No. 101 (81-2)

HANDLING SUMMARY INFORMATION IN A DATABASE: CATEGORIZATION AND SUMMARIZATION

by
Hideto Sato

January 1981

HANDLING SUMMARY INFORMATION IN A DATABASE: CATEGORIZATION AND SUMMARIZATION

Hideto Sato

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

ABSTRACT: "Categorization" is an abstraction which turns a group of atomic objects into a single object (category). "Summarization" is often accompanied with it. It maps attribute values of atomic objects to an attribute value of their category (summary). A schema of summary data is formalized by using these two notions. A database which collects summary data is considered. When an interrogator wants data under another categorization than that of collected data, it is needed to recategorize and resummarize the latter. It is shown that such resummarization is "compatible" with the direct summarization of atomic data if the summarizing operator is "associative."

1. Introduction

"Categorization" is an abstraction which turns a group of atomic objects into a single object called a "category." It is employed when one wish to take a wide view of things, neglecting differences among the things. "Summarization" is another type of abstraction which maps a group of values to a single value called a "summary." It is employed when one wish to obtain a brief account of things, finding a representation of the things. An important application of summarization is to attribute values of objects belonging to the same category. Such summarization is accompanied with categorization.

As for numerical information, total, average and maximum of attribute values are typical examples of summaries. As for non-numerical information, let us take an example such as a group of employees. Instead of listing all employees belonging to the group, at some time, a single employee who holds the highest

position in the group may represent the group. At another time, the number of the employees may represent the group. These are summaries of the group. Even in these cases, quantification such as ranking or counting is concerned, so that they can be treated in a similar manner to the case of numerical information.

Next, let us consider a database which deals with summary data. Since summary data have been compounded of atomic ones, any of them can be reproduced as far as the corresponding atomic ones are managed. However, in actual practices, many data are stored in databases just in the form of summaries, probably because of physical or economic reasons. Many examples of such summary databases are found in statistical fields.

Suppose some data are collected in the form of summaries in a database. We name the data in a database "collected data." Collected summary data concern certain categorization abstraction, while desirable categorization of objects may depend on a person's view. In effect, some interrogator may wish to obtain summary data under another categorization than that of collected data. We name such data "interrogator's data." Then the following questions may arise.

(1) Summarization is not reversible, that is, original atomic data cannot be derived from their summary. Summary data are incomplete data in this sense. Then, the judgement of derivability of interrogator's data from collected ones comes into question. How can it be judged ?

(2) When interrogator's data are judged to be derivable from collected ones, how can the derivation be carried out ?

(3) Whenever possible, an interrogator would want to obtain his

data by summarizing original atomic data directly. Hence, the following question is worth to ask. Is the derivation of interrogator's data from collected ones compatible with the direct summarization of original atomic data ? Or, do these two ways to obtain interrogator's data bring the same result ?

We presented the notion of categorization in [Sato 1980a] and [Sato 1980b], and discussed the questions of (1) and (2) in detail. However, the definition of categorization did not discriminate between a group of objects and its representation, and the notion of summarization was not formalized there. Consequently, the answer for (1) and (2) remained intuitive and the question (3) was left untouched. This paper gives a formal definition of categorization and summarization, and intends to answer the above three questions in a formal manner.

The notion of categorization has been referred as a part of "generalization" in [Smith+Smith 1977], as "correspondence" in [Santos+et al. 1979] and as "grouping/categorization" in [McLeod +King 1979]. However, they did not mention the derivation problem of interrogator's data. [Walker 1980] presented another type of derivation problem concerning categorization than that of ours. He discussed compatibility of relational operations on categorized data (abstraction-then-operation) with categorization of the results of relational operations on the corresponding atomic data (operation-then-abstraction).

On the other hand, the notion of summarization has been referred mainly in the context of "query languages" or "external views"; e.g. see [Shipman 1979] or [Bubenko+et al. 1976]. But they did not discuss summarization in connection with categoriza-

tion. Summarization with categorization was noticed in (CODASYL 1962), but it presented no futher than its definition.

2. Examples of Categorization and Summarization

In this section, we illustrate examples and present brief notions of categorization and summarization.

