A Survey of Multidimensional Modeling Methodologies

Oscar Romero, Universitat Politècnica de Catalunya, Spain

Alberto Abelló, Universitat Politècnica de Catalunya, Spain

ABSTRACT

150 words or less
[Article copies are available for purchase from InfoSci-on-Demand.com]

Keywords: *Comparison; Data Warehouse; Multidimensional Design; OLAP; Survey*

INTRODUCTION

Data Warehousing Systems were conceived to support decision making within organizations. These systems homogenize and integrate data of organizations in a huge repository of data (the *Data Warehouse*) in order to exploit this single and detailed representation of the organization and extract relevant knowledge for the organization's decision making. The data warehouse is a huge repository of data that does not tell us much by itself; like in the operational databases, we need auxiliary tools to query and analyze data stored. Without the appropriate exploitation tools, we will not be able to extract valuable knowledge of the organization from the data warehouse, and the whole system will fail in its aim of providing information for giving support to decision making. OLAP (O*n-line Analytical Processing*) tools were introduced to ease information analysis and navigation all through the data warehouse in order to extract relevant knowledge of the organization. This term was coined by E.F. Codd (1993), but it was more precisely defined by means of the *FASMI* test that stands for *fast analysis* of *shared* business *information* from a *multidimensional* point of view. This last feature is the most important

one since OLAP tools are conceived to exploit the data warehouse for analysis tasks based on *multidimensionality*.

The multidimensional conceptual view of data is distinguished by the *fact* / *dimension* dichotomy, and it is characterized by representing data as if placed in an n-dimensional space, allowing us to easily understand and analyze data in terms of facts (the subjects of analysis) and dimensions showing the different points of view where a subject can be analyzed from. One fact and several dimensions to analyze it give rise to what is known as the *data cube*. Multidimensionality provides a friendly, easy-to-understand and intuitive visualization of data for non-expert end-users. These characteristics are desirable since OLAP tools are aimed to enable analysts, managers, executives and in general those people involved in decision making, to gain insight into data through fast queries and analytical tasks, allowing them to make better decisions.

Developing a data warehousing system is never an easy job, and raises up some interesting challenges. One of these challenges focus on modeling multidimensionality. Nowadays, despite we still lack a standard multidimensional model, it is widely assumed that the data warehouse design must follow the multidimensional paradigm and it must be derived from the data sources, since a data warehouse is the result of homogenizing and integrating relevant data of the organization in a single and detailed view.

Terminology and Notation

Lots of efforts have been devoted to multidimensional modeling, and several methodologies and approaches have been developed and presented in the literature to support the multidimensional design of a data warehouse. However, since we lack a standard multidimensional terminology, terms used among methodologies to describe the multidimensional concepts may vary. To avoid misunderstandings, in this section we detail a specific terminology to establish a common framework where map and compare current multidimensional design methodologies.

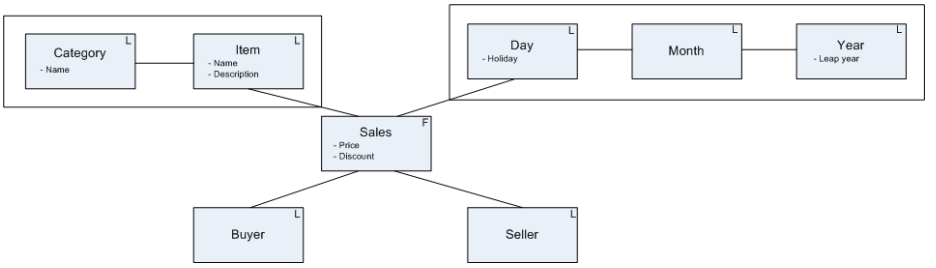
Multidimensionality is based on the fact/dimension dichotomy. **Dimensional concepts** give rise to the multidimensional space where the **fact** is placed. By **dimensional concepts** we refer to any concept likely to be used as a perspective of analysis. Traditionally, they have been classified as **dimensions**, **levels** and **descriptors**. Thus, we consider a **dimension** to contain a hierarchy of **levels** representing different granularities (or levels of detail) to study data, and a **level** to contain **descriptors**. On the other hand, a **fact** contains **measures** of analysis. One **fact** and several **dimensions** to analyze it give rise to a **multidimensional schema**. Finally, we denote by **base** a *minimal* set of **levels** functionally determining a **fact**. Thus, two different instances of data cannot be placed in the same point of the multidimensional space.

Let us consider now the example depicted in figure 1. There, one fact (sales) containing two measures (price and discount) is depicted. This fact has four different dimensions of analysis (buyer, seller, time and item sold). Two of these dimensions contain just one level of detail and two other have an aggregation hierarchy with more than one level. For instance, the time dimension has three levels of detail which contain, at the same time, some descriptors (for instance, the holiday attribute). Finally, if we consider {item X day X buyer X seller} to be the multidimensional base of sales it would mean that one value of each one of these levels identify an instance of factual data (i.e. a sale and its price and discount).

A Piece of History

Multidimensional modeling as it is known today was first introduced by Kimball in (Kimball, 1996). Kimball's approach was well received by the community and a deeper and advanced view of multidimensional modeling was presented in (Kimball et al., 1998). In these books they introduced the first methodology to derive the data warehouse logical schema. Like in most information systems, this methodology is *requirement-driven*: it starts eliciting business

Figure 1. Multidimensional Concepts



requirements of an organization and through a step-by-step guide we are able to derive the multidimensional schema from them. Only at the end of the process data sources are considered to map data from sources to target.

In the following years some other multidimensional modeling methodologies were presented in the literature. Like Kimball's methodology, these methodologies are step-by-step guides to be followed by a data warehouse expert that start gathering the end-user requirements. However, these approaches give more relevance to the data sources. According to the data warehouse definition, the data warehouse is the result of homogenizing and integrating relevant data of the organization (stored in the organization data sources) in a single and detailed view and consequently, data sources must be considered somehow along the design process. Involving the data sources in these approaches means that it is compulsory to have well-documented data sources (for instance, with up-to-date conceptual schemas) at the expert's disposal but it also entailed two main benefits. On the one hand the user may not know all the potential analysis contained in the data sources and analyzing them we may find unexpected potential analysis of interest for the user. On the other hand we should assure that the data warehouse will be able to be populated with data available within the organization.

As previously discussed, to carry out these approaches manually it is compulsory to have well-documented data sources. However, in a real organization the data sources documentation may be incomplete, incorrect or may not even exist and, in any case, it would be rather difficult for a non-expert designer to follow these guidelines. Furthermore, automating this process is essential to not depend on the expert's ability to properly apply the methodology chosen and to avoid the tedious and time-consuming task (even infeasible when working over large databases) of analyzing the data sources. In order to solve these problems several methodologies automating the design process were introduced in the literature. These approaches work directly over relational database logical schemas. Thus, despite they are restricted to relational data sources, they get up-to-date data which can be queried and managed by computers. Furthermore, they argue that restricting to relational technology makes sense since nowadays it is de facto standard for operational databases. About the process carried out, these methodologies follow a *data-driven* process focusing on a thorough analysis of the data sources to derive the data warehouse schema in a reengineering process. This reengineering process consists of techniques and design patterns that must be applied over the relational schema of the data sources to identify data likely to be analyzed from a multidimensional perspective.

However, a requirement analysis phase is crucial to meet the user needs and expectations. Otherwise, the user may find himself frustrated since s/he would not be able to analyze data of his / her interest, entailing the failure of the whole system. Nowadays, it is assumed that the ideal scenario to derive the data warehouse conceptual schema would embrace a hybrid approach (i.e. a combined data-driven and

requirement-driven approach). Therefore, the resulting multidimensional schema would satisfy the end-user requirements and it would have been conciliated with the data sources at once (i.e. capturing the analysis potential depicted in the data sources and being able to be populated with data within the organization).

