M. Rafanelli, A. Shoshani

A MODEL FOR THE GRAPHICAL REPRESENTATION OF AGGREGATE DATA

R. 348   Dicembre 1992

Maurizio Rafanelli - Istituto di Analisi dei Sistemi ed Informatica, del CNR, Viale Manzoni, 30, 00185 Roma, Italy
Tel. (+ 39 6) 7716437 Fax (+396) 7716461
e-mail rafanelli@irmiasi.bitnet
Arie Shoshani - Lawrence Berkeley Laboratory - University of California, 1 Cyclotron Road, Berkeley, CA 94720
Tel. + 1 510 486 9171 Fax + 1 510 486 6363
e-mail arie%csr.lbl.gov@lbl.bitnet

This work has been partially supported by the National Special Project of the C.N.R. "Statistical Databases"

Abstract

In this paper the structure and the semantic properties of the entities stored in databases, whose data are only aggregate-type data, are defined and discussed. This choice is justified by the widespread use of aggregated data without the corresponding raw data (i.e. micro-data, such as census data). Aggregated data are often derived by applying statistical aggregation (e.g. *sum, count)* and statistical analysis functions over micro-data, so that the relative databases are called *statistical databases.* For this reason in this paper such entities are called *statistical object* and a new *statistical object representation model (Storm)* based on a graph representation is proposed and discussed, as well as the canonical form of a statistical object. Also the well-formedness of a statistical object, that is its completeness and fullness of mapping and its summarizability, is discussed. Finally a brief conclusions and possible future work is given.

Key Words. Statistical database, conceptual model, aggregate data model.

1. Introduction

For the last several years, a number of researchers have been interested in the various problems which arise when modelling aggregate-type data or *macro data* [1SD81, 2SD83, 3SS86, RKS89, Mic90, Hin92]. These data are often obtained by applying statistical aggregation (e.g. *sum, count*) and statistical analysis functions over disaggregate-type data or *micro-data* [Won84]. They are often necessary for pratically, required by regulation, or desirable as a means of concentrating attention on relevant information. *Statistical data* are generally defined as data on which statistical functions are applied, so that both micro data and macro data fall within this category [RF92].

In this paper we will consider only aggregate-type data, a choice which is justified by the widespread use of aggregated data only, i.e. without the corresponding micro-data. The reason is that often it is too difficult to use the micro data directly (both in terms of storage space and computation time) and because of reasons of privacy (especially when the user is not the data owner) [RS90a] or more effective elaboration. Moreover, statistical macro data are the best or only alternative when there are resource contraints closely linked with physical limitations of capacity, power and storage, legal concerns required by law or by policy, and a narrowly defined purpose for the system.

We will call the aggregated data bases *statistical databases* (SDBs), similarly to previous papers [Sho82, SW85, Raf89, Mic91], in order to distinguish the different data structure, the different representation problems, the different operators for their manipulation, etc. .

In the Statistical Databases the entities stored are complex data structures (vectors, matrixes, relations, time series, etc., which are more complex than the conventional disaggregated data), which may have different possible ways of representation (e.g. tables, relations, vectors, pies, bar-charts, graphs, and so on). In this paper these complex structures will be called *statistical object* (SO) [RS90a], so as to stress these many possible configurations.

Each statistical object is characterized by having two different types of attribute: (a) one *summary attribute* (that is, the result of the application of aggregative functions on micro data), with an own *summary type*, depending on the type of function applied to the micro data, and whose instances, which consist of numeric values, are called *summary data*, and (b) a set of *category attributes*, which described the summary attribute. The former is often called *quantitative variable* and the latter *qualitative variable.*

The phenomenon described by it has always its *universe of definition*, generally expressed by means of its name (for example, "Fruit production in California"). Moreover, the summary data are always fixed in time (or *static* as they are often called). This means that, for example, the production of fruit in California in the years 1981, 1982,.., 1988 is quantified by a numeric value which does not change in time (also if new value can be added over time).

Various previous papers have dealt with the problem of how to logically represent an aggregated data reality (e.g. [CS81, RR83, Su83, OOM85]). Starting from those works, this paper will propose a new STatistical Object Representation Model (Storm) [RS90a, RS90b, SR91, RS92], based on a graph representation.

In the subsequent sections, we give some necessary definitions and discuss problems with current logical models

Then we present the Storm model, the SO data structure, the spaces and the levels of representation, the SO graphical representation (by different types of nodes and edges), and introduce a semantics for each edge (assignment, classification by, grouping and partitioning). Finally, we discuss some properties which characterize an SO, as well as some particular situations and the consequent need labeled edges.

The definition of the Storm model is followed by the definition of its *canonical form* and of classes of rules for the generation of this canonical form.

Again, the problem of the null values are discussed, and an investigation of a well-formed statistical object, developing concept of and conditions for its *summarizability*, which guarantees correct results of summary operations over statistical objects, is presented.

Finally a brief conclusion and future researches in this area are given.

2. Basic Concepts of Statistical Databases

This section starts by briefly presenting four basic concepts that are unique to SDBs [RS90a]:

1. Summary attributes - these are attributes that describe the quantitative data being measured or summarized (e.g. "income for socio-economic databases", or "production and consumption of energy data").

2. Category attributes - these are attributes that characterize the summary attributes (e.g. "Sex", "Race" and "Age" characterize "Population counts").

3. Multi-dimensionality - typically a multidimensional space defined by the category attributes is associated with a single summary attribute (for example, the three-dimensional space defined by "State", "Race" and "Year" can be associated with "Population". The implication is that a combination of values from "State", "Race" and "Year" (e.g. Alabama, Black, 1989) is necessary to characterize a single population value (e.g., 21,373)).

4. Classification hierarchies - a classification relationship often exists between categories. For example specific "Products" (e.g. "Fruits", "Vegetables", "Grains" can be classified as "Agricultural Products").

These basic concepts are addressed in different models currently used to describe statistical data by employing essentially two methodologies: a) 2-

dimensional tabular representation, and b) graph-oriented representation. This paper explores below some of the problems encountered using these methodologies in current models.

In this paper, we define a representation model which is independent from the above methodologies. As a consequence, a statistical object can then have a graphical representation, a 2-dimensional tabular representation, or any other representation preferred by the user (e.g. a "relation").

3. Problems with Current Logical Models

In this section two kinds of problems will be briefly discussed: the two-dimensional (2D) table representation and the current graph-oriented models.

3.1 Problems with two-dimentional tabular representation.

The 2D table representation exists historically because statistical data have been presented on paper. This representation, although it continues to be practiced by statisticians today, changes the semantic concepts discussed above. In particular this paper points out below several deficiencies.

3.1.1 The concept of multi-dimensionality is distorted

By necessity, suppose you need to squeeze the multi-dimensional space into two dimensions (action typically done by choosing several of the dimensions to be represented as rows and several as columns).