In Fig.1, R shows a relation R(EMPLOYEE,WAGES), which indicates wages paid to individual employees in an enterprise. Suppose a table f assigns each employee to a section he or she works for. Assume f is a function (an n to 1 mapping). Then the section in which each "wages" is paid is presented as in the table f(R) in the figure. The derivation of abstract data such as f(R) is called "categorization." The distinction among categorized objects, or the specification of employees belonging to the same section, is neglected through categorization.

However, categorization might not save much space for recording data. For example, f(R) in the figure has the same counts of tuples as in R in spite of the difference between their amounts of information. When one wish to obtain a brief account of information in this situation, another abstraction named "summarization" is often applied after categorization. In this example, it means transformation of f(R) to $\sum_f(R)$. The attribute values of objects, which are regarded as a single object after categorization, are mapped to a single attribute value through summarization. In this summarization, the summing operator has been employed as the "summarizing operator."

Summarization is not always accompanied with categorization. Fig.2 gives another example. Here wages in R' are determined by

R

EMPLOYEE	WAGES (IDX)
James	100 (01)
Smith	100 (02)
Adams	50 (03)
Jones	130 (04)
Parker	110 (05)

f

EMPLOYEE	SECTION
James	DP
Smith	DP
Adams	Account
Jones	Account
Parker	DP

f(R)

SECTION	WAGES (IDX)
DP	100 (01)
DP	100 (02)
Account	50 (03)
Account	130 (04)
DP	110 (05)

$\sum_f(R)$

SECTION	WAGES (IDX)
DP	310 (001)
Account	180 (002)

Note: Throughout this paper, numerical attributes are supposed to be indexed sets defined in Appendix.

Fig. 1 Categorization and Summarization

R'

EMPLOYEE	SECTION	WAGES (IDX)
James	DP	100 (01)
Smith	DP	100 (02)
Adams	Account	50 (03)
Jones	Account	130 (04)
Parker	DP	110 (05)
James	Account	40 (06)

$\sum(R';SECTION)$

SECTION	EMPLOYEE	WAGES (IDX)
DP	{James, Smith, Parker}	310 (001)
Account	{Adams, Jones, James}	220 (002)

Fig. 2 Summarization without Categorization

$\bar{R}$

EMPLOYEE	SKILL
James	programming
Smith	programming
Smith	typing
Adams	typing
Jones	typing
Jones	languages
Parker	typing

$f(\bar{R})$

SECTION	SKILL
DP	programming
DP	typing
Account	typing
Account	languages

$\sum_f(\bar{R})$

SECTION	SKILL
DP	{programming, typing}
Account	{typing, languages}

Note: f is the same as in Fig.1.

Fig. 3 Trivial Summarization of Non-Numerical Data

$\bar{R}'$

EMPLOYEE	SKILL	SIZE (IDX)
James	programming	1 (01)
Smith	programming	1 (02)
Smith	typing	1 (03)
Adams	typing	1 (04)
Jones	typing	1 (05)
Jones	languages	1 (06)
Parker	typing	1 (07)

$\sum_{f'}(\bar{R}')$

SECTION	SKILL	SIZE (IDX)
DP	programming	2 (001)
DP	typing	2 (002)
Account	typing	2 (003)
Account	languages	1 (004)

Note: f'=(f,id) where id is the identity function.

Fig. 4 Quantification of Non-Numerical Data and Summarization

g

EMPLOYEE	DEPARTMENT
James	Management
Smith	Management
Adams	Management
Jones	Management
Parker	Management

h

SECTION	DEPARTMENT
DP	Management
Account	Management

$\sum_g(R) = \sum_h(\sum_f(R))$

DEPARTMENT	WAGES (IDX)
Management	490 (0001)

Note: R and f are the same as in Fig.1.

Fig. 5 Recategorization and Resummarization

employee by section. In this case, we can also consider similar transformation with the previous case. Summarization of R' with respect to SECTION gives a summary table such as $\sum$(R';SECTION) in the figure.

Next, let us consider summarization of non-numerical information. If the summarizing operation means grouping of non-numerical values as in Fig.3, the summarization simply rearranges data in 1NF (f($\bar{R}$)) into data in non-1NF ($\sum_f(\bar{R})$). Both the data show what kinds of skill are provided in each section. This type of abstract information is often demanded to serve a top-decision such as to assign a certain section for a special type of business. However, set theoretical summarization does not discriminate one employee with a particular skill from a hundred of such employees. The loss of information is too great. In order to improve this situation, quantification of non-numerical information is often employed and then another type of summary table such as $\sum_{f'}(\bar{R}')$ in Fig.4 is compiled. This is a summary table which is obtained from $\bar{R}'$ in the figure through categorization and summarization with respect to (SECTION, SKILL), where $\bar{R}'$ is a quantified table of $\bar{R}$.