Later research lines aim to automate the process bearing in mind the organization data sources and requirements. However, automating requirement management is not an easy job since it requires to formalize the end-user requirements (i.e. translate them to a language understandable by computers) and nowadays, many of the current methodologies handle requirements mostly stated in languages (such as natural language) lacking any kind of formalization. On the other hand, a few new approaches have been introduced to automate multidimensional design from other sources that have gained relevance in the last years such as the semantic web (Berners-Lee, Hendler & Lassila, 2001) and they consider other kind of structured data sources based on web-related technologies such as ontologies or XML.

This paper is structured as follows: first, we present the criteria used to discuss about the multidimensional design methodologies surveyed. Furthermore, based on these criteria, we present a detailed comparison of the methodologies surveyed in this paper. This comparison summarizes results presented along the survey as well as it represents main items discussed in a graphical way that shows the evolution of this area.

COMPARISON CRITERIA

This section presents the criteria used to introduce the multidimensional design methodologies surveyed in next section. Setting a basis for discussion will facilitate the mapping of the surveyed methodologies to a common framework where to compare each approach and detect trends such as features in common or the evolution of assumptions made along the way.

These criteria were defined in an incremental analysis of the methodologies surveyed in this paper. For each methodology we captured its main features that were mapped onto different criteria. If a methodology introduced a new criterion, the rest of works were analyzed to know their assumptions with regard to this criterion. Therefore, criteria presented below were defined along an iterative process during the analysis of the multidimensional design methodologies.

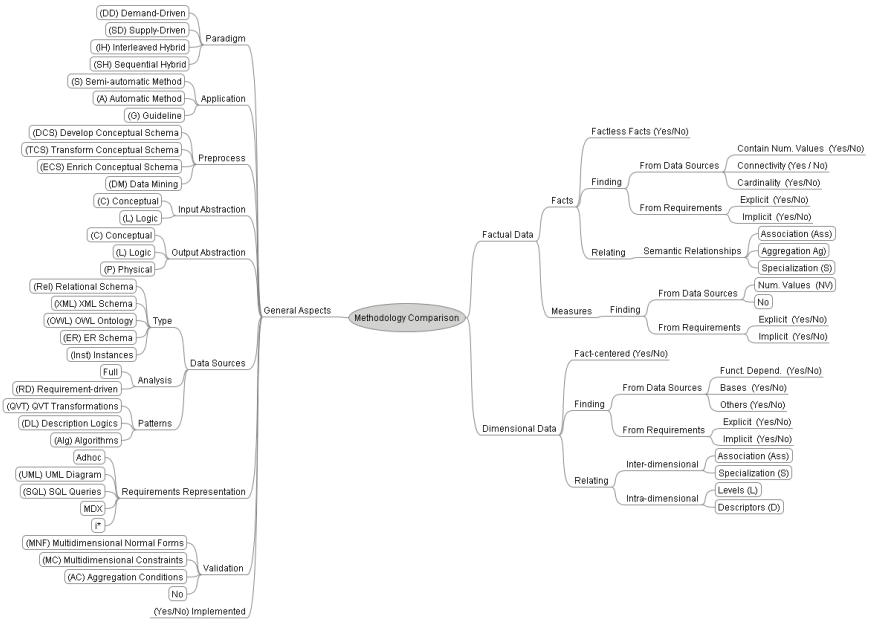
We have summarized these criteria in three main categories: general aspects, dimensional data and factual data. A graphical representation of these features may be found in figure 2. General aspects refer to those criteria regarding general assumptions made in the methodology and dimensional and factual data criteria refer to how dimensional data and factual data are identified and mapped onto multidimensional concepts.

General Aspects

General aspects are summarized into nine different items:

- Paradigm: According to Winter & Strauch (2003) multidimensional modeling methodologies may be classified within a *supply-driven* or *demand-driven* framework. Supply-driven approaches (also known as data-driven) start from a detailed analysis of the data sources to determine the multidimensional concepts in a reengineering process. Demand-driven approaches (also known as requirement-driven or goal-driven) focus on determining the user multidimensional requirements (as typically performed in other information systems) to later map them onto data sources. Finally, hybrid approaches propose to combine both paradigms in order to design the data warehouse from the data sources but bearing in mind the end-user requirements. We distinguish among interleaved hybrid approaches and sequential hybrid approaches. Main difference is that sequential approaches

Figure 2. Graphical view of the criteria used along the survey



perform the demand-driven and supply-driven stages independently and later on conciliate results got whereas interleaved approaches perform both stages in parallel benefiting from feedback retrieved by each stage along the whole process. The reader may found a slightly different classification in (List et al., 2002).

- Application: Most methodologies are semi-automatic. Thus, some stages of these methodologies must be performed manually by an expert (normally those stages aimed to identify factual data) and some others may be performed automatically (normally those aimed to identify dimensional data). In general, only a few methodologies fully automate the whole process. Oppositely, several methodologies present a detailed step-by-step guide that is assumed to be manually carried out by an expert.
- Pre-process: Some methodologies demand to adapt input data into a specific format that facilitates their work. For instance, these processes may ask to enrich a conceptual model with additional semantics or perform data mining over data instances to discover hidden relationships.
- Level of abstraction of the inputs: Most methodologies (mainly those automatable) work with inputs expressed at a logical level (e.g. relational schemas) whereas some others work with inputs at a conceptual level (e.g. from conceptual formalizations such as ER diagrams or from requirements in natural language).
- Level of abstraction of the outputs: Several methodologies choose to directly generate a star or snowflake schema and some others produce multidimensional conceptual schemas. Despite many approaches argue that the data warehouse methodology should span the three abstraction levels, only a few of them produce the conceptual, logical and physical schema of the data warehouse.

- Data sources: There are three items summarizing main features about how data sources are considered in the methodology.
  - Type of data sources: The input abstraction item informs about the abstraction level of the inputs whereas this item specifies the kind of technology of the data sources supported by the methodology. For instance, if the methodology works at a conceptual level it may work with UML or ER conceptual schemas or OWL ontologies, and if it works at a logical level it may work with relational schemas or XML schemas.
  - Analysis of the data sources: Most methodologies perform a full data-driven analysis of the data sources overlooking requirements. However, some of them perform a requirement-driven analysis of the data sources.
  - Pattern formalization: Supply-driven stages usually define design patterns to identify the potential multidimensional role that concepts depicted in the data sources may play. Some methodologies present this patterns in an informal way but most of them use some kind of structured language. For instance, ad-hoc algorithms are the most common representation but some other methodologies use description logic formulas or QVT transformations.
- Requirements representation: If requirements are considered, this item summarizes how requirements are represented. For instance, most methodologies use ad-hoc representations (like forms, sheets, tables or matrixes) whereas some others use UML diagrams or the *i** framework. Finally, some of them lower the level of abstraction of requirements to a logical level by means of SQL queries or MDX queries (Microsoft, 2008).
- Validation: Multidimensionality pays attention to two aspects: placement of data in a multidimensional space and correct summarizability of data. Consequently, some methodologies integrate a validation process to derive meaningful multidimensional schemas. For instance, restricting summarization of data to those dimensions and functions that preserve data semantics or giving rise to multidimensional spaces by means of orthogonal dimensions.
- Implementation: Some methodologies have been implemented in CASE tools or prototypes.

Factual Data

These criteria summarize how a given methodology identifies and handles factual data (i.e. facts and measures). First, criteria used to identify measures are summarized as follows:

- Data sources: Up to now, looking for numerical concepts is the only heuristic introduced to identify measures from the data sources.
- Requirements: Most approaches consider requirements to identify measures. We distinguish if the methodology only considers explicit measures or also implicit ones. Implicit measures are those explicitly stated in the requirements but implicit in the data sources (i.e. there is not a concept in the data sources that would correspond to it). For instance, derived measures. Therefore, some kind of transformations over the data sources must be performed.

Next, we introduce criteria used to identify facts. These criteria refer to how facts are identified from the data sources or from requirements, and how they may be semantically related in the resulting schema:

- Factless facts: This kind of facts were introduced by Kimball (1996).They are also known as empty facts and they are very useful to describe events and coverage and a lot of interesting questions may be asked from them.