For example, suppose you have to represent the "Average Income in California" by "Professional categories", "Sex" and "Year" and "Profes-sional categories" are further classified into "Profession".

Figure 1 is an example of a 2D tabular representation, but, obviously, one can choose (according to some other preferred criteria) other combinations by exchanging the dimensions.

Models using this tabular representation technique improperly consider the different tables to be different statistical objects, while in reality only the 2D representation has changed.

In general, the 2D representation of a multi-dimensional statistical object forces a (possibly arbitrary) choice of two hierarchies for the rows and columns.

The apparent conclusion is that a proper model should retain the concept of multi-dimensionality and represent it explicitly [RS90a].

3.1.2 The concept of classification relationship is lost

In the 2D representation, classification hierarchies are represented in the same manner as the multi-dimensional categories, while in the former the dimention is only one.

Consider, for example, "Professional Categories" and "Professions" shown in Figure 1.

It is obvious from this example that the values of average income are given for specific combinations of "Sex", "Year" and "Profession" only. Thus, Professional Category is not part of the multi-dimensional space of this statistical object, but part of a "classification hierarchy" which includes "Professional Categories" and "Professions".

This means that there is a fundamental difference between category relationship and multi-dimensionality.

Usually, only the low-level elements of the classification relationship participate in the multi-dimensional space. This fundamental difference should be explicitly represented in a semantically correct statistical data model.

3.1.3 Lack of metadata level

At present a 2D representation represents both the category names and the category instances together, that is there is no separate description of what the statistical database is about (meta-data).

Figure 1

For example, the meta-data for Figure 1 consists only of "Employment in California" by "Sex", "Year" and "Profession", however "Professional Category" is represented together with the data values. Consequently, this representation becomes very large for tables with high dimensionality, or when the categories have a large number of instances and, then, not comfortably fit on a page or a screen. In such cases, the representation spreads into multiple pages or screens. This confuses the global understanding of the statistical object.

It is therefore desirable to separate the representation of the categories and the category instances in order to achieve compactness of the semantic description of the database.

3.1.4 Problems with relational table representation

In a relational table representation no distinction between category and summary attribute is made.

Moreover, the classification concept is impractical and large redundencies of category attribute values are present, as shown in Figure 2, in which a comparison between two different ways to represent the same information by a table and a relation is shown, so that a high fanout produces unmanageable tables.

Another important element of distinction refers to the summary type of the aggregated data; this fact, together the different way to consider category and summary attributes, reflects itself also on the data manipulation [RR91].

3.2 Problems with current graph-oriented models

An attempt to correct some of the deficiencies of the 2D representation discussed above was made by introducing graph-oriented models. In these models the concepts of multi-dimensionality and classification hierarchies were introduced by having especially designated nodes.

For example, in GRASS [RR83] multi-dimensionality is represented by A-nodes (A stands for association) and C-nodes (C stands for classification), while in SUBJECT [CS81] it is represented by X-nodes (X stands for "cross product") and C-nodes (C stands for "clustering"). Thus, the statistical object of Figure 1 would be represented, for example, in GRASS as shown in Figure 3-a (note that the node of the type T represents a summary attribute) and in SUBJECT as shown in Figure 3-b (both X or C nodes can represent category and summary .

In SAM* [Su83] a semantic association model is proposed. In it a table is represented graphically by a graph and extensionally by a generalized relation, or G-relation, called in this way because the relation is enlarged to two different types of attributes (category and summary), separated by a double line. The problem of large redundency is not solved.

Figure 2

All these models do not solve many other problems discussed in the following.

3.2.1 Mixing categories and category instances

Consider again the classification hierarchy "Professional Category"-"Profession" of Figure 3-a and the intermediate node "Engineer". It has a dual function: on the one hand, it is an instance of the "Professional Category", on the other hand, it serves as the name of a category that contains "Chemical Engineer", "Civil Engineer", etc.

Figure 3

Note that the category "Profession" is missing in this representation. The reason is that after having expanded the first level ("Professional Category") into its instances, all the next levels can contain only instances. In this situation a high fanout produces large complex graphs.

Another consequence of this representation is similar to the problem of large 2D tables mentioned in section 3.1.3 . Here too, a large number of instances of categories produces large graphs that do not fit easily onto a page or a screen.

For the above reasons, the proposed model separates the categories and their instances into two different spaces, called "intentional" and "extensional".

For example, the statistical object of Figure 3-a will be represented at the meta-data level (intentional space) as shown in Figure 4 (the different types of edges will be discussed in the following).

Underlying this representation, the system stores and maintains the instances and their relationship.

The instances (extensional space) will be represented into another space, as it will be see later; in this space also the eventual classification hierarchies of the intentional space will be reported.

3.3 Problems with common reference to categories

It is often necessary to combine information from multiple statistical objects in order to correlate information or to obtain new statistical objects.

Figure 4

For example, suppose that we have two statistical objects on cancer incidence and pollution level, as shown in Figure 5, and that we wish to find the correlation between them. To achieve this we first need to bring the two SOs into common categories, so that they are comparable. In this case, the common categories are "cities" and "years", so we have to summarize the cancer incidence SO over "hospitals" and over "cancer sites".

The problem is that the categories "cities" and "years" can have different names in different SOs (e.g. "cities" may be called "town" or "city"), even though they contain exactly the same instances. In addition, even if the names are the same, they may contain different (or partially overlaping) sets of instances for the different SOs.

Figure 5

In general, there are five aspects to category correspondence:

a) *Correspondence between the same category names with different definition domains.* As mentioned in the previous example, category names may be the same but refer to different sets of instances (e.g. "states" in one SO may refer to the 50 states of the USA, and another SO to Guam and Puerto Rico in addition).

b) *Correspondence between different category names with the same* *definition domains.* Different names may be used, such as "nation", "country", etc., for categories which are defined over the same domain.

c) *Category instance value correspondence.* The format for instance values may be different for the same category of different SOs. For example, the different formats for the city Los Angeles may be "LA", "Los Angeles", "LOS ANGELES" or even a code assigned to Los Angeles. Codes are a common practice for large taxonomies, such as medicine, chemistry, and biology, and various business sectors.

d) *Overlap correspondence.* Sometimes it may happen that the same name of categories refers to different definition domains which have partial over-laps among them. Also different names of categories can refer to different definition domains which have partial overlaps among them. For example, you can have two category attributes called both "year" (referring to two different SOs) whose definition domains are respectively "1980, 1981, 1982, 1983, 1984, 1985" and "1983, 1984, 1985, 1986, 1987", or two category attributes called "year" and "years" with the above domains.

e) *Mixing category attributes name and category attributes instances value.* Sometimes it can happen that you have classification instances without class (category attribute) name.

