A survey on summarizability issues in multidimensional modeling

Jose-Norberto Mazón [a,*] Jens Lechtenbörger [b] Juan Trujillo [a]

[a] *Dept. of Software and Computing Systems*
University of Alicante, Spain

[b] *Dept. of Information Systems*
University of Münster, Germany

Abstract

The development of a data warehouse (DW) system is based on a conceptual multidimensional model, which provides a high level of abstraction in accurately and expressively describing real-world situations. Once this model is designed, the corresponding logical representation must be obtained as the basis of the implementation of the DW according to one specific technology. However, even though a good conceptual multidimensional model is designed underneath a DW, there is a semantic gap between this model and its logical representation. In particular, this gap complicates an adequate treatment of summarizability issues, which in turn may lead to erroneous results of data analysis tools. Research addressing this topic has produced only partial solutions, and individual terminology used by different parties hinders further progress. Consequently, based on a unifying vocabulary, this survey sheds light on (i) the weak and strong points of current approaches for modeling complex multidimensional structures that reflect real-world situations in a conceptual multidimensional model and (ii) existing mechanisms to avoid summarizability problems when conceptual multidimensional models are being implemented.

Key words: Multidimensional modeling, summarizability, data warehouse, data analysis

* Corresponding author. Tel.: +34-965-903772; fax: +34-965-909326.
Email addresses: jnmazon@dlsi.ua.es (Jose-Norberto Mazón), lechten@wi.uni-muenster.de (Jens Lechtenbörger), jtrujillo@dlsi.ua.es (Juan Trujillo).

1 Introduction

Nowadays, data warehouse (DW) systems play a decisive role in providing companies with many years of historical information in an accurate way for the decision making process. The development of these systems is based on multidimensional modeling [11], since (i) it is close to the way of thinking of human analysts and, therefore, it helps users to understand data; and (ii) it supports performance improvements as its simple structure allows designers to predict decision makers' intentions.

Multidimensional models structure data according to a multidimensional space, where dimensions specify different ways the data can be viewed, aggregated, and sorted (e.g., according to time, store, customer, product, etc.). Events of interest for an analyst (e.g., sales of products, treatments of patients, duration of processes, etc.) are represented as facts which are associated with cells or points in this multidimensional space and which are described in terms of a set of measures. Thus, every fact is based on a set of dimensions that determine the granularity adopted for representing the fact's measures. Dimensions, in turn, are organized as hierarchies of levels that allow analysts to aggregate data at different levels of detail.

Hence, two major issues must be faced by designers of multidimensional models:

(1) The adequate representation of interactions between dimensions and facts [37].
(2) The adequate representation of relationships between levels of aggregation within a dimension hierarchy [10].

In order to take these issues into consideration, the multidimensional constructs listed in Tab. 1 have to be used. Modeling by using the full potential of these constructs often results in complex multidimensional structures. These structures can be designed in a variety of ways in order to reflect real-world situations, and their accurate yet understandable design is a cornerstone to enable users to analyze large amounts of data stored in DWs to effectively and efficiently support decision-making processes. In particular, special attention should be paid to support an adequate treatment of summarizability issues in order to avoid erroneous results when data is aggregated in data analysis tools.

This introduction section is structured as follows. First, in Subsect. 1.1, we provide an example to be used throughout the survey. This running example allows us to describe summarizability challenges in Subsect. 1.2. Finally, in Subsect. 1.3 we state the focus of this survey. It is worth noting that, although other surveys about multidimensional modeling have been carried out (for example, [32] consists of a survey about using web data in data ware-

Table 1
Constructs for multidimensional modeling

Construct	Features
Level-Level association	roles, multiplicities, default: many-to-one
Level-Level generalization	default: disjoint and complete
Fact-Dimension association	multiplicities, default: many-to-one

houses and [36] provides a summary about general multidimensional modeling methodologies), our paper presents the first survey about summarizability issues in multidimensional modeling.

1.1 Running example

In order to provide a common vocabulary for multidimensional modeling, the UML profile proposed in [19] is used throughout this survey. With this profile, multidimensional models are specified as UML class diagrams, where facts and dimensions are represented by *Fact* (▦) and *Dimension* (⟀) classes respectively. More precisely, *Fact* classes are defined as composite classes in shared aggregation relationships with several *Dimension* classes. If multiplicities are not specified for those relationships, a default of many-to-one is assumed, i.e., each fact is associated with one coordinate in every dimension, and each of the coordinates can be used for many facts. Measures for *Fact* classes are represented as attributes with the *FactAttribute* stereotype (**FA**). With respect to dimensions, each level of a dimension hierarchy is specified by a *Base* class. Every *Base* class (🅱) can contain several dimension attributes (*DimensionAttribute* stereotype, **DA**), and must also contain a descriptor attribute (*Descriptor* stereotype, **D**). Associations (represented by the stereotype *Rolls-UpTo*, ◎) between pairs of *Base* classes form a dimension hierarchy. On the one hand, roles are used to indicate which level is assumed to provide a more detailed view than the other: Role r, for roll-up, (resp. d, for drill-down) represents the direction in which the level of detail decreases (resp. increases). These roles are used to disaggregate and aggregate data. On the other, UML multiplicities are used to specify associations more precisely. Also, we note that this UML profile allows designers to define constraints that indicate whether a specific measure may be aggregated using a specific function (e.g, SUM, COUNT, AVG, MIN, etc.) for specific dimensions or not.

For an introductory example consider the conceptual multidimensional model shown in Fig. 1. Roughly, this model represents a sample multidimensional scenario, where the facts of interest are sales. These sales are structured in a four-dimensional space and allow to analyze who (dimension *Customer*) bought and who sold (dimension *Salesperson*), what (dimension *Product*), and

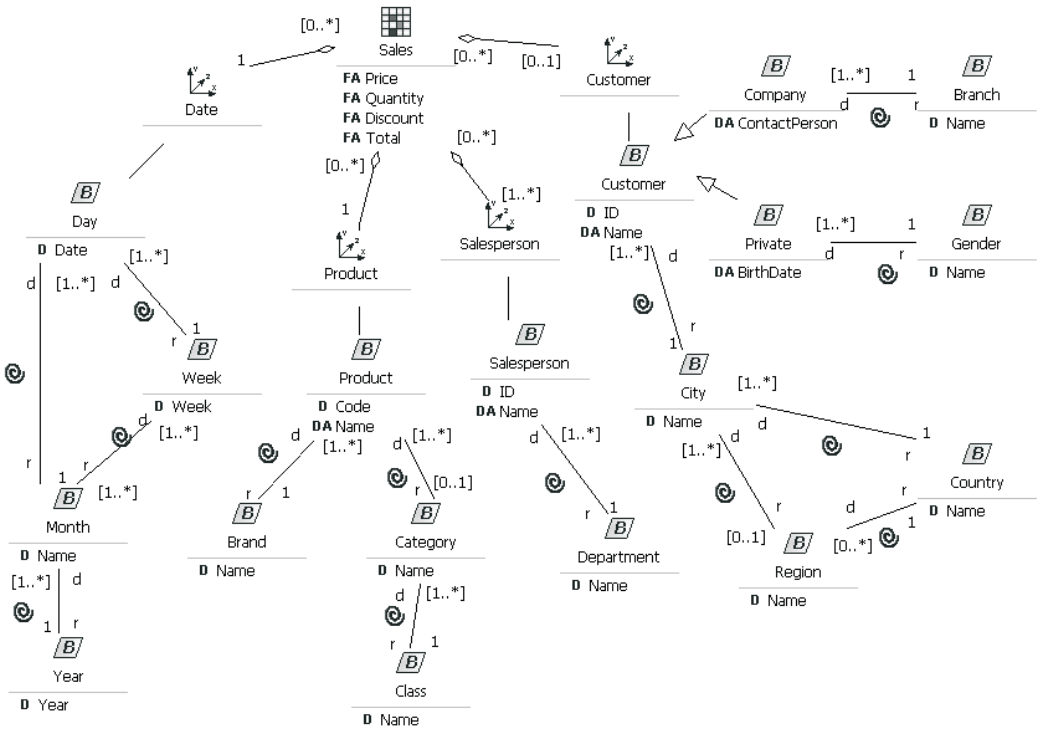


Fig. 1. Sample scenario

when (dimension *Date*). Concerning multiplicities for associations between levels we note that the "standard" case is many-to-one ("1..*" at role "d" and "1" at "r") such as in the case of levels *Day* and *Week*. Here every day belongs to exactly one week, and every week consists of several days. Other cases include the many-to-many relationship ("1..*" at both roles) between *Week* and *Month*, which is commonly referred to as *non-strict*, the many-to-at-most-one relationship ("1..*" at role "d" and "0..1" at "r") between *Product* and *Category*, which we call *roll-up incomplete*, as well as the zero-or-more-to-one relationship ("0..*" at role "d" and "1" at "r") between *Region* and *Country*, which we call *drill-down incomplete*. The "standard" case for associations between facts and dimensions is many-to-one ("0..*" at the role of the *Fact* class and "1..1" at the *Dimension* class), for example between *Sales* and *Date* to indicate that one sale is made at one certain day, and at a single day several sales can be made. "Non-standard" situations include the many-to-many relationship ("*" at the role of the *Fact* and *Dimension* class) between *Sales* and *Salesperson*, and the many-to-zero relationship ("0" at the role of the *Dimension* class) between *Sales* and *Customer*.