Finally, let us consider recategorization and resummarization. Assume summary data, such as $\sum_f(R)$ in Fig.1, are stored in a database. Suppose an interrogator wants another summary data such as $\sum_g(R)$ in Fig.5. The latter can be obtained from the former if a functional dependency SECTION ----> DEPARTMENT exists. Given a function h: SECTION ----> DEPARTMENT as in Fig.5, we can see the direct summarization of the original atomic data R under g ($\sum_g(R)$) is compatible with the resummarization of

the collected data $\sum_f(R)$ under h $(\sum_h(\sum_f(R)))$.

3. Categorization

3-1 Notions concerning Categorization

An element in a set is mapped, through categorization, to a category in which the element is categorized. Such a mapping is called a classification rule in this paper. The domain of this mapping is called an atomic set and the range is called a classification on the atomic set. Thus, each element in a classification is a category. We assume a classification rule is a function. Then the inverse image of a category under the function is a subset of the atomic set. The class of such inverse images forms a partition of the atomic set. A partition of A means a class of mutually disjoint subsets of A whose union is A. We distinguish a partition from a classification; the former means a substantial result of categorization (the groups of elements in the atomic set themselves) and the latter means a set of names each of which represents each group in the former. One may consider another type of categorization, through which a category in a classification is mapped to another category in another classification. This type of categorization corresponds with so-called reclassification from a minor (finer) classification to a major (coarser) classification. Such a mapping is called a reclassification rule in this paper. The notions of a classification rule and a reclassification rule are similar with $\in$-generalization and $\subset$-generalization in [Codd 1979], respectively.

A similar function as a reclassification rule was named an "abstractor" in [Walker 1980] and it is defined simply as an

arbitrary function there. However this definition may produce semantic ambiguity when two or more such functions are under consideration. For example, let X and Y be entity sets. We can suppose two or more functions from X to Y at the same time. If arbitrary functions can be abstractors, an entity in Y may represent more than one group of entities in X, simultaneously. This interpretation may result confusion. In order to avoid such ambiguity, we restrict reclassification rules to those that are consistent with a given classification hierarchy. The definition of notions introduced here are formalized in the next sub-section.

3-2 Definition on Categorization

Def. 1 Composition in a Class of Functions

Let S be a class of sets and F be a class of functions defined in S (or, each function in F is defined between a pair of sets in S). Suppose $X,Y \in S$. If a function $f: X \dashrightarrow Y$ satisfies the following condition, f is called to be composed in F, or to be a composition from X to Y in F.

(P1) $\exists f_1, \exists f_2, \ldots, \exists f_n \in F, \quad f = f_1 \circ f_2 \circ \ldots \circ f_n$.

The class of all compositions from X to Y in F is denoted by $CMP\langle X,F,Y\rangle$.

Def. 2 Classification

Let A be a set and S be a class of sets. Assume $A \in S$. Let F be a class of surjections defined in S. Assume F contains the identity functions 1_X $(X \in S)$. If F satisfies the following conditions, the triple $\langle A,F,S\rangle$ can be a classification hierarchy.

(P2) $\forall X \in S, \ CMP\langle A,F,X\rangle \neq \emptyset$,

(P3) $\forall X \in S$, $(u,u' \in CMP\langle A,F,X\rangle \Longrightarrow u=u')$.

Suppose $\langle A,F,S\rangle$ is a classification hierarchy. Then A is called an atomic set, each X in S a classification on A, and each x in X a category of A. S, the set of classifications on A, is denoted by CL(A). Thus, "$X \in CL(A)$" means "X is a classification on A." $u \in CMP\langle A,F,X\rangle$ is called the classification rule from A to X, denoted by $A \>\!\!-\!\!-u\!\!\rightarrow X$.