For example, "Energy sources" (class name) is classified by "oil", "coal" and "gas" (classification instances), and "oil" (class name) by "gasoline" and "naphtas" (classification instances), and so on. This means that you may find the same name used both for category attribute and for category attribute name.

4. The STORM Model

In this section we propose the Storm model, based on a graphical representation. Before discussing this model, we define its data structure: the Statistical Object.

4.1 Defining the data structure of a statistical object

Definition: A *Statistical Object* is a data structure defined by a quadruple < N, C, S, **f** >, where:

N is the name of the SO, which describes the universe of the phenomenon of interest (for example, Average Income in California).

C is a finite set of category attributes (sometimes called "characters" from statistician); each category attribute has a domain associated with it, and a "domain cardinality" which corresponds to the number of values (sometimes called "modalities" from statistician) of the domain for that category attribute.

S is a single *summary attribute* associated with the SO. Also the summary attribute has a domain and a domain cardinality associated with it.

f is a *function* which maps from the Cartesian product of the category attributes values to the summary attribute values of the SO.

Each category attribute has also a *primitive category attribute* (more category attributes can have the same primitive category attribute), to which all the attributes with the same semantic meaning are linked and whose domain consists of the union of all the domains of the previous category attributes linked to it. This is much important when, for example, the names of the different category attributes linked to the primitive attribute have different names. For example, in two different SO the category attribute*year* appears and in another SO the category attribute*years* also appears. Then they are linked to the same primitive category attribute *year*.

Each statistical object has different *properties*. Part of them are always specified (for example, summary type = percentage"), others not always appear (for example, it can happen that the "statistical source" is unknown, or that the unit of measure" hasn't sense).

Moreover, "marginal values" exist for each SO. They are connected to the problem of the summarizability of a category attribute in a SO which will be discussed in the following.

> ***Definition:*** Let D be a statistical object represented by a table of cross-tabulated additive statistics. The *marginal value R(i)* in a marginal cell *i* is the sum of the cell-values in the i-th row of D (analogously for the j-th column), assuming that D has at least two rows (and two columns) [MMR91].

The following notation to describe a SO is used in this paper:

N (C(1), C(2), ..., C(n) **:** *S),*

where N and S are respectively the name of the SO and the name of the summary type of the SO, and (C(1), C(2), ..., C(n) are the names of the components of the category attribute set C.

The function f is implied by the "**:**" notation. For example, the following describes a SO on various product sales in the USA, where the summary type (function) is SUM and the summary unit is DOLLARS:

PRODUCT_SALES_IN_THE_USA (TYPE, PRODUCT, YEAR, CITY, STATE, REGION **:** AMOUNT)

As already mentioned, a statistical object represents a summary over micro-data. This summary involves some statistical functions (count, average, etc.) which define the *summary type* of the SO, and, sometime, some *units of measure* of the phenomena of interest (gallons, tons, etc.). In the example above, the summary type is SUM and the unit of measure is DOLLARS.

Note that the above SO is presumed to be generated over some micro-data, such as the individual stores where the products were sold. We note also that the name of a SO is not necessarily a precise description of the SO universe.

In the example given above on "Product Sales", the sales levels are given "by year and by city". Depending on the complexity of the SO, the name may reflect part or all of the category attributes involved. However, it should always reflects the summary attribute intended meaning.

So far, we have described the SO in a form that resembles a relation description in a relational model, with the following structural semantics added: there is a single attribute designated as the summary attribute which has a *summary type* as property always expressed and a *unit of measure* (if it exists) associated with it, and there is a function which maps elements of the Cartesian product of the rest of the category attributes to the summary attribute.

In a following section, it will be shown that these structural semantics are not sufficient for describing a SO, since it needs to know the relationship between the category attributes as well.

In the example above on product sales", suppose that product type can assume the values: metal, plastic, and wood, and that product can assume the values: chair, table, bed. How is it possible to know if sales figures are given for products, product types, or both?

Further, suppose that it is known that figures are given for products, how is it possible to decide whether these figures can be summarized into product type?

Similarly, suppose you need to know whether sales figures for cities can be summarized to state levels and to regions.

In order to answer these type of questions, it needs to capture the structural semantics between category attributes and, to solve the above problems, the Authors propose and use the Storm model [RS90a].

4.2 The spaces and the levels of representation in the Storm model

In this section we propose and discuss the Storm model from the graphical point of view. In particular, we will visualize the representation of a SO in a graphical form as a tree. In order to solve all the problems discussed in the section 3, we define two *representation spaces*, called respectively *intentional* and*extensional* space.

Orthogonally to them, different *representation levels* are defined, as shown in Fig.6.

In the *intentional space* they are three and refer to an associative concept, to a statistical object concept and to a primitive attribute concept.

In the *extensional space* only two levels are present and refer to the instances of both the statistical objects concept and of the primitive attribute concept.

In the *intentional space* the statistical objects are described from the intentional point of view (i.e., without knowing the values of the category attributes instances and of the summary attribute instances). Instead, in the *extensional space* the values of the definition domain (both primitive, and referring to a category attribute of a particular statistical object) and the eventual links between different levels of a classification hierarchy or between attributes of a relation) are shown.

Figure 6

The three representation levels defined in the intentional space are called respectively "Topics" level (T level), "Statistical Objects" level (S level) and "Base" level (B level). We will discuss these spaces and their relative levels in the following sections. Now we discuss the SO data structure so as it is represented in the S level of the intentional space.

4.3 The graphical representation for a Statistical Object

The S (statistical object) level represents the "hearth" of the Storm model.

In it all the SOs of the DB are graphically represented by means of undirected trees. Each SO represents a complitely separated structure, in the sense that it is not possible that a node, anyone, may be connected to other nodes of different statistical objects.

From the graphical point of view each statistical object consists of a tree, whose root is an *S node* (the summary attribute). The S node has a "name" which is also the name of the SO. which is also the name of the SO. This S node is generally linked to an *A node* which represents the Cartesian product among the lying nodes. This A node hasn't a name (this is the only case of node without name in the Storm model). The configuration under this A node depends on the complexity of the SO.

In general, *C nodes* (the category attributes which describe the summary attribute) or other A nodes (linked to at least two C or A nodes andwhich represent the concept of aggregation of the lying nodes) are linked to the Cartesian product A node. All the C node(s) and the A node(s) (with the previous exception) have always a name. In particular cases it can happen that the edge, instead that an S node with an A node, linkes an S node with only one C node. In this case the statistical object represents a *vector*, that is, a statistical object with only one category attribute (e.g., *"Population of the United States in the year 1981, by race"*). Finally, an S node can also be *suspended*, i.e. it has only edge(s) between it and T node(s) (which are in the T level, as we will see in the following). In this case it is called a *scalar* statistical object, because the summary data consist on an only numeric value (e.g., *220.000.000* is such a numeric value with regard to the scalar statistical object, whose name is *Population of the United States in the year 1981*). A definition domain is always associated to each C node.