1.2 Summarizability challenges

Importantly, end-user tools for data analysis, such as OLAP (On-Line Analytical Processing) or "what-if" analysis, assume that multidimensional models

Table 2
Inconsistent totals for sales due to drill-down incompleteness.

(a) Sales by region.

Region	Sales
Westfalen	10
Bayern	5
Rheinland	10
Valencia	15
Murcia	20
Total	60

(b) Sales by country.

Country	Sales
Germany	25
Spain	35
Andorra	10
Total	70

ensure summarizability, which refers to the possibility of accurately computing aggregate values with a coarser level of detail from values with a finer level of detail. Every multidimensional model to be implemented must ensure summarizability because, otherwise, its violation can lead to incorrect results, and therefore erroneous analysis decisions [17,18]. In addition, summarizability is a necessary precondition for performance optimizations based on pre-aggregation [30].

Therefore, although the sample scenario shown in Fig. 1 takes advantage of highly expressive constructs to design an understandable conceptual multidimensional model, summarizability is guaranteed only in case of many-to-one relationships whereas the other cases require special treatment to ensure consistent query results. For example, consider the reports shown in Tab. 2. The sales by region are shown in Tab. 2(a) (each region belongs to exactly one country), where the total amount is 60. However, if we try to roll-up to country (see Tab. 2(b)), the total sales change to 70. Note that as an example for a violation of summarizability the total numbers obtained in both reports disagree since "Westfalen", "Bayern" and "Rheinland" are German regions, "Valencia" and "Murcia" are Spanish regions, but "Andorra" has no associated regions (as indicated in Fig. 1 by the minimum multiplicity "0" at role "d" for the association between region and country) and, therefore, its sales are not considered in aggregations by region. Importantly, an analyst, who rolls-up from the report in Tab. 2(a) towards the one in in Tab. 2(b), may perceive this difference as an inconsistency. Further details and examples for such situations are given in Sec. 2.

1.3 Focus of the survey

Bearing the aforementioned considerations in mind, the focus of this survey is on research that addresses how to conceptually model complex multidimensional structures in an understandable fashion for designers and users, whilst accurate query responses are allowed by enforcing summarizability when multidimensional models are being implemented. Although there is a general agreement concerning basic multidimensional modeling concepts, there is still a need for unifying terminology, in particular for a successful treatment of summarizability issues. Therefore, this survey provides a common vocabulary to integrate existing approaches for tackling summarizability issues in multidimensional modeling. Our discussion is structured by dividing the state-of-the-art into two main axes:

- General approaches for multidimensional modeling. These approaches focus on defining a structured design process for multidimensional modeling, giving guidelines to design complex multidimensional structures but considering summarizability issues only to a certain extent.
- Multidimensional modeling approaches with special emphasis on complex structures. These approaches focus on solving summarizability problems when complex multidimensional structures are defined.

As we discuss throughout the paper, considering these summarizability problems at the schema level in conceptual models would avoid tackling these problems when querying the data warehouse. Nevertheless, whenever the required data is stored in the DW schema then these summarizability problems could also be tackled ad-hoc when querying the data warehouse. What we firmly claim is that considering these issues in a conceptual schema and automatically generating the corresponding logical schema leads to a much cleaner approach to address these problems.

Finally, we point out that the specification of constraints that indicate whether a specific measure may be aggregated using a specific function is orthogonal to the summarizability issues considered in this paper. Summarizability problems can be then studied regardless of the chosen aggregation function. Therefore, for simplicity, we omit such constraints in our examples.

This survey is organized as follows. In the next section we provide an in-depth description of summarizability concepts and analyze to what extent complex multidimensional structures affect summarizability. Later, in Sect. 3 and 4 we present current approaches for modeling complex multidimensional structures and sketch how they tackle summarizability. We discuss our findings in Sect. 5 and conclude in Sect. 6.

2 Summarizability and multidimensional modeling

In this section we first recall the notion of summarizability introduced by Rafanelli and Shoshani [34] and analyzed by Lenz and Shoshani [18]. Afterwards, in Subsect. 2.1, we proceed by placing work on constraints, in particular, functional dependencies into perspective that allow designers to deal with complex scenarios where summarizability in the strict sense of [18,34] is violated. Then, in Subsect. 2.2, we focus on summarizability issues arising from different kinds of relationships between dimension levels, and we propose a systematic, unifying perspective based on the multiplicities of the involved relationships. In the same spirit we address fact-dimension relationships in Subsect. 2.3, and we briefly discuss the choice of the correct grain of facts in Subsect. 2.4.

The notion of *summarizability* was introduced by Rafanelli and Shoshani [34] in the context of statistical databases, where it refers to the correct computation of aggregate values with a coarser level of detail from aggregate values with a finer level of detail. Although this seminal work on summarizability is framed within the context of statistical databases, it is considered as a cornerstone in multidimensional modeling, because the authors lay the foundations for detecting and avoiding summarizability problems in a multidimensional space.

Concerning the notion of summarizability, consider a Rolls-UpTo association between two dimension levels, say the coarser level l_r and the finer level l_d, and aggregate values for l_d. According to Rafanelli and Shoshani [34], this association is *summarizable* if "using" this association "yields the correct summary values" for l_r, and they observe that many-to-one associations satisfy summarizability while many-to-many associations violate summarizability. For example, in the scenario of Fig. 1 if aggregate sales per *Week* ($= l_d$) are given, then there is no way to correctly compute aggregate sales per *Month* ($= l_r$) using the many-to-many association between *Week* and *Month* as it is unknown how the sales of weeks that partially lie in two months need to be divided.

Furthermore, Rafanelli and Shoshani state four necessary conditions for summarizability, whose essence translates as follows into our setting:

(1) Many-to-many associations must not be used, i.e., the maximum multiplicity of the coarser level (role "r") in Rolls-UpTo associations must be "1" (instead of "*").
(2) Existing many-to-one associations among levels must be modeled.
(3) Many-to-one associations must be "full," i.e., all values contributing to the coarser level must be recorded somewhere at the finer level. This condition can be enforced by including an additional value "other" at the

finer level, which records potentially missing values.

(4) There must not be missing values.

We emphasize that the first two of these conditions deal with the schema level whereas the remaining two are semantic conditions concerning the data level. Indeed, we neglect such completeness issues in this survey as they embody general data quality problems, which need to be addressed independently of aggregation: Clearly, if, e.g., one particular product is missing in the database then no sales concerning this product are recorded, which in turn implies that total sales numbers will be incorrect. Although strictly speaking summarizability is violated as incorrect aggregate values are obtained, the core of the problem does not lie in the use of aggregation.

In the spirit of [34], Lenz and Shoshani [18] argue that summarizability is of most importance for queries concerning multidimensional data, since violations of this property may lead to erroneous conclusions and decisions. Hence, users should be informed when performing non-summarizable operations. Importantly, Lenz and Shoshani show that summarizability is dependent on (i) types of measures and (ii) the specific dimensions under consideration. Moreover, they state three necessary conditions for summarizability. The first of these conditions, called *disjointness*, agrees with condition (1) of Rafanelli and Shoshani stated above. The second condition, called *completeness*, includes condition (4) above and requires in addition that the minimum multiplicity at both ends of an association is "1" (instead of "0"). The third condition, called *type compatibility*, ensures that the aggregate function applied to a measure is summarizable according to the type of the measure (stock, flow and value-per-unit) and the type of the related dimensions (temporal, non-temporal). For example, account balances or quantity on hand are of type stock and, hence, must not be summed over the time dimension. (In contrast, computing average balances over time or sums of balances over products is reasonable.)

For the special case where aggregate functions are restricted to the *sum* operator, the term *additivity* (instead of summarizability) is used frequently. An in-depth analysis of additivity and a taxonomy for reasons why additivity may not hold is presented by Horner and Song [7]. They distinguish schema problems, which are our focus in this survey, from data problems (e.g., inconsistencies and imprecision) and computational problems (e.g., type compatibility in the sense of [18]), give typical examples for each problematic case, and suggest guidelines for their management. Concerning schema problems, in particular roll-up and drill-down incomplete as well as non-strict associations, the general ideas are (i) to tolerate and display incorrect results and (ii) to measure and display inaccuracies, which enables analysts to estimate whether and how much results may suffer from summarizability problems.

Finally, although Lenz and Shoshani [18] only focus on the relationships be-

tween two levels of a dimension hierarchy, the relationships between facts and dimensions can also cause summarizability problems in multidimensional modeling. Specifically, to avoid erroneous results when a multidimensional model is queried, every measure in the fact must be determined by all dimensions, (i.e., the maximum and minimum multiplicity at the dimension end must be "1" in the relationship between a fact and a dimension), which we also address in the following.

2.1 Constraints for summarizability

It is instructive to note that the conditions for summarizability given in [34] and in [18] are incomparable: Condition (3) of [34] is not covered by [18], and condition (3) of [18] is not covered by [34]. Nevertheless, the conditions related to disjointness, completeness, and presence of many-to-one associations can all be understood as constraints (or dependencies) that are expressible at the schema level (e.g., in terms of multiplicities as done above). In fact, several authors advocate to represent summarizability conditions explicitly via dependencies at the schema level.