The conditions (P2) and (P3) insist that there exists a unique classification rule for each classification. This uniqueness ensures that, for each classification X on an atomic set A, a partition P of A is uniquely determined as the class of inverse images under the classification rule u from A to X, i.e. $P=u^{-1}\langle X\rangle = \{u^{-1}(x) \mid x \in X\}$. This partition P is regarded as the substantial result of categorization under u. We call this the partition of A associated with X. In addition, notice that A itself is also a classification on A in this definition and 1_A is the classification rule from A to A.

Def. 3 Reclassification Rule

Suppose $A \>\!\!-\!\!-u\!\!\rightarrow X$ and $A \>\!\!-\!\!-v\!\!\rightarrow Y$. If there exists a surjection $f: X \dashrightarrow Y$, such that $v=f\circ u$, then f is called the reclassification rule from X to Y, denoted by $X \dashrightarrow Y$.

The following proposition gives the substantial meaning of a reclassification rule.

Prop. 4 Suppose $A \>\!\!-\!\!-u\!\!\rightarrow X$ and $A \>\!\!-\!\!-v\!\!\rightarrow Y$. Assume $\exists f, X \dashrightarrow Y$. Then the partition of A associated with X is a refinement of that with Y, that is, $u^{-1}(x) \cap v^{-1}(y) \neq \emptyset \Longrightarrow u^{-1}(x) \subset v^{-1}(y)$.

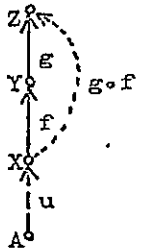
(Proof) $a \in u^{-1}(x)$, $u^{-1}(x) \cap v^{-1}(y) \neq \emptyset \Longrightarrow \exists a'$, $a,a' \in u^{-1}(x)$, $a' \in v^{-1}(y)$

$\Longrightarrow \exists a', \exists y'$, $a,a' \in (f\circ u)^{-1}(y')$ $(=v^{-1}(y'))$, $a' \in v^{-1}(y) \Longrightarrow$

$a \in v^{-1}(y)$ ($\because a' \in v^{-1}(y')$, $a' \in v^{-1}(y) \Rightarrow y'=y$).

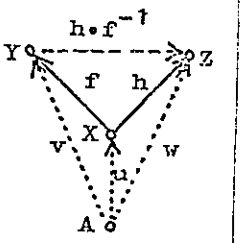
This means $u^{-1}(x) \cap v^{-1}(y) \neq \emptyset \Rightarrow u^{-1}(x) \subset v^{-1}(y)$. □

As to reclassification rule, the following propositions hold. Here we suppose $X,Y,Z \in CL(A)$.



Prop. 5 $X \text{ --}f\text{-}\!> Y$, $Y \text{ --}g\text{-}\!> Z \Rightarrow X \text{ --}g\circ f\text{-}\!> Z$.

(Proof) Suppose $A \text{ >--}u\text{-}\!> X$. Assume $X \text{ --}f\text{-}\!> Y$ and $Y \text{ --}g\text{-}\!> Z$. Then $A \text{ >--}f\circ u\text{-}\!> Y$, and then $A \text{ >--}g\circ(f\circ u)\text{-}\!> Z$. Since $g\circ(f\circ u)=(g\circ f)\circ u$, $g\circ f$ is the reclassification rule. □



Prop. 6 $X \text{ --}f\text{-}\!> Y$, $X \text{ --}h\text{-}\!> Z$, then

(1) $\exists g$, $Y \text{ --}g\text{-}\!> Z \Rightarrow g=h\circ f^{-1}$,

(2) $Y \text{ --}h\circ f^{-1}\text{-}\!> Z \Leftrightarrow h=h\circ f^{-1}\circ f$.

(Proof) (1) $Y \text{ --}g\text{-}\!> Z \Rightarrow h=g\circ f \Rightarrow h\circ f^{-1}=g\circ f\circ f^{-1}=g$.

Notice that "f is a surjection $\Rightarrow f\circ f^{-1}=1_Y$ (the identity function)."

($\because$) 1) $(f\circ f^{-1})(Y)=f(f^{-1}(Y))=f(X)=Y$,

2) $(y,y') \in f\circ f^{-1} \Rightarrow \exists x,\ y=f(x),\ y'=f(x) \Rightarrow y=y'$.

