*) [7, 10, 18] on *RS* is a constraint of the form $R_i[X] \supseteq R_j[Y]$ and an *exclusion dependency* (*ED*) [8] on *RS* is a constraint of the form $R_i[X] \cap R_j[Y]=\varnothing$ where $|X|=|Y|$, $R_i \supseteq X$ and $R_j \supseteq Y$. Let $r_i (r_j)$ be an instance of $\langle R_i, F_i\rangle$ $(\langle R_j, F_j\rangle)$. Then the *ID* $R_i[X] \supseteq R_j[Y]$ (resp. *ED* $R_i[X] \cap R_j[Y]=\varnothing$) is said to be *valid* if $\pi_X(r_i) \supseteq \pi_Y(r_j)$ (resp. $\pi_X(r_i) \cap \pi_Y(r_j)=\varnothing$). A *database scheme* is given by $D=(RS, I, E)$ where *RS* is a set of relation schemes, I is a set *IDs* on *RS* and E is a set of *EDs* on *RS*. A *database* $d=(r_1, \ldots, r_n)$, where r_i is an instance of $\langle R_i, F_i\rangle$, is an *instance* of D if each *ID* in I and each *ED* in E is valid in d. We denote the class of all instances of D by $\mathrm{Inst}(D)$.

We feel that *IDs* and *EDs* should primarily be used to express inter-relational constraints, *e.g.*, *IDs* can be used to express foreign keys and ISA hierarchies, and *EDs* together with *IDs* can be used to express partitioning. A simple example taken from the field of medical information systems demonstrates how to express foreign keys and an ISA hierarchy with partitioning. This example, sometimes suitably extended or modified, will be considered throughout the paper. We first express the application by an entity-relationship diagram:



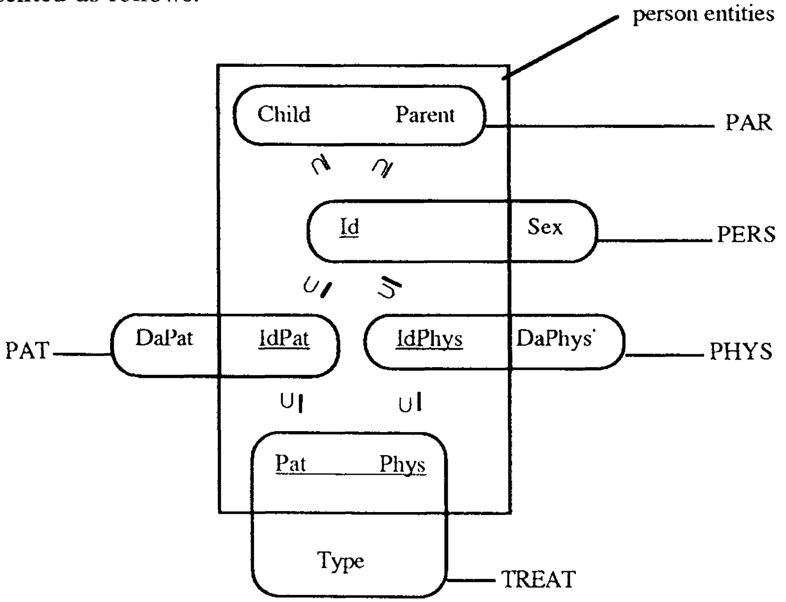
Then we define appropriate sequences of attributes in order to relationally represent the occuring entities and relationships:

PHYS = (IdPhys, DaPhys)
PAT = (IdPat, DaPat)
PERS = (Id, Sex)
PAR = (Child, Parent)
TREAT = (Pat, Phys, Type)

Finally we declare the pertinent constraints:

– *Functional dependencies:*
  IdPhys $\rightarrow$ DaPhys
  IdPat $\rightarrow$ DaPat
  Id $\rightarrow$ Sex
  Pat, Phys $\rightarrow$ Type
– *Inclusion dependencies:*
  $\left.\begin{array}{l}\text{PAT [IdPat]} \supseteq \text{TREAT [Pat]}\\ \text{PHYS [IdPhys]} \supseteq \text{TREAT [Phys]}\end{array}\right\}$ foreign keys
  $\left.\begin{array}{l}\text{PERS [Id]} \supseteq \text{PAR [Child]}\\ \text{PERS [Id]} \supseteq \text{PAR [Parent]}\end{array}\right\}$ foreign keys
  $\left.\begin{array}{l}\text{PERS [Id]} \supseteq \text{PHYS [IdPhys]}\\ \text{PERS [Id]} \supseteq \text{PAT [IdPat]}\\ \text{– } \textit{Exclusion dependencies:}\\ \text{PHYS [IdPhys]} \cap \text{PAT [IdPat]} = \varnothing\end{array}\right\}$ ISA hierarchy with partitioning

Now the resulting database scheme (without the *ED*) can be graphically represented as follows:



FDs, *IDs*, and *EDs* can interact among themselves and give rise to new dependencies [7, 8, 16]. This interaction may result in intra-relational *IDs* or *EDs*, *i.e.*, dependencies of the type $R[X] \supseteq R[Y]$ or $R[X] \cap R[Y] = \emptyset$. Sometimes, such interaction may result in *EDs* of the form $R[X] \cap R[X] = \emptyset$ [8]. Such *EDs* are called *vacuous* and their effect is to ensure that the only possible instance of R is the empty relation. Note that if R_i is involved in a vacuous *ED* then any *ED* of the form $R_j[X_j] \cap R_i[X_i] = \emptyset$, $|X_j| = |X_i|$, is always valid. Such *EDs* are called *trivial*. An *ID* is said to be *trivial* if it is of the form $R[X] \supseteq R[X]$. A trivial dependency whether it is an *ID* or an *ED* has not to be explicitly checked in verifying that a database is an instance. In the following we assume that I contains only non-trivial inter-relational dependencies and E contains only non-vacuous inter-relational dependencies.

Let $F = F_1 \cup F_2 \ldots \cup F_n$ then $(F \cup I \cup E)^+$ denotes the set of all *FDs*, *IDs*, and *EDs* implied by $F \cup I \cup E$ and $(F \cup I \cup E)_i^+$ denotes the set of all *FDs*, *IDs*, and *EDs* over R_i implied by $F \cup I \cup E$. Given a relation scheme $\langle R_i, F_i \rangle$, F_i^+ is the set of all *FDs* over R_i that are implied by F_i. An attribute sequence X_i *is said to be a key* of R_i *wrt* D if the *FD* $X_i \rightarrow R_i \in (F \cup I \cup E)_i^+$. Similarly, X_i is said to be a *key* of $\langle R_i, F_i \rangle$ if the *FD* $X_i \rightarrow R_i \in F_i^+$. A key X_i is said to be *minimal* if no proper subsequence of X_i is a key.

3. OBJECTS

DEFINITION (*objects*) [4]: An attribute sequence X is a *strong object wrt* $D = (\langle R_1, F_1 \rangle, \ldots, \langle R_n, F_n \rangle, I, E)$ if there exists an i with $R_i \supseteq X$ such that the following properties hold:

O 1. (*uniqueness*)

X is a key of R_i *wrt* D.

O 2. (*strong independent existence*)

For any instance $d = (r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$ and for *any* R_i-tuple t, where $t[X] \notin \pi_X(r_i)$, t can be inserted into r_i such that the resulting database $d^+ = (r_1, \ldots, r_i \cup \{t\}, \ldots, r_n) \in \mathrm{Inst}(D)$.

X is said to be a *weak object wrt* D if O 1 and the following property hold:

O 2*. (*weak independent existence*)

For any instance $d = (r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$ and for any X-tuple $x \notin \pi_X(r_i)$ there exists *an appropriate* R_i-tuple t, such that $t[X] = x$, which can be inserted into r_i to produce

$$d^+ = (r_1, \ldots, r_i \cup \{t\}, \ldots, r_n) \in \mathrm{Inst}(D).$$

In our medical example obviously the attribute sequence consisting of the single attribute Id is a strong object: Id is a key of PERS, and whenever an identification i has not been used before, *i.e.* $i \notin \pi_{\mathrm{Id}}(\mathrm{PERS})$, then either sex s can be chosen to successfully insert the new tuple (i, s) into the PERS relation.

However if we modify the relation scheme PERS by replacing attribute Id by attributes PeNo, Name, Birth (for person number, (full) name, birthdate and birthplace, respectively) then we could consider both PeNo and the sequence Name, Birth as keys. In that case, assuming that the other schemes and dependencies are also modified accordingly, the attribute sequences PeNo and Name, Birth respectively are only weak objects. For instance if we want to insert a new pair of values (n, b) for attributes Name and Birth then we must *appropriately* chose a person number that has not been used before.

Now it is clear that the above definitions capture the uniqueness property of an object. Then scheme R_i serves as some kind of *surrogate relation*, the values of which may be referenced in other relations that typically express relationships with other objects or properties of specific subsets of objects (as described by an ISA hierarchy).

However, independent existence of an object has two facets: independent insertion and independent survival. Independent insertion – as directly expressed in O 2 – requires that any new X-value $t[X]$, interpreted as a surrogate for a real world entity, can be inserted into the surrogate relation R_i where we can freely choose values for the additional attributes $R_i - X$, usually interpreted as some properties of the real world entity. In the weak version O 2*, however, only the existence of *appropriate* values is guaranteed.

Independent survival – not explicitly mentioned in O 2 – means that an object tuple should not get deleted due to the deletion of *any other* tuple. Formally, given $d = (r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$ and $t \in r_i$ then the deletion of t from r_i should result in an instance $d' = (r'_1, \ldots, r'_i, \ldots, r'_n) \in \mathrm{Inst}(D)$ such that the following conditions hold:

(i) $t \notin r'_i$.

(ii) $r_j \supseteq r'_j$, $1 \leq j \leq n$.

(iii) If some X_j, $R_j \supseteq X_j$ and $1 \leq j \neq i \leq n$, is an object (strong/weak) then $r_j = r'_j$. If some X_i, $R_i \supseteq X_i$, is an object then $r'_i = r_i - \{t\}$.

(iv) d' is maximal *wrt* the conditions (i)-(iii).

We claim that our definitions also capture the independent survival property. To see this, let $d=(r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$ and let $d^- = (r_1, \ldots, r_i - \{t\}, \ldots, r_n)$ where $t \in r_i$. Note that the deletion of t cannot lead to the violation of any *FD* or *ED* in $(F \cup I \cup E)^+$. However, if $R_i[X] \supseteq R_j[Y]$ is a non-trivial *ID* in $(F \cup I \cup E)^+$ then this *ID* may not hold in d^-. To ensure that this *ID* holds, we may have to recursively delete some tuple(*s*) from r_j (namely if now $t[X]$ is no longer an element of $\pi_X(r_i)$ then we have to delete all $t' \in r_j$ such that $t[X] = t'[Y]$) and this may in turn trigger off additional deletions. However, note that deletions are only required for those relations that occur on the right-hand side of some non-trivial *ID* in $(F \cup I \cup E)^+$. In what follows, we show that if X, $R \supseteq X$, satisfies O 1 and O 2 (resp. O 2*) then R cannot occur on the right-hand side of any non-trivial *ID* in $(F \cup I \cup E)^+$. Thus, our definitions also capture the independent survival property of an object.

We now attempt to characterize objects in terms of the dependencies specified for D. Let's first consider the effects of *IDs* and *EDs* on the defintion of an object. Let X be a subsequence of R_i such that R_i appears on the right-hand side (RHS) of a non-trivial *ID* $R_j[Z] \supseteq R_i[Y]$ in $(F \cup I \cup E)^+$. Consider a database instance in which all relations are empty. Clearly, this instance is in $\mathrm{Inst}(D)$. Given this instance, the *ID* $R_j[Z] \supseteq R_i[Y]$ prevents us from inserting any tuple in r_i. Hence X is not an object *wrt* D since the properties O 2 and O 2* are violated. Now consider the case when $(F \cup I \cup \mathrm{E})^+$ contains a vacuous *ED* $R_i[X] \cap R_i[X] = \varnothing$. This *ED* ensures that r_i must always be empty. Thus, no subsequence of R_i can be an object. It follows from the above discussion that an object X must satisfy the following properties:

A 1. R_i, $R_i \supseteq X$, does not occur on the RHS of any non-trivial *ID* in $(F \cup I \cup E)^+$.

A 2. $(F \cup I \cup E)^+$ contains no vacuous *EDs* involving R_i.

In our medical example property A 1 states that only subsequences of scheme PERS can act as objects because all other schemes occur on the RHS of non-trivial *IDs*.

We now consider the problem of deciding if these conditions hold. Since the inference problem for *FDs* and *IDs* is known to be undecidable [9, 16], it appears that there is no algorithm for testing these conditions. In what follows, we show that this is not the case. Note that if R_i does not occur on the RHS of any non-trivial *ID* in $(F \cup I \cup E)^+$ then R_i does not occur on the RHS of any *ID* in I (recall that I does not contain any trivial *ID*). Using this we show that in A 1 we can replace $(F \cup I \cup E)^+$ by I and that A 2 is redundant.

