No. 98 (80 - 36)

HANDLING SUMMARY INFORMATION IN DATABASES: DERIVABILITY

By
Hideto Sato

November 1980

HANDLING SUMMARY INFORMATION IN DATABASES: DERIVABILITY

Hideto Sato

Institute of Socio-Economic Planning
The University of Tsukuba
Sakura, Ibaraki 305, Japan

ABSTRACT: "Summary data" are representations of "groups of facts." Statistics are their typical examples. They are often major components of databases that deal with huge domains, such as objects in a whole country or events occurred in a long time range. Although any summary can be reproduced from the corresponding originals, they are often not available and required data may or may not be derived from given summary data. A schema of summary data is defined as a relationship between classifications on object types and attribute value types. Reclassification rules are introduced as semantic relations among classifications. Then set theoretical lemmata provide a procedure for judging derivability of summary data under a user's schema from data under an observer's one. It is also indicated that the former can be inferred, if they are derivable, by means of a certain sequence of natural joins among the observed data and the predefined reclassification rules. Discussions are made with examples in the statistical field.

KEYWORDS AND PHRASES: summary data, classification of objects, derivation by inference, statistical database, data abstraction.

1. Introduction

"Summary data" are representations of "groups of facts." Their typical examples are found in the statistical field. For example, take such data as "consumption expenditures by income groups of households." These data are summary data that represent some groups of atomic facts such as expenditures on individual commodities by individual households. Such summary data are often major components of databases that deal with huge domains, such as objects in a whole country or events occurred in a long time range. They are demanded not only by governments and international organizations but also by research and planning sections in enterprises and universities. In fact, the informa-

tion about economy and business, supplied by commercial databases today, consists most of summary data. However, few theoretical contributions have been made to this topic in the field of database analyses. This might be because most of the analysts image summary data whose corresponding original data (atomic data) are simultaneously available. Certainly, summary data have been compounded of atomic ones, so that, any of them can be reproduced as far as the corresponding atomic ones are managed. Although this understanding is conceptually correct, the condition is satisfied seldom in actual practices. Atomic data are often not available when a user wants their summary. In the above example, individual records on expenditures by households are usually not preserved for a long time by a physical or economical reason.

As another example, let us take such data as "yearly rainfall by region." Since a region can be infinitely partitioned into smaller regions, an atomic region cannot be defined in the strict sense. The same is true of time intervals. Therefore such data as "amount of rainfall" should be recorded as data related to certain subsets of the object types, region and time. This type of data may not be called summary data if each value of them corresponds to each observation, e.g. rainfall per unit region and unit time for an observation. But they have the same property as with summary data; i.e. each value of the data is related to certain subsets of object types. So that, we shall use the term "non-atomic data" instead of the term "summary data" to indicate such data that are reducible to finer information units, at least conceptually.

Even when dealing with non-atomic data, it is rather easy to handle them if there is consensus among the users about the classification of objects related to the data. But if desirable classifications differ from one user to another or if desirable classifications change with time, serious problems may arise. In fact, many users of statistics are vexed with such data handling as to derive their desirable data from given statistics and to ensure consistencies among statistics that they want to use. The author once suggested two notions, "derivability" and "comparability," corresponding to these two types of data handling, respectively (Sato 1980). "Derivability" concerns whether data subject to one subschema with a user's classification can be derived from data collected under a different subschema with an observer's classification. "Comparability" shows whether comparison is semantically possible between data collected under different observers' schemata. The following sections are devoted to a formal and detailed discussion about "derivability" of non-atomic data. The formal discussion about "comparability" will appear in (Sato 1981).

In section 2, the notion of classification of objects and the notion of derivability among the classifications are formally defined. In Section 3, derivation of non-atomic data by inference is introduced on the basis of set theoretical lemmata, and, in Section 4, its practical implications are illustrated. In Section 5, the notion of derivability is expanded to n-tuple non-atomic data. In Section 6, the conclusion is presented. Finally, in Appendix, the proofs of the lemmata in the sections are summarized.

The notion of classification defined in Section 2 is closely related to the discussion on data abstraction, especially "generalization" in [Smith+Smith 1977] and [Codd 1979]. In fact, the classification hierarchy in this paper is similar to the generalization hierarchy in those papers. However our discussion concerns another type of abstraction than that of Smith and Codd. They mainly handled composition of object types whose members are atomic objects, while we treat composition of grouped objects whose members are atomic objects and we also analyze still more abstract object types whose members are grouped objects. Such grouped objects and abstract object types are called here "categories" and "classifications," respectively.

As for the derivation by inference in Section 3, a similar but more general studies were made in [Bubenko+et al. 1976] and [Bubenko 1977]. But the formal discussion like in this paper was not presented there.

The discussion about classifications is well-known in mathematics as the theory of partitions; for example see [Borůvka 1976]. However, the notions and the terminology in it are neither familiar to most of practicians nor appropriate for ordinary types of database systems. In this paper, the analysis is developed on the basis of the notions and the terminology conventional for practicians in the field of statistics and database analyses.