Importantly, Lehner et al. [17] realized that many-to-one associations between dimension level are just functional dependencies (FDs) as known from standard relational database theory, and they distinguish strong FDs, which correspond to total functions, from weak ones, which correspond to partial functions and which give rise to roll-up incomplete associations in our terminology. For example, in Fig. 1, the association between *Customer* and *City* corresponds to a strong FD, and the association between *Product* and *Category* represents a weak FD.

Similarly, Niemi et al. [28,29] also observe that FDs in dimension hierarchies avoid summarizability problems. Moreover, they mention the use of Boolean dependencies to deal with special cases of many-to-many associations, and they propose novel dependencies to avoid roll-up and drill-incompleteness. For example, consider the well-known hierarchy *City-State-Country* that covers situations in which we have a city without state, e.g., "Washington DC", as well as several cities with the same name within the same country but within different states, e.g., "Springfield". In this approach, we have the following Boolean dependency: *City* $\rightarrow$ *State OR Country*. This dependency implies that if two different cities with the same name do not belong to any state, then they must disagree on the country. On the other hand, either only one of them belongs to a state or if they both belong to a state, then both states must be different.

Based on the work of [17], Lechtenbörger and Vossen [16] define three mul-

tidimensional normal forms (MNFs). Intuitively, the first one (1MNF) deals with the adequate representation of a multidimensional model based on the FDs that hold in the underlying data sources. Importantly, 1MNF implies the first two conditions for summarizability of Rafanelli and Shoshani [34]. Furthermore, the second and third MNF allow to model accurately when and why summarizability may not be given according to the conditions of [18,34] but can be ensured in a context-sensitive manner based on schema information. For example, the specialization of customers into private customers and companies shown in Fig. 1 cannot be represented directly in most multidimensional modeling approaches. Instead, those approaches would include roll-up incomplete associations from *Customer* to *Gender* as well as from *Customer* to *Branch*. Now, to enrich modeling approaches without explicit presence of specialization constructs, the work [16] introduces *context dependencies*, which enable an implicit representation of such specializations (and their reconstruction in relational implementations).

A context dependency of [16] can be regarded as a restricted kind of *dimension constraint* in the sense of [8]. Hurtado et al. [8] point out that there are two kinds of dimension hierarchies: homogeneous and heterogeneous. The former fulfills the summarizability conditions, whereas the latter does not. For example, a standard aggregation path such as the one from *Customer* via *City* to *Country* in Fig. 1 is called homogeneous in the terminology of [8] as all Rolls-UpTo associations are complete, i.e., every customer is located in some city which in turn is located in some country. Conversely, the path from *City* via *Region* to *Country* in Fig. 1, where the Rolls-UpTo association from *City* to *Region* is roll-up incomplete, is called heterogeneous in [8] as cities are heterogeneous in the sense that some of them are related to regions while others are not. The authors argue that it is easier for designers to model heterogeneous dimensions because they are closer to real-world (they represent more naturally and cleanly many practical situations). In our context, heterogeneous dimensions correspond to roll-up/drill-down incomplete hierarchies (since the mapping between levels in heterogeneous dimensions is defined as partial), whereas non-strict associations are not addressed in [8]. The aim of Hurtado et al. is to reason about summarizability of heterogeneous dimensions via a new kind of integrity constraints, called *dimension constraints*, for which they derive a summarizability test. Moreover, they introduce the notion of frozen dimensions, which represent minimal homogeneous dimensions mixed up in a heterogeneous dimension, and they provide an algorithm for the implication problem of dimension constraints based on frozen dimensions.

The constraints for summarizability presented so far aim at a careful definition of dimensions and their hierarchies. Hence, they are called intra-dimensional constraints in [17]. However, within the full scope of multidimensional modeling, further inter-dimensional constraints [17] for relationships between facts and dimensions are needed to ensure summarizability. These interdimensional

constraints are related to the grain of the fact in such a way that, to avoid erroneous results when a multidimensional model is queried, every measure in the fact must be determined by all dimensions, which is reflected in the common relational implementation of a star schema, where the primary key of the fact table is composed of foreign keys of the dimension tables [13]. These constraints are made formally precise by the 1MNF proposed in [16]. Indeed, 1MNF makes sure that the terminal dimension levels of all dimensions of a fact form a key for every measure and implies that all measures are recorded with the same granularity.

In order to obtain a complete picture of the problem space, which allows to pinpoint and differentiate previous approaches, we next revisit each of the UML constructs listed in Tab. 1 that are used to model complex multidimensional structures. For each construct, we systematically discuss the arising summarizability issues based on a complete enumeration of cases.

2.2 Relationships between dimension levels

Dimension hierarchies are among the most important multidimensional structures to be modeled, since they are used by data analysis tools to accurately aggregate or disaggregate data, depending on levels of aggregation. These levels of aggregation must be explicitly specified by organizing the members of a given dimension into hierarchies of levels, in particular in the presence of various kinds of "irregularity" or "heterogeneity" for which early examples can be found in [17].

Importantly, all kinds of relationships between pairs of dimension levels that have been proposed so far can be represented either by associations or by generalizations of dimension levels as listed in Tab. 1. Concerning associations, we have already seen that multiplicities play a crucial role for summarizability. In view of that observation in Tab. 3 we present a complete characterization of associations based on the minimum and maximum multiplicities used in the roles "d" and "r". In that table "regular" and "unusual" denote association types without summarizability problems, the latter being rarely used, whereas the remaining entries form a selection of terms used in the literature for a particular irregularity; the ones used in this paper are *emphasized* and explained in the following. In particular, we propose the novel terms "drill-down incomplete" and "roll-up incomplete," which convey a figurative meaning that we hope to be easy to remember.

Moreover, we emphasize that our discussion deals with single associations in contrast to entire hierarchies, which allows for a more precise classification and treatment of summarizability issues (in particular, a single hierarchy may con-

Table 3
Classification of associations between dimension levels

	Minimum Multiplicity		Maximum Multiplicity	
	0	1	1	*
Role d	*drill-down incomplete*, asymmetric, non-onto, unbalanced	regular	unusual	regular
Role r	*roll-up incomplete*, incomplete, non-covering, ragged	regular	regular	*non-strict*

tain unproblematic associations as well as problematic associations of different kinds).

Concerning generalization we briefly observe that disjointness of generalizations bears similarity with strictness, whereas completeness corresponds to roll-up completeness. Details are presented below.

2.2.1 Regular relationships between dimension levels

We note that summarizability of the "regular" entries follows from the necessary conditions "disjointness" and "completeness" for summarizability stated in [18]. Indeed, disjointness implies that the maximum multiplicity at role "r" is "1" while completeness implies that the minimum multiplicities at both roles are "1". Furthermore, if the maximum multiplicity at role "d" is "1" an unusual situation arises; however, this situation does not contradict the necessary conditions of [18].

2.2.2 Drill-down completeness

A Rolls-UpTo association involving a pair of dimension levels is *drill-down complete* if for every element e of the coarser level (i.e., role r, such as *Country* for the association between *City* and *Country*) there exists an element at the finer level (i.e., role d, here *City*) which is associated with element e; otherwise, it is called *drill-down incomplete*. In other words, a Rolls-UpTo association is drill-down incomplete if the minimum multiplicity at role "d" is 0; otherwise, it is drill-down complete. For example, the association between *Country* and *Region* in Fig. 1 is drill-down incomplete as there are countries (such as Andorra, Monaco, etc.) without associated regions. As explained in the Introduction and illustrated in Tab. 2, drill-down incompleteness violates summarizability since it may yield inconsistent totals.

2.2.3 Roll-up completeness

A Rolls-UpTo association involving a pair of dimension levels is *roll-up complete* if for every element e of the finer level (i.e., role d, such as *Product* for the association between *Product* and *Brand*) there exists an element at the coarser level (i.e., role r, here *Brand*) which is associated with element e; otherwise, it is called *roll-up incomplete*. In other words, a Rolls-UpTo association

Table 4
Inconsistent totals for sales due to roll-up incompleteness.

(a) By product.

Product	Sales
Milk	10
Beer	5
Bread	10
Tuna	15
Napkin	20
Total	60

(b) By category.

Category	Sales
Drink	15
Food	25
Total	40

is roll-up incomplete if the minimum multiplicity at role "r" is 0; otherwise, it is roll-up complete. For example, the association between *Product* and *Category* in Fig. 1 is roll-up incomplete, and faces the problem of inconsistent totals as shown in Tab. 4, where we assume that "milk" and "beer" belong to category "drink", "bread" and "tuna" to category "food", and "napkin" has no category. Therefore, when factual data is aggregated by product the sales are 60 (see Tab. 4(a)). However, special attention should be paid when data is aggregated by category, since "napkin" sales are not taken into account and the total sales decrease to 40 (as shown in Tab. 4(b)).