(2) ($\Rightarrow$) is trivial from Prop.5. ($\Leftarrow$) $h=h\circ f^{-1}\circ f \Rightarrow h\circ f^{-1}$ is a surjection. ($\because$) 1) $(h\circ f^{-1})(Y)=Z$, $(h\circ f^{-1})^{-1}(Z)=Y$,

2) $(y,z),(y,z') \in h\circ f^{-1} \Rightarrow \exists x,\ (x,y) \in f,\ (y,z),(y,z') \in h\circ f^{-1}$ $\Rightarrow (x,z),(x,z') \in h\circ f^{-1}\circ f\ (=h) \Rightarrow z=z'$.

Next, suppose $A \text{ >--}u\text{-}\!> X$, $A \text{ >--}v\text{-}\!> Y$, $A \text{ >--}w\text{-}\!> Z$. Then, $h=h\circ f^{-1}\circ f \Longrightarrow w=h\circ u=h\circ f^{-1}\circ f\circ u=(h\circ f^{-1})\circ(f\circ u)=(h\circ f^{-1})\circ v$.

Hence, $Y \text{ --}h\circ f^{-1}\text{-}\!> Z$. □

Prop. 7 Let $\langle A,F,S\rangle$ be a classification hierarchy. Suppose $f \in F$ and $f: X \dashrightarrow Y$. Then $X \text{ --}f\text{-}\!> Y$.

(Proof) Suppose $A \text{ >--}u\text{-}\!> X$, $A \text{ >--}v\text{-}\!> Y$. Assume $f \in F$. Then $f\circ u=v$ should hold from the definition of the classification

hierarchy. This indicates f is a reclassification rule. □

Notice that reclassification rules are not necessarily restricted to those defined explicitly in a classification hierarchy. Some of them may be inferred by using Prop.5 and/or Prop.6.

Lemma 8 Let $\langle A,F,S\rangle$ be a classification hierarchy. Suppose $X,Y \in S$. Then, "$\exists f,\ X \text{ -- } f \rightarrow Y$" can be checked only by applying Prop.5 and Prop.6 to the reclassification rules explicitly defined in F. In addition, if such f exists, f can be obtained by applying these propositions to the reclassification rules in F.

(Proof) Let $B \in S$ such that $CMP\langle B,F,X\rangle \neq \emptyset$ and $CMP\langle B,F,Y\rangle \neq \emptyset$. For any X and Y in S, such B always exists since at least A satisfies these conditions. From Prop.7, f' and g', such that $B \text{ -- } f' \rightarrow X$ and $B \text{ -- } g' \rightarrow Y$, can be composed in F by repeating application of Prop.5. Then, by using Prop.6, "$\exists f,\ X \text{ -- } f \rightarrow Y$" can be checked and f can be obtained as $g' \cdot f'^{-1}$ if such f exists. □

3-3 Categorization of Relational Data

Def. 9 Relation

A set of values (or names) is called an entity set. The cartesian product of entity sets is called a compound entity set. A subset of a compound entity set is called a relation.

Note Let $X_1, X_2, \ldots, X_n$ be entity sets and $I=\{1,2,\ldots,n\}$. Then $X = (X_i \mid i \in I) = X_1 \times X_2 \times \ldots \times X_n$ is a compound entity set. A relation R in X, denoted by R(X), satisfies $R(X) \subset X = (X_i \mid i \in I)$. Note that $(X_i \mid i \in \{1\}) = X_1$ and if $I \cap J = \emptyset$ then

$(X_i|i\in I\cup J)=(X_i|i\in I)x(X_i|i\in J)=((X_i|i\in I),(X_i|i\in J))$.

Hence we also denote a relation R in X x Y as R(X,Y) where X and Y are either entity sets or compound entity sets.

Then categorization of relational data is defined as follows:

Def. 10 Categorization of Relational Data

Let R(X,Y) be a relation. Suppose $X,X'\in CL(A)$ and $X \text{ --f-> } X'$. Then, the categorized relation of R under f is defined as:

(P4) $f[R] = \{(x',y)|\exists x,\ (x,y)\in R,\ x'=f(x)\}$.

The process of categorization can be implemented by means of the join operation and the projection operation as shown in [Walker 1980] and [Sato 1980].

4. Summarization

4-1 Definition of Summarization

A group of values is mapped to its summary through summarization. Usually, the domain of summary coincides with the domain of values to be summarized. Such mapping is given as an operator defined as follows:

Def. 11 Summarization

Let D be a set and $\sum$ be a function such that $\sum: 2^D \text{ ---> } D$, where 2^D is the power set of D. Then the value of $\sum$ at d ($\subset$ D), denoted by $\sum d$, is called the summary of d by $\sum$. $\sum$ is called the summarizing operator over D and D the domain of summarization by $\sum$.