2. Derivability of Non-Atomic Data

Table 1 shows an example of non-atomic data in a statistical field. C1 in the table indicates a classification on the land

Table 1. Non-Atomic Data under Observer's Data Scheme

D1	
Land Transport Industry (C1)	Number of Employees (N)
a) Railroad Passenger Transport	68
b) Railroad Freight Transport	23
c) Other Passenger Land Transport	586
d) Local Trucking	493
e) Long-Distance Trucking	295

Table 2. User's Data Scheme

D2	
Land Transport Industry (C2)	Number of Employees (N)
a') Passenger Transport	?
b') Freight Transport	?

transport industry, and the numerical values show the number of employees who work for the corresponding industrial categories. The terminology used in this paper is defined as follows:

Def. 1 Object Type, Category, Classification

An object type A is a set A of atomic objects.

A category x of A is a subset x of A.

A classification X on A is a family X of mutually disjoint categories of A and we denote $X \in CL(A)$.

In Table 1, a, b, c, d, e are categories of the object type "Land Transport Industry," and C1 = {a, b, c, d, e} is a classification on the same object type. We assume this classification was made by the observer who collected the data, so that C1 is called an "observer's classification." Next, suppose a user wants data under such a data scheme with another classification (user's classification) as shown in Table 2. In this case, if each category in C2 in Table 2 coincides with a union of some categories in C1 in Table 1, the summary data D2 corresponding to the data scheme in Table 2 can be inferred from the data D1 in Table 1. When C2 satisfies this condition, we say C2 is derivable from C1. Formally:

Def. 2 Derivability of Classification

Let $X, Y \in CL(A)$. If they satisfy the following conditions,

then we say Y is derivable from X and denote $X \dashrightarrow Y$.

(P1) $\bigcup Y \subset \bigcup X$

(P2) $x \in X,\ y \in Y,\ x \cap y \neq \phi \Longrightarrow x \subset y$.

The meaning of the conditions in Def. 2 would be intuitively understandable. But the notations in the conditions may not be familiar to practicians and the verification of them can not be easily done in a direct manner by an ordinary type of DBMS unless the classifications are time-invariable. In order to obtain a simpler expression of the conditions, let us introduce a semantic relation between classifications as follows:

Def. 3 Connection Relation

Let $X, Y \in CL(A)$. The connection relation F of X with Y is defined as:

$$F = \{ (x,y) \mid x \in X,\ y \in Y,\ x \cap y \neq \phi \} .$$

Then the following lemma holds.

Lemma 4 Let $X, Y \in CL(A)$ and F be the connection relation of X with Y. If F satisfies the following condition, then $X \dashrightarrow Y$. The converse is also true.

(P3) $\forall y \in Y, \bigcup F^{-1}(y) = y$,

where $F^{-1}(y) = \{ x \mid (x,y) \in F \}$.

Remark $x \in F^{-1}(y)$ is a category (subset) of A, and $\bigcup F^{-1}(y)$ means the union of categories belonging to $F^{-1}(y)$.

Using this lemma, the definition of derivability in Def. 2 can be rewritten as in the following definition.

Def. 5 Reclassification Rule

In Lemma 4, when F satisfies the condition P3, F is called the reclassification rule from X to Y. In this situation, we say Y is derivable from X under F and denote $X \overset{F}{\dashrightarrow} Y$.

Remark The reclassification rule from X to Y is also an inclusion relation of X in Y, i.e. $\{ (x,y) \mid x \in X, y \in Y, x \subset y \}$, because of the condition P2 in Def. 2.

The reclassification rule defined above is the same type of table as called a "classified catalogue" or a "correspondence table" in the practical fields.

In our working example, suppose the reclassification rule from C1 to C2 is given by R1 in Table 3. This table indicates that the category a' in C2 includes the categories a and c in C1 and b' in C2 includes b, d and e in C1. Then the data corresponding to Table 2 can be obtained as in Table 4 by arranging in non-1NF the result of the natural join (or composition of relation) of R1 in Table 3 with D1 in Table 1 on the common domain C1. In Table 4, the values in the column N are represented as lists of the corresponding values in Table 1. Although each value in this example might be expressed as the sum of the values in the corresponding list, the more general method is to represent the values by lists as in Table 4, since the average, maximum or minimum values or the collections of character strings can be alternatively derived from these lists.

Table 3. Reclassification Rule (1)

R1	
C1	C2
a) Railroad Passenger Transport	a') Passenger Transport
b) Railroad Freight Transport	b') Freight Transport
c) Other Passenger Land Transport	a') Passenger Transport
d) Local Trucking	b') Freight Transport
e) Long-Distance Trucking	b') Freight Transport

Table 4. Derived Data under User's Data Scheme

D2	
Land Transport Industry (C2)	Number of Employees (N)
a') Passenger Transport	(68, 586)
b') Freight Transport	(23, 493, 295)

3. Determination of Derivability by Inference

The situation described in the previous section has the following shortcomings.