2.2.4 Strictness

A Rolls-UpTo association involving a pair of dimension levels is *strict* if for every element e of the finer level (i.e., role d, such as *Day* for the association between *Day* and *Week*) there exists at most one element at the coarser level (i.e., role r, here *Week*) which is associated with element e; otherwise, it is called *non-strict*. In other words, a Rolls-UpTo association is strict if the maximum multiplicity at role "r" is 1; otherwise, it is non-strict. For example, the association between *Week* and *Month* in Fig. 1 is non-strict, and requires special care to avoid the well-known double counting problem, which is illustrated in Tab. 5: As week "5-2008" partially belongs to "January" as well as "February" (see Tab. 5(a)), the sales for week "5-2008" should not be counted twice (as is done in Tab. 5(b)) but should be divided appropriately among both months.

2.2.5 Generalization

As observed by Lehner et al. [17], dimension levels in multidimensional models may exhibit heterogeneity in the sense that certain properties may only be ap-

Table 5
Double counting problem for sales due to non-strictness.

(a) Sales by week.

Week	Sales
4-2008	10
5-2008	20
6-2008	10
7-2008	10
8-2008	10
9-2008	10
Total	70

(b) Sales by month.

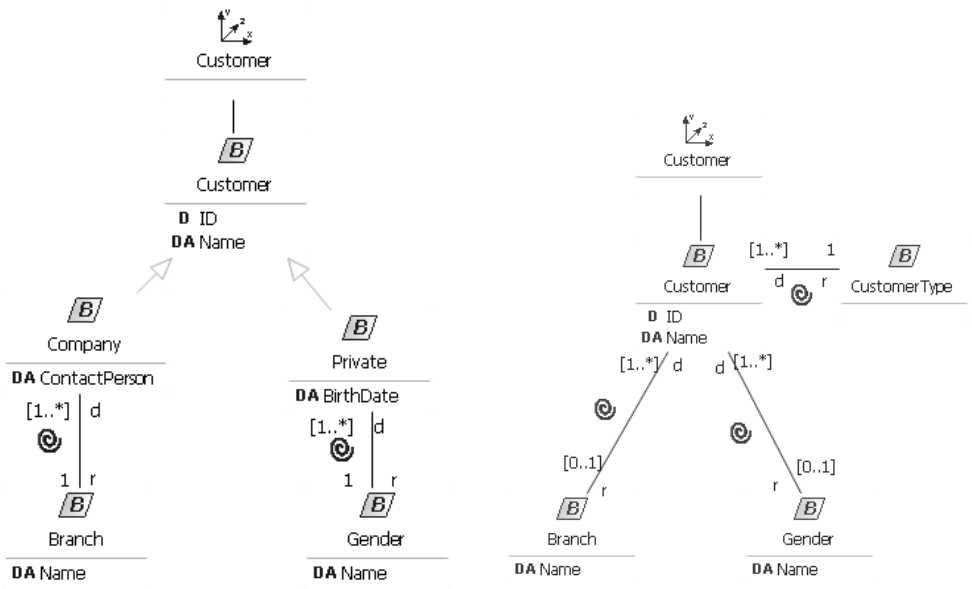
Month	Sales
Jan-2008	30
Feb-2008	60
Total	90

plicable to a subset of elements. For example, for the dimension level *Customer* in the scenario shown in Fig. 1, some customers can be categorized by their *Gender* (if they are human beings) and others by *ContactPerson* (if they are companies). If those properties are modeled as levels, which are reachable from *Customer* via Rolls-UpTo associations, then those associations will necessarily suffer from roll-up incompleteness and the inherent summarizability problems. Moreover, as explained in [17], sparse data cubes may result, and users may pose inconsistent queries ("show sales by Gender and ContactPerson usage").

From an object-oriented perspective such heterogeneity indicates the existence of sub-classes where the individual properties are either applicable (as attributes) or not. In fact, this perspective guides the definition of the third multidimensional normal form of [16], where context dependencies explain the applicability of dimension levels; moreover, these dependencies can be used to construct class hierarchies that contain the applicable attributes, which avoids roll-up incompleteness and associated summarizability problems. It is instructive to note, however, that this approach does *not* allow generalizations as modeling constructs. Moreover, although several multidimensional design proposals are based on object-oriented modeling and, in particular, the UML [1,19,33] none of them explicitly suggests the use of generalization to avoid roll-up incompleteness. In line with [3] we argue that optional constructs such multiplicities of "0" should be avoided in *understandable* conceptual models whenever possible. In particular, generalization relationships for dimension levels embody an attractive alternative for roll-up incomplete associations.

As explained in [16], new analysis potential is unleashed with generalization in two ways:

(1) Traditional roll-up operations may be performed within a certain sub-



(a) Use of generalization. (b) Use of roll-up incompleteness.

Fig. 2. Modeling heterogeneity.

class by using only the applicable levels, which now provides context-sensitive summarizability.

(2) A novel type of drill-down operation allows to switch from Base classes to their immediate sub-classes.

An example can be seen in Fig. 2(a) where sales for all *Customer*s are split into sales for *Company* and *Private* customers (or in the opposite direction via a novel type of roll-up). In traditional multidimensional modeling approaches (see Fig. 2(b)) the drill-down operation towards sub-classes can be simulated by introducing a new dimension level that captures the immediate sub-classes and that is connected from the Base class via a Rolls-UpTo association (e.g., there is a Rolls-UpTo association from *Customer* –role "d"– to the new level *CustomerType* –role "r"– with elements "company" and "private"). It is easy to see that this association has

- multiplicity "1" at role "r" if the generalization is complete and disjoint,
- multiplicity "0..1" at role "r" if the generalization is incomplete and disjoint,
- multiplicity "1..*" at role "r" if the generalization is complete and overlapping, and
- multiplicity "0..*" at role "r" if the generalization is incomplete and overlapping.

Consequently, the novel type of drill-down and roll-up operations is summarizable for complete and disjoint generalizations, whereas other cases exhibit the summarizability issues of roll-up incomplete and/or non-strict associations seen above.

Table 6
Classification of associations between facts and dimensions

	Minimum Multiplicity		Maximum Multiplicity	
	0	1	1	*
Fact	regular	unusual	unusual	regular
Dimension	*incomplete dimensioning*	regular	regular	*non-strict dimensioning*, many-to-many

2.3 Fact-dimension relationships

A crucial decision for designing multidimensional models concerns the grain of facts [13], i.e., the list of dimensions which defines the scope of the measures in a fact. Therefore, the grain of the fact is determined by fact-dimension relationships. To avoid erroneous results, a multidimensional model must have a consistent granularity, which means that every measure in the fact must be determined by all dimensions. As we have explained in Sect. 2.1 this assumption is made formally precise by the first multidimensional normal form proposed in [16]. This assumption implies that the relationship between a fact and a dimension must be many-to-one, which avoids summarizability problems. Therefore, multidimensional models are usually defined according to this multiplicity constraint in order to enforce summarizability in fact-dimension relationships. However, these constraints are too strict for modeling many real-world situations in an easy and understandable way. Designers must deal with scenarios in which different granularities are necessary and where relationships between a fact and a dimension can have different multiplicities. Such situations can lead to multidimensional models with summarizability problems. Specifically, Tab. 6 presents different kinds of fact-dimension relationships, which are characterized by the multiplicities of the association between fact and dimension.

Again, "regular" and "unusual" means that summarizability is ensured, whilst the other terms denote situations which violate summarizability.

2.3.1 Regular fact-dimension relationships

Summarizability is ensured in these relationships, because the fact is functionally determined by the dimension. The common feature of these regular situations is that the minimum and maximum multiplicity in the role of the *Dimension* class is "1" to indicate that every instance of the *Fact* class must be always related to one and only one instance of the *Dimension* class. The other multiplicities are described as follows:

- Minimum multiplicity of "0" at the end of the fact. This multiplicity allows the existence of dimension instances that are not related with any fact instance. This is the most common option for multidimensional modeling, e.g., consider a *Product* dimension where some products have not been sold

so far.

- Minimum multiplicity of "1" at the end of the fact. This multiplicity requires that every dimension instance is related with at least one fact instance. For practical purposes this multiplicity is usually ignored, since it introduces additional restrictions in the multidimensional model that make ETL (Extraction-Transformation-Load) processes more complex and prone to fail.
- Maximum multiplicity of "1" at the end of the fact. This multiplicity requires that every dimension instance is related with at most one fact instance. For practical purposes this multiplicity is usually ignored as well, since it prevents the orthogonal use of dimensions. For example, consider a *Time* dimension where every date can only be used once.
- Maximum multiplicity of "*" at the end of the fact. This multiplicity allows dimension instances to be related with many fact instances. It is the most desirable option within the regular situations.

2.3.2 Incomplete fact-dimension relationships

An association between a fact f and a dimension d is complete if for every fact instance of f, there exists a dimension instance of d which is related to that fact instance; otherwise, the association is *incomplete*. This situation can fall into a summarizability violation since there is a granularity mismatch in the instances of the fact. In other words, a fact-dimension association is incomplete if the minimum multiplicity at the end of the dimension is "0". For example, the association between *Customer* dimension and *Sales* fact in Fig. 1 exhibits an incomplete relationship. This sample faces the problem of inconsistent totals, as shown in Tab. 7(a), where we assume that John and Anna buy some products in January, and George goes shopping in April. The totals arise in a supermarket where some customers have loyalty cards to get discounts. For those customers, sales are recorded directly together with their personal information (e.g., city of residence). In contrast, sales of (anonymous) customers without cards are recorded without considering any personal information (see Tab. 7(b)). Consequently, when the sales are analyzed by customer and date some sales are missing (those from anonymous customers). Only when the customer is not taken into account, the total sales are correct. The problem of inconsistency is shown in Tab. 7 where some anonymous sales made in January are not shown when customer dimension is considered.