- Data sources: Most of the methodologies demand to explicitly identify facts by means of the requirements but some others use heuristics to identify facts from the data sources. For instance, in case of relational sources, most used heuristics are table cardinalities and the number of numerical attributes that a table contains. Furthermore, some works also look for concepts with high to-one connectivity (i.e. with many potential dimensional concepts).
- Requirements: Similar to measures, if requirements are considered, we distinguish among explicit and implicit facts. However, implicit facts have a slightly different meaning. Implicit facts are those that have not been explicitly stated in the requirements but can be identified from a requirement-driven analysis of the sources.
- Semantic relationships: In case of producing a conceptual schema, some methodologies are able to identify semantic relationships among facts. We distinguish among associations, aggregations (also called drill-down relationships) and generalizations. In the multidimensional model, it means that we may perform multidimensional operators such as drill-across or drill-down along them.

Dimensional Data

These criteria analyze how the methodology identifies and handles dimensional data (i.e. dimensions, levels and descriptors). We have two main groups of items. Those referring to how dimensional data is identified from the data sources or from requirements, and how they are semantically related in the resulting schema. The process to identify dimensions, levels and descriptors must be understood as a whole and unlike criteria used to identify factual data we do not distinguish among criteria to look for different dimensional concepts. Roughly speaking, most approaches start looking for concepts representing interesting perspectives of analysis and from these concepts they look for aggregation hierarchies (i.e. levels). The whole hierarchy is then identified as a dimension and level attributes are considered to play a descriptor role:

- Fact-centered: Most methodologies look for dimensional data once they have identified facts. From each fact, dimensional concepts are identified using varied techniques according to the methodology inputs but always looking for functional dependencies from the fact.
- Data sources: There are several techniques to identify dimensional concepts from data sources. We classify these techniques in three main groups: identification of functional dependencies, use of bases and others. At a conceptual level, functional dependencies are modeled as to-one relationships, and at a logical level it depends on the technology. For instance, in the relational model, dimensional concepts are identified by means of foreign keys and candidate keys. Bases (see section terminology and notation for further information) are used to identify dimensional concepts as well. In this case, the methodology looks for candidate multidimensional bases in order to identify interesting perspectives of analysis (i.e. levels).
- Requirements: Dimensional concepts are mostly identified from data sources once we have identified facts and measures. However, demand-driven approaches rely on requirements to identify dimensional concepts and some hybrid approaches also enrich their supply-driven stages with requirements. Like facts, we distinguish among explicit dimensional concepts and implicit ones.
- Intra-dimensional: Most of the methodologies differentiate among descriptors and levels, but some others do not.
- Inter-dimensional: Some approaches are able to identify semantic relationships between dimensions. In this case, we consider associations and generalizations as possible relationships.

Survey of Multidimensional Design Methodologies

This section presents an insight of the multidimensional design methodologies that we have selected for this survey. These methodologies were selected according to three factors: reference papers with a high number of citations according to Google Scholar (Google, 2008), papers with novelty contributions and in case of papers of the same authors we have included the latest version of their works. Each methodology is described following the notation presented in section *terminology and notation* and taking into account the criteria presented in previous section. To present each approach we follow a chronological order that shows the evolution of the approaches along the way:

Kimball et al. (1998) introduced multidimensional modeling as we know it nowadays. Moreover, They also introduced a methodology to derive the multidimensional schema. Being the first approach, it does not present an explicit design procedure but a detailed guide of tips to identify the multidimensional concepts and give rise to the multidimensional schema. The presentation of the methodology is quite informal and it relies on examples rather than on explicit rules. Kimball's approach follows a requirement-driven framework to derive a logical design of the data warehouse.

First, the methodology forces us to name all the data marts we could possibly build. Data marts are defined as pragmatic collections of related facts, but it does not have to be exclusive. Data sources are not considered an it is just suggested to take a look to the data sources to find which data marts may be of our interest.

Next step aims to list all conceivable dimensions for each data mart. At this point it is suggested to build an ad-hoc matrix to capture our multidimensional requirements. Rows represent the data marts whereas the columns represent the dimensions. A given intersection is marked where a dimension exists for a data mart. This matrix is also used to show the associations among data marts if we look to which dimensions they share.

Third step uses a four-step method to design each fact table once we have chosen a data mart:

- First, we declare the grain of detail. It is suggested to be declared by the design team at the beginning despite it can be reconsidered during this process. Normally, it would be determined by primary dimensions.
- Next, we choose the dimensions for the particular fact table that should be tested against the grain selected. This must be a creative step since we need to look for the pieces (i.e. levels and descriptors) of the dimension in different models and through different documents which, in the end, results in a time-consuming task. At this point, it is also suggested to choose a large number of descriptors to populate dimensions.
- Finally, the last step is to add as many measures as possible within the context of the declared grain.

Cabibbo & Torlone (1998) present one of the most cited design methodologies. This approach generates a logical schema from ER diagrams. Moreover, it may produce multidimensional schemas in terms of relational databases or multidimensional arrays. At a first sight, this methodology may be thought as supply-driven since it performs an in-depth analysis of the data sources but no formal rules are given to identify the multidimensional concepts from the data sources. In fact, multidimensional concepts must be manually identified by the user and therefore, from requirements. For this reason, we have considered it to follow a hybrid framework. In general, like Kimball's approach, this approach is quite informal. However, these methodologies set up the foundations that were later used by the rest of methodologies.

This methodology consists of four steps. First and second steps aim to identify facts and dimensions and restructure the ER diagram.

Both steps may be performed in parallel and benefit from the feedback retrieved by each step. Indeed the authors suggest to perform them in an iterative way to refine results got. However, no clue about how to identify facts, measures and dimensions are given and they must be identified from the end-user requirements. Once they have been identified, each fact is represented as an entity. Next, we add dimensions of interest that may be missing in the schema but could be derived from external sources or metadata associated to our data sources. At this point, it is also compulsory to refine the levels of each dimension by means of the following transformations: replacing many-to-many relationships, adding new concepts to represent new levels of interest, selecting a simple identifier for each level entity and removing irrelevant concepts. Finally, steps three and four aim to derive the multidimensional schema. To do so, some clues are given to derive a multidimensional graph that will be directly mapped into the multidimensional schema.

Golfarelli & Rizzi (1998) present one of the reference methodologies in this area. They present a generic overview of the multidimensional design process that embraces their previous works such as (Golfarelli & Rizzi, 1998a). This approach presents a formal and structured methodology partially automatable that consists of six well-defined steps. However, the fourth step aims to estimate the data warehouse workload which goes beyond the scope of this survey:

- First step analyzes the underlying information system and produces a conceptual schema (i.e. a ER diagram) or a logical schema (i.e. a relational schema).
- Second step collects and filters requirements. In this step it is important to identify facts. The authors give some tips to identify them from ER diagrams (entities or n-ary relationships) or relational schemas (tables frequently updated are good candidates).
- Next step derives the multidimensional conceptual schema from requirements and facts identified in previous steps. This step may be carried out semi-automatically as follows:
  - Building the attribute tree: From the primary key of the fact we create a tree by means of functional dependencies. Thus, a given node (i.e. an attribute) of the tree functionally determines its descendants.
  - Pruning and grafting the attribute tree: The tree attribute must be pruned and grafted in order to eliminate unnecessary levels of detail.
  - Defining dimensions: Dimensions must be chosen in the attribute tree among the vertices of the root.
  - Defining measures: Measures are defined by applying to numerical attributes of the tree, aggregation functions at the root level.
  - Defining hierarchies: The attribute tree shows a plausible organization for hierarchies. Hierarchies must be derived from to-one relationships that hold between each node and its descendants.
- Finally, the last two steps derive the logical (by translating each fact and dimension into one relational table) and physical schemas (the authors give some tips regarding indexes to implement the logical schema in a ROLAP tool).

The fourth step of this methodology aims to estimate the workload of the data warehouse. The authors argue that this process may be used to check the correctness of the conceptual schema produced in the third step since queries could only be expressed if measures and hierarchies have been properly defined. However, no more information is provided.