A C node may be linked to an A node (with the semantics of *classified by*), an S node (case of a *vector*), or another C node (with the semantics of *grouping* or *classification hierarchy*). It is also possible to have a *partitioning* (approach top-down) or *union* (approach bottom-up) of a C node in different C nodes (or, of different C nodes into a C node). For this reason three different types of edges appear at this level: single-continue line, double-continue line and dotted line. Their semantics will be discussed in the following section. All these situations are shown in Fig. 7.

Figure 7

4.3.1 The semantics of the edges of a statistical object

In order to understand the meaning of the edges which link between them the different nodes, we now examine all the situations and represent them by different types of lines, or edges (continue, broken, double) [RS92].different nodes, we now examine all the situations and represent them by different types of lines, or edges (continue, broken, double) [RS92].

Assignment

This semantics is assigned to the edge which links an S node to an A node and whose representation consists of a *single continue line*. The meaning of this edge is that to assign the correspondent value of the summary data to each tuple of the A node resulting from the Cartesian product of the category attributes which characterize the above summary values.

Classification by

This semantics is assigned to the edges which link the A node (without name), which is under the S node, to the C nodes and to the eventual other A node(s) (with name) and whose representation consists of a *single continue line.* In every part of an SO structure (except that between the S and the without name A nodes previously seen) a continue line assumes the semantics of "classification by". It is assigned also to the edge which links an S node to an only C node (vector case).

Grouping* or *hierarchical classification

This semantics is assigned to the edge which links two C nodes in the following way: part of the domain values of a C node are grouped in one domain value of the other C node, (for example, *city-state*, where a group of cities refers to one state). Its representation consists of a *double-continue line.* Note that the C node representing the highest level aggregation concepts (in our case, "state") can be deleted without lose of information. This one depends from the fact that the dimention is the same.

Subset* or *Union* or *Partitioning

This semantics is represented by a *broken line* and represents a subset or a partitioning (or union). We distinguish different situations. turn, different criteria of classification, i.e., a *partitioning* (approach top-down) of a category attribute is made over different category attributes, at the lower level, or, in the other hand, a *union* (approach bottom-up) of different subsets (category attributes) into the same set (category attribute) is carried out. This means that one or more instances of the partitioned category attribute are classified by different criteria. The different classifications are represented by the *name* of the category attribute which "classified by" the *part* (one or more instances) of the category attribute at the higher level written (between parentheses, with before a "for") after the above name.

For example, suppose you have the SO "Hospitalization in California" described by the category attributes *Sex, Race* (grouped in the concept *Demographic information* by the A node with the same name), *City* and *Disease type*, and suppose that this last category attribute is partitioned in two parts. The former refers to the instance *Infection*, "classified by" *Age range*, while the latter refers to the instance *Non-infection* "classified by" *Type of occupation.* (see Fig. 7). Then, the instance "*Infection* of the category attribute *Disease type* will be "classified by" *Age range"* and represented by a C node, whose name is *Age range (***for** *Infection)*, and the instance "*Non-infection* of the category attribute *Disease type* will be "classified by" *Type of occupation"* and represented by a C node, whose name is *Type of occupation (***for** *Non-infection).*

The second situation may represent both the case of *non-symmetric development* of the tree which represents a statistical object, and the case of an *incomplete* classification hierarchy.

For the former, suppose, for example, you have the statistical object "Employment in the United States" of Fig. 7, described by the category attributes *Race*, *Sex* and *Employment status.*

Suppose that this last attribute has the instance *Unemployed* "classified by" *State* and the instance *Employed* is transformed in category attribute. Still suppose that this last category attribute has, as definition domain values <Public, Private>, and whose instance *Private* is "classified by" *Area of Employment.* Then, the instance "*Unemployed* of *Employment status* will be "classified by" *State* " and represented by a C node, whose name is *State (**for** Unemployed),* the attribute *Employed* is a subset of the attribute *Employment status,* whose instance *Private* will be "classified by" *Area of Employment* and represented by a C node, whose name is *Area of Employment (**for** Private).* This a typical non-symmetric classification (only "Private" is classified by "Area of employment").

For the latter, suppose , for example, you still have the SO "Employment in California in 1980" of Fig.7 classified by *Employment status, Sex* and *Race.*

Suppose that *Employment status* is the top of the *classification hierarchy* "Employment - Type of employment - Area of employment" defined in the B level (described in the following section). Suppose the definition domain values of "Employment" are <Employed, Part-time employed, Un-employed>. The partitioning shown refers only to the values which, seen as category attributes, have own criteria of "classification by *somethings"* (in this case, only "Employed" and "Unemployed"). Moreover, suppose, for example, that "Type of employment" refers only to the subset "Employed" of the partitioning of the category attribute "Employed" and let the values of the "Type of employment" definition domain <Public, Private>. Suppose also that only *Private* is classified by "Area of employment", as well as "Unemployed" is classified by "State". In this situation "Public" does not appear as category attribute because it has no criteria of classification and the partitioning of "Employed" consists of only the subset "Private". Analo-gously, if the category attribute "Unemployed" had not been classified by "State", it should disappeared as category attribute (remaining only as value of the definition domain of "Employment".

This is a typical incomplete classification.

Note that all the C nodes of a partitioning may be followed by any other type of line: another partitioning (broken line), or a grouping (double continue line), or an A node, etc.

Finally, a partitioning generally can be carried out for multiple instances (for example, "Year of experience (for engineers, phisics, mathematicians)").

4.3.2 The properties of a statistical object

In this section we discuss about the properties of a statistical object.

Such properties are divided in two classes: a) always present; b) may be present.

In the former we find the *summary type* and the *cardinality* of an SO, in the latter the *information source* and the *unit of measure.*

The *summary type* is defined by the type of function applied to the disaggregated data to obtain the aggregated data (for example, "average").

The *cardinality* of a statistical object (that is, its dimension) is defined by the number of edges linked to the A node under the S node, plus the eventual other "classification by" edges, but only on condition that between it and the S node does not there be a partitioning.

In fact, suppose you have the situation of Fig. 7 with regard to the Statistical Objects "Hospitalization in California" and "Agricultural production in the United States".

The structure "*sex,* patitioned by *male* and *female,* classified respectively by *Type of occupation* and *Age range* " of the first Statistical Object determines only one dimension, because each classification by refers to only one part of the partitioned category attribute, while the structure "*City* classified by *Area of activity* " of the second Statistical Object determines two dimensions, because each state (and city) is classified by *Area of activity*.

The *information source* represents the organization which gives the data and it is important to know who is it both to evaluate its degree of reliability, and because may happen that the same date are coming from two or more different statistical sources. Finally, the *unit of measure,* if present, qualifies the summary data from the quantitative point of view (the numeric value "170" could represent *kilograms,* or *tons,* or *pounds*).