LEMMA 3.1: *Let D be a database scheme and let $\langle R_i, F_i \rangle$ be a relation scheme in D such that R_i does not occur on the* RHS *of any ID in I. Then R_i does not occur on the* RHS *of any non-trivial ID in $(F \cup I \cup E)^+$.*

Proof: The proof is by contradiction. Let *id* be a non-trivial *ID* in $(F \cup I \cup E)^+$ which has R_i on its RHS. Let $d=(r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$. Let t be an R_i-tuple constructed by using distinct constants that do not occur in d. Since no tuple in r_i matches with t on any attribute of R_i, $r_i \cup \{t\}$ is an instance of $\langle R_i, F_i \rangle$. Let $d^+=(r_1, \ldots, r_i \cup \{t\}, \ldots, r_n)$. It follows from the construction of t that each *ED* in E is valid in d^+. Similarly, each *ID* in I is valid in d^+ since R_i does not occur on the RHS of any *ID* in I. However, the *ID id* is not valid in d^+ because the constants occuring in t do not occur in any tuple in d. Thus, *id* is not in $(F \cup I \cup E)^+$ which is a contradiction. ■

COROLLARY 3.1: *Let D be a database scheme and let $\langle R_i, F_i \rangle$ be a relation scheme in D such that R_i does not occur on the* RHS *of any ID in I. Then R_i does not occur in any vacuous ED in $(F \cup I \cup E)^+$.*

Proof: The proof of Lemma 3.1 shows that it is always possible to produce a valid instance by inserting a tuple in r_i. Hence R_i cannot occur in any vacuous *ED* in $(F \cup I \cup E)^+$. ■

It turns out that if R_i does not occur on the RHS of any *ID* in I then the dependencies in I and E do not induce any additional *FDs* on R_i, *i.e.*, each *FD* in $(F \cup I \cup E)_i^+$ is also in F_i^+. This property in very useful since F_i^+ is computable whereas $(F \cup I \cup E)_i^+$ is not [9, 16]. In particular, it makes the problem of testing for property O 1 algorithmically tractable.

LEMMA 3.2: *Given a database scheme D, let $\langle R_i, F_i \rangle$ be a relation scheme such that R_i does not appear on the* RHS *of any ID in I. Then F_i^+ is the same as the set of FDs in $(F \cup I \cup E)_i^+$.*

Proof: It suffices to show that each *FD* $X \rightarrow Y$ which is not in F_i^+ is also not in $(F \cup I \cup E)_i^+$. Let $\mathrm{RHS}(X)=\{Z : Z$ is a single attribute and $X \rightarrow Z \in F_i^+\}$. Let X^+ be a sequence formed by using each member of $\mathrm{RHS}(X)$ exactly once. It follows from the proof of Lemma 3.1 and Corollary 3.1 that the instance d in which $r_i=\{(11\ldots1)\}$ and all other relations are empty is in $\mathrm{Inst}(D)$. Let t be an R_i-tuple such that $t[X^+]=11\ldots1$ and $t[R_i-X^+]=00\ldots0$. Consider the relation $r_i \cup \{t\}$. Since $X \rightarrow Y \notin F_i^+$, Y is not a subsequence of X^+. Thus $X \rightarrow Y$ is not valid in $r_i \cup \{t\}$. It can be shown that each *FD* in F_i is valid in $r_i \cup \{t\}$ (*see* the proof of Theorem 7.1 in [19], p. 219-220). Since R_i does not occur on the RHS of any *ID* in I, all *IDs* in I are valid in d^+. Since $r_i \cup \{t\}$ is the only non-empty relation in d^+

and E only contains non-vacuous inter-relational dependencies if follows that each ED in E is valid in d^+. Hence $d^+ \in \mathrm{Inst}(D)$ but the $FD\, X \to Y$ is not valid in r_i. Thus, $X \to Y$ is not in $(F \cup I \cup E)^+$. ∎

With this background, we are ready to characterize weak and strong objects in terms of the dependency structure of D. We consider six conditions for X, $R_i \supseteq X$:

C 1. (*not referencing*)

R_i does not occur on the RHS of any ID in I.

C 2. (*not directly affected by partitioning*)

If $R_i[Z]$ occurs in a non-trivial ED in $(F \cup I \cup E)^+$ then Z is not a subsequence of X.

C 3. (*minimal key*)

X is a minimal key of $\langle R_i, F_i \rangle$, *i.e.*, $X \to R_i \in F_i^+$ and $Y \to R_i \notin F_i^+$ for all $Y \subset X$.

C 4. (*Boyce-Codd normal form*)

$\langle R_i, F_i \rangle$ is in Boyce-Codd normal formal (BCNF), *i.e.*, if $Z \to A \in F_i^+$ and A is not a subsequence of Z then $Z \to R_i \in F_i^+$ for all $Z \subset R_i$ and $A \subset R_i$.

C 5. (*unique minimal key*)

X is the unique minimal key of $\langle R_i, F_i \rangle$.

C 6. (*not affected by partitioning*)

R_i does not occur in any non-trivial ED in $(F \cup I \cup E)^+$.

Conditions C 3, C 4, C 5 deal with functional dependencies where C 5 strengthens C 3. Condition C 1 deals with inclusion dependencies. And conditions C 2 and C 6 deal with exclusion dependencies where C 6 strengthens C 2.

Note that in general we would have to provide a more complicated definition of BCNF in terms of FDs in $(F \cup I \cup E)_i^+$ rather than F_i^+ only. Since, however, we will consider BCNF only for schemes $\langle R_i, F_i \rangle$ such that R_i does not appear on the RHS of any ID in I Lemma 3.2 allows to use the presented version.

We first show that the conditions C 1-C 3 are basic in the sense that for each kind of object they necessarily hold.

LEMMA 3.3: *If X, $R_i \supseteq X$, is a weak object then* C 1, C 2 *and* C 3 *hold.*

Proof: The discussion preceding the postulation of A 1 and A 2 shows that if X is a weak object then C 1 must hold. We now use contradiction to prove that C 2 also holds. Let $X \supseteq Z$ and let $R_i[Z] \cap R_j[W] = \varnothing$ be a non-trivial

ED in $(F \cup I \cup E)^+$. Since this ED is non-trivial, $(F \cup I \cup E)^+$ contains no vacuous ED involving R_j and thus we may find an instance with non empty r_j. Arguing along the lines of the proof of Lemma 3.1 there actually exists $d=(r_1, \ldots, r_i, \ldots, r_j, \ldots, r_n) \in \mathrm{Inst}(D)$ such that r_j is not empty. Let $w \in \pi_W(r_j)$ and let x be an X-tuple such that $x[Z]=w$. If $x \in \pi_X(r_i)$ then we derive the necessary contradiction since the $ED\, R_i[Z] \cap R_j[W]=\varnothing$ is not valid in d. If $x \notin \pi_X(r_i)$ then it follows from O2* that there exists an R_i-tuple t, $t[X]=x$, such that $d^+=(r_1, \ldots, r_i \cup \{t\}, \ldots, r_n) \in \mathrm{Inst}(D)$. Since $R_i[Z] \cap R_j[W]=\varnothing$ is not valid in d^+ we derive the contradiction that this ED is not in $(F \cup I \cup E)^+$. Finally we show that C3 also holds. Since we have proved that C1 holds, there is an instance $d=(r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$ such that $r_i=\{u\}$ and all the other relations are empty. Let $Y \subset X$. We choose a suitable R_i-tuple t such that $u[Y]=t[Y]$ but $u[X-Y] \neq t[X-Y]$. Since $t[X] \notin \pi_X(r_i)$, it follows from O2* that $d^+=(r_1, \ldots, r_i \cup \{t\}, \ldots, r_n) \in \mathrm{Inst}(D)$. However, the FD $Y \rightarrow R_i$ is not valid in d^+ and thus $Y \rightarrow R_i \notin \mathrm{F}_i^+$. ■

The basic conditions C1 (not referencing), C2 (not directly affected by partitioning), and C3 (minimal key) are not sufficient for being an object. We get a sufficient condition for a weak object, however, if we add condition C4 (Boyce-Codd normal form). But an example in [4] demonstrates that condition C4 in turn is not necessary.

LEMMA 3.4: *If conditions* C1-C4 *hold then X is a weak object.*

Proof: Let $d=(r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$ where r_i is non-empty (Corollary 3.1). Let $x \notin \pi_X(r_i)$ be an X-tuple. We construct an R_i-tuple t such that $t[\mathrm{X}]=x$ and the remaining attributes of t contain distinct constants that do not occur in d. Let $d^+=(r_1, \ldots, r_i \cup \{t\}, \ldots, r_n)$. We show that $d^+ \in \mathrm{Inst}(D)$ and hence X is a weak object. Note that each ID in $(F \cup I \cup E)^+$ is valid in d^+ since C1 holds. We claim that each ED in $(F \cup I \cup E)^+$ is also valid in d^+. To see this, assume that d^+ violates some ED ed in $(F \cup I \cup E)^+$. Since $d \in \mathrm{Inst}(D)$ and d^+ is obtained from d by only inserting t into r_i, it follows that R_i must occur in ed. Let ed be $R_i[Y] \cap R_j[W]=\varnothing$. Since $d \in \mathrm{Inst}(D)$ and $t[R_i-X]$ contains values that do not occur in d, it follows that $X \supseteq Y$. But this is not possible since C2 is assumed to hold.

Finally we show that no FDs are violated in d^+. Due to Lemma 3.2, it suffices to only consider the FDs in F_i^+. Consider $Z \rightarrow A \in F_i^+$, A is not a subsequence of Z, and assume that t and some tuple $u \in r_i$ violate this FD. Then $u[Z]=t[Z]$ and, it follows from the construction of t, $X \supseteq Z$. Since $t[X] \notin \pi_X(r_i)$, $Z \subset X$. It now follows from condition C3 that $Z \rightarrow R_i \notin F_i^+$.

However, the BCNF-condition C 4 implies that $Z \to R_i \in F_i^+$. Thus we derive a contradiction. ■

For strong objects the situation is less complicated. For we can give a set of necessary and sufficient conditions for X to be a strong object. This set comprises the basic condition C 1 (not referencing) dealing with inclusion dependencies, the strengthened condition C 5 (unique minimal key) and condition C 4 (Boyce-Codd normal form) dealing with functional dependencies, and the strengthened condition C 6 (not affected by partitioning) dealing with exclusion dependencies.

LEMMA 3.5: *X is a strong object iff* C 1, C 4, C 5 *and* C 6 *hold.*

Proof: If: Let $d=(r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$. Let t be any R_i-tuple such that $t[X] \notin \pi_X(r_i)$. We claim that $d^+=(r_1, \ldots, r_i \cup \{t\}, \ldots, r_n) \in \mathrm{Inst}(D)$. C 1 and the fact that $d \in \mathrm{Inst}(D)$ ensure that each *ID* in $(F \cup I \cup E)^+$ is valid in d^+. Since $(F \cup I \cup E)^+$ contains no vacuous *ED* involving R_i (Corollary 3.1) and C 6 holds, it follows that each *ED* in $(\mathrm{F} \cup \mathrm{I} \cup E)^+$ is valid in d^+. We now show that C 4 and C 5 ensure that each *FD* in F_i^+ is valid in $r_i \cup \{t\}$. To see this, let $Z \to A \in F_i^+$ be violated by $r_i \cup \{t\}$. Then there exists a tuple $u \in r_i$ such that u and t violate this *FD*. It follows that $t[Z]=u[Z]$. The BCNF assumption C 4 implies that $Z \to R \in F_i^+$. Now the unique minimal key assumption C 5 implies that $Z \supseteq X$. Thus $t[X]=u[X] \in \pi_X(r_i)$ which is a contradiction.

Only If: Since a strong object is also a weak object, the fact that C 1 holds follows from Lemma 3.3. To see that C 6 also holds, consider the case when $R_i[Y] \cap R_j[W]=\varnothing$ is a non-trivial *ED* in $(F \cup I \cup E)^+$. In this case, $(F \cup I \cup E)^+$ contain no vacuous *ED* on R_j. Thus there exists $d=(r_1, \ldots, r_i, \ldots, r_j, \ldots, r_n) \in \mathrm{Inst}(D)$ such that r_j is non-empty. Due to $R_i[Y] \cap R_j[W]=\varnothing$, it is not possible to insert any tuple t in r_i such that $t[Y] \in \pi_W(r_j)$. This violates property O 2 and hence we derive the contradiction that X is not a strong object.

To verify C 4, the BCNF condition, it suffices to consider the *FDs* in F_i^+ because of the simplified definition of BCNF. Let $Z \to A \in F_i^+$ where A is not a subsequence of Z. If $Z \supseteq X$ then $Z \to R_i \in F_i^+$ since X is a key of $\langle R_i, F_i \rangle$. We now show that the other case when $X-Z \neq \varnothing$ is impossible. Consider $d=(r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$ where $r_i=\{u\}$. Let v be an X-tuple where $u[X \cap Z]=v[X \cap Z]$ but $u[B] \neq v[B]$ for all attributes B in $X-Z$. Then $v \notin \pi_X(r_i)$. Select an R_i-tuple w such that $w[X]=v[X]$ and $w[Z-X]=u[Z-X]$ but $w(B) \neq u(B)$ for all attributes B in R_i-XZ. It follows from O 2 that

$d^+ = (r_1, \ldots, r_i \cup \{w\}, \ldots, r_n) \in \mathrm{Inst}(D)$. Since A is not a subsequence of Z, the tuples u and w violate the *FD* $Z \to A$ which is a contradiction.