2.3.3 Non-strict fact-dimension relationships

An association between a fact f and a dimension d is strict if for every instance of the fact f there exists at most one instance of the dimension d which is related to that fact instance; otherwise, it is called *non-strict*. In other words,

Table 7
Inconsistent totals for sales due to incompleteness.

(a) Sales by customer and time.

Customer	Date	Sales
John	January-2001	10
Anna	January-2001	5
George	April-2001	15
Total		30

(b) Sales by time.

Date	Sales
January-2001	25
April-2001	15
Total	40

Table 8
Double counting problem for sales due to non-strictness.

(a) Sales by salesperson.

Date	Salesperson	Sales
17/01/2001	Bill	10
17/01/2001	Peter	10
18/01/2001	Bill	5
18/01/2001	Peter	5
Total		30

(b) Shared sales by salesperson.

Date	Salesperson	Sales
17/01/2001	Bill, Peter	10
18/01/2001	Bill	5
18/01/2001	Peter	5
Total		20

non-strict associations between a fact and a specific dimension are specified by means of the multiplicity "*" in the role of the corresponding dimension. This situation may cause summarizability problems in the same way as the double counting problem for non-strict dimension associations explained above. For example, in Fig. 1, the association between the *Sales* fact and the *Salesperson* dimension is non-strict, which means that more than one salesperson could be involved in the same sale. This problem is illustrated in Tab. 8: as the sale made on January 17 is shared by Bill and Peter, it should be counted once.

2.4 Beyond multiplicities

In addition to summarizability issues arising from problematic multiplicities, there is another more subtle way in which an inadequate choice for the grain of facts can lead to summarizability problems. As an example, consider the scenario shown in Fig. 1 and assume that there is an additional measure recording the age of customers. If this measure is included in the fact *Sales*, then as shown in Table 9 duplicate age values will be recorded for every product

Table 9
Double counting problem for sales due to incorrect grain.

Date	Product	Salesperson	Customer	CustomerAge
17/01/2001	PC	Bill	Alice	30
17/01/2001	Printer	Bill	Alice	30
17/01/2001	TV	Bill	Bob	20
AvgCustomerAge				26.6

bought by a customer within the same year. In that table, the age of *Customer* Alice is counted twice (once for every sale), leading to an incorrect average customer age of 26.6 (instead of 25).

Formally, these duplicates arise as the age is functionally dependent on only two of the three dimensions, namely on the levels *Date* and *Customer*. Indeed, such a scenario violates the first multidimensional normal form defined in [16] (see also Sec. 2.1). Notice that while in this example it may seem obvious *not* to record customer ages for customer sales, in more complex scenarios such situations may be more difficult to detect. In this survey, we assume that multidimensional models satisfy the first multidimensional normal form, which can be guaranteed based on an analysis of functional dependencies occurring in the application domain, for example using the tool Data Warehouse Detective [6].

3 General approaches for multidimensional modeling

When implementing the multidimensional model in a relational platform, the most common representation is the star schema [13]. This schema consists of one central fact table and several related dimension tables. The levels of aggregation are implicitly contained within the dimension tables. As the star schema is a *logical* representation that aims at a relational implementation, it does not consider the modeling of complex multidimensional structures and it only ensures summarizability if the relationships between multidimensional elements are many-to-one. Therefore, several *conceptual* modeling approaches have been proposed to design multidimensional models that address the design of complex multidimensional structures to accurately model the real world.

In this section, we first review early *conceptual* modeling approaches such as [5] and [9,15], which focus on a structured design process starting from requirements and ending in a tool-specific implementation but which lack support for defining complex multidimensional structures. Afterwards, we consider the recent approaches [1,19,33], which exhibit more expressiveness based on an

object-oriented perspective but which lack in a detailed proposal for ensuring summarizability.

In [5], the authors present a graphical conceptual model for designing multidimensional models (Dimensional Fact Model). This conceptual model consists of a set of fact schemata whose basic elements are facts, measures, attributes, dimensions, and hierarchies. The relationship between a fact and a specific dimension is always many-to-one, and only strict hierarchies can be explicitly modeled by means of many-to-one links between dimension attributes to explicitly indicate how to aggregate and disaggregate data. Furthermore, the existence of optional relationships between attributes in a hierarchy and type compatibility (the third condition for summarizability of [18]) of fact attributes along dimensions is explicitly considered. This approach does not deal with obtaining an implementation of the conceptual multidimensional model without summarizability problems because it only uses constructs that do not violate summarizability conditions.

The multidimensional model proposed in [9,15] stresses the design of summarizable dimension hierarchies in the presence of optional dimension levels and advocates the specification of summarizability constraints. The authors propose a graphical notation for conceptual multidimensional design and also show how to apply multidimensional normal forms [17,9,16] in order to guarantee summarizability. Dimension hierarchies are classified into two basic types: (i) a simple hierarchy consists of exactly one aggregation path within a dimension, and (ii) a multiple dimension hierarchy contains at least two different aggregation paths in a dimension. Alternative aggregation paths occur in multiple hierarchies if elements of split levels belong to exactly one element of a higher level. Moreover, optional groups of aggregation paths are also allowed if some element of a dimension level does not belong to an element of some higher level. Therefore, this approach considers not only strict hierarchies but also generalization relationships between dimension levels. Regarding the relationships between facts and dimensions, only many-to-one relations are considered, and multidimensional normal forms are used in this approach to ensure that the measures are determined by the set of dimensions.

Concerning semantically richer proposals, in [33], Prat et al. first build a general UML (Unified Modeling Language) diagram which is enriched with multidimensional concepts that facilitate the derivation of a logical representation. Although, the authors describe in another paper [2] how to model different kinds of dimension hierarchies, the transformations of the hierarchies into a logical level are not as rich, since they are only defined from many-to-one and one-to-one associations of the conceptual model to fulfill summarizability conditions. Moreover, many-to-many relationships in the general UML diagram are mapped to facts directly, whereas roll-up/drill-down incompleteness is not addressed. Type compatibility is considered in this work, since the link

between a measure and a dimension is characterized by a set of aggregate functions depending on the type of the measure and the corresponding dimension.

Another approach which uses UML constructs is described in [1], where the authors define the multidimensional data model YAM^2. Here, a dimension is a connected, directed graph, where every vertex corresponds to a hierarchy level, and an edge reflects that every instance of target level can be decomposed into a collection of instances of the source level (a relation between levels reflects a part-whole relationship among instances of levels). Although rich constructs are provided, hierarchies must conform with the three conditions for summarizability of [18] (disjointness, completeness and type compatibility) at the conceptual level. The authors argue that the first two conditions depend on constraints specified over multiplicities of relationships between levels, and they propose to allow different kinds of hierarchies provided that multiplicity information is taken into account to decide whether summarizability is given (allowing non-strict hierarchies). However, neither roll-up incomplete nor drill-down incomplete hierarchies are allowed because this approach assumes that every instance of a dimension must have the same structure. In order to address this shortcoming, the authors propose to use dummy values at the instance level as a solution. Concerning the third condition, YAM^2 allows to specify information related to type compatibility directly in the metamodel. Moreover, YAM^2 covers multiple hierarchies in each dimension as well as generalization relationships between levels. Modeling non-strict relationships between facts and dimensions is considered by means of multiplicities in the UML association, whilst incompleteness is addressed by defining generalization between facts in order to express optionality. Unfortunately, the approach does not provide mechanisms to avoid summarizability problems that arise during schema implementation.

Finally, in [19], the UML profile for multidimensional modeling that is used in Fig. 1 is proposed. In this approach, a dimension is composed of hierarchy levels. An association between levels specifies the relationship between two levels of a hierarchy. The only prerequisite is that these levels must define a Directed Acyclic Graph (DAG) rooted in the dimension. A dimension contains a unique first hierarchy level called terminal dimension level. An aggregation path is a subsequence of hierarchy levels, which starts in a terminal level (lower level of detail). The definition of a dimension hierarchy is very expressive, since there are no restrictions concerning associations between dimension levels, provided that every hierarchy fulfills the DAG condition. Consequently, every kind of relationship between levels of a dimension can be represented by using the corresponding multiplicity in the association between levels. Furthermore, it is worth mentioning that this work considers the definition of non-strict and incomplete relationships between facts and dimensions via the definition of different multiplicities between facts and dimensions. However, this work neither offers guidelines to help the designer model different kinds

of complex multidimensional structures nor to ensure summarizability. Concerning type compatibility, all measures are considered as additive by default, i.e., measure values can be summed along all dimensions. Non-additivity and semi-additivity are considered by defining constraints on measures between brackets and placing them somewhere around the fact. These constraints have formal underlying formulae and contain the allowed operators, if any, along the dimension that the measure is not additive. Finally, in [25,26], this approach is integrated in a model driven framework in order to obtain a logical representation of the conceptual multidimensional model in an automatic way.