4.4 The S level in the extensional space

In the *extensional space,* which regards both the S level and the B level (but not the T level, for which it should haven't any sense), the domains of all the category attributes represented by C nodes in the intentional space are stored.
The extensional space of the S level is called "Statistical Object Instances Level". At this level the domains of all the category attributes of each statistical object represented at S level are shown, as well as the links between different levels of the same classification hierarchy.
Here it is possible to know, in case of overlapping, if you have the correct total (for example, "Number of patients" by disease , where a patient can have more than one disease), as well as to know if you have different totals between different instances levels (for example, in classification hierarchy, such as "County - State" of the Statistical Object "Production in the United States" of Fig. 7, where "County" refers only to *Orange* and *Richmond* and "State" to all the California (and this for each state of such a category attribute). Note that in this level we have one definition domain for each C node, also if the relative category attribute appears in different Statistical Objects.

4.5 The labeled edges (at the S level)

Both at the S level and at the B level it is possible to have labeled edges. We will discuss the situations in which we need a labeled edge.

4.5.1 Identification (ID) dependency

It can happen that, in a classification hierarchy one instance of a category attribute is not sufficient to recognize exactly the object characterized by it.

For example, suppose you have the Statistical Object "Agricultural production in the United States" of Fig. 7, in which the classification hierarchy "State - City" appears.

Suppose also that one instance of the category attribute "City" refers to different instances of the category attribute "State" (for example, *Buffalo* is the name of two different cities, one in the state of Wyoming and another in the state of New York, as shown in Fig. 9). This means that we need to know not only the name of the city, but also the name of the state to which this city refers. We call this dependency *identification dependency* (or ID dependency).

Obviously, such a dependency can appear in both the S and the B level, but if it "exists" at B level, may "not exists" at S level (for example, because "Buffalo" does not appear at this level), while if it "exists" at S level, must "exist" at B level. This means that the ID dependency is inherited from S to B level, but it is not true the countrary.

In the Storm model this dependency is represented by a label " ID " on the edge which links the two category attributes (in the case of Figure 7, the edge between "city" and "state") [RS90a].

Note that it is possible that different labels appear on the same label.

4.5.2 Non summarizability (NS) and describing marginals (M and Mo)

Often it is not possible to summarize one or more category attributes with regard to a given statistical object, both because this will not have sense, and because the sum of the summary numeric values carries out an uncorrect value.

The former refers, for example, to time series: the population of the states of the United States, counted along the time (for example, every year), is not summable along this category attribute (in fact, the population of the "years

'80" is not the sum of the population of the single years 1980, 1981, . . . , 1989).

In this case a label " NS " appears near the edge of the C node relative to the time (in this case, "year").

The latter refers, instead, to two different situations:

a) when an overlap exists in the summary attribute depending on the category attribute not summarizable (for example, "sicks" classified by "type of disease"; obviously, a person can have two or more diseases, so that the marginal total is lower than the sum of each number of "sicks by disease"). If the correct value of the marginal is available, then this category attribute is summarizable and the edge will be flagged by the label " NS (Mo) ".

b) when the marginal of a given category attribute is larger than the sum of the single values.

Consider, for example, the SO "Population of the United States", in which the classification hierarchy "State - City" exists.

If the population of the state "California" refers to all its population, while the population of the cities of California refers only to San Francisco, San Diego and Los Angeles, the non-summarizability depends on the fact that the marginal value is higher then the sum of the numeric values of the single cities of California.

In this case, if the correct value of the marginal is available (in our case, the population of all the states which appear in such a SO), then this category attribute is summarizable and the edge between the C nodes "City" and "State" is flagged by the label " NS (M) ". Obviously, speaking of "summarizability", the only level in which the above discussion has sense is the S level.

5. Canonical form of a Statistical Object

Now we define the *key* of an SO:

Definition: The *key* of a statistical object is the n-ple (with n ≥ 2) consisting of the minimum set of the names of category attributes which:

a) identifies univocally each summary data value;

28.

b) does not have any redundancy; c) each other category attribute of the SO is obtainable from it.

For example, the category attributes of the SO of Fig. 8-a are "year, sex, street, city, Zip-code, Employment-category, Employment-profession", but the key is "year, sex, street, city, Employment-profession", because "Zip-code" is infered from "street" and "city".

Figure 8

29.

Likewise, "Employment-category" is infered from "Employment-profes-sion".

Definition: The *canonical form* of a SO is the graphical repre-sentation of the SO key.

It is shown in Figure 8-b.

From the above definition we deduce that only one C nodes level appears in the canonical form (then without classification hierarchies, or partitioning, etc.).

5.1 Equivalence in the canonical form

In this section we discuss about the canonical equivalence between a given graphical representation and the correspondent canonical form. This is important in order to understand which part of the model is useful (a facility of the model) but not strictly need.

The first case refers to a *hierarchical classification,* in which one or more ID labels appear.

The equivalent canonical form is a C node whose name is the union of the C nodes names involved by the ID label(s), separated by a "/". This means that the contribution to the dimension of a SO in this case is 1.

The second case refers to a *classification by*.

The equivalent structure consists of a C node, followed by an A node (without "name"), to which the two previous C nodes are linked.

This structure is also equivalent to an A node (with "name"), to which the two previous C nodes are linked.

The equivalent canonical form is a C node whose name is the union of the C nodes names involved in the "classification by".

Examples of the previous situations are shown in Fig. 9.

It is important to note that the A node with label is not necessary, but it expresses the concept by means of which the C nodes under it are grouped.

Figure 9

For example, in Fig. 8 "Address" is the cross product between "Street-City" and "Zip_code", where "Zip_code" is functionally dependent by "Street-City", but "City" is functionally dependent by "Zip_code" (this situation can be seen at the B level).

Moreover, only a subset of the previous cross product exists (the values of this subset can be seen in the extensional space of the same S level).

5.2 Generation of a canonical form

In this section we describe in which way it is possible, given a statistical object, to generate or to transform or to semplify the graphical structure of a Statistical Object in order to obtain its canonical form.

First of all we give the definition of *Complex* statistical object, of *Multi-summary* statistical object and of *Multi-summary Complex* statistical object. These definitions are important to explain that these structures are not present in the Storm model, because they are transformed (splitted) in statistical objects, also if the statistician may need them as structures on screen or on paper.

Definition: A *Complex* statistical object is an SO in which a partitioning (with the relative "classification by") appears.

This means that an SO with partitioning may have different dimentions depending on the structures which there are under the nodes involved in the partitioning.

For instance, in Fig. 10-a an example of complex statistical object using the graphical representation by the Storm model is shown, as well as the relative representation by table (Fig. 10-b).

Figure 10-a

Figure 10-b

In this case the SO has to be splitted in two (or more) different SOs, as shown in Fig. 11.