Finally we verify C5. Assume to the contrary that there exists another minimal key Z, $Z \neq X$. Then $Z \to R_i \in F_i^+$. Since X and Z are minimal keys, $X - Z \neq \emptyset$ and $Z - X \neq \emptyset$. Consider $d = (r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$ where $r_i = \{u\}$. Let v be an R_i-tuple such that $u[X-Z] \neq v[X-Z]$ and $u[X \cap Z] = v[X \cap Z]$. Clearly $v[X] \notin \pi_X(r_i)$. Further, let $v[R_i - X] = u[R_i - X]$. It follows from O2 that $d^+ = (r_1, \ldots, r_i \cup \{v\}, \ldots, r_n) \in \mathrm{Inst}(D)$. However, since $u[Z] = v[Z]$ the tuples u and v violate the *FD* $Z \to R_i$ which is a contradiction. ■

Now we can also formally verify that in our medical example the attribute sequende Id in scheme PERS is a strong object. Firstly we observe that the declared dependencies imply the following further non-trivial dependencies:

Child $\to$ Sex

Parent $\to$ Sex

IdPhys $\to$ Sex	Phys $\to$ Sex
IdPat $\to$ Sex	Pat $\to$ Sex

TREAT [Pat] $\cap$ TREAT [Phys] $= \emptyset$

Secondly we check the conditions: PERS does not occur on the RHS of any declared *ID* (condition C1), $\langle$ PERS, Id $\to$ Sex $\rangle$ is in BCNF (condition C4) with unique minimal key Id (condition C5), and PERS does not occur in any implied non-trivial *ED* (condition C6).

In the modified example the structure of implied functional dependencies is slightly more complex since PeNo and Name, Birth are both minimal keys and thus condition C5 does not hold for PeNo and Name, Birth, respectively. However we can easily confirm that both subsequences are weak objects: condition C1 holds as before; PERS does not occur in any implied non-trivial *ED* and thus condition C2 trivially holds; both subsequences are minimal keys (condition C3) in PERS which is in BCNF (condition C4).

4. OBJECT NORMAL FORMS

Let $D = (\langle R_1, F_1 \rangle, \ldots, \langle R_n, F_n \rangle, I, E)$ be a database scheme. We assume *wlg* that each F_i only contains *FDs* of the form $L \to R$ where R is a single attribute, R does not occur in L and L is minimal. Let $X \to Y$ be an *FD* in F_i. It has been argued in [4] that by specifying this *FD* a designer intends to say that X must be an object *wrt* D, *i.e.*, X-values should be unique and should serve as surrogates for real-world objects. Note that if some F_i does not contain any *FDs* then R_i itself can be regarded as an object.

We argue that by specifying an *ID* $R_i[X] \supseteq R_j[Z]$, $i \neq j$, the designer intends to say that no subsequence of R_j can be an object since in any instance the existence of tuples in r_j depends on the existence of tuples in r_i. Note that this argument is valid only for the case where a set of real-world objects is assigned to a single relation scheme. To see this, note that the inverse *ID* $R_j[Z] \supseteq R_i[X]$ may also be specified and R_i and R_j may represent two different views of the same object. In the next section, we shall develop the notion of high-order objects to handle such cases.

Now we consider both functional and inclusion dependencies and thus the argument of [4] has to be refined as follows. Specifying an *FD* $X \to Y$ a designer intends to say that X must be an object provided X is not a subsequence of a scheme R_j that appears on the RHS of an inclusion dependency. We now define

$$\mathrm{RHS}(I) = \{ R_i : R_i \text{ occurs on the RHS of some } ID \text{ in } I \}$$

and

$$\mathrm{LHS}(F) = \{ (X, i) \text{: If } F_i \text{ is not empty then } R_i \supseteq X,\ R_i \notin \mathrm{RHS}(I) \text{ and for some } A \in R_i - X \text{ we have } X \to A \in F_i. \text{ Otherwise } X = R_i \text{ and } R_i \notin \mathrm{RHS}(I) \}.$$

D is said to be in *strong* (resp. *weak*) *object normal form* (ONF) iff for each $(X, i) \in \mathrm{LHS}(F)$, X is a strong (resp. weak) object *wrt* D, *i.e.*, X satisfies O 1 and O 2 (resp. O 2*).

In our medical example the specification of the functional dependencies makes the subsequences Id, IdPhys, IdPat, (Pat, Phys) and the scheme (Child, Parent) *candidates* of being an object. The specification of the inclusion dependencies, however, says that only Id must be an object.

More formally:

$$\mathrm{RHS}(I) = \{ \mathrm{TREAT, PAR, PHYS, PAT} \}.$$

The following set describes those (X, i) such that X is a candidate to be an object in R_i (where we identify i and R_i):

{(Id, PERS),

(IdPat, PAT),

(IdPhys, PHYS),

((Pat, Phys), TREAT),

((Child, Parent), PAR)}

Finally we have

$$\mathrm{LHS}(F) = \{(\mathrm{Id}, \mathrm{PERS})\}.$$

As we have seen above Id is an object indeed and hence the database scheme is in object normal form.

We now show that there is a strong connection between ONF and BCNF.

THEOREM 4.1: *D is in weak* ONF *iff for each $(X, i) \in \mathrm{LHS}(F)$, $\langle R_i, F_i \rangle$ is in* BCNF *and if $R_i[Z]$ occurs in a non-trivial ED in $(F \cup I \cup E)^+$ then Z is not a subsequence of X.*

Proof: If: Since $\langle R_i, F_i \rangle$ is in BCNF and according to the presuppositions of this section, it follows that if $(X, i) \in \mathrm{LHS}(F)$ then X is a minimal key of $\langle R_i, F_i \rangle$. Now, since the conditions C1-C4 are satisfied, the proof follows from Lemma 3.4.

Only If: Since X is a weak object, it follows from Lemma 3.3 that R_i satisfies C1 and C2. Let $Z \to A$ be a non-trivial *FD* in F_i^+. It follows from the inference axioms for *FDs* [15, 19] that there exists an *FD* $Y \to B$ in F_i such that $Z \supseteq Y$. Thus $(Y, i) \in \mathrm{LHS}(F)$ since C1 holds. Since D is in weak ONF, Y is a weak object. It now follows from Lemma 3.3 that $Y \to R_i \in F_i^+$. Since $Z \supseteq Y$, $Z \to R_i \in F_i^+$. Hence, it follows that $\langle R_i, F_i \rangle$ is also in BCNF. ■

THEOREM 4.2: *D is in strong* ONF *iff for each $(X, i) \in \mathrm{LHS}(F)$, $\langle R_i, F_i \rangle$ is in* BCNF, *X is the unique minimal key of $\langle R_i, F_i \rangle$ and R_i does not occur in any non-trivial ED in $(F \cup I \cup E)^+$.*

Proof: The proof follows immediately from Lemma 3.5. ■

We can easily verify that the database scheme for our medical example satisfies the stated conditions indeed. For we have seen above that (Id, PERS) is the single element of $\mathrm{LHS}(F)$, $\langle \mathrm{PERS}, \mathrm{Id} \to \mathrm{Sex} \rangle$ is in BCNF with unique key Id, and PERS does not occur in any implied non-trivial *ED*.

We now consider the computational complexity of testing if a given database scheme is in weak (resp. strong) ONF. It turns out that these problems are undecidable in general. We prove this assertion in two stages. We first show that the following problem is undecidable:

THE VACUOUS-ED PROBLEM. — *Given a database scheme* $D=(\langle R_1, F_1\rangle, \ldots, \langle R_n, F_n\rangle, I, E)$, *verify whether* $(F\cup I\cup E)^+$ *contains a vacuous ED on some* R_i, $1\leqq i\leqq n$. ■

Next, we complete the proof by reducing the vacuous-*ED* problem to the problem of testing whether a given database scheme is in weak (resp. strong) ONF. We use the following restricted version of the implication problem for *FDs* and *IDs* to show that the vacuous-*ED* problem is undecidable.

THE RESTRICTED IMPLICATION PROBLEM. — *Given a database scheme* $D=(\langle R_1, F_1\rangle, \ldots, \langle R_n, F_n\rangle, I)$, *such that* R_n *does not occur on the left-hand side (LHS) of any ID in* I, *verify whether* $(F\cup I)^+$ *contains the ID* $R_1[X_1]\supseteqq R_n[X_n]$. ■

The following lemma shows that the restricted implication problem is undecidable.

LEMMA 4.1: *The restricted implication problem is undecidable.*

Proof: We reduce the implication problem for *FDs* and *IDs*, which is known to be undecidable [9, 16], to the restricted implication problem. Let $D'=(\langle R_1, F_1\rangle, \ldots, \langle R_n, F_n\rangle, I'\rangle$ and let *id* be the *ID* $R_1[X1]\supseteqq R_n[Xn]$. We construct a new database scheme $D=(\langle R_1, F_1\rangle, \ldots, \langle R_n, F_n\rangle, \langle R, \varnothing\rangle, I\rangle)$ where R is a new relation scheme and $I=I'\cup\{R_n[Xn]\supseteqq R[Y1]\}$. Note that, by construction, R does not occur on the LHS of any *ID* in I. We now show that $(F\cup I')^+$ contains *id* iff $(F\cup I)^+$ contains the *ID* $R_1[X1]\supseteqq R[Y1]$.

If: Suppose that $(F\cup I')^+$ does not contain *id*. Then there exists a $d'=(r_1, \ldots, r_n)\in \mathrm{Inst}(D')$ such that $\pi_{X1}(r_1)\supseteqq\pi_{Xn}(r_n)$ does not hold. Let r be a relation on R such that $\pi_{Y1}(r)=\pi_{Xn}(r_n)$. Clearly, $d=(r_1, \ldots, r_n, r)\in \mathrm{Inst}(D)$. Since d violates the *ID* $R_1[X1]\supseteqq R[Y1]$, if follows that $(F\cup I)^+$ does not contain $R_1[X1]\supseteqq R[Y1]$.

Only If: Since $F\cup I\supseteqq F\cup I'$, it follows that $(F\cup I)^+$ also contains *id*. It now follows from the transitivity axiom for *IDs* that $(F\cup I)^+$ also contains $R_1[X1]\supseteqq R[Y1]$. ■

Using the above lemma, we show

LEMMA 4.2: *The vacuous-ED problem is undecidable.*

Proof: We reduce the restricted implication problem for *FDs* and *IDs* to the vacuous-*ED* problem. Let $D=(\langle R_1, F_1\rangle, \ldots, \langle R_n, F_n\rangle, I)$ such that R_n does not occur on the LHS of any *ID* in I. Let *id* be the *ID* $R_1[X]\supseteqq R_n[Y]$.

We construct a new database scheme $D' = (\langle S, \varnothing \rangle, \langle R_1, F_1 \rangle, \ldots, \langle R_n, F_n \rangle, I', E')$ where S is a new relation scheme, $I' = I \cup \{ S[W] \supseteq R_n[Y] \}$ and $E' = \{ R_1[X] \cap S[W] = \varnothing \}$. We now show that $(F \cup I)^+$ contains *id* iff $(F \cup I' \cup E')^+$ contains a vacuous *ED* on R_n.

If: Suppose that $(F \cup I)^+$ does not contain *id*. Since D has no *EDs*, there exists $d = (r_1, \ldots, r_n) \in \mathrm{Inst}(D)$ in which r_n is not empty and there exists a $y \in \pi_Y(r_n)$ such that $y \notin \pi_X(r_1)$. We delete all tuples t from r_n such that $t[Y] \neq y$ to obtain r_n'. Next, we construct an S-tuple s such that $s[W] = y$ and claim that $d' = (\{ s \}, r_1, \ldots, r_n') \in \mathrm{Inst}(D')$. To see this, note that $S[W] \supseteq R_n[Y]$ and $R_1[X] \cap S[W] = \varnothing$ are both valid in d'. Further, the deletion of tuples from r_n does not lead to the violation of any *ID* in I' since R_n does not occur on the LHS of any *ID* in I'. Since $d \in \mathrm{Inst}(D)$, it now follows that $d' \in \mathrm{Inst}(D')$. However, d' violates each vacuous *ED* on R_n and thus $(F \cup I' \cup E')^+$ does not contain a vacuous *ED* on R_n.