(1) A user who wants to obtain data from a DB would need to know every observer's classification under which the data have been described in the DB. This condition puts a heavy burden on the user, who may wish to use many sets of data observed by many different observers.

(2) An observer's classification is usually time-varying. Since any reclassification rule given by a user depends on the corresponding observer's classification, the rule would have to be rewritten whenever the observer's classification is revised.

In order to improve these shortcomings, we introduce an inferential process defined over a set of reclassification rules on the basis of the following lemmata. Suppose X, Y, Z are classifications in an object type and F, G, H are reclassification rules.

Lemma 6 $X \xrightarrow{F} Y,\ Y \xrightarrow{G} Z \Longrightarrow X \xrightarrow{G \cdot F} Z,$

where $G \cdot F$ means the composition of the relations F and G; i.e. $G \cdot F = \{ (x,z) \mid \exists y,\ (x,y) \in F \wedge (y,z) \in G \}$.

Lemma 7 Suppose $X \xrightarrow{F} Y,\ X \xrightarrow{H} Z.$

(1) $H = H \cdot F^{-1} \cdot F \Longrightarrow Y \xrightarrow{H \cdot F^{-1}} Z$ (Z is derivable from Y).

(2) $H \neq H \cdot F^{-1} \cdot F \Longrightarrow$ Z is not derivable from Y.

Again we return to our working example. Let us introduce a new classification C3 whose categories are likely to be finer and more stable (relatively time-invariable) than those of the observer's classification C1. Suppose that two reclassification rules are given as follows: R2 from C3 to C1 by the observer and

Table 5. Reclassification Rule (2)

R2	
C3	C1
a) Railroad Passenger Transport	a) Railroad Passenger Transport
b) Railroad Freight Transport	b) Railroad Freight Transport
c) Urban Railway and Subway Trans.	c) Other Passenger Land Trans.
d) Bus and Coach Transport	c) Other Passenger Land Trans.
e) Other Passenger Land Trans.	c) Other Passenger Land Trans.
f) Local Trucking	d) Local Trucking
g) Long-Distance Trucking	e) Long-Distance Trucking

R3	
C3	C2
a	a') Passenger Transport
b	b') Freight Transport
c	a') Passenger Transport
d	a') Passenger Transport
e	a') Passenger Transport
f	b') Freight Transport
g	b') Freight Transport

R3 from C3 to C2 by the user, as in Table 5. Since R3 is independent of the observer's classification, the user's reclassification rule R3 is easily defined. The direct reclassification rule R1 from C1 to C2 can be inferred from the result of the natural join of R2 with R3 on C3. The derivation of the data under the user's data scheme is illustrated in Fig. 6. In this figure, a similar diagram as used in (Nijssen 1977) is employed for representing semantic connections. The circles indicate domains of names or values; in this case, classifications on an object type and an attribute value domain. The rectangles indicate relationships among the domains; for example, the observed data D1 is expressed there as a relationship between the observer's classification C1 and the attribute value domain N. The rectangles drawn with broken lines indicate relations obtained by using natural joins.

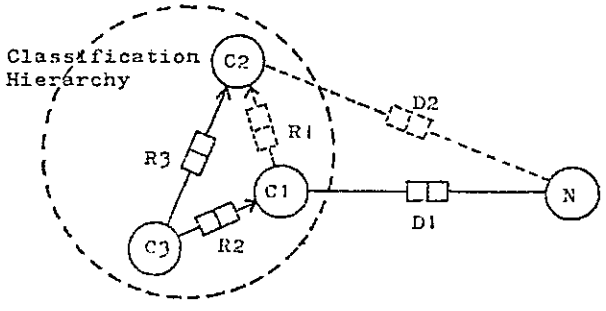

Fig. 6 Derivation of Data under User's Data Scheme

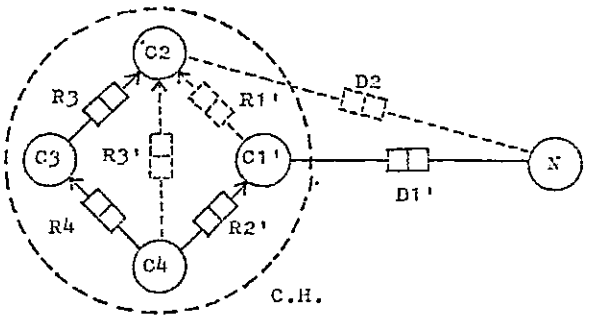

Fig. 7 Revision of Observer's Data Scheme

We can construct a DBMS that recognizes a classification hierarchy as shown in the figure and automatically carries out such inferential processes (implementation of natural joins) as mentioned above. Suppose that R2 and D1 exist in a DB and a user gives R3 to the DB as additional information and requests such data under a user's data scheme D2. Then the DBMS checks whether C2 in D2 is derivable from C1 in D1 by using Lemma 7 above, computes R1, if C2 is derivable, by means of the natural join of R2 with R3, and presents D2 to the user after joining R1 with D1.