Definition: A *Multi-summary* statistical object is an SO in which different summary types appear.

This means that this type of SO was obtained by the union of two (or more) SOs in which the phenomenon described was the same, but the summary type was different.

For example, in Fig. 12 an example of Multi-summary SO (represented by a table) is shown.

Also in this case the SO is splitted in two (or more) SOs (in the S level of the intentional space), each of them with a unique summary type.

Definition: A *Multi-summary Complex* statistical object is an SO in which different summary types and a partitioning (with the relative "classification by") contemporarely appear.

For example, in Fig. 13 an example of this situation is shown.

Figure 11

34.

Figure 12

Figure 13

Now we give some rules, called *production rules*, to simplify the representation of a Statistical Object in order to obtain easierly the canonical form of a given Statistical Object.
These rules are divided in three subtypes, called respectively 'transformation", "reduction" and "partitioning" rules.

5.2.1 Transformation rules

These rules regard the transformation of a structure in another equivalent.

RULE T1

A "classification by" between two C nodes is equivalent to another "classification by" between an A node and the two previous C nodes (see Fig. 14).

Figure 14

RULE T2

A "classification by" between a C node and an A node is equivalent to another "classification by" between an A node and the two previous A and C nodes (see Fig. 15).

RULE T3

A "classification by" between two A nodes can be deleted and the remaining A node is that higher (nearer to the S node) (see Fig. 16).

Figure 15

Figure 16

In Fig. 17 we give some examples of application of the previous rules, showing the transformation of some structures in others simplier.

(a)

(b)

Figure 17

5.2.2 Reduction rules

These rules regard the types of possible reductions for a grouping (or hierarchical classification) structure.

RULE R1

A hierarchical classification between two (or more) C nodes is always reduceble to only one C node (see Fig. 18)

Figure 18

RULE R2

A hierarchical classification between one C node and one A node is always reduceble to only one A node (see Fig. 19).

Figure 19

RULE R3

A hierarchical classification between two C nodes linked by an ID label is always reduceble to only one C node whose name is the combination of the names of the two previous C nodes ordered in the "bottom-up" sense and separated by a "slash" (see Fig. 20).

Figure 20

RULE R4

A hierarchical classification between two C nodes linked by an ID label, followed by a "clas-sification by" and another C node, is reduceble to two C nodes linked to an A node by a "classification by" line (see Fig. 21).

In Fig. 22 and Fig. 23 we give some examples of application of the previous rules, showing the transformation of some structures in others simplier.

Figure 21

5.2.3 Partitioning rules

Finally, these rules regard the partitioning of a structure and the generation of different structures (and, consequently, different SOs) deriving by it.

Figure 22

RULE P1

A partitioning of a C node C2, linked to another C node C1, into two A and/or C nodes *x* and *y* produces three structures which consists of a "classification by" between the C node C_1 and the C node *x* (first node of the partitioning), plus another "classification by" between the C node C_1 and the C node *y* (second node of the partitioning), plus another "classification by" between the C node C_2 and the C node C_1 (root of the partitioning).

Figure 23

This last structure represents the other values of the partitioning (see Fig. 24). Note that these three new structures produce three new SOs.

Figure 24

RULE P2

A partitioning of a C node C_1, linked to an A node A_1, into two A and/or C nodes *x* and *y* produces three structures which consists of a "classification by" between the A node A_1 and the C node *x* (first node of the partitioning), plus another "classification by" between the C node C_2 and the C node *y* (second node of the partitioning), plus another "classification by" between the C node C_1 (root of the partitioning) and the A node A_1.

This last structure represents the other values of the partitioning (see Fig. 24). Note that also in this case these three new structures produce three new SOs (see Fig. 25).

Figure 25

In Fig. 26 we give some examples of application of the previous rules.

5.3 The other (B and T) levels of the intentional space

As previously shown in Fig.6, different *representation levels* are defined in the intentional space. Now we illustrate the remaining two levels, the T and the B levels.

5.3.1 The T level

At the *T level* a conceptual representation of grouping of different statistical objects (or sub-concepts under the same topic) is represented by a direct, acyclic, connected graph.

Only *T nodes* (representing the above-mentioned topics) appear in this level and the edges are always oriented. All the T nodes at the lower level (leaves of the graph) point towards one or more S nodes in the S level. This means that an S node is linked to at least one T node (link between the T level and the S level) and the edges relative to these inter-level links are non-oriented. An example of T level is shown in Fig. 27.

Figure 26

Figure 27

5.3.2 The B level

At *B level* all the *primitive* (or *base*) category attributes, to which each category attribute (C node) of each SO in the S level refers, are represented. In this level only C and A nodes appear, as both single C nodes, and linked by oriented or non-oriented lines, such as primitive classification hierarchies (double line), or functional dependencies (oriented line), or equivalences (bi-oriented line), etc.

It is possible to represent also *multiple classification hierarchies*.

The names of the category attributes at S level may be different from the name of the primitive attribute at B level from which they are extracted.

In this level only C and A nodes appear, as well as the above-mentioned three types of edges. In particular, the the single line can be oriented (case of a functional dependency) or bi-oriented (case of an equivalence).

An example of B level representation is shown in Fig. 28.

Figure 28

5.4 The extensional space at the B level

In this level of the extensional space all the primitive definition domains are represented.

At the *S instance level* the domains which refer to the category attributes of each statistical objects appear. In particular, if the same category attribute appears in three different statistical objects and in each of them the definition domain is different, three different domains will appear at this level.

These domains, which can have the same name or different names, will be linked to the same *primitive domain* at the *B instance level.* At this level each domain which appears at the S instance level, will appear with its maximum cardinality. This means that all the values which appear at the S instance level will appear at the B instance level ((the contrary can be not true).

An example of this space and of its two levels is shown in Fig. 29.

Figure 29

6. Representing and interpreting nulls in Statistical Objects

The presence of scarse data or null values is frequent in the statistical objects. In the case of statistical databases, the refined subdivision made in the theory of null values [ANS75, Vas80] is not necessary. In fact, the statisticians basically use only two types of null values: Unknown (or non-available) and non-existing (the latter are often called "structural zero entries" [Bis78]). In this model we also assume, for the same reasons of semplicity, the two above mentioned types of null values.

An example of the null value "unknown" arises when in a SO only the data relative to the production of fruit in the USA are reported; in this case, if the data relative to the state of California for the years "80, 81, 82" are missing, they will be "unknown" (presumably not recorded after a survey and, in any case, not available).

An example of "non-existing" data, instead, can be found in an SO where the data relative to the reports of illness subdivided into illness and sex are illustrated; in this case the missing data regard "cancer of the prostate gland" for "sex = female" or "breast cancer" for "sex = male" (the value will be zero, but it will be a structural zero, in that it can never assume a value which is different from zero). It should be noted that there is an important difference between the two types of null values, especially with reference to the relative marginal value (the total with respect to the category attribute).