Boehnlein & Ulbrich-vom Ende (1999) present a hybrid approach to derive logical schemas from SER (Structured Entity Relationship) diagrams. SER is an extension of ER that visualizes existency dependencies between objects. For this reason, the authors argue that

SER is a better alternative to identify multidimensional structures. This approach has three main stages:

- Step 0: First, we must transform the ER diagram into a SER diagram.
- Step 1: Business measures must be identified from goals. For instance, the authors suggest to look for business events for discovering adequate measures. Once business measures have been identified, they are mapped to one or more objects in the SER diagram. Eventually, these measures will give rise to facts.
- Step 2: The hierarchical structure of the SER diagrams is helpful to identify potential aggregation hierarchies. Dimensions and aggregation hierarchies are identified by means of direct and transitive functional dependencies. The authors argue that discovering dimensions is a creative task that must be complemented with a good knowledge of the application domain.
- Step 3: Finally, a star or snowflake schema is derived creating fact tables with the primary keys of their dimensions of analysis and denormalizing or normalizing aggregation hierarchies accordingly.

Hüsemann et al. (2000) present a requirement-driven methodology to derive multidimensional schemas in *multidimensional normal form* (MNF). This work introduces a set of constraints that any multidimensional schema produced by this methodology will satisfy. Furthermore, despite this approach produces conceptual schemas they also argue that the design process must comprise four sequential phases (requirements elicitation and conceptual, logical and physical design) like in any classical database design process:

- Requirement analysis and specification: Despite it is argued that the operational ER schema should deliver basic information to determine the multidimensional analysis potential, no clue about how to identify the multidimensional concepts in the the data sources is given. Business domain experts must select strategically relevant operational database attributes and specify the purpose to use them as dimensions or measures.
- Conceptual design: This step transforms the semi-formal business requirements into a formalized conceptual schema. This process is divided in three sequential phases:
  - Context definition of measures: This approach requires to determine a base for each measure (i.e. a minimal set of dimension levels functionally determining the measure values). Furthermore, measures sharing the same bases are grouped into the same fact, as they share the same dimensional context.
  - Dimensional hierarchy design: From each atomic dimension level identified this step gradually develops the dimension hierarchies by means of functional dependencies. Descriptors and levels are distinguished according to requirements and according to this classification, they distinguish between simple and multiple (containing, at least, two different aggregation path) hierarchies as well. Moreover, specialization of dimensions must be considered to avoid structural NULL values when aggregating data.
  - Definition of summarizability constraints: The authors argue that some measure aggregations along certain dimensions do not make sense. Therefore, they propose to distinguish meaningful aggregations of measures from meaningless ones in an appendix of the conceptual schema.

Finally, the authors argue that a multidimensional schema derived by means of this methodology is in multidimensional normal form (MNF) (Lehner, Albrecht & Wedekind, 1998) and therefore it makes full multidimensional sense; that is, we can give rise to a data

cube (i.e. a multidimensional space) free of summarizability problems.

Moody & Kortink (2000) present a methodology to develop multidimensional schemas from ER schemas that was one of the first supply-driven approaches introduced in the literature and one of the most cited papers in this area. Despite it is not the first approach that worked over ER schemas, they present a structured and formal methodology to develop logical schemas. Their methodology is divided into four steps:

- Pre-process: This step develops the enterprise data model if it doesn't exist yet.
- First step: This step classifies the ER entities in three main groups:
  - Transactional entities: These entities record details about particular events that occur in the business (orders, sales, etc). They argue that these are the most important entities in a data warehouse and form the basis of fact tables in star schemas since these are the events that decision makers want to analyze. Despite the authors do not consider requirements, they underline the relevance of requirements to identify facts, since not all transactional entities will be of interest for the user. Moreover, they provide the key characteristics to find this kind of entities: it describes an event that happens at a point in time and it contains measures or quantities that may be summarized.
  - Component entities: These entities are directly related to a transaction entity via a one-to-many relationship and they define details or components of each business event. These entities will give rise to dimension tables in star schemas.
  - Classification entities: These entities are related to component entities by a chain of one-to-many relationships. Said in other words, they are functionally dependent on a component entity directly or transitively. They will represent dimension hierarchies in the multidimensional schema.
- Second step: Next step aims to shape dimension hierarchies. The authors provide some formal rules to indentify them. Specifically, a dimension hierarchy is defined as a sequence of entities joined together by one-to-many relationships all aligned in the same direction.
- Third step: Transactional entities will give rise to facts whereas dimension hierarchies will give rise to their analysis perspectives. The authors introduce two different operators to produce logical schemas:
  - Collapse hierarchy: Higher levels within hierarchies can be collapsed into lower levels. It is a form of denormalization used in data warehousing used to improve query performance.
  - Aggregation: Can be applied to a transaction entity to create a new entity containing summarised data. To do so, some attributes are chosen to be aggregated and other to aggregate by.

According to these operators, this approach introduces five different dimensional design options. According to the level of denormalization of the resulting schema and granularity of data they introduce rules to derive flat schemas, terraced schemas, star schemas, snowflake schemas or star cluster schemas. They also introduce the notion of constellation schema that is defined as a set of star schemas with hierarchically linked fact tables.

Bonifati et al. (2001) present a hybrid semi-automatic approach consisting of three basic steps: a demand-driven stage, a supply-driven stage and a third stage of integration. The final step aims to integrate and conciliate both paradigms and generate a feasible solution that best reflects the user's needs. This method generates a logical multidimensional schema and it was the first to introduce a formal

hybrid approach with an integration step that conciliates both paradigms. Moreover, this methodology have been applied and validated in a real case study:

- In this approach we start collecting the end-user requirements through interviews and expressing user expectations through the Goal/Question/Metrics (GQM) paradigm. GQM is composed of a set of forms and guidelines along four steps: a first vague approach to formulate the goals in abstract terms, a second approach using forms and a detailed guide to identify goals by means of interviews, a step to integrate and reduce the number of goals identified by collapsing those with similarities and finally, a deeper analysis and a detailed description of each goal. Next, the authors present an informal guideline to derive a logical multidimensional schema from requirements. Some clues and tips are given to identify facts, dimensions and measures from the forms and sheets used along the process.
- Second step aims to carry out a supply-driven approach from ER diagrams depicting the operational sources. This step, to be performed in parallel to the previous one may be automated and it performs an exhaustive analysis of the data-sources. From the ER diagram, a set of graphs that will give rise to star-schemas are created as follows:
  - They label potential fact entities according to the number of additive attributes they have. Each identified fact is taken as the center node of a graph.
  - Dimensions are identified by means of many-to-one and one-to-one relationships from the center node.

Next, they introduce an algorithm to derive snowflake schemas from each graph. This transformation is immediate and once it is done, they transform the snowflake schemas into star schemas by flattening the dimension hierarchies (i.e. denormalizing dimensions).

- Third step aims to integrate star-schemas derived from the first step with those identified from the second step. In short, they try to map demand-driven schemas into supply-driven schemas by means of three steps:
  - Terminology analysis: Before integration, demand-driven and supply-driven schemas must be converted to a common terminological idiom. A mapping between GQM concepts and ER concepts must be provided.
  - Schema matching: Supply-driven schemas are compared one-by-one to demand-driven schemas. A match occurs if both have the same fact and some metrics with regard to the number of measures and dimensions are calculated.
  - Ranking and selection: Supply-driven schemas are ranked according to the metrics calculated in the previous step and presented to the user.