Examples

(1) sum D: integers $\sum\{1,3,10\}=14$.

(2) union D: a power set $\sum\{\{a,b\},\{b,c\},\{d\}\}=\{a,b,c,d\}$.

(3) median D: real numbers $\sum\{1.0,3.0,6.0\}=3.0$.

A special type of summarizing operator, so-called "associative," is of great importance to practical applications. This property indicates giving an equivalent result when the order, in which the operator is applied, is changed. Formally:

Def. 12 Associative Summarizing Operator

Let $\{x_i | i \in I\}$ where $I = \bigcup_{j \in J} I(j)$ be an arbitrary subset of an indexed set D. Let Σ be a summarizing operator over D. If Σ always satisfies the following condition, then Σ is said to be associative.

(P5) $\Sigma\{\Sigma\{x_i | i \in I(j)\} | j \in J\} = \Sigma\{x_i | i \in I\}$ where $I = \bigcup_{j \in J} I(j)$.

In the above examples, (1) and (2) are associative, but not (3). The ordinary notations of (P5) for (1) and (2) are:

(1) $\sum_{j \in J} \sum_{i \in I(j)} x_i = \sum_{i \in I} x_i$,

(2) $\bigcup_{j \in J} \bigcup_{i \in I(j)} X_i = \bigcup_{i \in I} X_i$, where $I = \bigcup_{j \in J} I(j)$.

4-2 Summarization of Relational Data

Summarization of a relation means to extract such tuples as judged to concern the same object from a certain point of view, and to summarize them into one tuple. When such a point of view is given as the sameness of values in a particular entity set (or a particular compound entity set) participating in the relation, the summarization of the relation is defined as follows:

Def. 13 Summarization of Relational Data

Given R(X,Y) and a summarizing operator Σ, the summary of R with respect to X by Σ is defined as:

(P6) $\Sigma[R;X] = \{(x,y') | y' = \Sigma\{y | (x,y) \in R\}\}$.

X is called the key of this summarization.

Note Since $\Sigma[R;X]$ is an n to 1 mapping from X to Y, (P6) can be

rewritten as:

(P6') $\forall x \in X, \sum(R;X)(x) = \sum\{y|(x,y)\in R\}$.

As mentioned in Section 1, summarization is often accompanied with categorization. Such summarization is more general, because the identity function can be regarded as a reclassification rule and hence summarization without categorization can be considered as summarization with categorization under the identity function. Summarization with categorization can be defined as in the following definition with reference to Def.10 and Def.13.

Def. 14 Summarization of Relational Data with Categorization

Let $X,X' \in CL(A)$. Suppose $X \xrightarrow{f} X'$. Given $R(X,Y)$ and a summarizing operator $\sum$, the summary of R by $\sum$ under f is defined as:

(P7) $\sum_f(R) = \sum(f(R);X')$

$= \{(x',y')|y' = \sum\{y|\exists x, (x,y)\in R, x'=f(x)\}\}$.

Note For a similar reason to (P6), (P7) can be rewritten as:

(P7') $\forall x' \in X, \sum_f(R)(x') = \sum\{y|\exists x, (x,y)\in R, x'=f(x)\}$.

A similar transformation of data is defined in [CODASYL 1962] and is called a "function of glumps."

5. Recategorization and Resummarization

5-1 Lemma of Recategorization and Resummarization

Using the notions prepared in the previous sections, let us consider the compatibility of recategorization and resummarization with direct categorization and direct summarization. In this section, we assume $\sum$ is an associative summarizing operator.

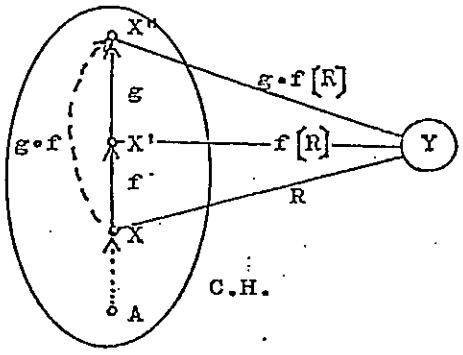
Lemma 15 Let R(X,Y) be a relation.