In particular, in the former (unknown) case the total with respect to the category attribute, for which there are unknown values, is not the total of the attribute itself. In the latter (non-existing) case, although it is true that, for example, the total of cancer of the prostate cases is the total reported in the marginal (summarizing with respect to "sex"), in this way the information that these cases are the total of males alone and not the entire population is lost.

For this reason, we distinguish, for the tables in output (extensional representation of the summary attribute), two null values: non-available value (symbol " NA ") and structural zero (symbol " - ").

The structural zeroes are important also in the case in which the cross product among the different category attributes which describe the summary data are incomplete (for example, in the case of a relation).

In fact, suppose you have the statistical object *Occupation in the Computer Science Department in 1990 at UCLA* and suppose the *structure* of this Department is the following: < Research, Technical, Administrative >.

Moreover, suppose the *Professional/ Academic titles* admitted for each area are respectively <PhD, Degree>, <PhD, Degree, Diploma (General Certificate of Education)> and <Degree, Diploma (General Certificate of Education)>. This situation is shown in Fig. 30 by a bipartite graph and by a G-relation [Su83], where the last column is the summary data.

The same situation can be represented by a Storm graph (at the S level) and by a statistical table as in Fig. 30, where the symbol " - " represents the structural zero.

The relation between *structure* and *titles* , if it is a *primitive* relation, will appear at the B level, both in the intentional and in the extensional space (and will be inherited at the S level), otherwise will appear only at the S level.

7. Well-formedness of a Statistical Object

In this section we complete the description of the Storm model giving some concepts and definitions which regard the summarizability of a category attribute (for a given statistical object), the fullness and the completeness of a mapping between C nodes, and, finally, the well-formedness of a statistical object.

7.1 Completeness and fullness of mapping

A C node represents a class of values, such as "state" or "year". We call "category attribute" this class and "category attribute instances" or "class instances" the single indivizible values in this class.

A *classification instance* is a C-mapping (i.e., mapping between C nodes) between one class instance of the higher level and one or more class instances at lower level.

Figure 30

Definition: A classification instance is *full* if all the class instances of the lower level exist (according to the semantics of the classification as determined by the Data Base Administrator).

Note that fullness is a semantic concept, no absolute.

Corollary: A C-mapping is *full* if each classification instance is full. To visualize this definition, consider the mapping between "city" and "state" of Fig. 6.

In order to summarize correctly over cities we need to know that there are no missing values. However, it is reasonable to assume that some small towns or other villages were not included in the list of cities, and therefore sales figures for them are not included. If we summarize to the state level, we will get incorrect results.

Figure 30

To compensate for such situations, data-base designers often include another value for cities, which we will label "other".

If a sales figure for "other" was available, then we could claim that the summary can be done correctly. We will call a mapping that satisfies this condition a "full" mapping.

Obviously, this is a semantic condition that depends on the specific mapping. Some mappings may be naturally full.

For example, the mapping between states and regions (e.g. west, mid-west, ... in Fig. 33) can be expected to be full because all the states will be partitioned into disjoint sets that belong to regions.

We call *S mapping* the mapping between an A node and the relative S node; then, the definitions of completeness, quasi-completeness and incompleteness at S level of an S mapping are the following:

> ***Definition:*** A S mapping is *complete* if for every tuple of values in A exists a unique, defined (i.e. not "non-available" or "structural zeroes") value in S.

For example, the mapping of the Statistical Object in Fig.1 is a complete mapping.

> ***Definition:*** An S mapping is *quasi-complete* if for some tuple in A exists a "structural zero", that is non existent value, (represented by " - " into the summary attribute) in S.

For example, in a database on cancer rates, a value for breast cancer for males (regardless of any other category attributes) does not exist.

> ***Definition:*** An S mapping is *incomplete* if for every tuple of values in A exists a unique, defined or non-available (represented by " NA " into the summary attribute), value in S.

Note that an incomplete S mapping can begin complete (for example, it can happen that not all the data referring to a given phenomenon is *at present* available, but within a given time they will be all available). On the other hand missing values can occur for many other reasons. It is not possible to get correct results when missing values exist.

The condition that there are no missing values is a global condition of the

SO, and not unique to each mapping. Although this condition is obviously required for summarizability, it could be tolerated if the information on the missing values is added to the response to the summary operation. Thus, in the case that only a small number of values are missing, most of the results will be correct or near correct.

Definition: A category attribute C is *summarizable* with regard to the statistical object that it describes if:

a) there are no overlapping among the values of its definition domain;

b) there are no overlapping among the numeric values of the summary attribute;

c) there are no overlapping along the time among the population (people, cars, etc.) considered in the phenomenon studied by this statistical object;

d) the edge starting from it, if part of a classification hierarchy, defines a complete C-mapping toward the upper level C node;

e) the unit of measure is the same for each instance of the definition domain and for each category attribute of an eventual partition of which it is part;

f) no unknown (NA, not available) value appears in the summary attribute.

Starting from this definition, we can give the definition of well-formedness of a statistical object.

Definition: A statistical object is *well-formed* if each category attribute, which characterizes it, is summarizable.

Every time that one of the previous conditions of summarizability is not satisfied we have a condition of non-summarizability.

Now we examine with one example for each above condition the real situations in which these conditions determine the non-summarizability of a statistical object:

a) overlapping among the numeric values of the summary attribute.

Suppose you have a statistical object in which the examined fenomenon is "people with a given disease", and that this fenomenon is classified by "state", "sex" and "disease".

For a given state and a given sex, the total number of people that had almost one disease can be lower than the total obtained doing the sum of the number of people that had a given disease (it is sufficient to think to a person that had two or more diseases).

b) overlapping among the values of a definition domain of a category attribute.

Suppose you have a statistical object in which a category attribute is, for example, "age range".

If the domain instances are, for example, < 0-10, 6-18, 19-29, 30-60, over 60 >, an overlapping exists among these values.

This fact does not necessarily determine an overlap-ping among the numeric values of the summary attribute (we do not know if, with regard to that statistical object, somebody exists into the range "6-10"), but the unknowing of it deter-mines the non-summarizability of the category attribute.

c) no overlapping along the time among the population considered.

It is sufficient to consider, for this case, the two time series which describe, along the years, the different phenomena "Production of car" and "Moving cars", by model.

In the former there is no overlapping, in the latter a large part of cars which moved in a given year, move also during the next year, and so on.

d) non-complete C-mapping between two C nodes (in a classification hierarchy)

This is the case in which a sub-area of a category attribute is complitely lacking (for example, all the states of "west" in the statistical object "Car production" by "year" and "state"); obviously the summarization of the category attribute "state" will not be possible with regard to the upper C node "country", also if, in general, it will be possible, by an estimation based on more or less complex criteria, evaluate the lacking values.