4 Multidimensional modeling of complex structures

In this section, we review approaches that improve the proposals previously mentioned via the definition of mechanisms to facilitate the modeling of complex multidimensional structures and their implementation ensuring summarizability. These approaches are concerned with either (i) how to model complex dimension hierarchies or (ii) how to model relationships between facts and dimensions.

4.1 Dimension hierarchies

One fundamental work has been carried out by Pedersen et al. [30,31], in which the authors argue that summarizability occurs when dimension hierarchies are "normalized," i.e., roll-up and drill-down complete as well as strict. Importantly, starting from a multidimensional data model allowing multiple, drill-down and roll-up incomplete, as well as non-strict hierarchies, the authors show how to (i) transform dimension *instances* to enforce summarizability and (ii) implement transformed hierarchies using relational database technology. The authors argue that a multidimensional modeling approach should support the explicit design of every kind of hierarchy at a conceptual level to model real-world scenarios accurately and at the same time easily. Only later at a logical phase, summarizability constraints must be enforced by transforming hierarchies into well-behaved logical structures that enable summarizability when data analysis tools are used. For doing so, instance level algorithms are presented to automatically transform dimension hierarchies to achieve summarizability for hierarchies that are roll-up/drill-down incomplete or non-strict. As this proposal works at the instance level, it is necessary to transform the data that will populate the DW, which may involve considerable efforts of preprocessing. In particular, ETL processes become more complex, as summarizability checks must be incorporated and executed for every update. In addition, as the data transformations produce artificial data values,

data analysis becomes more complex.

In [20,21] the authors present a classification of different kinds of complex real-world dimension hierarchies, and they define the MultiDimER model for the conceptual design of complex multidimensional models based on an extension of the well-known Entity-Relationships (ER) model. The idea is that this classification guides developers to properly capture at a conceptual level the precise semantic of different kinds of hierarchies without being limited by current data analysis tools. Modeling of so-called generalized hierarchies is allowed, but somewhat surprisingly the authors do not consider generalization constructs (although extended ER models typically include support for generalization relationships). Furthermore, the authors discuss how to map these conceptual hierarchies to the relational model (enabling implementation in commercial tools). However, the mapping between the conceptual and the logical is described informally. In addition, the commented mapping is tool-dependent and it may vary depending on the scenario. Finally, each mapping is defined independently from the others and the combination of multiple mappings in a process is not addressed at all. In particular, it remains open which mapping needs to be applied first if several of them are applicable (e.g., if a hierarchy is at the same time roll-up incomplete and non-strict, and both kinds of heterogeneity need to be resolved). As no ordering for applying mappings for different hierarchies is specified, applicability problems of the overall approach arise.

Similarly, in [22,23] the authors argue that OLAP tools could fail when dealing with complex hierarchies for real-world situations, since they only admit homogeneous dimension hierarchies. Hence, hierarchies need to be modeled precisely at a conceptual level and then complex hierarchies should be transformed to make them navigable in a uniform manner. To this end, the authors present a framework for conceptual modeling of complex hierarchies and their transformation into a set of well-behaved sub-hierarchies without summarizability problems. They present how to deal with generalization hierarchies at a conceptual model by using informal guidelines, and they use the algorithms from Pedersen et al. [30] (slightly modified) to eliminate roll-up/drill-down incomplete and non-strict hierarchies at the instance level. They focus on visualization of data and every of the proposed transformation aims at incorporating a different kind of hierarchies into a visual OLAP interface to query complex data properly.

In the same spirit, in [2] the authors argue that modeling hierarchies directly at the logical level (by using, for example star or snowflake schemas) can be misleading; hence careful conceptual design is necessary, which then requires a non-trivial transformation to derive a logical representation. The authors advocate the use of aggregation and generalization associations to model hierarchies in UML. Nevertheless, they see problems with the use of generaliza-

tion in hierarchies in multidimensional models, and they aim to preserve the information contained in UML generalizations by transforming them into aggregations following the proposal of [27]. The transformations are formalized with OCL rules in [33].

4.2 Fact-dimension relationships

Few efforts address the proper design of relationships between facts and dimensions and their summarizability issues. Surprisingly, every work is only concerned with the many-to-many relationships between facts and dimensions, i.e. non-strictness, thus ignoring incomplete relationships.

The first proposal considers so-called multivalued dimensions [13], which permit a star schema to have non-strict relationships between facts and dimensions by means of a bridge table. This bridge table captures the non-strict fact-dimension relationship by using foreign keys that refer to the tables that represent the dimension and the fact. These foreign keys also form a compound primary key for the bridge table. Song et al. [37] focus on defining several methods to improve the use of a bridge table. They advocate the representation of many-to-many relationships with correct semantics, maintaining at the same time the star schema structure by defining six different approaches. They also give advantages and disadvantages of each approach and recommendations for their use. Apart from summarizability issues, the authors consider other challenges such as storage or performance requirements. Unfortunately, both approaches [13,37] are defined at the logical level, which requires a lot of expertise to model real-world situations in too complex schemas.

Pedersen et al. [31,30] state that non-strict relationships between facts and dimensions are necessary in many real-world situations, therefore, these relationships must be directly captured in a conceptual model. Nevertheless, in [30] summarizability is tackled at the instance level by modifying data instances, which raises several problems as mentioned in Sect.4.1. Another point of view is described in [31], where a relational approach is described for representing fact-dimension non-strictness by means of alternatives that broaden the use of bridge tables with more expressive solutions.

5 Discussion

Early approaches for multidimensional modeling exhibited a lack of rich mechanisms to specify different kinds of complex multidimensional structures. For example, the well-known star schema [13] does not explicitly define dimension

hierarchies, while other approaches only consider conceptual multidimensional models with limited expressiveness (such as [5,9,15]), where summarizability is guaranteed since problematic complex structures are ignored. However, this lack of modeling support increases the modeling efforts necessary to reflect complex real-world scenarios.

Due to this fact, several approaches [31,20,8,22] arose to define more expressive multidimensional formalisms for modeling complex real-world scenarios, whilst ensuring summarizability. The common foundation of these works is the definition of a classification of different kinds of complex multidimensional structures in order to ease the task of designers about identifying different real-world situations. Most of these approaches present a set of informal guidelines to transform the defined complex multidimensional structures into multidimensional structures which enforce summarizability but which require manual decisions and a lot of expertise when dealing with complex structures, which reduces their applicability. Furthermore, these approaches do not provide enough expressivity to specify every complex multidimensional structure, thus only providing partial solutions. Hence, subsequent research dealt with the definition of more expressive multidimensional formalisms for defining real-world scenarios [19,1,33]. These approaches are based upon an object-oriented approach to allow designers to model more complex multidimensional structures, but they neither offer guidelines for using those more expressive features properly nor formal mechanisms to avoid summarizability problems of complex multidimensional models. Actually, even full expressiveness of object-orientation (generalization) is not exploited so far for considering summarizability on multidimensional modeling.

Therefore, the discussion revolves around two key issues in multidimensional modeling [35]: defining complex structures in an explicit way, and giving mechanisms to support their implementation to avoid the semantic gap regarding summarizability problems. Moreover, suggestions for improving the state-of-the-art are also provided.

5.1 Modeling complex multidimensional structures

Several properties related to the definition of multidimensional models are addressed in Table 10. Specifically, this table focuses on showing (i) the technique or notation used to specify the multidimensional model (technique column), (ii) the supported kind of hierarchies according to our unified notation (roll-up incompleteness, drill-down incompleteness, non-strictness, and generalization columns), (iii) the different kinds of relationships between a fact and a dimension (incomplete and non-strict fact-dimension relationships), and (iv) some kind of guidelines or classification framework provided in support of modeling

Table 10
Properties considered for modeling complex multidimensional structures

	Technique	Hierarchies				F-D relationships		Guidelines
		Drill-down inc.	Roll-up inc.	Non-strictness	Generalization	Incompleteness	Non-strictness	
[19]	UML	Yes	Yes	Yes	Yes	Yes	Yes	No
[1]	UML	No	No	Yes	Yes	Yes	Yes	No
[2,33]	UML	No	Yes	Yes	Yes	No	No	No
[20,21]	ER	Yes	Yes	Yes	Yes	No	No	Yes
[30,31]	Formalism	Yes	Yes	Yes	No	No	Yes	Yes
[22,23]	ER	Yes	Yes	Yes	Yes	No	Yes	Yes
[5]	DFM	No	No	No	No	No	No	No
[9,15,16]	MNFs	No	Yes	No	Yes	No	No	Yes
[13]	Relational	No	No	No	No	No	Yes	Yes
[8]	Formalism	Yes	Yes	No	No	No	No	Yes
[37]	Relational	No	No	No	No	No	Yes	Yes

complex multidimensional structures. At first sight, state-of-the-art in multidimensional modeling lacks an overall approach that defines mechanisms to guide designers to define every complex multidimensional structure, since current research only offers partial solutions either for dimension hierarchies or for fact-dimension relationships.