Only If: Since $(F \cup I' \cup E') \supseteq (F \cup I)$, it follows that if $(F \cup I)^+$ contains *id* then $(F \cup I' \cup E')^+$ also contains *id*. Since I' also contains $S[W] \supseteq R_n[Y]$, it follows that each $d' = (s, r_1, \ldots, r_n) \in \mathrm{Inst}(D')$ satisfies $\pi_X(r_1) \cap \pi_W(s) \supseteq \pi_Y(r_n)$. However, it follows from the *ED* in E' that $\pi_X(r_1) \cap \pi_W(s) = \varnothing$. Thus it follows that $(F \cup I' \cup E')^+$ contains a vacuous *ED* on R_n since r_n is always empty. ■

We now reduce the vacuous-*ED* problem to the problem of testing whether a given database scheme is in strong (resp. weak) ONF. We first give the reduction to the strong ONF problem. Consider the following instance of the vacuous-*ED* problem. Given $D = (\langle R_1, F_1 \rangle, \ldots, \langle R_n, F_n \rangle, I, E)$, we wish to know whether $(F \cup I \cup E)^+$ contains a vacuous *ED* on R_1. Note that if R_1 does not occur on the RHS of any *ID* in I then it follows from Corollary 3.1 that $(F \cup I \cup E)^+$ does not contain any vacuous *ED* on R_1. So we only need to consider the case when R_1 occurs on the RHS of some *ID* in I. To solve this problem, we construct a new relation scheme $\langle R_{n+1}, F_{n+1} \rangle$ such that R_{n+1} has the same number of attributes as R_1, $\langle R_{n+1}, F_{n+1} \rangle$ is in BCNF and it has a unique minimal key. We also construct relation schemes $\langle R_{n+i}, F_{n+i} \rangle$, $2 \leq i \leq n$, where each $\langle R_{n+i}, F_{n+i} \rangle$ is a copy of $\langle R_i, F_i \rangle$, *i.e.*, it is obtained from $\langle R_i, F_i \rangle$ by assigning new names to the attributes of R_i. Consider $D' = (\langle R_1, F_1 \rangle, \ldots, \langle R_n, F_n \rangle, \langle R_{n+1}, F_{n+1} \rangle, \ldots, \langle R_{2n}, F_{2n} \rangle, I', E')$ where

$$I' = I \cup \{ R_i \supseteq R_{n+i},\ R_{n+i} \supseteq R_i : 2 \leq i \leq n \}$$

and

$$E' = E \cup \{ R_{n+1} \cap R_1 = \varnothing \}.$$

The following lemma shows that D' is in strong ONF iff $(F \cup I \cup E)^+$ contains a vacuous ED on R_1.

LEMMA 4.3: *D' is in strong* ONF *iff $(F \cup I \cup E)^+$ contains a vacuous ED on R_1.*

Proof: Recall that R_1 occurs on the RHS of some ID in I. It now follows from the construction that each R_i and R_{n+i}, $2 \leqq i \leqq n$, occurs on the RHS of some ID in I'. Let $F' = F \cup F_{n+1} \ldots \cup F_{2n}$. Thus $\mathrm{LHS}(F') = \{ (X_{n+1}, n+1) : X_{n+1}$ is the unique minimal key of $R_{n+1} \}$.

If: Since $(F' \cup I' \cup E') \supseteq (F \cup I \cup E)$, it follows that $(F' \cup I' \cup E')^+$ also contains a vacuous ED on R_1. Thus, $R_{n+1} \cap R_1 = \varnothing$ is a trivial ED. It now follows from Theorem 4.2 that D' is in strong ONF.

Only If: Since D' is in strong ONF, it follows from Theorem 4.2 that $R_{n+1} \cap R_1 = \varnothing$ is a trivial ED. Hence $(F' \cup I' \cup E')^+$ contains a vacuous ED on R_{n+1} or a vacuous ED on R_1. The former case is impossible by Corollary 3.1 since R_{n+1} does not appear in any ID of D', and thus the latter case holds. We now claim that $(F \cup I \cup E)^+$ also contains this vacuous ED. To see this, note that given any $d = (r_1, \ldots, r_n) \in \mathrm{Inst}(D)$ we can construct $d' = (r_1, \ldots, r_n, r_{n+1}, \ldots, r_{2n})$ such that $r_1 \cap r_{n+1} = \varnothing$ and $r_i = r_{n+i}$, $2 \leqq i \leqq n$. It is obvious that $d' \in \mathrm{Inst}(D')$ and hence $r_1 = \varnothing$. Therefore, $(F \cup I \cup E)^+$ also contains a vacuous ED on R_1. ∎

The following modification of the above reduction works for the weak ONF problem. Choose $R_{n+1} = XW$ and $F_{n+1} = \{ X \to W \}$ where the number of attributes in X is the same as that in R_1. Further, set $E' = E \cup \{ R_{n+1}[X] \cap R_1 = \varnothing \}$. The construction of R_{n+i}, $2 \leqq i \leqq n$, and I' remains unchanged. Then $\mathrm{LHS}(F')$ becomes $\{ (X, n+1) \}$. Now, using Theorem 4.1 instead of Theorem 4.2, it is easy to see that D' is in weak ONF iff $(F \cup I \cup E)^+$ contains a vacuous ED on R_1. Thus, we have shown

THEOREM 4.3: *Given a database scheme $D = (\langle R_1, F_1 \rangle, \ldots, \langle R_n, F_n \rangle, I, E)$, it is undecidable to test whether D is in weak (resp. strong)* ONF. ∎

In view of Theorem 4.3, it is of interest to develop heuristics for testing whether a database scheme is in weak or strong ONF. We now show that there exists a sound, but incomplete, procedure to test whether

$D=(\langle R_1, F_1\rangle, \ldots, \langle R_n, F_n\rangle, I, E)$ is in weak (resp. strong) ONF. For weak ONF, this procedure can be obtained by modifying the statement of Theorem 4.1 as follows:

THEOREM 4.4 (Modification of Theorem 4.1): *D is in weak* ONF *if for each $(X, i) \in$ LHS(F), $\langle R_i, F_i\rangle$ is in* BCNF *and if $R_i[Z]$ occurs in an ED in E then Z is not a subsequence of X.*

Proof: It suffices to show that X is a weak object *wrt* D. Let $d=(r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$ and let $x \notin \pi_X(r_i)$. Let t be an R_i-tuple such that $t[X]=x$ and t contains distinct constants, that do not occur in d, in the remaining attributes. Note that the insertion of t into r_i does not violate any *EDs* in E since if $R_i[Z]$ occurs in an *ED* in E then Z is not a subsequence of X. Now the arguments used in the proof of Lemma 3.4 can be applied to show that X is a weak object *wrt D*. ■

Finally, a sound, but incomplete, procedure to test whether $D=(\langle R_1, F_1\rangle, \ldots, \langle R_n, F_n\rangle, I, E)$ is in strong ONF can be obtained by modifying the statement of Theorem 4.2 as follows:

THEOREM 4.5 (Modification of Theorem 4.2): *D is in strong ONF if for each $(X, i) \in$ LHS(F), $\langle R_i, F_i\rangle$ is in* BCNF, *X is the unique minimal key of R_i and R_i does not occur in any ED in E.*

Proof: Let $d=(r_1, \ldots, r_i, \ldots, r_n) \in \mathrm{Inst}(D)$. Since R_i does not occur in any *ED* in E, insertion of an arbitrary tuple in r_i will not violate any *EDs* in E. Now the arguments used in the "If" part of the proof of Lemma 3.5 can be applied to show that X is a strong object *wrt D*. ■

We now show that the conditions specified in Theorems 4.4 and 4.5 can be checked in polynomial time. It has been shown by Biskup *et al.* [6] that given a relation scheme $\langle R_i, F_i\rangle$ and $R_i \supseteq X$, it is possible to test in polynomial time whether X is the unique minimal key of $\langle R_i, F_i\rangle$. It is also possible to check in polynomial time whether a relation scheme $\langle R_i, F_i\rangle$ is in BCNF (*see* Lemma 6.1 in [17] or Theorem 13.7 in [20]). Thus, we obtain

THEOREM 4.6: *There exists a sound, but incomplete, polynomial-time procedure to test whether $D=(\langle R_1, F_1\rangle, \ldots, \langle R_n, F_n\rangle, I, E)$ is in weak (resp. strong) ONF.* ■

However, if we restrict ourselves to database schemes of the form $D=(RS, I)$, *i.e.*, with no exclusion dependencies, then both these problems are decidable in polynomial time. To see this, note that to check if D is in strong ONF it suffices to check that for each $(X, i) \in \mathrm{LHS}(F)$, $\langle R_i, F_i \rangle$ is in BCNF and X is the unique minimal key of $\langle R_i, F_i \rangle$. Similarly, to check if D is in weak ONF it suffices to verify that $\langle R_i, F_i \rangle$ is in BCNF for each $(X, i) \in \mathrm{LHS}(F)$. Since we have already shown that these conditions can be checked in polynomial time, we obtain:

THEOREM 4.7: *Given a database scheme $D = (\langle R_1, F_1 \rangle, \ldots, \langle R_n, F_n \rangle, I)$, it is possible to check in polynomial time whether D is in weak (resp. strong) ONF.* ■

5. HIGH-ORDER OBJECTS

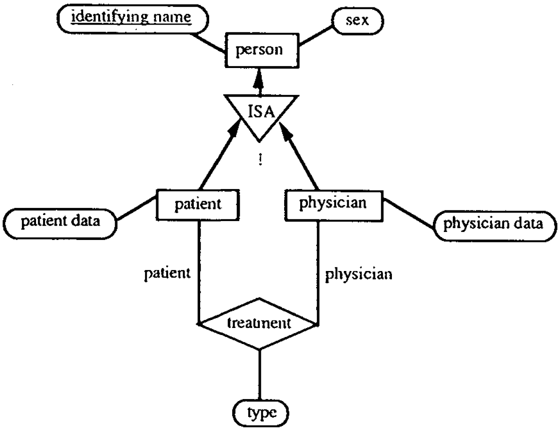
In our study of objects, we only considered the case where a set of real-world objects is assigned to a single relation scheme. This assumption makes sense when the database scheme has been designed using a global point of view. However, in the view-integration approach to database design [2, 5, 8, 11], a designer starts with several user-views and attempts to integrate them into a single global view. During the integration process, it is of interest to know whether a set of relation schemes actually represent different views of the same set of real-world objects. For example, consider a database scheme

$$D = (\langle R1, F1 \rangle, \langle R2, F2 \rangle, \{R1[X1] \supseteq R2[X2], R2[X2] \supseteq R1[X1]\})$$

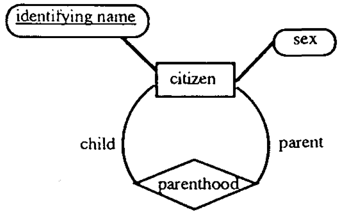
where $X1$ (resp. $X2$) is a minimal key of $R1$ (resp. $R2$) *wrt* D and $R1$ (resp. $R2$) is in BCNF. The two *IDs* ensure that $\pi_{X1}(r1) = \pi_{X2}(r2)$ where $(r1, r2) \in \mathrm{Inst}(D)$, *i.e.*, the surrogate values in $r1$ and $r2$ are always same. Note that the insertion (resp. deletion) of an $X1$-object must be accompanied by the insertion (resp. deletion) of the corresponding $X2$-object. Thus, we may regard $\{X1, X2\}$ as a high-order object. Hence, during the view integration process, we can replace D by $D' = (\langle R1 \cup R2 - X2, F1 \cup F2' \rangle)$ where $F2'$ is obtained from $F2$ by appropriately replacing the attributes of $X2$ by those of $X1$.

Coming back to our medical example our database scheme could be the result of integrating a first view dealing with treatments and a second view considering parenthoods.

treatment view:



parenthood view:



PERS = (Id, Sex)	Id $\rightarrow$ Sex	CIT = (IdCit, SexCit) IdCit $\rightarrow$ SexCit
PAT = (IdPat, DaPat)	IdPat $\rightarrow$ DaPat	PAR = (Child, Parent)
PHYS = (IdPhys, DaPhys)	IdPhys $\rightarrow$ DaPhys	
TREAT = (Pat, Phys, Type)	Pat, Phys $\rightarrow$ Type	

PAT [IdPat] $\supseteq$ TREAT [Pat]
PHYS [IdPhys] $\supseteq$ TREAT [Phys]
PERS [Id] $\supseteq$ PHYS [IdPhys]
PERS [Id] $\supseteq$ PAT [IdPat]

CIT [IdCit] $\supseteq$ PAR [Child]
CIT [IdCit] $\supseteq$ PAR [Parent]

PHYS [IdPhys] $\cap$ PAT [IdPat] = $\varnothing$

integration constraints: PERS [Id, Sex] $\supseteq$ CIT [IdCit, SexCit]
CIT [IdCit, SexCit] $\supseteq$ PERS [Id, Sex]

In what follows, we formalize the notion of a high-order object. Let $D=(\langle R1, F1\rangle, \ldots, \langle Rn, Fn\rangle, I, E)$ be a database scheme. As usual, we assume that I only contains non-trivial inter-relational dependencies and that E only contains non-vacuous inter-relational dependencies. Let $Ri \supseteq Xi$, $1 \leq i \leq k$, and $|Xi| = |Xj|$, $1 \leq i, j \leq k$. We define $\{X1, \ldots, Xk\}$ to be a *strong high-order object* if $\{X1, \ldots, Xk\}$ is a maximal set with the following properties:

O 1. (*uniqueness*).