Phipps & Davis (2002) introduced one of the first methodologies automating part of the design process. This approach proposes a supply-driven method to be validated, a posteriori, by means of a demand-driven stage. It is assumed to work over relational schemas (i.e. at a logical level) and a conceptual multidimensional schema is produced. In this approach, their main objective is the automation of the supply-driven process with two basics premises: numeric fields represent measures and the more numeric fields a relational table has the more likely it is that the table play a fact role. Furthermore, any table related with a to-many relationship is likely to play a dimensional role. In general, they go one step beyond in the formalization of their approach since a detailed pseudo-algorithm is presented in this paper (and therefore, automation is immediate). However, this approach generates too many results and a demand-driven stage is needed to filter results according to the end-user requirements. Thus, the demand-driven stage within this approach

is rather different to the rest of demand-driven approaches since they do not derive the multidimensional schema from requirements but they use requirements to filter results. This method consists of five steps:

- First step finds tables with numeric fields and create a fact node for each table identified. Tables with numeric fields are sorted in descending order of number of numeric fields.
- Second step creates measures based on numeric fields within fact tables.
- Third step creates date and / or time dimension levels with any date/time fields per fact node.
- Fourth step creates dimensions (consisting of one level) for each remaining table attribute that is non-numeric, non-key and non date field.
- Fifth step recursively examines the relationships of the tables to add additional levels in a hierarchical manner. To do so, it looks for many-to-one relationships (according to foreign keys and candidate keys) all over the schema.

The heuristics used to find facts and determine dimensional concepts within a fact table are rather generic and they give rise to results containing too much noise. Consequently, the authors propose a final requirement-driven step to filter results got. This step presents a step-by-step guide to analyze the end-user requirements expressed as MDX queries and guide the selection of candidate schemas most likely to meet user needs. This last step must be performed manually.

Winter & Strauch (2003) present a detailed demand-driven approach. This is a reference work in the area since it presents a detailed discussion between different multidimensional design paradigms. In this work, the authors also present a design methodology developed from the analysis of several data warehouse projects in participating companies. However, their approach is rather different from the rest of methodologies. Despite they argue that the multidimensional model has gained relevance in the last years, they do not choose any specific data model to express the conceptual schema developed and the authors present a high-level step-by-step guideline independent of any data model. This guideline identifies the best practices that a data warehouse design project must include according to their analysis task. The design process must be iterative and it is divided into four stages:

- First step embraces the analysis of the information supply (i.e. from the sources) and the analysis of the information needed.
- Next, we must match requirements demanded with current information supply and order requirements accordingly.
- In a third step, information supply and information demand must be synchronized on a full level of detail (i.e. considering data granularity selected).
- In the last step we must develop the data schema. This schema must be evaluated and if needed, reformulate the process from first step to develop it in an iterative way. They do not choose any specific data model but they suggest to use specific data models for data wareouses (like the multidimensional model) instead of general purpose semantic models such as ER.

Finally, despite this approach gives relevance to the data sources and requires to synchronize data demanded with the sources, we consider it to be a demand-driven approach since no clue about how to analyze the data sources is given.

Vrdoljak et al. (2003) present a semi-automatic supply-driven approach to derive logical schemas from XML schemas. This approach considers XML schemas as sources. Therefore, the authors propose to integrate XML data in the data warehouse since XML is now a de facto standard for the exchange of semi-structured data. Their approach works as follows:

- Preprocessing the XML schema: The schema is simplified to avoid complex and redundant specifications of relationships.
- Creating and transforming the schema graph: Every XML schema can be represented as a graph. Two transformations are carried out at this point: functional dependencies are explicitly stated (by means of key attributes) and nodes not storing any value are eliminated.
- Choosing facts: Facts must be chosen among all vertexes and arcs of the graph. An arc can be chosen only if it represents a many-to-many relationship.
- Building the dependency graph: For each fact, a dependency graph is built. The graphical representation of the XML schema facilitates finding the functional dependencies. The graph must be examined in the direction expressed by arcs and according to cardinalities included in the dependency graph. It may happen that no cardinality is provided. In this case, XML documents are queried by means of XQueries to look for to-one relationships. The authors also consider many-to-many relationships to be of interest in some cases. However, these cases must be manually identified by the user. Finally, the dependency graph will give rise to aggregation hierarchies.
- Creating the logical schema: Facts and measures are directly depicted from vertexes and arcs chosen whereas dimensions are derived from the aggregation hierarchies identified.

Jensen et al. (2004) present a supply-driven methodology from relational databases. They present data-mining techniques to be applied over the database instances to discover functional and inclusion dependencies and derive snowflake schemas.

Their method starts collecting metadata such as table and attribute names, cardinality of attributes, frequency, etc. Later, data is divided into three groups according to the role of the attribute: measure, keys and descriptive data. Next, integrity constraints such as functional and inclusion dependencies are identified between attributes and finally, the snowflake schema is produced.

First two steps are performed consulting the database catalog and the role of each attribute is derived with a bayesian network which takes as input metadata collected for each attribute. Third step discovers the database structure by identifying functional and inclusion dependencies that represent many-to-one relationships that will give rise to dimensions. Candidate keys and foreign keys are identified assuming that there are no composite keys in the database. Furthermore, inclusion dependencies among foreign keys and candidate keys are identified in this step. These dependencies will be mainly used to identify dimensions. This step is critical, since all permutations of candidate keys and foreign keys are constructed with the consequent computational cost. To pair two keys, both must have the same attribute type and the candidate key must have, at least, as many distinct values for the attribute as the table containing the foreign key. If these constraints hold, a SQL statement is issued to check if the join of both tables by these attributes have the same cardinality as the table containing the candidate foreign key. If so, an inclusion dependency is identified between both keys. Next, they propose an algorithm to derive snowflake schema from this metadata:

- Fact tables are identified in a semi-automatic process involving the user by means of the table cardinality and the presence of measures that have been identified by the bayesian network.
- Inclusion dependencies found conform different connected graphs. A connected graph is considered to be a dimension if exists a inclusion dependency among a fact table and one node of the graph. In this case, that node will be the atomic level of the dimension. The authors propose an algorithm to break potential cycles and give rise to the aggregation hierarchy from the graph. When giving shape to the aggregation hierarchy, two consecutive levels are

analyzed to avoid aggregation problems (i.e. duplicated or lost values).

Giorgini et al. (2005) present a hybrid approach to derive a conceptual multidimensional schema. They propose to gather multidimensional requirements and later map them onto the data sources in a conciliation process. However, they also suggest that their approach could be also considered demand-driven if the user does not want to take into account the data sources.

The authors introduce an agent-oriented methodology based on the *i** framework. They argue that it is important to model the organization setting in which the data warehouse will operate (organization modeling) and to capture the functional and non-functional requirements of the data warehouse (decisional modeling).

If we consider their hybrid approach, then next step is to match requirements with the schema of the operational sources. In this approach both ER diagrams and relational schemas are allowed as inputs describing the data sources. This matching stage consists of three steps:

- Requirement mapping: Facts, dimensions and measures identified during the requirement analysis are now mapped over the data sources. According to the kind of data sources considered, the authors introduce a set of hints to map each concept. For instance, facts are mapped onto entities or n-ary associations in ER diagrams and onto relations in relational schemas.
- Hierarchy construction: For each fact identified, the data sources are analyzed looking for functional dependencies based on the algorithm already discussed in (Golfarelli & Rizzi, 1998b).
- Refinement: This step aims to rearrange the fact schema in order to better fit the user's needs. Along the process, we may distinguish among concepts available (mapped from requirements), unavailable (demanded in the requirements but not mappable to the data sources) and what is available and not needed. The authors propose to use this information to reorder dimensions (grafting and pruning the aggregation hierarchies) or try to find new directions of analysis.

Prat et al. (2006) present a methodology to derive the conceptual, logical and physical schema of the data warehouses according to the three abstraction levels recommended by ANSI/X3/SPARC. Starting from end-user requirements, the conceptual phase leads to a UML schema. To this end, UML is enriched with concepts relevant to multidimensionality that will facilitate the generation of the logical schema. The logical phase maps the enriched UML schema into a multidimensional schema and finally, the physical phase maps the multidimensional schema into a physical database schema depending on the target implementation tool (in this case Oracle MOLAP). At each phase, they introduce a metamodel and a set of transformations to perform the mapping between metamodels.