Next, let us consider the case in which the observer's classification has been revised from C1 to C1'. Then, on introducing a new classification C4 from which both C1' and C3 are derivable, the classification hierarchy becomes connected again as in Fig. 7. Such a revision of contents in the DB is easy since it is independent of user's reclassification rules such as R3. R3' in the figure can be inferred from R4 and R3 by using Lemma 6, so that automatic inference of D2 can be carried out in a similar manner to the previous case in Fig. 6.

4. Practical Implication of Non-Atomic Data Approach

The non-atomic data approach discussed in the previous sections free us from a vexing problem of identifying atomic objects. For example, in the previous case, it is not necessary for the user to have explicit knowledge of individual activities belonging to his category such as "Passenger Transport Industry." In fact, almost all users of the statistics may not recognize whether "Passenger Transport Industry" includes "Dining Service in Train" or not. Without such trifling knowledge, he can make a

query to the DB. It is only needed for him to clarify a reclassification rule to his classification from a classification predefined in the DB, such as C3 in Fig. 6. The reclassification rule is the inclusion relation between the classifications as shown in Section 2, so that it can be defined without any knowledge of atomic objects. In actual life, we are often satisfied with communication in such abstract information. However, in order that the semantics of data can be properly communicated, there should exists a classification common to both the user and the observer, such as C3 in Fig. 6. In the actual world, "standard classifications" defined by the authorities play the role of such common classifications; for example, the standard industrial classification and the administrative regional sections. A "standard classification system" means a set of classifications that are organized in a systematic way. They are usually constructed in a hierarchical structure of classifications, such as a decimal classification system.

Now, let us consider implication of our previous discussion in the case with a standard classification system. Fig. 8 illustrates connections of a standard classification system with actual sets of non-atomic data (observed data). S1, S2, S3, and S4 indicate classifications in the standard classification system. O1, O2, O3, O4 and O5 indicate classifications related to the actual non-atomic data, that is, each classification corresponds to each actual non-atomic data set. Arrows in the figure show the directions of derivability. Such a situation is frequently observed in the statistical field.

Then suppose that a user requests to retrieve the data

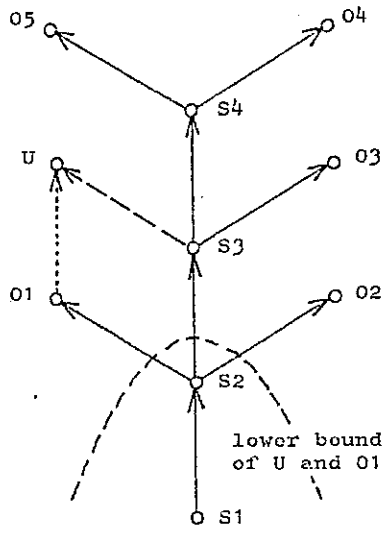

Fig. 8 Classification Hierarchy with Standard Classifications

related to the classification 01, and that he wants them to be classified under his scheme, that is, he has a user's classification such as U in the figure. In order to check whether U is derivable from 01 or not, one has to trace back the tree to the common trunk. Lemma 6 and Lemma 7 imply that it is not necessary to identify the corresponding atomic objects but it is only needed to find a classification in the given classification system from which both U and 01 are commonly derivable. In this example, the classifications S1 and S2 have this property. The set of such classifications is called the "lower bound" of U and 01 in the classification system. Although an arbitrary classification in the lower bound is sufficient enough to specify the difference between U and 01, the coarser the classification, the easier the comparison between them. Therefore the best choice is the coarsest classification in the lower bound, that we name the "coarsest common decomposition" (for short, CCD). In the above example, the CCD of U and 01 in the classification system is S2. With the CCD it is easy to check the derivability of U from 01

and the direct reclassification rule from O1 to U is given according to Lemma 6 and Lemma 7 if U is derivable from O1.

Next recall that the finest classification S1 in the figure is not supposed to be a set of atomic objects but only one of the classifications in the standard classification system. Even standard classifications may be time-varying. For example, the standard industrial classification system in Japan has been revised eight times in the past 25 years. Usually, the coarser a classification the more it is stable to changes in the real world, because a change in a detail may be ignorable at higher level of abstraction. Therefore the revision of a classification system raises a demand for connecting new classifications with old ones. This situation is illustrated in Fig. 9, where S1' and S2' indicate new classifications. If the distinction between old and new ones are recognizable, usually there exists a finer classification, such as S0, with which the distinction can be specified[1/]. In fact, such specification like S0 has been usually announced by the authority at the revision of standard classification systems.

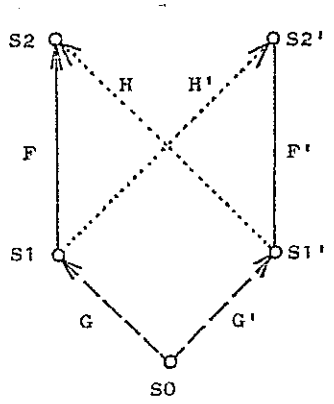

Fig. 9 Revision of Standard Classifications

Consequently, this approach allows, if necessary, a DB based on non-atomic data to approach to a DB based on atomic data after repeating revisions of standard classifications as mentioned above.