5.1.1 Techniques for defining multidimensional structures

The most popular techniques used to define Multidimensional structures are somehow related to a relational viewpoint. Thus, some approaches [13,37] directly define multidimensional structures as relational concepts (such as tables, columns keys, and so on) at the logical level. Other approaches [5,9,15] define several multidimensional structures at a conceptual level by using their own notation, but they are highly influenced by a subsequent relational implementation, since they use well-known concepts from relational databases such as functional dependencies. Moreover, other conceptual approaches [21,23] are based on Entity-Relationship modeling, which is suitably extended by additional notation to accommodate specifics of multidimensional modeling. Finally, some approaches use either the UML [19,1,33] or their own formalisms [8,31].

In any case, it is apparent that a highly expressive modeling technique or language must be used to be able to reflect any real-world situation to model complex multidimensional structures in an easy way at the conceptual level.

5.1.2 Modeling different kinds of hierarchies

Approaches at the logical level such as [13] fail in providing mechanisms to model different kinds of hierarchies. Although, the approach proposed in [5] is defined at the conceptual level, it only considers regular hierarchies. The conceptual approach defined in [9,15,16] increases the level of expressiveness, by allowing to model roll-up incomplete and generalization hierarchies, while [8]

only addresses roll-up and drill-down incompleteness.

Thanks to the expressiveness of UML the approach described in [19] covers every possible kind of hierarchy. Other UML-based approaches lack some features: [1] does not deal with roll-up and drill-down incompleteness, while [33] does not give support for drill-down incompleteness.

Some non-UML approaches are also very expressive. The proposal of [31] only misses generalization relationships within a dimension hierarchy, while in [21] a form of generalization is considered but without using generalization constructs. Finally, in [23], every kind of dimension hierarchy is addressed.

Hence, few approaches are able to represent every kind of dimension hierarchy to model every possible real-world situation, which must be a desirable property for any multidimensional modeling approach.

5.1.3 Modeling different kinds fact-dimension relationships

Several approaches point out the necessity for support of many-to-many relationships between fact and dimensions (non-strictness), e.g., [19]. However, regarding incompleteness, there are only two works that allow its definition [19,1]. Other works are somehow related to this issue, e.g., [30] stresses the necessity for having facts with at least one dimension value in every dimension in order to avoid complex and misleading models. Then, incomplete relationships between facts and dimensions are not allowed, and certain real-world situations cannot be specified.

5.1.4 Guidelines for defining complex multidimensional structures

Apart from a specific notation, several approaches provide guidance for defining complex multidimensional structures. A classification framework of different kinds of dimension hierarchies is defined in [21,31,23] in order to help designers to discover situations in the real-world that can be modeled according to a certain dimension hierarchy type. Other work [9] defines a process to apply multidimensional normal forms in modeling dimension hierarchies. On the other hand, thanks to the definition of dimension constraints in [8], summarizability can be characterized and checked. Finally, how to use bridge tables for designing non-strict dimensioning is proposed by [13]. Furthermore, there are several proposals that broaden the use of bridge tables with more expressive solutions [31,37].

Table 11
Properties considered for ensuring summarizability

	Kind	Automation	Level	Tool
[19]	none	-	-	-
[1]	none	-	-	-
[2,33]	Rules	Semiautomatic	Schema/instance	No
[20,21]	Guidelines	Manual	Schema/instance	No
[30,31]	Algorithm	Automatic	Instance	Yes
[22,23]	Algorithm	Semiautomatic	Schema/instance	Yes
[5]	Direct	-	-	-
[9,15,16]	Rules	Automatic	Schema/instance	No
[13]	Direct	-	-	-
[8]	Algorithm	Automatic	Schema	Yes
[37]	Direct	-	-	-

5.2 Ensuring summarizability in multidimensional models

In order to check and enforce summarizability in multidimensional models, state-of-the-art offers different kinds of approaches. Table 11 focuses on providing the properties related to the transformation mechanisms between complex multidimensional structures and their counterparts without summarizability problems. The following properties have been studied: (i) mechanism used to check the summarizability conditions (none, guidelines, algorithm, rules) in the kind column, (ii) level of automation (manual, semiautomatic, automatic), (iii) level in which the summarizability is enforced (schema, instance), and (iv) tool support for summarizability fulfillment.

5.2.1 Kind of summarizability checking

Importantly, at one end of the spectrum some of the studied approaches lack mechanisms to check summarizability constraints: those approaches only address how to model complex multidimensional structures ("none" in Tab. 11). At the other end, there are approaches that focus on directly defining a summarizability-compliant model, so they do not need to check summarizability conditions ("direct" in Tab. 11). In between, we can find approaches that either use informal guidelines to help designers check summarizability [21] or a set of rules or algorithms to formalize summarizability checking [9,31,23,33,8].

5.2.2 Level of automation in modeling complex multidimensional structures

Every modeling technique comes with specific aims, guidelines, procedures, and algorithms, which determine the level of automation. We first consider those approaches that include automatic procedures. The proposal by Pedersen et al. [31] to deal with complex dimension structures is based on algorithms to transform dimension instances that violate summarizability into unproblematic ones [30]. In contrast, Hüsemann et al. [9] consider a simplistic

multidimensional data model, where non-strict associations are not allowed and incomplete associations are only allowed if context dependencies in the sense of [16] are available. In this setting, they show how to design dimension hierarchies based on the analysis of functional dependencies. Finally, Hurtado et al. [8] present algorithms with again a different focus: They do not consider the problem of deriving schemas without summarizability problems but they show how to reason about summarizability of heterogeneous dimensions in the presence of dimension constraints.

With respect to manual or semiautomatic proposals that address the design of complex multidimensional structures, the approach presented in [21] is based on informal guidelines that cannot be implemented immediately. Other approaches are based on rules [33] or algorithms [23] but they need human interaction to validate their application or to provide further information (e.g., determining the applicable aggregation functions).

Moreover, the level of automation is also influenced by the complexity in applying the corresponding mechanisms. We argue that entire dimension hierarchies can be complex and might include several structures involving summarizability challenges. For example, the *Customer* dimension in Fig. 1 involves subclassing as well as roll-up and drill-down incompleteness. So far, there are no formal mechanisms to check summarizability constraints for entire hierarchies. Hence, the proposals described in [21,23], which aim to address summarizability for entire dimension hierarchies, lack a completely automatic procedure. In contrast, other approaches focus on sub-structures of entire hierarchies, e.g., the proposals presented in [31,8] consider relationships between dimension levels to obtain a summarizable version of the dimension hierarchy. These last solutions also allow the definition of modular and easy-to-apply algorithms that can be applied automatically.

5.2.3 Level of multidimensional modeling

Summarizability checking and enforcing can be done at two different levels. Some approaches directly transform the data instances to ensure summarizability [31], e.g., adding some special values, or requiring information from data instances [9,37]. The advantage of these approaches lies in their algorithmic mechanisms to check summarizability; however, non-trivial effort is required to preprocess the huge amounts of data instances before checking the summarizability conditions. Other approaches combine the instance level with information extracted from the schema to decrease the level of required preprocessing and improve performance [33,23,21]. Anyway, the most desirable situation is working only with information from the schema, as stated by [8] (their dimension constraints allow us to test summarizability at the schema level), which avoids exploring potentially huge data instances.

5.2.4 Tool support

Only three approaches present some kind of implementation that helps in checking and enforcing the summarizability conditions. The approach for modeling and transforming complex multidimensional structures of [23] was implemented as part of an OLAP tool in order to improve the visualization expressiveness. The algorithms for enforcing summarizability proposed in [8,31], were implemented and tested. Nevertheless, it is worth noting that, even though several implementations are described, there is neither a prototype nor a tool that supports checking and enforcing the summarizability conditions in a modular and easy-to-use fashion for designers.

5.3 Suggestions for future work

There are three fundamental areas that need to be covered: (i) ensuring summarizability in a comprehensive DW design process, (ii) addressing every complex multidimensional structure in an integrated way, and (iii) developing a tool that supports the design of complex multidimensional structures without summarizability problems. In addition, DWs are more and more used in other novel areas, such as biological, multimedia, or spatio-temporal [35], rather than the classical enterprise domain. Interestingly, this scenario poses new research challenges for tackling summarizability, since even if a carefully designed multidimensional model is obtained, summarizability may not be ensured at all without considering specific semantics of each domain. For example, geographical DWs should use spatial semantics to face up with partial containment dimension hierarchies [12], as well as imprecise and uncertain data [4]. Further advanced issues regarding summarizability are to study and analyze how to solve the problems of summarizability in these non-traditional DWs. For example, new operators are introduced for aggregating these data, such as the combination of OLAP and data mining techniques. Therefore, specific semantics of each domain together with the influence of complex data types and data mining techniques should be further investigated to help ensuring summarizability in the next generation of DWs.

Summarizability needs to be addressed as integrated aspect of a comprehensive design process (rather than as problem concerning isolated multidimensional concepts). A good way to achieve this goal is to follow the conceptual/logical/physical design phases, which allows the designer to take advantage of complex multidimensional structures at a conceptual level without taking summarizability into account for the initial design. Afterwards, this rich conceptual model should be transformed (semi-) automatically into another model based on summarizability considerations (e.g., non-strict dimension-fact relationships may be replaced with more complex schema structures that explicitly

represent different granularities). Whether this transformation should work at the conceptual level or should lead to another model at the logical level is subject to future work. In any case, the transformed model then serves as basis for an implementation of the multidimensional model without summarizability problems, thus bridging the inherent semantic gap. Also, summarizability may be considered in the data sources that will populate the data warehouse, for example by detecting different kind of hierarchies in transactional databases [14].