- Conceptual phase: The authors embrace under this phase requirements elicitation and conceptual representation of requirements. First, requirements should be captured by means of a UML-compliant system analysis method. Requirements engineering techniques used in transactional design processes may be applied and for instance they mention interviews, joint sessions, study of existing reports and prototyping of future reports as potential techniques to be used. Next, requirements are represented in a UML class diagram that needs to be enriched to capture multidimensional semantics. To do so, they present an extension of the UML metamodel.
- Logical phase: Creating the logical schema from the enriched conceptual model produced in the first phase is immediate and a set of transformations expressed in OCL are presented. They also introduce an adhoc multidimensional metamodel to represent the logical schema.
- Physical phase: A set of transformations

are presented to map the logical schema into the Oracle MOLAP tool.

Romero & Abelló (2006) present a method to derive conceptual multidimensional schemas from requirements expressed in SQL queries and relational models. This approach is fully automatic and follows a hybrid paradigm. Furthermore, unlike other hybrid approaches, this approach does not carry out two well-differentiated phases (i.e. data-driven and requirement-driven) that need to be conciliated a posteriori, but carry out both phases at once. Thus, both paradigms benefit from feedback returned by each other and eventually, it is able to derive more valuable information than carrying out both phases sequentially. On the other hand this is the first method automating its demand-driven stage. Said in other words, automating the analysis of the end-user requirements (Romero & Abelló, in press).

This method is fully automatic and it produces constellation schemas from the requirements (i.e. the SQL queries) and the data sources logical schema. Moreover, this method is able to cope with denormalization in the input relational schemas and get equivalent outputs when applied over normalized (up-to third normal form) and denormalized relational sources. The multidimensional schema is derived along two different stages:

- For each input query, first stage extracts the multidimensional knowledge contained in the query (i.e. the multidimensional role played by each concept in the query as well as the conceptual relationships among concepts), that is properly stored in a graph. Along this stage, the role played by the data sources will be crucial to infer the conceptual relationships among concepts.
- Second stage validates each multidimensional graph according to multidimensionality. To do so, this method defines a set of constraints that must be preserved in order to place data in a multidimensional space and give rise to a data cube free of summarizability problems. The objective along this step is to enforce that concepts and relationships stated in the graph give rise as a whole to a data cube. If the validation process fails, the method ends since data demanded could not be analyzed from a multidimensional point of view. Otherwise, the resulting multidimensional schema is directly derived form the multidimensional graph.

Unlike data-driven methods, this approach focuses on data of interest for the end-user. However, the user may not know all the potential analysis contained in the data sources and, unlike requirement-driven approaches, it is able to propose new interesting multidimensional knowledge related to concepts already queried by the user. That is, it does not analyze the whole data sources but those concepts closely related to the end-user requirements. Finally, multidimensional schemas derived from a validation process are proposed. Therefore, like in (Hüsemann, Lechtenbörger & Vossen, 2000) and (Mazón, Trujillo & Lechtenbörger, 2007), schemas proposed are sound and meaningful.

Mazón et al. (2007) present a semi-automatic hybrid approach that firstly obtains the conceptual schema from user requirements and later verifies and enforces its correctness against data sources by means of Query/View/Transformation QVT relations. Their approach work over relational sources and requirements expressed in the *i** framework. This approach starts with a requirement analysis phase. They introduce a detailed demand-driven stage where it is argued that the user should state his / her requirements at a high abstraction level according to business goals and derive the information requirements from the information business goals. Goals and information requirements should be modeled by an adaptation of the *i** framework. The multidimensional conceptual schema must be derived from requirements and expressed in a UML extension that the authors provide.

Next, they propose a final step to check correctness of the conceptual multidimensional

model. The objective of this step is twofold: they present a set of QVT relations based on the *multidimensional normal forms* to align the conceptual schema derived from requirements with the relational schema of the data sources. Thus, output schemas will capture the analysis potential of the sources and moreover, they will be validated according to the MNF. The MNF used in this paper are an evolution of the ones used in (Hüsemann, Lechtenbörger & Vossen, 2000) and they share the same objective. Along five QVT relations that may be semi-automated, this paper describes how the conceptual multidimensional schema should be aligned to the underlying relational schema:

- 1MNF (a): A functional dependency in the conceptual schema must have a corresponding functional dependency in the relational schema.
- 1MNF (b): Functional dependencies among dimension levels contained in the source databases must be represented as aggregation relationships in the conceptual schema. Therefore, they complement the conceptual schema with additional aggregation hierarchies contained in the sources.
- 1MNF (c): Measures that can be computed from other measures must be identified in the conceptual schema. Therefore, they support derived measures.
- 1MNF (d): Measures must be assigned to facts in such a way that the atomic levels of the fact form a key. Said in other words, they demand to place the measure in a fact with the correct base.
- 2MNF and 3MNF: These constraints demand to use specializations of concepts when structural NULLs in the data sources do not guarantee completeness.

Song et al. (2007) present an automatic data-driven methodology that derives logical schemas from ER models. These approach presents a novel approach to automatically identify facts from ER diagrams by means of the *connection topology value* (CTV). The main idea underlying this approach is that facts and dimensions are usually related by means of many-to-one relationships. Concepts at the many-side are fact candidates and concepts in the one-side are dimension candidates. Moreover, it distinguishes among direct and transitive many-to-one relationships:

- First, this approach demands a preprocess to transform ER diagrams into binary (i.e. without ternary nor many-to-many relationships) ER diagrams.
- The CTV of an entity is a composite function of the topology value of direct and indirect many-to-one relationships where direct relationships have a higher weighting factor with regard to transitive ones. Thus, all those entities with a CTV value higher than a threshold are proposed as facts. Notice that facts are identified by their CTV and therefore, it would be possible to consider factless facts.
- For each fact entity dimensions are identified by means of many-to-one relationships. Moreover, the authors propose to use Wordnet and annotated dimensions (that represent commonly used dimensions in business processes) to enrich aggregation hierarchies depicted.

Romero & Abelló (2007) present a semi-automated supply-driven approach. This approach derives conceptual schemas from OWL ontologies that may represent different and potentially heterogeneous data sources. Thus, this method will derive multidimensional schemas from data sources of our domain that do not have anything in common but that they are all described by the same domain ontology. This approach consists of three well-differentiated tasks. In each step it automatically looks for a given multidimensional concept (facts, bases and aggregation hierarchies) by means of a fully supply-driven stage. A formal pattern expressed in Description Logics is presented at each step. Finally, at the end of each step the user selects results of his / her interest and this will trigger next steps:

- The first task looks for potential facts. Those concepts related to most potential dimensional concepts and measures are good candidates. At the end of this task, the user chooses his / her subjects of interest among those concepts proposed by the method. The rest of the tasks will be carried out once for each fact identified in this step (i.e., each fact will give rise to a multidimensional schema).
- The second task points out sets of concepts likely to be used as bases for each fact identified. Candidate bases giving rise to denser data cubes will be presented first to the user. Finally, it would be up to the user to select those bases making more sense to him / her.
- The third task gives rise to dimension hierarchies. For every concept identified as a dimension its hierarchy of levels is conformed from those concepts related to it by typical part-whole relationships. In this step, this approach builds up graphs giving shape to each dimension hierarchy and again, it will be up to the user to modify them to fit his / her needs.

Finally, this approach uses the same criteria as (Romero & Abelló, 2006) to validate the multidimensional schema.

COMPARISON OF METHODOLOGIES

In this section we present a detailed summarization of the main features of each methodology by means of two tables (see tables 1 and 2). Methodologies surveyed are distributed in these tables according to a chronological order. There, rows correspond to criteria introduced in section *comparison criteria* and columns correspond to each methodology studied. A given cell contains information for a methodology for a certain criterion. Most of the criteria are evaluated as *yes / no*, but some other have alternatives. Acronyms used to represent these alternatives may be found in figure 1 but two general alternatives can be found for any criterion: "-" means that this criterion does not make sense for that methodology whereas "*none*" means that none of the alternatives are considered.