Suppose $X,X',X'' \in CL(A)$ and

$X \xrightarrow{f} X'$, $X' \xrightarrow{g} X''$. Then

(1) $g[f[R]] = g \circ f[R]$,

(2) $\sum_g[\sum_f[R]] = \sum_{g \circ f}[R]$.



(Proof)

(1) $g[f[R]] = \{(x'',y) \mid \exists x',\ (x',y) \in f[R],\ x''=g(x')\}$

$= \{(x'',y) \mid \exists x', \exists x,\ (x,y) \in R,\ x'=f(x),\ x''=g(x')\}$

$= \{(x'',y) \mid \exists x,\ (x,y) \in R,\ x''=g \circ f(x)\}$

$= g \circ f[R]$ ($\because$ $g \circ f$ is a reclassification rule from Prop.5).

(2) $\sum_g[\sum_f[R]](x'')$

$= \sum \{y' \mid \exists x',\ (x',y') \in \sum_f[R] \quad ,\ x''=g(x')\}$

$= \sum \{\sum_f[R](x') \mid x''=g(x')\}$

$= \sum \{\sum \{y \mid \exists x,\ (x,y) \in R,\ x'=f(x)\} \mid x''=g(x')\}$

$= \sum \{\sum \{y \mid y \in R(f^{-1}(x')) \mid x' \in g^{-1}(x'')\}\}$

$= \sum \{y \mid y \in \bigcup_{x' \in g^{-1}(x'')} R(f^{-1}(x'))\}$ ($\because \sum$ is associative)

$= \sum \{y \mid \exists x' \in g^{-1}(x''),\ y \in R(f^{-1}(x'))\}$

$= \sum \{y \mid \exists x,\ (x,y) \in R,\ x''=g \circ f(x)\} = \sum_{g \circ f}[R](x'')$. □

5-2 Derivation of Summary Data in a Database

Lemma 8 and Lemma 15 afford a foundation of a database management system which deals with summary data. A database consists of two parts; one is a class of classification hierarchies and another is a class of relations including summary data. We name the schema of summary data in a database a collector's schema. It must clarify what kind of atomic data concerns the summary data and what type of categorization and summarization has been applied to the atomic data. They can be specified by a relational schema RA(A,V) of the original atomic

data (an atomic data schema), a reclassification rule r(A,C) from A to C where C is the collector's classification on A, and a summarizing operator Σ. Thus, the collector's schema can be denoted as "$\Sigma_{r(A,C)}$[RA(A,V)]." Notice that RA(A,V) is not supposed to be stored in the database and r(A,C) is not needed to be explicitly defined in a classification hierarchy in the database. However, the actual summary data corresponding to the collector's schema should coincide with the result of operations expressed in the schema if the atomic data and the reclassification rule were materialized. As mentioned in Section 1, an interrogator may wish to obtain summary data under another categorization. Let us call his schema an interrogator's schema, and use a similar notation with collector's one for its representation. Suppose "$\Sigma_{r(A,I)}$[RA(A,V)]" is an interrogator's schema where I is the interrogator's classification on A. Hereinafter we suppose RC(C,V)=$\Sigma_{r(A,C)}$[RA(A,V)] and RI(I,V)=$\Sigma_{r(A,I)}$[RA(A,V)]. Let us assume C and I are actually contained in a classification hierarchy in the database whose atomic set is A. Fig.6 illustrates the situation above.

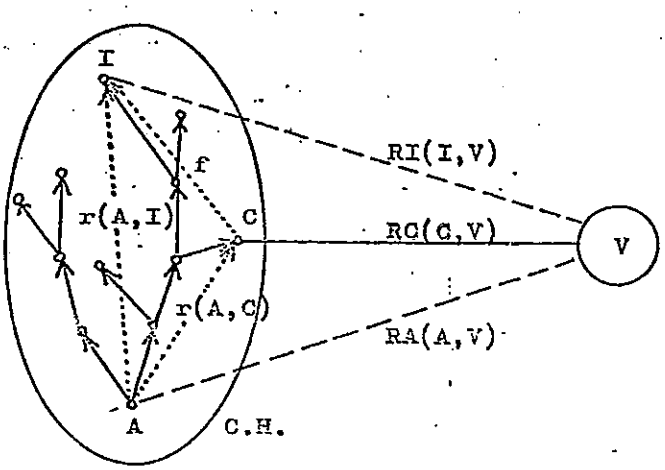


Fig. 6 Derivation of Summary Data

In the situation in Fig.6, Lemma 8 can be applied and we can judge whether a reclassification rule exists from C to I or not. If such a rule exists we can obtain it according to the lemma. Let f be such a rule. Then the interrogator's data RI can be derived from the collector's data RC in such a manner as $RI=\sum_f(RC)$, because, by using Lemma 15 and Prop.5,