Xi is a key of Ri *wrt* D, $1 \leq i \leq k$.

O 2. (*equality*).

Given any $d=(r_1, \ldots, r_k, \ldots, r_n) \in \text{Inst}(D)$ then

$$\pi_{X1}(r_1)=\pi_{X2}(r_2)= \ldots =\pi_{Xk}(r_k).$$

O 3. (*strong independent existence*).

Given any $d=(r_1, \ldots, r_k, \ldots, r_n) \in \text{Inst}(D)$ and *any* Rj-tuple t_j such that $t_j[Xj] \notin \pi_{Xj}(r_j)$ then for *all* Ri-tuples t_i, $1 \leqq i \neq j \leqq k$, such that $t_i[Xi]=t_j[Xj]$ the instance

$$d^+=(r_1 \cup \{t_1\}, \ldots, r_k \cup \{t_k\}, r_{k+1}, \ldots, r_n) \in \text{Inst}(D).$$

O 4. (*unit deletion*).

Given any $d=(r_1, \ldots, r_k, \ldots, r_n) \in \text{Inst}(D)$, consider any relation r_j, $1 \leqq j \leqq k$, and let $t_j \in r_j$. Let $r'_j=r_j-\{t_j\}$. Then there exists a member of $\text{Inst}(D)$ in which r'_j occurs and each such member is of the form $d^-=(r_1-\{t_1\}, \ldots, r_j-\{t_j\}, \ldots, r_k-\{t_k\}, r'_{k+1}, \ldots, r'_n) \in \text{Inst}(D)$ where $t_i \in r_i$ and $t_i(Xi]=t_j[Xj]$, $1 \leqq i \leqq k$, and $r_p \supseteqq r'_p$, $k+1 \leqq p \leqq n$.

$\{X1, \ldots, Xk\}$ is said to be a *partially-strong high-order object wrt D* if it is a maximal set for which O 1, O 2, O 4 and the following hold:

O 3*. (*partially-strong independent existence*).

Given any $d=(r_1, \ldots, r_k, \ldots, r_n) \in \text{Inst}(D)$ and *any* Rj-tuple t_j such that $t_j[Xj] \notin \pi_{Xj}(r_j)$, $1 \leqq j \leqq k$, then there exist *appropriate* Ri-tuples t_i, $1 \leqq i \neq j \leqq k$, where $t_i[Xi]=t_j[Xj]$ such that

$$d^+=(r_1 \cup \{t_1\}, \ldots, r_k \cup \{t_k\}, r_{k+1}, \ldots, r_n) \in \text{Inst}(D).$$

Obviously {Id, IdCit} is a partially-strong high-order object:

O 1. Id $\rightarrow$ Sex and IdCit $\rightarrow$ SexCit imply that Id is key of PERS and that IdCit is key of CIT.

O 2. The integration constraints ensure that $\pi_{\text{Id}}(\text{pers})=\pi_{\text{IdCit}}(\text{cit})$ for any instance $d=$(per, cit, pat, phys, treat, par).

O 3*. Let database $d=$(per, cit, pat, phys, treat, par) be an instance and assume that for tuple $t_1=(\text{Id}:i,\ \text{Sex}:s)$, $t_1[\text{Id}]=i \notin \pi_{\text{Id}}(\text{pers})$. Defining t_2 *appropriately*, namely by $t_2=(\text{IdCit}:i,\ \text{SexCit}:s)$ we have $t_1[\text{Id}]=t_2[\text{Id}]$ and $d^+=(\text{pers} \cup \{t_1\}, \text{cit} \cup \{t_2\}$, pat, phys, treat, par) is an instance. The analogous observation holds if we start with t_2.

O 4. Using the same notations as above we now assume $t_1 \in$ pers. If we delete t_1 from pers, t_2 from cit, and every tuple in the other relations containing the key value i then we get the required instance d^-. The analogous observation holds if we start with t_2.

{ Id, IdCit } is not, however, a strong high-order object. For in inserting the tuple t_1 into the relation pers we are not free to choose the SexCit component of t_2 arbitrarily, on the contrary the key dependencies Id $\rightarrow$ Sex and IdCit $\rightarrow$ SexCit together with the integration constraints require that the Sex and SexCit values associated with the key value i are identical.

Only if we dropped the attributes Sex and SexCit then { Id, IdCit } would be even a strong high-order object because then the above mentioned problem would disappear. This problem cannot be avoided, however, by considering { (Id, Sex), (IdCit, SexCit) } because in Section 3 we have already learnt that objects must be minimal keys.

$\{X1, \ldots, Xk\}$ is said to be a *weak high-order object wrt* D if it is a maximal set for which O 1, O 2, O 4 and the following hold:

O 3**. (*weak independent existence*).

Given any $d=(r_1, \ldots, r_k, \ldots, r_n) \in \text{Inst}(D)$, let $x \notin \pi_{Xi}(r_i)$, $1 \leqq i \leqq k$. Then there exist *appropriate* Ri-tuples t_i, $1 \leqq i \leqq k$, satisfying $t_i[Xi]=x$ such that

$$d^+ = (r_1 \cup \{t_1\}, \ldots, r_k \cup \{t_k\}, r_{k+1}, \ldots, r_n) \in \text{Inst}(D).$$

If we modify the relation schemes PERS and CIT by replacing attribute Id by attributes PeNo, Name, Birth and attribute IdCit by PeNoCit, NameCit, BirthCit (as already discussed in Section 2) and if we further assume that all schemes and dependencies are also modified accordingly and if we then use

$$\text{PERS}\,[\text{Name, Birth}] \supseteq \text{CIT}\,[\text{NameCit, BirthCit}]$$

$$\text{CIT}\,[\text{NameCit, BirthCit}] \supseteq \text{PERS}\,[\text{Name, Birth}]$$

as integration constraints, then { (Name, Birth), (NameCit, BirthCit) } would be a weak high-order object.

We now argue that the above definitions also capture the independent survival property, *i. e.*, if $\{X1, \ldots, Xk\}$ is a high-order object then given any $d=(r_1, \ldots, r_k, r_{k+1}, \ldots, r_n) \in \text{Inst}(D)$, the deletion of any tuple from some r_i, $k+1 \leqq i \leqq n$, should not cause the deletion of any tuple from r_j, $1 \leqq j \leqq k$. The argument is essentially the same as that for objects (Section 3). We shall show [*see* condition C 1 (*a*) below] that if $\{X1, \ldots, Xk\}$ is a high-order object then $(F \cup I \cup E)^+$, where $F=F1 \cup \ldots \cup Fn$, contains no *IDs* of the form $R[Y] \supseteq R_j[Yj]$ where $R \notin \{R1, \ldots, Rk\}$. Thus the deletion of a tuple from some r_i, $k+1 \leqq i \leqq n$, will never result in the deletion of a tuple from any r_j, $1 \leqq j \leqq k$.

In the future, we use the term *high-order object* as a generic term to denote a strong, partially-strong or weak high-order object. We now attempt to provide a precise characterization of these three notions of a high-order object. First, we once again state several conditions for $\{X1, \ldots, Xk\}$, $Ri \supseteqq Xi$, $1 \leqq i \leqq k$:

C 1. (*only self referencing*)

(*a*) $(F \cup I \cup E)^+$ does not contain any *ID* of the form $R[Y] \supseteqq Ri[Yi]$ where $R \notin \{R1, \ldots, Rk\}$.

(*b*) $(F \cup I \cup E)^+$ contains the non-trivial *IDs* $Ri[Xi] \supseteqq Rj\,[Xj]$, $1 \leqq i \neq j \leqq k$.

C 2. (*not directly involved in intra-relational IDs*)

$(F \cup I \cup E)^+$ does not contain any non-trivial *ID* of the form $Ri[X_{i1}] \supseteqq Ri[X_{i2}]$, $1 \leqq i \leqq k$, where $Xi \supseteqq X_{i2}$.

C 3. (*all referencing is connected to self referencing*)

(*a*) If $(F \cup I \cup E)^+$ contains an *ID* of the form $Ri[Yi] \supseteqq Rj[Yj]$ then it also contains the *ID* $Rj[Yj] \supseteqq Ri[Yi]$, $1 \leqq i, j \leqq k$.

(*b*) If $(F \cup I \cup E)^+$ contains an *ID* of the form $Ri[Yi] \supseteqq Rj[Yj]$, $1 \leqq i, j \leqq k$, such that $Yi \cap Xi = \varnothing$ and $Yj \cap Xj = \varnothing$ then $(F \cup I \cup E)^+$ also contains the *ID* $Ri[Xi\,Yi] \supseteqq Rj[Xj\,Yj]$.

C 4. (*insertable*)

$(F \cup I \cup E)^+$ does not contain any vacuous *ED* involving any Ri, $1 \leqq i \leqq k$.

C 5. (*not directly affected by partitioning*)

$(F \cup I \cup E)^+$ does not contain any non-trivial *ED* of the form $Ri[Yi] \cap Rp[Yp] = \varnothing$, $1 \leqq i \leqq k$, where $Xi \supseteqq Yi$.

C 6. (*minimal key*)

Xi is a minimal key of Ri *wrt* D.

C 7. (*Boyce-Codd normal form*)

Each Ri in BCNF *wrt* D, $1 \leqq i \leqq k$.

C 8. (*no intra-relational referencing*)

For each i, $1 \leqq i \leqq k$, $(F \cup I \cup E)_i^+$ does not contain any non-trivial intra-relational *IDs*.

C 9. (*not affected by partitioning*)

Each Ri, $1 \leqq i \leqq k$, is not involved in any non-trivial *ED* in $(F \cup I \cup E)^+$.

C 10. (*unique minimal key*)

Each Xi is the unique minimal key of Ri *wrt* D, $1 \leqq i \leqq k$.

C 11. (*only key referencing*)

If $(F \cup I \cup E)^+$ contains an *ID* of the form $Ri[Yi] \supseteqq Rj[Yj]$, $1 \leqq i \neq j \leqq k$, then there exists a projection ρ and a permutation γ such that $Yi = \gamma(\rho(Xi))$ and $Xj = \gamma(\rho(Xj))$.

Conditions C 6, C 7, C 10 deal with functional dependencies where C 10 strengthens C 6. Conditions C 1, C 2, C 3, C 8, C 11 deal with inclusion dependencies where C 8 strengthens C 2 and C 11 strengthens C 3. And conditions C 4, C 5, C 9 deal with exclusion dependencies where C 9 strengthens C 5.

We first show that the conditions C 1-C 6 are basic in the sense that for each kind of high-order object they necessarily hold.

For our integrated database scheme we can verify that $\{$ Id, IdCit $\}$ satisfies these conditions:

C 1. (*a*) PERS and CIT occur on the right-hand side of non-trivial *IDs* only in the integration constraints.

(*b*) The integration constraints are just the required *IDs*.

C 2. There are no non-trivial *IDs* between PERS [Id] and PERS [Sex], respectively between CIT [IdCit] and CIT [SexCit].

C 3. (*a*) The integration constraints are stated symmetrically.

(*b*) For the implied *IDs* PERS [Sex] $\supseteq$ CIT [SexCit] and CIT [SexCity] $\supseteq$ PERS [Sex] the integration constraints are just the required *IDs*.

C 4. There are no vacuous *EDs*.

C 5. Neither PERS nor CIT occur in any implied non-trivial *ED*.

C 6. Id is minimal key of PERS, and IdCit is minimal key of CIT.

Basically the same arguments show that in the modified integrated database the schemes of $\{$(Name, Birth), (NameCit, BirthCit)$\}$ also satisfy these conditions.

THEOREM 5.1: *If* $\{Xi, \ldots, Xk\}$ *is a weak high-order object then* C 1-C 6 *must hold.*

Proof: C 1 (*a*) follows immediately by applying O 3** to the empty instance and C 1 (*b*) follows directly from O 2. C 2 also follows from O 3**. To see this, suppose that C 2 does not hold. Consider any $d=(r_1, \ldots, r_k, \ldots, r_n) \in \mathrm{Inst}(D)$. Then we can insert t_i into r_i only if $t_i[X_{i1}]=t_i[X_{i2}]$ or there exists a $t \in r_i$ such that $t[X_{i1}]=t_i[X_{i2}]$. This violates O 3** and hence it follows that C 2 must hold.