Analyzing these tables we can find some interesting trends as well as assumptions that have been considered in most of the methodologies surveyed. First approaches tried to give context to multidimensional modeling providing tips and informal rules about how to design a multidimensional data warehouse. In other words, they presented the first guidelines to support multidimensional design. Later, when main concepts with regard to multidimensional modeling were set up new formal and powerful methodologies were developed. These new methodologies focused on formalizing and automating the process. Automation is an important feature along the whole data warehouse lifecycle and multidimensional design has not been an exception. Indeed, first methodologies were step-by-step guidelines but in the course of time many semi-automatic and automatic approaches have been presented. These evolution also conditioned the kind of inputs used, and logical schemas were considered instead of conceptual schemas. Nowadays, last methodologies introduced present a high degree of automation. Moreover, we may say that this trend also motivated a change of paradigm. At the beginning, most methodologies where demand-driven or, in case of being hybrid approaches, they gave much more weight to requirements than to data sources. However, with the time data sources gained relevance. This makes sense since automation has been tightly related to focusing on data sources instead of requirements. Consequently, last methodologies (which are highly automatable) mostly follow a supply-driven framework. Nevertheless, it is well assumed that the ideal approach to design multidimensional data warehouses must be a hybrid approach. In this line, some works that automate somehow their demand-driven stage have been presented. In these tables we can also realize the evolution of how the multidimensional model has been considered. First

Table 1. Summary of the comparison of multidimensional design methodologies

	[KRTR98]	[CT98]	[GR98]	[BvE99]	[HLV00]	[MK00]	[BCC+01]	[PD02]
General Aspects								
Paradigm	DD	IH	SH	SH	DD	SD	SH	SH
Application	G	G	S	G	G	G	S	S
Preprocess	None	DCS	None	ECS	None	DCS	None	None
Input Abstr.	C	C	C/L	C	C	C	C/L	L
Output Abstr.	L	L	C/L/P	L	C	L	L	C
Data Sources								
↪ Type	-	ER	ER/Rel	SER	-	ER	ER	Rel
↪ Analysis	-	RD	Full	Full	-	Full	Full	Full
↪ Patterns F.	-	None	Alg	None	-	None	Alg	Alg
Req. Expr.	Adhoc	Adhoc	Adhoc	Adhoc	Adhoc	-	Adhoc	MDX
Validation	No	No	No	No	MNF	No	No	No
Tool	No	No	Yes	Yes	No	No	No	No
Factual Data								
Facts								
Factless Facts	Yes	No	No	No	No	No	No	No
Requirements	Expl	Expl	Expl	Expl	Expl	-	Expl	No
Data Sources								
↪ C.Num.Val.	-	No	No	No	-	Yes	Yes	Yes
↪ Connectivity	-	No	No	No	-	No	No	No
↪ Cardinality	-	No	No	No	-	No	No	Yes
Semantic Rels.	-	-	Ass	-	Ag	-	-	None
Measures								
Requirements	Impl	Expl	Expl	Impl	Expl	-	Expl	No
Data Sources	-	No	NV	No	-	NV	NV	NV
Dimensional Data								
Fact-centered	No	No	Yes	Yes	No	No	Yes	Yes
Requirements	Expl	Expl	Expl	Expl	Expl	-	Expl	No
Data Sources								
↪ Func. Depend.	-	No	Yes	Yes	-	Yes	Yes	Yes
↪ Bases	-	No	No	No	-	No	No	No
↪ Others	-	No	No	No	-	No	No	Yes
Related								
Interdim.	None	-	None	-	None	-	-	None
Intradim.	L/D	L/D	L/D	L	L/D	L/D	L/D	L

approaches used to produce logical multidimensional schemas but with the time, most of them generate conceptual schemas. One reason for this situation could be that Kimball introduced multidimensional modeling at a logical level as a specific relational implementation. With the course of time it has been argued that it is necessary to generate schemas at a platform-independent level and in fact, the multidimensional design should span the three abstraction levels (conceptual, logical and physical) like in the relational databases field.

About the kind of data sources handled, most of the first approaches chose conceptual entity-relationships diagrams describing the data sources. ER diagrams were the most spread way to represent operational databases (the most common kind of data source to populate the data warehouse) but the necessity to automate this process and the need to provide up-to-date conceptual schemas to the data warehouse designer motivated that many methodologies worked over relational schemas instead of conceptual schemas. Almost every methodology either consider ER diagrams or relational schemas to describe the data sources. Lately, with the relevance gained by the semantic web area, some other works automating the process from XML schemas or OWL ontologies have been presented. About requirements, their representation have varied considerably. At the beginning, adhoc representations such as forms, tables, sheets or matrixes were proposed but lately, many methodologies propose to formalize requirements representation with frameworks such as UML diagrams or *i**. Moreover, some works have also proposed to lower the level of abstraction of requirements to the logical level by means of SQL or MDX queries which opens new possibilities of automation.

Finally, we can also identify a trend to validate the resulting multidimensional schema as well as the importance to provide a tool supporting the methodology.

About how to identify factual data, there are some trends that most approaches follow. Looking at the data sources, numerical concepts are likely to play a measure role whereas concepts containing numerical attributes or those with a high table cardinality once implemented are likely to play a fact role. First methodologies were mainly demand-driven but later, most of them used these heuristics to identify factual concepts within supply-driven stages. However, these heuristics do not identify facts or measures but concepts likely to play that role. Thus, requirements must be considered and in the last years requirements have gained relevance again to identify these concepts. Moreover, with the course of the time relationships among facts have gained relevance as well, since they open new analysis perspectives when considering multidimensional algebras. Finally, the reader may notice that despite Kimball introduced the concept of factless facts from the beginning it has been traditionally overlooked. Lately, some methodologies considered them again. One of the reasons could be that it is difficult to automate the identification of facts that do not have measures.

According to our study, dimensional concepts have been traditionally identified by means of functional dependencies. From the very beginning, some methodologies proposed to automate the identification of aggregation hierarchies. In fact, many methodologies use requirements to identify factual data and later they analyze the data sources looking for functional dependencies to identify dimensional data and maybe for this reason, the use of requirements to identify dimensional concepts has not been that relevant as to identify factual data. Another

Table 2. Summary of the comparison of multidimensional design methodologies

	[WS03]	[VBR03]	[JHP04]	[GRG05]	[PACW06]	[RA06]	[MTL07]	[SKD07]	[RA07]
General Aspects									
Paradigm	DD	SH	SD	SH	DD	IH	SH	SD	SD
Application	G	S	A	S	G	A	S	A	S
Preprocess	None	TCS	DM	None	ECS	None	None	TCS	None
Input Abstr.	C	L	L	C	C	L	C/L	C	C
Output Abstr.	C	L	L	C	C/L/P	C	C	L	C
Data Sources									
↪ Type	-	XML	Inst	ER/Rel	-	Rel	Rel	Rel	OWL
↪ Analysis	-	Full	Full	RD	-	RD	RD	Full	Full
↪ Patterns F.	-	None	Alg	None	-	Alg	QVT	None	DL
Req. Expr.	Adhoc	Adhoc	-	i^*	UML	SQL	i^*	-	-
Validation	No	No	AC	No	AC	MC	MNF	No	MC
Tool	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Factual Data									
Facts									
Factless Facts	No	No	No	No	Yes	Yes	No	Yes	No
Requirements	Expl	Expl	-	Expl	Expl	Impl	Expl	-	-
Data Sources									
↪ C.Num.Val.	-	No	Yes	No	-	No	No	No	Yes
↪ Connectivity	-	No	No	No	-	No	No	Yes	Yes
↪ Cardinality	-	No	Yes	No	-	No	No	No	No
Semantic Rels.	None	-	-	None	None	Ass/S	Ass/S	None	Ass/Ag
Measures									
Requirements	Expl	Expl	-	Expl	Expl	Impl	Impl	-	-
Data Sources	-	No	NV	No	-	No	No	No	NV
Dimensional Data									
Fact-centered	No	Yes	Yes	Yes	No	No	No	Yes	Yes
Requirements	Expl	No	-	Expl	Expl	Impl	Impl	-	-
Data Sources									
↪ Func. Depend.	-	Yes	Yes	Yes	-	Yes	Yes	Yes	Yes
↪ Bases	-	No	No	No	-	No	No	No	Yes
↪ Others	-	No	No	No	-	No	No	No	No
Related									
Interdim.	None	-	-	None	None	Ass/S	S	None	Ass
Intradim.	-	L/D	L/D	L/D	L/D	L/D	L/D	L	L/D

clear trend with regard to dimensional concepts is that, in general, the more automatable a methodology is, the more fact-centered it is. About relationships among dimensional concepts, interdimensional relationships (like relationships between facts) open new perspectives of analysis when considering multidimensional algebras. However, in this case they have been traditionally more overlooked than relationships between facts. Oppositely, intradimensional relationships have gained relevance from the very beginning. Most methodologies agree that distinguishing among dimensions, levels and descriptors is relevant for analysis purposes.