* 1/ Note that in Fig. 9 we do not consider direct reclassification rules such as H or H'. Such rules can be inferred from the rules F, G and G' or F', G' and G respectively by using Lemma 6 and Lemma 7. If such connections as H and H' were defined at each time of the revision, the system would become complicated and confusion might result after many revisions.

5. Extension to n-Tuple Non-Atomic Data

So far, the type of non-atomic data has been limited to a binary relation, one domain of which is a classification on an object type and another is an attribute value type. It is needed to extend it to an n-tuple relation in order to deal with actual data. Nevertheless, so far as non-atomic data have a single key domain, that is, each n-tuple of the data is identified by a category of a single object type, they can be decomposed into binary ones, to which the previous discussions can be directly applied. Hence, let us consider the case of non-atomic data which are related to two or more object types. For example, "amount of rainfall" is usually related to two object types, "Region" and "Time." In this case, we say the data are identified by an "aggregate object type" of "Region" and "Time," i.e. "Region" x "Time."

Suppose an aggregate object type $A \times B$. Let $X \in CL(A)$ and $Y \in CL(B)$. Then the following question may arise. How does the classification on $A \times B$, implied by the classifications X and Y, classify the atomic objects $(a, b) \in A \times B$? As example, let us

	y1	y2	
x1	(a1,b1)	(a1,b2)	(a1,b3)
	(a2,b1)	(a2,b2)	(a2,b3)
x2	(a3,b1)	(a3,b2)	(a3,b3)

Fig. 10 Classification of Aggregate Objects

consider the case where X = {x1, x2} = {{a1, a2}, {a3}}, Y = {y1, y2} = {{b1}, {b2, b3}} as in Fig. 10. In this case, it is natural to consider that the objects belonging to A × B are classified into four categories x1 × y1, x1 × y2, x2 × y1, and x2 × y2 by the classifications X and Y, where x1 × y1 = {(a1, b1), (a2, b1)} and so on. This consideration leads to the following definition of a classification on an aggregate object type.

Def. 8 Product of Classifications

Let $X \in CL(A)$ and $Y \in CL(B)$. The product of classifications X and Y, denoted $X \boxtimes Y$, is defined as:

$$X \boxtimes Y = \{ x \times y \mid x \in X \wedge y \in Y \},$$

where $x \times y$ is an ordinary cartesian product of x and y.

By defining the product of classifications on object types as above, we can prove it is a classification on the aggregate object type in the meaning of Def. 1.

Lemma 9 Let $X \in CL(A)$ and $Y \in CL(B)$. Then $X \boxtimes Y$ is a classification on $A \times B$, i.e. $X \boxtimes Y \in CL(A \times B)$.

Remark $X \times Y$ is not a classification on $A \times B$.

Next, let us consider the reclassification rules among classifications on an aggregate object type. As for them, the following lemma holds.

Lemma 10 Let $X, X' \in CL(A)$ and $Y, Y' \in CL(B)$. Suppose $X \xrightarrow{F} X'$ and $Y \xrightarrow{G} Y'$. Then $X \boxtimes Y \xrightarrow{H} X' \boxtimes Y'$ where

$$H = \{ (x \times y, x' \times y') \mid (x, x') \in F \wedge (y, y') \in G \}.$$

This lemma implies two things. First, one product of classifications on an aggregate object type is derivable from another product of classifications on the same aggregate object type if every classification composing the former product is derivable from the counterpart of the latter. Secondly, the reclassification rule to a product of classifications can be computed from the reclassification rules to the classifications composing the product. Note that H in the lemma has one to one correspondence with the cartesian product of F and G, $\{ ((x, x'), (y, y')) \mid (x, x') \in F \wedge (y, y') \in G \}$, hence H can be easily inferred from F and G.

Using the above lemma, derivation of 3-tuple non-atomic data can be implemented in a similar manner to the case of binary non-atomic data. Suppose an observer's data scheme such as R(X, Y, V) where $X \in CL(A)$ and $Y \in CL(B)$; e.g. "Rainfall"($X \in$ CL("Region"), $Y \in$ CL("Time"), V = "Amount of Rainfall"). Next, suppose a user's sheme R'(X', Y', V) is given as in Fig. 11. Then the DBMS computes H from F and G by using Lemma 10 and presents the required data R' after joining R with H.