We now show that C 3 (*a*) also holds. Let $d=(r_1, \ldots, r_i, \ldots, r_j, \ldots, r_k, \ldots, r_n)$ be any instance in Inst(D). It follows from O 2 that $\pi_{Xi}(r_i)$ and $\pi_{Xj}(r_j)$ have the same number of tuples, and by O 1 so do r_i and r_j. Now, suppose that $Ri[Yi] \supseteq Rj[Yj]$ is a non-trivial *ID* which is valid in d but $Rj[Yj] \supseteq Ri[Yi]$ is not valid in d (note that $i=j$ is also possible). Thus, there exists $x \in \pi_{Yi}(r_i)$ such that $x \notin \pi_{Yj}(r_j)$. Since $|r_i| = |r_j|$, it now follows that there exists a value y, $y \in \pi_{Yi}(r_i)$ and $y \in \pi_{Yj}(r_j)$, such that $|\sigma_{Yi=y}(r_i)| < |\sigma_{Yj=y}(r_j)|$. Consider the situation when we start deleting, one by one, those tuples from r_i that are also in $\sigma_{Yi=y}(r_i)$. Since $|\sigma_{Yi=y}(r_i)| < |\sigma_{Yj=y}(r_j)|$, at some stage in this deletion the *ID* $Ri[Yi] \supseteqq Rj[Yj]$ will force us to delete *more than one* tuple from r_j. Thus, we derive the contradiction that $\{X1, \ldots, Xk\}$ is not a high-order object since O 4 does not hold.

To see that C 3 (*b*) also holds, let $d=(r_1, \ldots, r_i, \ldots, r_j, \ldots, r_k, \ldots, r_n) \in \mathrm{Inst}(D)$ such that $Ri[Yi] \supseteqq Rj[Yj]$ is a non-trivial *ID* which is valid in d but the *ID* $Ri[Xi\,Yi] \supseteqq Rj[Xj\,Yj]$ is not valid in d. Let $t_i \in r_i$ and $t_j \in r_j$ such that $t_i[Xi]=t_j[Xj]$ but $t_i[Yi] \neq t_j[Yj]$. Let $t_i[Yi]=y$. Since C 3 (*a*) holds, $Rj[Yj] \supseteqq Ri[Yi]$ is also valid in d. Hence, there exists $t'_j \in r_j$, $t'_j \neq t_j$, such that $t'_j[Xj]=y$. If $\sigma_{Yi=y}(r_i)=\{t_i\}$ then the deletion of t_i from r_i would force the deletion of both t_j and t'_j from r_j. Thus, we derive the contradiction that $\{X1, \ldots, Xk\}$ is not a high-order object since O 4 is violated. Otherwise, we first delete from r_i all tuples in $\sigma_{Yi=y}(r_i)-\{t_i\}$. If O 4 is violated during these deletions then we are done. If not, then the deletion of t_i will now violate O 4.

C 4 follows immediately by applying O 3** to the empty instance. We now use contradiction to show that C 5 holds. Suppose that $Ri[Yi] \cap Rp[Yp]=\emptyset$, $Xi \supseteqq Yi$, is a non-trivial *ED* in $(F \cup I \cup E)^+$. Since this *ED* is non-trivial, $(F \cup I \cup E)^+$ does not contain any vacuous *ED* on Rp. Thus, there exists a $d=(r_1, \ldots, r_k, r_{k+1}, \ldots, r_n) \in \mathrm{Inst}(D)$ such that r_p is not empty. Then there exists a $y \in \pi_{Yp}(r_p)$ such that $y \notin \pi_{Yi}(r_i)$. We now construct a suitable Ri-tuple t_i such that $t_i[Y_i]=y$ and $t_i[Xi] \notin \pi_{Xi}(r_i)$. It now follows from O 3** that there exist appropriate Rj-tuples t_j, $1 \leqq i \neq j \leqq k$, satisfying $t_j[Xj] = t_i[Xi]$ such that $d^+=(r_1 \cup \{t_1\}, \ldots, r_k \cup \{t_k\}, r_{k+1}, \ldots, r_n) \in \mathrm{Inst}(D)$. However, by construction, $Ri[Yi] \cap Rp[Yp]=\emptyset$ is not valid in d^+, a contradiction.

C 6 can be shown to hold by employing arguments similar to those used in the last part of the proof of Lemma 3.3. ■

We now show that $\{X1, \ldots, Xk\}$ is a weak high-order object if C 1-C 6 and additionally C 7 (Boyce-Codd normal form) hold.

Before giving the proof of this assertion, we show that C 2, in conjunction with C 1 (*b*), also rules out the existence os some more *IDs*. Given a projection ρ and a permutation γ, we say $X=\gamma(\rho(Y))$ if the sequence X can be obtained from Y by first applying the projection ρ and then applying the permutation

γ. We now show:

PROPOSITION 5.1: *If* C1 (*b*) *and* C2 *hold then* $(F \cup I \cup E)^+$ *does not contain any IDs of the form*

(*a*) $Ri[Yi] \supseteq Rj[Yj]$, $1 \leqq i \neq j \leqq k$, *where* $Xj \supseteq Yj$ *and* Yi *is* not *a subsequence of* Xi.

(*b*) $Ri[Yi] \supseteq Rj[Yj]$, $1 \leqq i \neq j \leqq k$, *where* $Xi \supseteq Yi$, $Xj \supseteq Yj$ *and there exists a projection* ρ *and a permutation* γ *such that* $Yi = \gamma(\rho(Xi))$ *and* $Yj \neq \gamma(\rho(Xj))$.

Proof: (*a*) The proof is by contradiction. Suppose that $(F \cup I \cup E)^+$ contains such an *ID*. Since $Xj \supseteq Yj$, there exist ρ and γ such that $Xj = \gamma(\rho(Xj))$. Let $Yi' = \gamma(\rho(Xi))$. It follows from C1 (*b*) that $Rj[Xj] \supseteq Ri[Xi]$ is in $(F \cup I \cup E)^+$. By using the projection and permutation axiom for *IDs* [7, 16, 18] it follows that $Rj[Yj] \supseteq Ri[Yi']$ is in $(F \cup I \cup E)^+$. Since $Ri[Yi] \supseteq Rj[Yj]$ is assumed to be in $(F \cup I \cup E)^+$, it follows from the transitivity axiom for *IDs* [7, 16, 18] that $Ri[Yi] \supseteq Ri[Yi']$ is a non-trivial *ID* in $(F \cup I \cup E)^+$. Since $Xi \supseteq Yi'$, the *ID* $Ri[Yi] \supseteq Ri[Yi']$ violates C2 which is the desired contradiction.

(*b*) The proof is again by contradiction. Suppose that $(F \cup I \cup E)^+$ contains such an *ID*. Since $Xj \supseteq Yj$ it follows that there exists a projection $\rho 1$, $\rho 1 \neq \rho$, and a permutation $\gamma 1$ such that $Yj = \gamma 1(\rho 1(Xj))$. Let $Yi' = \gamma 1(\rho 1(Xi))$. It now follows from C1 (*b*) and the projection and permutation axiom for *IDs* that $Rj[Yj] \supseteq Ri[Yi']$ is in $(F \cup I \cup E)^+$. Using the transitivity axiom we conclude that the non-trivial *ID* $Ri[Yi] \supseteq Ri[Yi']$ is in $(F \cup I \cup E)^+$. Since $Xi \supseteq Yi'$, $Ri[Yi] \supseteq Ri[Yi']$ violates C2 which is the desired contradiction ■

THEOREM 5.2: *If* C1-C7 *hold then* $\{X1, \ldots, Xk\}$ *is a weak high-order object.*

Proof: Note that C6 implies that O1 holds and C1 (*b*) implies that O2 holds. We now show that $\{X1, \ldots, Xk\}$ also satisfies O3**. Let $d = (r_1, \ldots, r_k, r_{k+1}, \ldots, r_n) \in \text{Inst}(D)$ and let $x \notin \pi_{Xi}(r_i)$, $1 \leqq i \leqq k$. We claim that there exists *Ri*-tuples t_i, $1 \leqq i \leqq k$, where $t_i[Xi] = x$ and the remaining attributes of t_i contain values that do not occur in d, such that $d_1 = (\{t_1\}, \ldots, \{t_k\}, \varnothing, \ldots, \varnothing) \in \text{Inst}(D)$. This is so for the following reasons.

Since there are no vacuous *EDs* involving any $R\,i$, $1 \leqq i \leqq k$, (condition C4) there exists $d_3 = (s_1, \ldots, s_n) \in \text{Inst}(D)$ such that $s_i \neq \varnothing$, $1 \leqq i \leqq k$. We can even choose d_3 such that the values of d_3 do not occur in d. Then also $d_2 = (s_1, \ldots, s_k, \varnothing, \ldots, \varnothing) \in \text{Inst}(D)$ since C1 (*a*) rules out any *ID* of the form $R[Y] \supseteq Ri[Yi]$ where $R \notin \{R1, \ldots, Rk\}$. Let $t_1 \in s_1$. According to C 1 (*b*) and C 6 there are uniquely determined tuples $t_2, \ldots, t_k$ such that $t_1[X1] = \ldots = t_k[Xk]$. Then $d_1 := (\{t_1\}, \ldots, \{t_k\}, \varnothing, \ldots, \varnothing) \in \mathrm{Inst}(D)$.

For consider any *ID* $Ri[Yi] \supseteq Rj[Yj]$. If $Xj \supseteq Yj$ then C 2 implies $i \neq j$ and Proposition 5.1 ensures that $Yi = \gamma(\rho(Xi))$ and $Xj = \gamma(\rho(Xj))$ for some projection ρ and some permutation γ; hence the *ID* is valid since $t_i[Xi] = t_j[Xj]$. If $Yj - Xj \neq \varnothing$ then the *ID* $Ri[Yi - Xi] \supseteq Rj[Yj - Xj]$ is implied because otherwise, using C 1 (*b*) and C 3 (*a*), we would find an intra-relatonal *ID* which contradicts C 2; then C 3 (*b*) ensures that $Ri[Xi\,Yi] \supseteq Rj[Xj\,Yj]$ is implied too; since $d_2 \in \mathrm{Inst}(D)$ and since the tuples t_i and t_j are uniquely determined within s_i and s_j, respectively, by their key values the *ID* $Ri[Xi\,Yi] \supseteq Rj[Xj\,Yj]$ and thus also the *ID* $Ri[Yi] \supseteq Rj[Yj]$ is valid in d_1.

If $t_1[X1] = \ldots = t_k[Xk] = x$ then we are done. Otherwise, we modify the value in the Xi attributes of each t_i so that $t_i[Xi] = x$ holds. We now show that even after this modification, $d_1 \in \mathrm{Inst}(D)$.

First, note that setting each $t_i[Xi]$ to x cannot lead to the violation of any *ED* since C 5 is assumed to hold. We now show that no *ID* is violated due to this modification. Since we only modify the value in the Xi-attributes of each t_i, it suffices to show that no *ID* of the form $Ri[Yi] \supseteq Rj[Yj]$, $1 \leq i, j \leq k$, such that $Xi \supseteq Yi$ or $Xj \supseteq Yj$, is violated. We first consider intra-relational *IDs*. Since C 2 rules out non-trivial *IDs* of the form $Ri[Yi] \supseteq Ri[Zi]$, $Xi \supseteq Zi$, it suffices to consider *IDs* of the $Ri[Yi] \supseteq Ri[Zi]$, $Xi \supseteq Yi$ and $Zi - Xi \neq \varnothing$. We now show that $(F \cup I \cup E)^+$ does not contain any such *ID*. To see this, note that if $(F \cup I \cup E)^+$ contains such an *ID* then it follows from C 3 (*a*) that $Ri[Zi] \supseteq Ri[Yi]$ is also in $(F \cup I \cup E)^+$. But this contradicts C 2 which rules out the presence of $Ri[Zi] \supseteq Ri[Yi]$ in $(F \cup I \cup E)^+$. Next, we consider inter-relational *IDs*. Due to C 3 (*a*), it suffices to show that each *ID* of the form $Rp[Yp] \supseteq Rq[Yq]$, $1 \leq p \neq q \leq k$, such that $Xq \supseteq Yq$ is still valid. Since $Xq \supseteq Yq$, $Yq = \gamma(\rho(Xq))$. It now follows from Proposition 5.1 that $Yp = \gamma(\rho(Xp))$. Since $t_i[X1] = \ldots = t_k[Xk]$, all such *IDs* are clearly valid in d_1.

We now claim that $d^+ = (r_1 \cup \{t_1\}, \ldots, r_k \cup \{t_k\}, r_{k+1}, \ldots, r_n) \in \mathrm{Inst}(D)$. Since Xi is a minimal key of Ri and Ri is in BCNF (C 6 and C 7), the arguments used in the proof of Lemma 3.4 can be applied to show that each *FD* in $(F \cup I \cup E)^+$ is valid in d^+. We also claim that each *ID* in $(F \cup I \cup E)^+$ is valid in d^+. To see this, recall that $d \in \mathrm{Inst}(D)$ and d^+ is obtained from d by inserting t_i in r_i, $1 \leq i \leq k$. Thus, we only need to verify that each *ID*, in which some Ri occurs on the RHS, is valid in d^+. C 1 (*a*) assures us that $(F \cup I \cup E)^+$ contains no *IDs* of the form $R[Y] \supseteqq Ri[Yi]$, $R \notin \{R1, \ldots, Rk]$. The construction of $t_1, \ldots, t_k$ ensures that each *ID* of the form $Rp[Yp] \supseteqq Rq[Yq]$, $1 \leqq p,\ q \leqq k$, is valid in d^+. Finally, we show that each *ED* in $(F \cup I \cup E)^+$ is also valid in d^+. Recall that due to C 5 no attribute of any Xi, $1 \leqq i \leqq k$, is involved in any non-trivial *ED*. Further, the construction of each t_i is such that $t_i[Ri - Xi]$ contains values that do not occur in *d*. Since $d \in \mathrm{Inst}(D)$ and $d_1 \in \mathrm{Inst}(D)$, it follows immediately that each *ED* in $(F \cup I \cup E)^+$ is valid in d^+.