$$RI=\sum_{r(A,I)}(RA)=\sum_{f\cdot r(A,C)}(RA)=\sum_f(\sum_{r(A,C)}(RA))=\sum_f(RC).$$

Namely, RI is obtained by summarizing RC under f. In addition, the above transformation also shows that this summarization is compatible with the direct summarization of the atomic data.

Notice that the derivation of RI from RC does not require explicit knowledge about the reclassification rules, such as r(A,I) and r(A,C). The role of them played in the schema expressions is merely to notify the database management system (and the users of the database) of such specification that "A is an entity set (or a compound entity set) to be categorized within entity sets participating in the atomic relation RA and I (or C) is the resultant classification of the categorization."

6. Conclusion and Acknowledgment

A schema of summary data is specified by the original atomic data schema, the specification of the atomic set and its classification concerning categorization, and the summarizing operator. As far as the summarizing operator is associative, the questions raised in Section 1 are answered as:

(1) The derivability of interrogator's data from collected one can be judged from the connection of the interrogator's classification with the collector's one in a classification

hierarchy.

(2) The resummarization with recategorization, needed to derive interrogator's data, are assigned by the reclassification rule inferred from the above connection.

(3) The resummarization assigned in (2) is compatible with the direct summarization of the atomic data.

The author is very grateful to Professor Ryosuke Hotaka for helpful discussions on this subject.

APPENDIX Indexed Set

An indexed set is a set, each element of which has an index (or name) independent to the content expressed by the element. We denote an indexed set X with an index set I as $X=\{x_i|i\in I\}$. We assume for any indexed set that $i=j \Longrightarrow x_i=x_j$ but the converse is not necessarily true. $i=j$ indicates the sameness of elements x_i and x_j, and $x_i=x_j$ means equivalence of the contents expressed by the elements. In order to apply the ordinary set theory to this type of set, let us define set operations as:

Given $X=\{x_i|i\in I\}$ and $Y=\{x_i|i\in J\}$,

(1) $X\cup Y = \{x_i|i\in I\cup J\}$, (2) $X\cap Y = \{x_i|i\in I\cap J\}$.

Indexed sets are employed for expressing sets of numerical values in this paper. In this case, we assume that the indices are unchanged through set operations but they are reset after summarization, as shown in Section 2.

REFERENCES:

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

[CODASYL 1962] Language Structure Group of the CODASYL Development Committee, "An Information Algebra Phase I Report," Communications of ACM, 5(4), 1962, pp.190-204.

[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.

[Entity-Relationship 1979] P.P.Chen (ed.), Proceedings of the International Conference on Entity-Relationship Approach to Systems Analysis and Design, Los Angeles, Dec. 10-12, 1979.

[McLeod+King 1979] D.McLeod and R.King, "Applying A Semantic Database Model," in [Entity-Relationship 1979], pp.172-189.

[Santos+et al. 1979] C.S.dos Santos, E.J.Neuhold, U.Stuttgart, and A.L.Furtado, "A Data Type Approach to the Entity-Relationship Model," in [Entity-Relationship 1979], pp.114-130.

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

[Sato 1980b] H.Sato, "Handling Summary Information in a Database: Derivability," Institute of Socio-Economic Planning, DP-98, The University of Tsukuba, 1980.

[Shipman 1979] D.Shipman, "The Functional Data Model and the Data Language DAPLEX," ACM Transactions on Database Systems, (to appear).

[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.

[VLDB 1980] Proceedings of the Sixth International Conference on Very Large Data Bases, Montreal, Canada, Oct. 1-3, 1980.

[Walker 1980] A.Walker, "On Retrieval from a Small Version of a Large Data Base," in [VLDB 1980], pp.47-54.