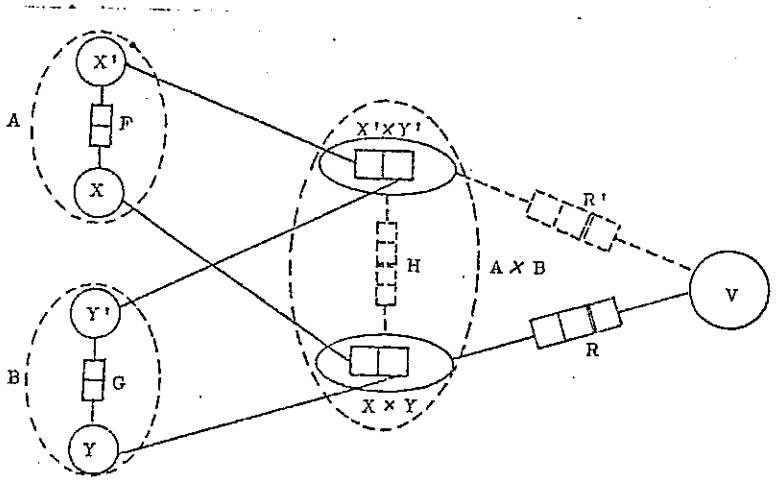

Fig. 11 Derivation of n-Tuple Non-Atomic Data

The definition and the lemmata introduced here can be easily extended to the case of an aggregate object type compounded of n object types.

6. Conclusion

In this paper, we have formally defined derivability of a classification of objects and pointed out that non-atomic data (summary data) under user's schema can be inferred from the data under observer's schema by means of a certain sequence of natural joins among the data and predefined reclassification rules. The notion and the lemmata introduced in this paper are not newly revealed things but formal follow-ups after data handlings manipulated by practicians, especially by statisticians. A formal approach provides such powerful supporting factors as:

(1) a standard for judging whether notions used in one field are applicable to other fields, and

(2) a foundation for translating abstract notions into such active notions as can be implemented by computers.

This paper was initially intended for a statistical database that serves for integrating various types of statistics into a system of statistics; for such a system, see [SSDS 1975], and for data handling in the field, see Chapter III-D in it. However, examples of summary data are found everywhere. For example, even in enterprises, information related to the long past is often preserved in the form of summary data. Hence, the notion of derivability in this paper may be practicable for database management systems for general purposes.

Finally, let us mention a bit of the notion of "deriv-

ability." In the statistical field, it has been often used to imply existence of an inclusion relation between classifications, like in this paper. On the other hand, in the database field, it has been used to imply possibility of composing a relation of other relations by using natural joins; for example, see [Schenk+ Pinkert 1977] or [Lozinskii 1980]. Although these two notions of derivability are explained in a natural language as if they were independent, the former is, as we have seen, a special case of the latter concerning a database with semantic relations such as reclassification rules. This may imply that the problems of non-atomic data will be resolved in the context of general derivability of data in database analyses. However, the procedures to find a derivation path presented in the papers have some problems yet for our purpose. First, the judgement of derivability depends just on functional dependencies explicitly declared in a database, so that it concerns only Lemma 6 in our context but not Lemma 7. Secondly, they draw their inference from the closure of functional dependencies in the database. However, inference from functional dependencies may not result a unique solution, such as the "multiple access path problem" in [Carlson+Kaplan 1976]. Therefore, the procedures provide no sufficient guarantee of semantics against derived data. Discussions about derivation of data may become fruitful when the mathematical lemmata are applied to a certain range where the semantics of derived data are obvious. Though our discussion concerns only a special type of data, it provides an example in which such a range is confined to a classification hierarchy.

ACKNOWLEDGMENT: The auther is very grateful to Professor Ryosuke Hotaka for helpful and useful discussions on the subject. He also thanks Professor Shuntaro Shishido who encouraged him to work on the problems of statistical databases.

APPENDIX: Proofs of Lemmata

Proof of Lemma 4

($\Longrightarrow$) (P3) $\Rightarrow \forall y \in Y,\ y = \bigcup \{ x \mid (x,y) \in F \}$

$\Rightarrow \forall y \in Y,\ y = \bigcup \{ x \mid x \in X,\ x \cap y \neq \phi \}$ (*)

(1) (*) $\Rightarrow \forall y \in Y,\ y \subset \bigcup X \Rightarrow \bigcup Y \subset \bigcup X$, and

(2) (*) $\Rightarrow (x \cap y \neq \phi \Rightarrow x \subset y)$.

Hence X and Y satisfy the condition (P1) and (P2) in Def. 2, so X ----> Y.

($\Longleftarrow$) X ----$\not>$ Y $\Rightarrow$ (P1) and (P2) holds. Then

(1) $\bigcup Y \subset \bigcup X \Rightarrow \forall y \in Y,\ \exists x \in X,\ x \cap y \neq \phi \Rightarrow \forall y \in Y,\ F^{-1}(y) \neq \phi$,

(2) $\bigcup F^{-1}(y) = \{ a \mid \exists x,\ (x,y) \in F \wedge a \in x \}$

$= \{ a \mid \exists x \in X,\ x \cap y \neq \phi \wedge a \in x \} = \{ a \mid \exists x \in X,\ x \subset y \wedge a \in x \}$

((P2) is used. Note always $x \subset y \Rightarrow x \cap y \neq \phi$)

$= y$ (Since $\bigcup Y \subset \bigcup X$, $\forall a \in y, \exists x \in X,\ a \in x$).

Hence, if X ----> Y then (P3) holds. (Q.E.D.)