Finally, we show that O 4 is also satisfied. Let $d = (r_1, \ldots, r_k, r_{k+1}, \ldots, r_n) \in \mathrm{Inst}(D)$. Consider any r_j, $1 \leqq j \leqq k$, and let $t_j \in r_j$ such that $t_j[Xj] = x$. It now follows from C 1 (*b*) that for each *i*, $1 \leqq i \leqq k$, there exists $t_i \in r_i$ such that $t_i[Xi] = x$. We now delete the tuple t_i from each r_i, $1 \leqq i \leqq k$, and claim that these deletions do not lead to the violation of any *ID* of the form $Rp[Yp] \supseteqq Rq[Yq]$, $1 \leqq p,\ q \leqq k$. To see this, first consider an intra-relational *ID* $Rp[Yp] \supseteqq Rp[Zp]$. It follows from C 2 and C 3 (*a*) that $Xp \supseteqq Zp$ or $Xp \supseteqq Yp$ is not possible. Hence, we only need to consider the case when $Yp \cap Xp = \emptyset$ and $Zp \cap Xp = \emptyset$. It now follows from C 3 (*b*) that $Rp[Xp\ Yp] \supseteqq Rp[Xp\ Zp] \in (F \cup I \cup E)^+$. Thus, no intra-relational *ID* is violated. Now consider the case of inter-relational *IDs*. Note that $\pi_{X1}(r_1 - \{t_1\}) = \ldots = \pi_{Xk}(r_k - \{t_k\})$. Thus it follows immediately from Proposition 5.1 and C 3 that no *IDs* of the form $Rp[Yp] \supseteqq Rq[Yq]$, $1 \leqq p \neq q \leqq k$, are violated.

However, the deletion of t_i from each r_i may violate *IDs* of the form $Ri[Yi] \supseteqq Rp[Y]$, $Rp \notin \{R1, \ldots, Rk\}$. Thus, in order to obtain a valid instance, we may have to remove some tuples from r_p. Since C 1 (*a*) holds, the deletion of tuples from r_p will not result in the deletion of any tuple from r_i. Since deletion of tuples cannot lead to the violation of any *FD* or *ED*, each valid instance that results due to the deletions of t_j from r_j is of the form

$$d^- = (r_1 - \{t_1\}, \ldots, r_k - \{t_k\}, r'_{k+1}, \ldots, r'_n)$$

$$\text{where} \quad r_p \supseteqq r'_p, \quad k+1 \leqq p \leqq n. \quad \blacksquare$$

Theorem 5.2 confirms that $\{$(Name, Birth), (NameCit, BirthCit)$\}$ is a weak high-order object in the modified integrated database scheme because according to the discussion above conditions C 1-C 6 are satisfied and PERS and CIT are in BCNF (condition C 7). By the same reasons $\{$Id, IdCit$\}$ is a weak high-order object in the integrated database scheme.

We now give a set of necessary and sufficient conditions for $\{X1, \ldots, Xk\}$ to be partially-strong high-order object. One can easily verify that $\{\mathrm{Id}, \mathrm{IdCit}\}$ in the integrated database scheme satisfies all these conditions but $\{(\mathrm{Name}, \mathrm{Birth}), (\mathrm{NameCit}, \mathrm{BirthCit})\}$ in the modified version fails in condition C 10 because the keys are not unique.

THEOREM 5.3: *$\{X1, \ldots, Xk\}$ is a partially-strong high-order object iff* C 1, C 3, C 4 *and* C 7-C 10 *hold.*

Proof: If: Let $d=(r_1, \ldots, r_k, \ldots, r_n)\in \mathrm{Inst}(D)$. Property O 1 follows from C 10 and property O 2 follows from C 1 (*b*). We now show that O 3* also holds. Let t_j be *any* Rj-tuple such that $t_j[Xj]\notin \pi_{Xj}(r_j)$, $1\leqq j\leqq k$. Given t_j, we claim that there exist suitable Ri-tuples t_i satisfying $t_i[Xi]=t_j[Xj]$, $1\leqq i\neq j\leqq k$, such that $d_1=(\{t_1\}, \ldots, \{t_k\}, \varnothing, \ldots, \varnothing)\in \mathrm{Inst}(D)$. We show that t_j can be chosen arbitrarily. Note that C 8 implies C 2. Hence we can still use Proposition 5.1 to claim that $t_j[Xj]$ can be chosen arbitrarily. Further, since C 8 and C 9 hold, $t_j[Rj-Xj]$ can also be arbitrarily constructed. Now, by employing the arguments used in the proof of Theorem 5.2, we can show that $d_1\in \mathrm{Inst}(D)$.

We now claim that

$$d^+=(r_1\cup\{t_1\}, \ldots, r_k\cup\{t_k\}, r_{k+1}, \ldots, r_n)\in \mathrm{Inst}(D).$$

First, we show that each *FD* in $(F\cup I\cup E)^+$ is valid in d^+. To see this, note that it follows from C 1 (*b*) that $t_i[Xi]\notin \pi_{Xi}(r_i)$, $1\leqq i\leqq k$. Further, since Ri is in BCNF and Xi is the unique minimal key of Ri (C 7 and C 10), the arguments used in the "If" part of Lemma 3.5 can be used to show that $r_i\cup\{t_i\}$ satisfies each *FD* in $(F\cup I\cup E)_i^+$. Next, we show that each *ID* in $(F\cup I\cup E)^+$ is valid in d^+. Since $d\in \mathrm{Inst}(D)$ and d^+ is obtained from d by inserting t_i into r_i, we only need to consider those *IDs* in which Ri occurs on the *RHS*. Since $d_1\in \mathrm{Inst}(D)$ and $d\in \mathrm{Inst}(D)$, it follows that each *ID* of the form $Rp[Yp]\supseteq Rq[Yq]$, $1\leqq p\neq q\leqq k$, is valid in d^+. Since C 8 says that $(F\cup I\cup E)^+$ does not contain any non-trivial intra-relational *IDs* on any Ri, $1\leqq i\leqq k$, and C 1 (*a*) holds, it follows that each *ID* in $(F\cup I\cup E)^+$ is valid in d^+. Finally, we show that each *ED* in $(F\cup I\cup E)^+$ is also valid in d^+. Since *C* 4 says that $(F\cup I\cup E)^+$ contains no vacuous *EDs* involving any Ri and C 9 says that each Ri does not occur in any non-trivial *ED* in $(F\cup I\cup E)^+$, it follows that each *ED* in $(F\cup I\cup E)^+$ is valid in d^+.

Finally, the arguments used in the proof of Theorem 5.2 can be used to show that O 4 is also satisfied.

Only If: The proof that C1, C3 and C4 hold follows from Theorem 5.1. The BCNF condition C7 and the unique minimal key condition C10 can be shown to hold by using the arguments in the "Only If" part of the proof of Lemma 3.5. To see that C8 holds, assume that $Rj[Y] \supseteq Rj[Z]$ is a non-trivial *ID* in $(F \cup I \cup E)_j^+$. Thus, we are constrained to choose t_j such that either $t_j[Y]=t_j[Z]$ or $t_j[Z] \in \pi_Y(r_j)$. In any case, O3* is violated. Similarly, it can be shown that if C9 does not hold then O3* is again violated. ■

Finally, we characterize strong high-order objects. Condition C11 now formally explains why {Id, IdCit} is not a strong high-order object. For the integration constraints imply the *IDs* PERS[Sex] $\supseteq$ CIT[SexCit] and CIT[SexCit] $\supseteq$ PERS[Sex] but obviously Sex is not contained in Id and SexCit is not contained IdCit. If we dropped the attributes Sex and SexCit, however, then condition C11 would trivially hold and hence {Id, IdCit} would become even a strong high-order object.

THEOREM 5.4: $\{X1, \ldots, Xk\}$ *is a strong high-order object iff* C1, C4 *and* C7-C11 *hold.*

Proof: If: Similar to that of the "If" part of Theorem 5.3. Since C11 also holds, we can now choose *any* tuples t_i, $1 \leq i \leq k$, for insertion into r_i provided $t_1[X1] = \ldots = t_k[Xk]$.

Only If: It follows from the "Only If" part of the proof of Theorem 5.3 that C1, C4 and C7-C10 hold. By using contradiction, we can show that C11 also holds. ■

6. TESTING FOR HIGH-ORDER OBJECTS

Recall that high-order objects are a generalization of objects. Thus, in view of Theorem 4.3, it is not surprising that, in general, it is undecidable to test if $\{X1, \ldots, Xk\}$ is a high-order object. Conditions C5 and C9, which indirectly involve testing for vacuous *EDs*, are the main stumbling blocks towards finding an algorithm for recognizing high-order objects. A straight-forward reduction from the vacuous-*ED* problem gives us the following result:

THEOREM 6.1: *Given a database scheme* $D=(\langle R1, F1\rangle, \ldots, \langle Rn, Fn\rangle, I, E)$, *it is undecidable to test if* $\{X1, \ldots, Xk\}$, $Ri \supseteq Xi$, $1 \leq i \leq k$, *is a high-order object.*

Proof: Given $D=\langle R_1, F_1\rangle, \ldots, \langle R_n, F_n\rangle, I, E)$ we wish to know whether $(F \cup I \cup E)^+$ contains a vacuous *ED* on R_1. To solve this problem we construct two new relation schemes $\langle R_{n+1}, \varnothing \rangle$ and $\langle R_{n+2}, \varnothing \rangle$ which have the same number of attributes as R_1. Consider

$$D' = (\langle R_1, F_1 \rangle, \ldots, \langle R_n, F_n \rangle, \langle R_{n+1}, \varnothing \rangle, \langle R_{n+2}, \varnothing \rangle,$$
$$I \cup \{ R_{n+1} \supseteqq R_{n+2}, R_{n+2} \supseteqq R_{n+1} \}, E \cup \{ R_1 \cap R_{n+1} = \varnothing \}).$$

Then $\{ R_{n+1}, R_{n+2} \}$ is a high-order object iff $(F \cup I \cup E)^+$ contains a vacuous *ED* on R_1. For the *ED* $R_1 \cap R_{n+1} = \varnothing$ is trivial iff $(F \cup I \cup E)^+$ contains a vacuous *ED* on R_1. By Lemma 4.2 the vacuous-*ED* problem is undecidable and hence the high-order-object problem is undecidable too. ■

In view of the above theorem, it is of interest to find heuristics for recognizing high-order objects and also to see whether recognition algorithms can be developed for database schemes without *EDs*. In what follows, we develop a heuristic for recognizing strong high-order objects and show that this heuristic leads to a recognition algorithm for database schemes without *EDs*.

It follows from Theorem 5.4 that if C1, C4 and C7-C11 hold then $\{X1, \ldots, Xk\}$ is a strong high-order object. We show that a heuristic for recognizing strong high-order objects can be obtained by replacing C4 and C9, the two *ED* conditions, by the following simple condition:

C4*. Ri, $1 \leqq i \leqq k$, does not occur in any *ED* in *E*.

First, we claim that if C1 and C4* hold then C4 is redundant. Clearly, since C1 holds, $(F \cup I \cup E)^+$ does not contain any *ID* of the form $R[Y] \supseteqq Ri[Z]$ where $R \notin \{R1, \ldots, Rk\}$. Let each $r_i = \{(11\ldots1)\}$, $1 \leqq i \leqq k$. Since C4* holds, $d = (r_1, \ldots, r_k, \varnothing, \ldots, \varnothing) \in \mathrm{Inst}(D)$. Thus $(F \cup I \cup E)^+$ cannot contain a vacuous *ED* on any Ri.

Second, the inference rules of [8] show that C4* together with C1 and C4 imply C9. Now, the following lemma follows immediately from Theorem 5.4:

LEMMA 6.1: *If* C1, C4*, C7, C8, C10 *and* C11 *hold then* $\{X1, \ldots, Xk\}$ *is a strong high-order object.* ■

It is not obvious if the above conditions can be checked algorithmically since the inference problem for *FDs* and *IDs* is, in general, undecidable [9, 16]. However, we give an algorithm for testing the conditions mentioned in the above lemma. C4* is trivial to test. We now show that the remaining conditions, which only involve *FDs* or *IDs*, can also be checked. Note that each *FD* in *F* and each *ID* in *I* is also present in $(F \cup I \cup E)^+$. Thus, if some Fi, $1 \leqq i \leqq k$, contains an *FD* of the form $W \to A$, $Xj - W \neq \varnothing$, then we know that $\{X1, \ldots, Xk\}$ cannot be a strong high-order object since the presence of this FD in $(F \cup I \cup E)^+$ leads to the violation of C 7 or C 10. Similarly, if I contains an ID which violates C 1 (a), C 8 or C 11 then again we know that $\{X1, \ldots, Xk\}$ is not a strong high-order object. Since such violations are easy to detect, in what follows we assume that

1. if $W \to A \in Fi$, $1 \leqq i \leqq k$, then $W \supseteq Xi$ and

2. I does not contain any ID which violates C 1 (a), C 8 or C 11.

We use these assumptions to derive the following important fact:

FACT 6.1: If $Rp[Vp] \supseteq Rq[Vq] \in I$, $1 \leq p \neq q \leq k$, and $W \to A \in Fp$ then $W \supseteq Vp$.