In addition, while current work gives more importance to dimension hierarchies, we are not aware of research that (i) integrates the definition of all types of complex multidimensional structures presented in this survey and (ii) at the same time defines guidelines to ease the task of designer. For example, non-strict relations between facts and dimensions do occur in real-world scenarios, hence need to be modeled conceptually and then transformed correctly into their corresponding implementation, addressing summarizability problems [37,31]. Towards this direction, the novel work presented in [24] focuses on identifying problematic situations in fact-dimension relationships, defining these relationships in a conceptual multidimensional model, and applying a normalization process with which to transform this conceptual multidimensional model into a summarizability-compliant model that avoids erroneous analysis of data. Furthermore, we argue that object-based approaches are a good choice to proceed: Importantly, approaches that use UML constructs [1,19,33] can represent generalization relationships explicitly to ensure context-sensitive summarizability. Thus, there is no need for designers to explicitly deal with dimension constraints [8] or context dependencies [9,16], while retaining their advantages.

We note that in the related literature two kinds of completeness are identified [18]. First, some instances may simply be missing in the database (e.g., a customer may not be recorded in the DW), which is called incompleteness of type "omitted" in [7]. Second, some instance at a lower level may not be assigned to an instance of a higher level, which is called "orphaned" in [7]. While the later kind of completeness has been addressed in this paper, we argue that the former one is a general problem of data quality, which is unrelated to the design of conceptual or logical models. Incompleteness of type "omitted" should be further investigated in order to ensure summarizability when this situation arises.

Finally, advanced modeling and transformation approaches need to be supported via tools that help designers in checking and enforcing summarizability. Such support is even more pressing when dealing with complex multidimensional models.

6 Conclusions

Multidimensional modeling stresses the definition of complex multidimensional structures, allowing designers to deal with real-world situations, such as roll-up incomplete or non-strict hierarchies. Specifically, powerful modeling constructs must be used to define rich conceptual multidimensional structures, such as (i) relationships between levels within a dimension hierarchy, and (ii) relationships between facts and dimensions.

One key issue when a multidimensional model is being defined is dealing with summarizability. Summarizability guaranties correct aggregation of data. However in real-world scenarios summarizability does not arise in a natural way, and frequently multidimensional design starts with non-summarizable but easily specified and understood conceptual models. Then, an equivalent summarizable model must be obtained before the implementation.

This survey provides researchers with an overall understanding about current approaches for modeling complex multidimensional structures that reflect real-world situations and the mechanisms for enforcing summarizability. We conclude that further research is needed to tackle summarizability issues in multidimensional modeling in a comprehensive way which remains as an open research problem.

7 Acknowledgements

This work has been partially supported by the ESPIA project (TIN2007-67078) from the Spanish Ministry of Education and Science, and by the QUASIMODO project (PAC08-0157-0668) from the Castilla-La Mancha Ministry of Education and Science (Spain). Jose-Norberto Mazón is funded by the Spanish Ministry of Education and Science under a FPU grant (AP2005-1360).

References

[1] A. Abelló, J. Samos, F. Saltor, YAM^2: a multidimensional conceptual model extending UML., Inf. Syst. 31 (6) (2006) 541–567.

[2] J. Akoka, I. Comyn-Wattiau, N. Prat, Dimension hierarchies design from UML generalizations and aggregations., in: ER, 2001.

[3] F. Bodart, A. Patel, M. Sim, R. Weber, Should optional properties be used in conceptual modelling? a theory and three empirical tests, Inf. Syst. Research 12 (4) (2001) 384–405.

[4] D. Burdick, P. M. Deshpande, T. S. Jayram, R. Ramakrishnan, S. Vaithyanathan, OLAP over uncertain and imprecise data, VLDB J. 16 (1) (2007) 123–144.

[5] M. Golfarelli, D. Maio, S. Rizzi, The Dimensional Fact Model: A conceptual model for data warehouses., Int. J. Cooperative Inf. Syst. 7 (2-3) (1998) 215–247.

[6] T. Haselmann, J. Lechtenbörger, G. Vossen, Datawarehouse detective: Schema design made easy, in: BTW, 2007.

[7] J. Horner, I.-Y. Song, A taxonomy of inaccurate summaries and their management in OLAP systems, in: ER, 2005.

[8] C. A. Hurtado, C. Gutiérrez, A. O. Mendelzon, Capturing summarizability with integrity constraints in OLAP., ACM Trans. Database Syst. 30 (3) (2005) 854–886.

[9] B. Hüsemann, J. Lechtenbörger, G. Vossen, Conceptual data warehouse modeling., in: DMDW, 2000.

[10] H. V. Jagadish, L. V. S. Lakshmanan, D. Srivastava, What can hierarchies do for data warehouses?, in: VLDB, 1999.

[11] M. Jarke, M. Lenzerini, Y. Vassiliou, P. Vassiliadis, Fundamentals of Data Warehouses, Springer, 2003.

[12] C. S. Jensen, A. Kligys, T. B. Pedersen, I. Timko, Multidimensional data modeling for location-based services, VLDB J. 13 (1) (2004) 1–21.

[13] R. Kimball, M. Ross, The Data Warehouse Toolkit, Wiley & Sons, 2002.

[14] N. Lammari, I. Comyn-Wattiau, J. Akoka, Extracting generalization hierarchies from relational databases: A reverse engineering approach, Data Knowl. Eng. 63 (2) (2007) 568–589.

[15] J. Lechtenbörger, Data warehouse schema design, in: BTW, 2003.

[16] J. Lechtenbörger, G. Vossen, Multidimensional normal forms for data warehouse design., Inf. Syst. 28 (5) (2003) 415–434.

[17] W. Lehner, J. Albrecht, H. Wedekind, Normal forms for multidimensional databases., in: SSDBM, 1998.

[18] H.-J. Lenz, A. Shoshani, Summarizability in OLAP and statistical data bases., in: SSDBM, 1997.

[19] S. Luján-Mora, J. Trujillo, I.-Y. Song, A UML profile for multidimensional modeling in data warehouses, Data Knowl. Eng. 59 (3) (2006) 725–769.

[20] E. Malinowski, E. Zimányi, OLAP hierarchies: A conceptual perspective., in: CAiSE, 2004.

[21] E. Malinowski, E. Zimányi, Hierarchies in a multidimensional model: From conceptual modeling to logical representation., Data Knowl. Eng. 59 (2) (2006) 348–377.

[22] S. Mansmann, M. H. Scholl, Extending visual OLAP for handling irregular dimensional hierarchies., in: DaWaK, 2006.

[23] S. Mansmann, M. H. Scholl, Empowering the OLAP technology to support complex dimension hierarchies., Int J. Data Warehous. Min. 3 (4) (2007) 31–50.

[24] J.-N. Mazón, J. Lechtenbörger, J. Trujillo, Solving summarizability problems in fact-dimension relationships for multidimensional models, in: DOLAP, 2008.

[25] J.-N. Mazón, J. Trujillo, An MDA approach for the development of data warehouses, Decis. Support Syst. 45 (1) (2008) 41–58.

[26] J.-N. Mazón, J. Trujillo, J. Lechtenbörger, Reconciling requirement-driven data warehouses with data sources via multidimensional normal forms, Data Knowl. Eng. 63 (3) (2007) 725–751.

[27] D. L. Moody, M. A. R. Kortink, From enterprise models to dimensional models: a methodology for data warehouse and data mart design, in: DMDW, 2000.

[28] T. Niemi, J. Nummenmaa, P. Thanisch, Logical multidimensional database design for ragged and unbalanced aggregation, in: DMDW, 2001.

[29] T. Niemi, J. Nummenmaa, P. Thanisch, Normalising OLAP cubes for controlling sparsity, Data Knowl. Eng. 46 (3) (2003) 317–343.

[30] T. B. Pedersen, C. S. Jensen, C. E. Dyreson, Extending practical pre-aggregation in on-line analytical processing., in: VLDB, 1999.

[31] T. B. Pedersen, C. S. Jensen, C. E. Dyreson, A foundation for capturing and querying complex multidimensional data., Inf. Syst. 26 (5) (2001) 383–423.

[32] J. M. Pérez, R. B. Llavori, M. J. Aramburu, T. B. Pedersen, Integrating data warehouses with web data: A survey, IEEE Trans. Knowl. Data Eng. 20 (7) (2008) 940–955.

[33] N. Prat, J. Akoka, I. Comyn-Wattiau, A UML-based data warehouse design method., Decis. Support Syst. 42 (3) (2006) 1449–1473.

[34] M. Rafanelli, A. Shoshani, STORM: A statistical object representation model, in: SSDBM, 1990.

[35] S. Rizzi, A. Abelló, J. Lechtenbörger, J. Trujillo, Research in data warehouse modeling and design: dead or alive?, in: DOLAP, 2006.

[36] O. Romero, A. Abelló, A survey of multidimensional modeling methodologies, IJDWM 5 (2) (2009) 1–23.

[37] I.-Y. Song, W. Rowen, C. Medsker, E. F. Ewen, An analysis of many-to-many relationships between fact and dimension tables in dimensional modeling, in: DMDW, 2001.