Another situation is the classification hierarchy "state"-"city", in the statistical object "population of the USA" by state and year. If only the population of San Francisco and Los Angeles appears in the statistical object, you cannot summarize "city", because the population of the California obtained in such a way should be an erroneous value.

e) different units of measure in the same definition domain of a category attribute.

This case has been previously discussed (the category attribute "dairy" in which you can have two definition domains, each of them with different units of measure).

f) unknown (NA, not available) value appears in the summary attribute.

In this case, non-available (NA) values appear in the summary attribute of a statistical object as "holes" (for example, you can have the car production in California for the years 1985, 1986, 1987 and 1990, but do not have it for the years 1988 and 1989. Then, the value of the production of car in California for the years 1985-1990 is "NA"). This situation is like the previous point d) and also it is generally solved by estimated values (for example, by an interpolation).

8. Conclusion

The work described here was motivated by limitations of current models for describing Statistical Databases. We have defined a new model, called the STatistical Object Representation Model (STORM), and showed how it overcomes these limitations.

In particular, after a brief introduction on the basic concepts regarding the statistical databases, we spoke on the main problems which arise with the current logical models, with particular regard to the concepts of multidimentionality, of classification relationship, of metadata level, of relational table representation, of current graph oriented models and of common reference to categories.

Then we defined the Statistical Object data structure and the different spaces (intentional and extensional), as well as the different levels (T, S and B) of the model. Moreover, we described the graphical representation of a statistical object, its different nodes (T, S, A and C) and the semantics of the different edges (single continue line, double continue line and dotted line), as well as the properties of a Statistical Object.

In the following we discussed about the representation and the interpretation of null values, distinguishing only two types of them.

In addition, we defined the condition for which an edge can be labeled, intruducing four different labels (ID, NS, NS(M) and NS(Mo)).

We defined also the canonical form of a SO, as well as the transformation rules to obtain simple structures of Statistical Objects in canonical form.

Finally, we defined the conditions for a well-formed Statistical Object, for the completeness and the fullness of mapping and the conditions for the "summarizability" of a Statistical Object.

References

[1SD81] Proceedings of the 1st International Workshop on Statistical Data Base Management, Menlo Park, CA, December 2-4, 1981

[2SD83] Proceedings of the 2nd International Workshop on Statistical Data Base Management, Los Altos, CA, September 27-29, 1983

[3SS86] Proceedings of the 3rd International Workshop on Statistical and Scientific Data Base Management, Grand Duchy of Luxembourg, July 22-24, 1986

[ANS75] ANSI/X3/SPARC Study Group on Data Base Manag.Systems Interim Report 75-02-08, EDT-Bull. ACM SIGMOD, 7-2, 1975

[Bis78] Bishop Y. "Discrete Multivariate Analysis. Theory and Practice" MIT Press, 1978

[CS81] Chan P., Shoshani A. "SUBJECT: A Directory Driven System for Organizing and Accessing Large Statistical Databases" Proc. of the 7th Intern. Confer. on Very Large Data Bases (VLDB), 1981

[Hin92] Hinterberger H. Ed. "Statistical and Scientific Database Management" Proceedings of the VI Intern. Conference on Scientific and Statistic Database Management, ETH Publ., Ascona, Switzerland, June 6-10, 1992

[Mic90] Michalewicz Z. Ed. "Statistical and Scientific Database Management" Lecture Notes in Computer Science, N.420, Springer Verlag Publ., 1990

[Mic91] Michalewicz Z. Ed. "Statistical and Scientific Databases" E.Horwood Lim. Publ., Imp.Coll.Comp.Center, Univ. of London, 1991

[MMR91] Malvestuto F.M., Moscarini M., Rafanelli M. "Suppressing marginal cells to protect sensitive information in a two-dimensional statistical table" Proceed. of the 10th ACM Symposium on Principles of Database Systems, ACM Press, Denver, CO, May 29-31, 1991

[OOM85] Ozsoyoglu G., Ozsoyoglu M., Matos F. "A language and a physical organization technique for summary tables" Proceed. of ACM-SIGMOD Intern. Conference on Management of Data, Austin, TX, May 28-31, 1985

[Raf89] Rafanelli M. "Statistical and Scientific Database Management Systems" Encyclopedia of Computer Science and Technology, Kent A. & Williams J.G. Eds, M.Dekker Publ., 1989.

[RF92] Rafanelli M., Ferri F. "VIDDEL: an object oriented visual data definition language for statistical data" in [Hin92]

[RKS89] Rafanelli M., Klensin J.C., Svensson P. Ed.s "Statistical and Scientific Database Management" Lecture Notes in Computer Science, N.339, Springer Verlag Publ., 1989

[RR83] Rafanelli M., Ricci F.L. "Proposal of a Logical Model for Statistical Database" Proceed. of the second International Workshop on Statistical Database Management, Los Altos, CA, 1983.

[RR91] Rafanelli M., Ricci F.L. "A functional model for macro-databases" ACM-Sigmod Recors Vol. 20, N°1, 1991

[RS90a] M.Rafanelli, A.Shoshani :"STORM: a statistical object representation model" in "Statistical and Scientific Database Management" Lecture Notes in Computer Science, N. 420, Springer-Verlag Pub., 1990
and IEEE Data Engineering, Vol. 13, N°3, Sept. 1990

[RS90b] M.Rafanelli, A.Shoshani :"On the representation problems of statistical object", International Conference on Engineering Information in Data Base and Knowledge Based Systems, Berlin, Dec. 4-6, 1990

[RS92] Rafanelli M., Shoshani A. "A model for the graphical representation of aggregated data" Technical Report IASI, R.367, July 1992

[Sho82] Shoshani A. "Statistical Databases: Characteristics, Problems and Solutions" Proc. of the 7th Intern. Confer. on Very Large Data Bases (VLDB), Mexico city, 1982.

[SR91]Shoshani A., Rafanelli M. "A Model for Representing Statistical Objects" Proceed. of the 3rd Internat. Confer. on Management of Data, COMAD '91, Bombay, India, December 12-14, 1991

[Su83] Su S.Y.W. "SAM*: A Semantic Association Model for Corporate and Scientific/Statistical Databases" Inform. Sciences, Vol. 29, N. 2 and 3, May and June 1983.

[SW85] Shoshani A., Wong H.K.T. "Statistical and Scientific Database Issues" IEEE Transactions on Software Engineering, Vol.SE-11, N.10, October 1985.

[Vas80] Vassiliou Y. "Functional dependencies and incomplete information" Proceed. 6th International Conference on Very Large Data Base, Montreal, 1980

[Won84] Wong H.K.T. "Micro and Macro Statistical/Scientific Database Management" Proc. of the 1st IEEE Intern. Confer. on Data Engineering, Los Angeles, CA, 1984.