Proof: Since $Rp[Vp] \supseteq Rq[Vq]$ does not violate C 11, $Xp \supseteq Vp$. Further, since $W \supseteq Xp$ is assumed to hold, it follows that $W \supseteq Vp$. ■

We use Fact 6.1 to show:

LEMMA 6.2: 1. *Each FD involving* Ri, $1 \leqq i \leqq k$, *in* $(F \cup I \cup E)^+$ *is also in* Fi^+.

2. *Let* $I_s = \{id: id \in I$ *and some* Ri, $1 \leqq i \leqq k$, *occurs on the* RHS *of* $id\}$. *Then each ID in* $(F \cup I \cup E)^+$ *which has some* Ri, $1 \leqq i \leqq k$, *on its* RHS *is in* I_s^+.

Proof: The proof is similar to that of Lemma 5.1 and Theorem 5.3 in [8]. However, since I satisfies C 1 (a), the chase procedure for *IDs*, used in [8], can be restricted to work only with the *IDs* in I_s. Although Lemma 5.1 and Theorem 5.3 in [8] are stated without *EDs*, condition C 4* ensures that the presence of *EDs* does not cause any problems. ■

An immediate consequence of Lemma 6.2 is that C 7 and C 10 can be checked in polynomial time (*see* Section 4). Similarly, C 1, C 8 and C 11 can also be checked by examining I_s^+ which can be obtained from I_s by using the reflexivity, projection and permutation, and the transitivity axioms for *IDs* [7, 18]. However, it is not obvious that this can be done in polynomial time since the inference problem for *IDs* is PSPACE-complete [7]. In what follows, we show that C 1, C 8 and C 11 can be checked in polynomial time. Recall that we have assumed that I does not violate C 1 (a), C 8 or C 11. First we state some facts about the *IDs* in I_s^+.

FACT 6.2: If $Rp[Y] \supseteq Rq[Z]$, $1 \leq p, q \leq k$, is a non-trivial ID in I_s^+ then $Rp[Y']$, $Y' \supseteq Y$, occurs on the LHS of some ID in I_s and $Rq[Z']$, $Z' \supseteq Z$, occurs on the RHS of some ID in I_s.

Proof: Follows from the three inference axioms for *IDs*. ■

FACT 6.3: No ID in I_s^+ violates C 11.

Proof: The proof is by induction. Recall that no *ID* in I violates C 11. It is easy to see that an *ID* derived from I, by a single application of the reflexivity, projection and permutation or transitivity axiom [7, 18], also satisfies C 11. ■

Since I does not violate C 1 (a), it follows from Fact 6.2 that no *ID* in I_s^+ violates C 1 (a). To check for C 1 (b), we construct a directed graph G_d with k vertices as follows. G_d has an edge (p, q), *i.e.*, an edge from the vertex p to vertex q, if $Rp[Xp] \supseteq Rq[Xq]$, $1 \leq p \neq q \leq k$, is an *ID* in I_s or it can be derived from an *ID* in I_s by a single application of the permutation and projection axiom. We now show:

PROPOSITION 6.1: *C 1 (b) holds iff G_d is strongly connected.*

Proof: *If*: Is is easy to see that if G_d is strongly connected then C 1 (b) holds.

Only If: We show that if $Rp[Xp] \supseteq Rq[Xq]$, $1 \leq p \neq q \leq k$, is an *ID* in I_s^+ then G_d has a path from p to q. The proof is by induction on the number of applications of the transitivity axiom in the derivation of $Rp[Xp] \supseteq Rq[Xq]$. Clearly, if $Rp[Xp] \supseteq Rq[Xq]$ can be derived without using the transitivity axiom then, by construction, (p, q) is a path in G_d. Inductively, assume that the assertion holds for all relevant *IDs* whose derivations involve i applications of the transivity axiom, $i \geq 0$. Let $Rp[Xp] \supseteq Rq[Xq]$ be an *ID* whose derivation uses $i+1$ applications of the transitivity axiom. Let the $i+1$-th application use $Rp[Xp] \supseteq Rj[Yj]$ and $Rj[Yj] \supseteq Rq[Xq]$ to derive $Rp[Xp] \supseteq Rq[Xq]$. Since each *ID* in I_s^+ satisfies C 11 (Fact 6.3), $Yj = Xj$. Now by the inductive assumption, G_d has a path from p to j and from j to q. Hence, G_d has a path from p to q. ■

Thus, C 1 (b) can be checked in polynomial time since G_d can be constructed in polynomial time and a linear-time algorithm for testing if a graph is strongly connected is well-known [1]. Since each *ID* in I_s^+ satisfies C 11, it is easy to see that I_s^+ does not contain any non-trivial *IDs*. Thus, the three *ID* conditions can be checked in polynomial time. Hence, we obtain:

THEOREM 6.2: *Given a database scheme $D = (\langle R1, F1 \rangle, \ldots, \langle Rn, Fn \rangle, I, E)$, a sound, but incomplete, polynomial-time procedure to test if $\{X1, \ldots, Xk\}$, $Ri \supseteq Xi$, $1 \leq i \leq k$, is a strong high-order object can be obtained by checking the conditions* C 1, C 4*, C 7, C 8, C 10 *and* C 11. ■

It follows from Theorem 5.4 that, in the absence of *EDs*, conditions C 1, C 7, C 8 and C 11 are necessary and sufficient to ensure that $\{X1, \ldots, Xk\}$

is a strong high-order object. Hence we obtain:

THEOREM 6.3: *Given a database scheme* $D=(\langle R1, F1\rangle, \langle Rn, Fn\rangle, I)$, *there exists a polynomial-time algorithm to test if* $\{X1, \ldots, Xk\}$, $Ri \supseteq Xi$, $1 \leq i \leq k$, *is a strong high-order object.* ■

Unfortunalety, we have not been able to obtain similar results for partially-strong and weak high-order objects. In both these cases, condition C 3 allows for some additional *IDs*. Due to these *IDs*, *FDs* and *IDs* interact to produce new dependencies, *i.e.*, Lemma 6.2 no longer holds.

Example 6.1: Let

$$D=(\langle R(ABC), \{A \to BC\}\rangle, \langle S(EF), \{R[AB] \supseteq S[EF], S[EF] \supseteq R[AB]\}\rangle).$$

Note that R has a unique minimal key, *i.e.* A, and is in BCNF. Although S has no *FDs* defined on it, $A \to BC$ and $R[AB] \supseteq S[EF]$ imply the new *FD* $E \to F$. Now it is easy to see that $\{A, E\}$ is a partially-strong high-order object. Also note that both the *IDs* violate C 11 but satisfy C 3. ■

Further, we do not even have a set of necessary and sufficient conditions to characterize weak high-order objects. Thus, the recognition problem for weak and partially-strong high-order objects in the presence of *FDs* and *IDs* remains open.

7. CONCLUSIONS

Ascribing uniqueness and independent existence to objects, we formally defined the notions of weak and strong objects for relational database schemes with *FDs*, *IDs*, and *EDs*. Next, we characterized weak and strong objects in terms of the specified dependencies. Arguing that X, $R \supseteq X$, should be treated as an object if X is the minimal left-hand side of an *FD* and R does not occur on the RHS of any *ID*, we defined weak and strong *ONFs* for database schemes. We showed that testing whether a given database scheme is in *ONF* is, in general, undecidable. However, we presented polynomial-time heuristics for this problem and also developed polynomial-time algorithms for database schemes without *EDs*.

We generalized the notion of objects and developed the notion of high-order objects to capture situations, which may arise during view-integration, where several relation schemes represent different views of the same set of real-world objects. We defined the notions of weak, partially-strong and strong high-order objects for database schemes with *FDs*, *IDs* and *EDs*. We showed that the recognition problem for high-order objects was, in general, undecidable. However, we presented a polynomial-time heuristic for strong high-order objects and used it to develop a polynomial-time recognition algorithm for database schemes without *EDs*.

However, several problems still remain open. We have not been able to give a sound and complete characterizations of weak objects and weak high-order objects. Besides being of theoretical interest, such characterizations may also be useful in deriving efficient recognition algorithms. It is also of interest to know whether our heuristics for *ONFs* (*see* Theorems 4.4 and 4.5) can be improved. Our attempts to find polynomial-time heuristics (resp. algorithms) for recognizing high-order objects have not been very successful. We have obtained a polynomial-time heuristic for recognizing strong high-order objects and showed that in the absence of *EDs* this heuristic is actually an algorithm. However, at present, we do not even know whether the absence of *EDs* makes the recognition problem for partially-strong (resp. weak) high-order objects decidable. Thus, the recognition problem for partially-strong (resp. weak) high-order objects needs further investigations.

ACKNOWLEDGEMENTS

We would like to thank the anonymous referees for their constructive comments on the first version of this paper.

REFERENCES

1. A. V. Aho, J. E. Hopcroft and J. D. Ullman, The Design and Analysis of Computer Algorithms, *Addison-Wesley*, 1974.
2. C. Batini, M. Lenzerini and M. Moscarini, Views Integration, in *Methodology and Tools for Data Base Design*, S. Ceri Ed., North-Holland, 1983, pp. 57-84.
3. P. A. Bernstein and N. Goodman, What Does Boyce-Codd Normal Form Do?, *Proc. 6th Internat. Conf. on Very Large Data Bases*, Montreal, 1980, pp. 245-259.
4. J. Biskup, Boyce-Codd Normal Form and Object Normal Forms, *Information Processing Letters*, 1989, *32*, pp. 29-33.
5. J. Biskup and B. Convent, A Formal View Integration Method, *Proc. of ACM-SIGMOD Intl. Conf. on Management of Data*, 1986, pp. 398-407.
6. J. Biskup, J. Demetrovics, L. O. Libkin and I. B. Muchnik, On the Relational Database Schemes having Unique Minimal Key, *Journal Information Processing and Cybernetics* (EIK), 1991, *27*, 4, pp. 217-225.
7. M. A. Casanova, R. Fagin and C. H. Papadimitriou, Inclusion Dependencies and their Interaction with Functional Dependencies, *Journal of Computer and System Sciences*, 1984, *28*, 1, pp. 29-59.

8. M. A. CASANOVA and V. M. P. VIDAL, Towards a Sound View Integration Methodology, *Proc. of 2nd ACM SIGACT-SIGMOD Symp. on Principles of Database Systems*, 1983, pp. 36-47.
9. A. K. CHANDRA and M. Y. VARDI, The Implication Problem for Functional and Inclusion Dependencies is Undecidable, *S.I.A.M. Journal on Computing*, 1985, *14*, 3, pp. 671-677.
10. C. J. DATE, Referential Integrity, *Proc. of the 7th International Conf. on Very Large Data Bases*, 1981, pp. 2-12.
11. R. ELMASRI and G. WIEDERHOLD, Data Model Integration using Structural Model, *Proc. of ACM-SIGMOD Intl. Conf. on Management of Data*, 1979, pp. 191-202.
12. H. J. HERNÁNDEZ and E. P. F. CHAN, A Characterization of Constant-Time Maintainability for BCNF Database Schemes, *Proc. ACM-SIGMOD Internat. Conf. on Management of Data*, 1988, pp. 209-217.
13. W. KENT, A Simple Guide to Five Normal Forms in Relational Database Theory, *Comm. A.C.M.* 26, 1983, pp. 120-125.
14. C. H. LE DOUX and D. S. PARKER, Reflections on Boyce-Codd Normal Form, *Proc. 8th Internat. Conf. on Very Large Data Bases*, Mexico City, 1982, pp. 131-141.
15. D. MAIER, The Theory of Relational Databases, *Computer Science Press*, 1983.
16. J. C. MITCHELL, The Implication Problem for Functional and Inclusion Dependencies, *Information and Control*, 1983, *56*, 3, pp. 154-173.
17. S. L. OSBORN, Normal Forms for Relational Data Bases, *Ph. D. Thesis*, Dept. of Computer Science, University of Waterloo, 1978.
18. J. PAREDAENS *et al.*, The Structure of the Relational Database Model, *Springer-Verlag*, 1989.
19. J. D. ULLMAN, Principles of Database Systems, *Computer Science Press*, 1982.
20. G. VOSSEN, Datenmodelle, Datenbanksprachen und Datenbank-Management-Systeme (in Germain), *Addison-Wesley*, 1987.
21. G. VOSSEN, Entwurf und Bearbeitung von Datenbanken im Universalrelationen-Datenmodell, *Dissertation*, RWTH Aachen, 1986.