Proof of Lemma 6

1) First, we shall prove G•F is the connection relation of X with Z. It is sufficient to prove, for $x \in X$ and $z \in Z$, "$\exists y \in Y,\ x \cap y \neq \phi \wedge y \cap z \neq \phi \Longleftrightarrow x \cap z \neq \phi$," because, if it is holds, then

$(x,z) \in G \cdot F \Longleftrightarrow \exists y \in Y,\ (x,y) \in F \wedge (y,z) \in G$

$\Longleftrightarrow \exists y \in Y, x \cap y \neq \phi \wedge y \cap z \neq \phi \Longleftrightarrow x \cap z \neq \phi$ (from the assumption above)

$\Longleftrightarrow$ (x,z) is an element of the connection relation.

($\Longrightarrow$) $\exists y \in Y,\ x \cap y \neq \phi \wedge y \cap z \neq \phi$

$\Longrightarrow x \subset y \subset z$ ($\because X \dashrightarrow Y \wedge Y \dashrightarrow Z$) $\Longrightarrow x \cap z \neq \phi$.

($\Longleftarrow$) $x \cap z \neq \phi \Longrightarrow \exists y \in Y,\ x \cap y \neq \phi \wedge y \cap z \neq \phi$ ($\because \cup Z \subset \cup Y$).

2) Next, we shall prove (P3) in Lemma 4 holds with respect to G•F.

$\cup (G \cdot F)^{-1}(z) = \{\ a \mid \exists x,\ (x,z) \in G \cdot F \wedge a \in x\ \}$

$= \{\ a \mid \exists x, \exists y,\ (x,y) \in F \wedge (y,z) \in G \wedge a \in x\ \}$

$= \{\ a \mid \exists y,\ (y,z) \in G \wedge a \in \cup F^{-1}(y)\ \}$

$= \{\ a \mid \exists y,\ (y,z) \in G \wedge a \in y\ \}$ ($\because X \xrightarrow{F} Y$ and (P3))

$= \{\ a \mid a \in \cup G^{-1}(z)\ \} = \cup G^{-1}(z) = z$ ($\because Y \xrightarrow{G} Z$ and (P3)).

Since (P3) in Lemma 4 holds, $X \xrightarrow{G \cdot F} Z$. (Q.E.D.)

Proof of Lemma 7

1) First, we shall prove $H \cdot F^{-1}$ is the connection relation of Y with Z. It is sufficient to prove, for $y \in Y$ and $z \in Z$, "$\exists x \in X,\ x \cap y \neq \phi \wedge x \cap z \neq \phi \Longleftrightarrow y \cap z \neq \phi$," because of the reason similar to Lemma 6.

($\Longrightarrow$) $\exists x \in X,\ x \cap y \neq \phi \wedge x \cap z \neq \phi$

$\Longrightarrow x \subset y \wedge x \subset z$ ($\because X \dashrightarrow Y \wedge X \dashrightarrow Z$) $\Longrightarrow y \cap z \neq \phi$.

($\Longleftarrow$) $y \cap z \neq \phi \Longrightarrow \exists x \in X,\ x \cap y \neq \phi \wedge x \cap z \neq \phi$ ($\because \cup Z \subset \cup X \wedge \cup Y \subset \cup X$).

2) Next, we shall prove (P3) in Lemma 4 holds with respect to $H \cdot F^{-1}$ if and only if $H = H \cdot F^{-1} \cdot F$.

$\cup (H \cdot F^{-1})^{-1}(z) = \{\ a \mid \exists y,\ (y,z) \in H \cdot F^{-1} \wedge a \in y\ \}$

$= \{\ a \mid \exists y,\ (y,z) \in H \cdot F^{-1} \wedge a \in \cup F^{-1}(y)\ \}$ ($\because X \xrightarrow{F} Y$ and (P3))

$= \{\ a \mid \exists x, \exists y,\ (y,z) \in H \cdot F^{-1} \wedge (x,y) \in F \wedge a \in x\ \}$

$= \{\ a \mid \exists x,\ (x,z) \in H \cdot F^{-1} \cdot F \wedge a \in x\ \} = \cup (H \cdot F^{-1} \cdot F)^{-1}(z)$.

On the other hand, $z = \cup H^{-1}(z)$ since $X \xrightarrow{H} Z$. Hence $\forall z \in Z$, $z = \cup (H \cdot F^{-1})^{-1}(z)$ if and only if $H = H \cdot F^{-1} \cdot F$, because both domains of $H \cdot F^{-1} \cdot F$ and H are subsets (sub-families) of X whose members (categories) are mutually disjoint. That is,

Lemma 7 holds since the connection relation is unique between classifications. (Q.E.D.)

Proof of Lemma 9

1) $x \times y \in X \boxtimes Y \Longrightarrow x \in X \wedge y \in Y \Longrightarrow x \subset A \wedge y \subset B \Longrightarrow x \times y \subset A \times B$,

2) $x \times y,\ x' \times y' \in X \boxtimes Y,\ x \times y \cap x' \times y' \neq \emptyset$

$\Longrightarrow x \cap x' \neq \emptyset \wedge y \cap y' \neq \emptyset \Longrightarrow x = x' \wedge y = y' \Longrightarrow x \times y = x' \times y'$.

Hence, $X \boxtimes Y$ is a family of mutually disjoint categories of $A \times B$, that is, $X \boxtimes Y \in CL(A \times B)$. (Q.E.D.)

Proof of Lemma 10

1) $(x \times y,\ x' \times y') \in H \Longleftrightarrow (x,x') \in F \wedge (y,y') \in G$

$\Longleftrightarrow x \in X,\ x' \in X',\ y \in Y,\ y' \in Y',\ x \cap x' \neq \emptyset \wedge y \cap y' \neq \emptyset$

$\Longleftrightarrow x \times y \in X \boxtimes Y,\ x' \times y' \in X' \boxtimes Y',\ x \times y \cap x' \times y' \neq \emptyset$.

Hence, H is the connection relation of $X \boxtimes Y$ with $X' \boxtimes Y'$.

2) $\bigcup H^{-1}(x' \times y')$

$= \{ (a,b) \mid \exists (x \times y),\ (x \times y,\ x' \times y') \in H \wedge (a,b) \in x \times y \}$

$= \{ (a,b) \mid \exists x, \exists y,\ (x,x') \in F \wedge (y,y') \in G \wedge a \in x \wedge b \in y \}$

$= \{ (a,b) \mid a \in \bigcup F^{-1}(x') \wedge b \in \bigcup G^{-1}(y') \}$

$= \{ (a,b) \mid a \in x' \wedge b \in y' \}$ ($\because X \xrightarrow{F} X' \wedge Y \xrightarrow{G} Y'$ and (P3))

$= x' \times y'$.

Hence (P3) in Lemma 4 holds with respect to H, so $X \boxtimes Y \xrightarrow{H} X' \boxtimes Y'$. (Q.E.D.)

REFERENCES:

[Borůvka 1976] O.Borůvka, Foundations of the Theory of Groupoids and Groups, Birkhäuser Verlag Basel, Berlin, 1976.

[Bubenko+et al. 1976] J.A.Bubenko and et al., "From Information Requirements to DBTG-Data Structures," in [SIGPLAN+SIGMOD 1976], pp.73-85.

[Bubenko 1977] J.A.Bubenko Jr, "IAM: An Inferential Abstract Modeling Approach to Design of Conceptual Schema," in [SIGMOD 1977], pp.62-74.

[Carlson+Kaplan 1976] C.R.Carlson and R.S.Kaplan, "A Generalized Access Path Model and Its Application to a Relational Database System," in [SIGMOD 1976], pp.143-156.

[CODATA 1980] Proceedings of the 7th International CODATA Conference, Kyoto, Japan, Oct. 8-11, 1980, Pergamon Press, 1981 (to appear).

[Codd 1979] E.F.Codd, "Extending the Database Relational Model to Capture More Meaning," ACM Transactions on Database Systems, 4(4), Dec. 1979, pp.397-434.

[IFIP 1977] G.M.Nijssen (ed.), Architecture and Models in Data Base Management Systems, (Proceedings of the IFIP Working Conference on Modelling in Data Base Management Systems, Nice, France, Jan. 3-7, 1977), North-Holland, Amsterdam, 1977

[Lozinskii 1980] E.L.Lozinskii, "Construction of Relations in Relational Databases," ACM Transactions on Database Systems, 5(2), June 1980, pp.208-224.

[Nijssen 1977] G.M.Nijssen, "Current Issues in Conceptual Schema Concepts," in [IFIP 1977], pp.31-65.

[Sato 1980] H.Sato,"Derivability and Comparability among Non-Atomic Data," in [CODATA 1980].

[Sato 1981] H.Sato, "Handling Summary Information in Databases: Derivability and Comparability," (in preparation).

[Schenk+Pinkert 1977] K.L.Schenk and J.R.Pinkert, "An Algorithm for Servicing Multi-Relational Queries," in [SIGMOD 1977], pp.10-19.

[SIGMOD 1976] J.B.Rothnie (ed.), Proceedings of the ACM SIGMOD International Conference on Management of Data, Washington D.C., June 2-4, 1976.

[SIGMOD 1977] D.C.P.Smith (ed.), Proceedings of the ACM SIGMOD International Conference on Management of Data, Toronto, Canada, Aug. 3-5, 1977.

[SIGPLAN+SIGMOD 1976] Proceedings of the ACM SIGPLAN/SIGMOD International Joint Conference on Data: Abstraction, Definition and Structure, Salt Lake City, Utah, March 22-24, 1976.

[Smith+Smith 1977] J.M.Smith and D.C.P.Smith, "Database Abstractions: Aggregation and Generalization," ACM Transactions on Database Systems, 2(2), June 1977, pp.105-133.

[SSDS 1975] United Nations, Towards a System of Social and Demographic Statistics, ST/ESA/STAT/SER.F/18, U.N., New York, 1975.