CONCLUSION

In this paper we provide an insight to the most relevant multidimensional design methodologies. This paper surveys 17 works that have been selected according to three factors: reference papers with a high number of citations, papers with novelty contributions and in case of papers of the same authors we have included the latest version of their works.

Since we still lack a standard multidimensional terminology and terms used among methodologies to describe the multidimensional concepts may vary, we have introduced a common multidimensional notation to avoid misunderstandings and facilitate the mapping of the surveyed methodologies to a common framework where to compare each approach.

We have also introduced a set of criteria to set a basis for discussion and detect trends such as features in common or the evolution of assumptions made along the way. These criteria were defined in an incremental analysis of the methodologies surveyed in this paper. For each methodology we captured its main features that were mapped onto different criteria. If a methodology introduced a new criterion, the rest of works were analyzed to know their assumptions with regard to this criterion. Therefore, criteria presented were defined along an iterative process during the analysis of the multidimensional design methodologies. We have summarized these criteria in three main categories: general aspects, dimensional data and factual data. General aspects refer to those criteria regarding general assumptions made in the methodology and dimensional and factual data criteria refer to how dimensional data and factual data are identified and mapped onto multidimensional concepts.

All in all, we have provided a comprehensive framework to better understand the current state of the area as well as its evolution.

ACKNOWLEDGEMENT

This work has been partly supported by the Ministerio de Educación y Ciencia under project TIN 2005-05406.

REFERENCES

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web. *Scientific American*.

Böhnlein, M., & Ulbrich-vom Ende, A. (1999). Deriving Initial Data Warehouse Structures from the Conceptual Data Models of the Underlying Operational Information Systems. In I. Song, T. J. Teorey (Eds.), *Proceedings of 2nd International Workshop on Data Warehousing and OLAP;* pp, 15-21. Kansas City, USA: ACM Press.

Bonifati, A., Cattaneo, F., Ceri, S., Fuggetta, A., & Paraboschi, S. (2001). Designing Data Marts for Data Warehouses. *ACM Transactions Software Engineering and Methodology*, *10(4):452-483.*

Cabibbo L., & Torlone, R. (1998). A Logical Approach to Multidimensional Databases. In H. Schek, F. Saltor, I. Ramos, G. Alonso (Eds.), *Proceedings of 6th International Conference on Extending Database Technology; Vol. 1377, Lecture Notes of Computer Science* (pp, 183-197). Valencia, Spain: Springer.

Codd, E. F., Codd, S.B., & Salley, C.T. (1993). Providing OLAP (On Line Analytical Processing) to Users-Analysts: an IT Mandate. *E. F. Codd and Associates*.

Giorgini, P., Rizzi, S., Garzetti, M. (2005). Goal-oriented Requirement Analysis for Data Warehouse

Design. In I. Song, J. Trujillo (Eds.), *Proceedings of 8th International Workshop on Data Warehousing and OLAP*; pp, 47-56. Bremen, Germany: ACM Press.

Golfarelli, M., Maio, D., & Rizzi, S. (1998a). The Dimensional Fact Model: A Conceptual Model for Data Warehouses. *International Journal of Cooperative Information Systems, 7(2-3):215-247*.

Golfarelli, M., Rizzi, S. (1998b). Methodological Framework for Data Warehouse Design. In In I. Song, T. J. Teorey (Eds.), *Proceedings of 1st ACM International Workshop on Data Warehousing and OLAP;* pp, 3-9. Bethesda, USA: ACM Press.

Google (Google, 2008). Google Scholar. Retrieved August, 8, 2008, from http://scholar.google.com/.

Hüsemann, B., Lechtenbörger, J., & Vossen, G. (2000). Conceptual Data Warehouse Modeling. In M. A. Jeusfeld, H. Shu, M. Staudt, G. Vossen (Eds.), *Proceedings of 2nd International Workshop on Design and Management of Data Warehouses*; pp 6. Stockholm, Sweden: CEUR-WS.org.

Jensen, M. R., Holmgren, T., & Pedersen, T. B. (2004). Discovering Multidimensional Structure in Relational Data. In Y. Kambayashi, M. K. Mohania, W. Wöß (Eds.), Proceedings of 6th International Conference on Data Warehousing and Knowledge Discovery; Vol. 3181, Lecture Notes of Computer Science (pp 138-148). Zaragoza, Spain: Springer.

Kimball, R. (1996). *The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses*. John Wiley & Sons, Inc.

Kimball, R., Reeves, L., Thornthwaite, W., & Ross, M. (1998). *The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing and Deploying Data Warehouses*. John Wiley & Sons, Inc.

Lehner, W., Albrecht, J., & Wedekind, H. (1998). Normal Forms for Multidimensional Databases. In M. Rafanelli, M. Jarke (Eds.), *Proceedings of 10th International Conference on Statistical and Scientific Database Management*; pp 63-72, Capri, Italy: IEEE.

List, B., Bruckner, R. M., Machaczek, K., & Schiefer, J. (2002). A Comparison of Data Warehouse Development Methodologies Case Study of the Process Warehouse. In A. Hameurlain, R. Cicchetti, R. Traunmüller (Eds.) *Proceedings of 13th International Conference on Database and Expert Systems Applications; Vol. 2453, Lecture Notes in Computer Science (pp* 203-215). Aix-en-Provence, France: Springer.

Mazón, J. N., Trujillo, J., & Lechtenborger, J. (2007). Reconciling Requirement-Driven Data Warehouses with Data Sources Via Multidimensional Normal Forms. *Data & Knowledge Engineering, 23(3):725-751*.

Microsoft (Microsoft, 2008). MDX Specification. Retrieved August, 8, 2008, from http://msdn.microsoft.com/en-us/library/aa216767.aspx.

Moody, D. L., & Kortink, M. A. (2000). From Enterprise Models to Dimensional Models: A Methodology for Data Warehouse and Data Mart Design. In M. A. Jeusfeld, H. Shu, M. Staudt, G. Vossen (Eds.), *Proceedings of 2nd International Workshop on Design and Management of Data Warehouses*; pp 6. Stockholm, Sweden: CEUR-WS.org.

Phipps, C., & Davis, K. C. (2002). Automating Data Warehouse Conceptual Schema Design and Evaluation. In L. V. S. Lakshmanan (Ed.), *Proceedings of 4th International Workshop on Design and Management of Data Warehouses*; pp 23-32, Toronto, Canada: CEUR-WS.org.

Prat, N., Akoka, J., & Comyn-Wattiau, I. (2006). A UML-based Data Warehouse Design Method. *Decision Support Systems*, *42(3):1449-1473*.

Romero, O., & Abelló, A. (2006). Multidimensional Design by Examples. In A. M. Tjoa, J. Trujillo (Eds.), *Proceedings of 8th International Conference on Data Warehousing and Knowledge Discovery; Vol. Lecture Notes of Computer Science (pp* 85-94). Krakow, Poland: Springer.

Romero, O., & Abelló, A. (2007). Automating Multidimensional Design from Ontologies. In I. Song, T. B. Pedersen (Eds.), *Proceedings of ACM 10th International Workshop on Data Warehousing and OLAP*; pp 1-8, Lisbon, Portugal: ACM Press.

Romero, O., & Abelló, A. (in press). MDBE: Automatic Multidimensional Modeling. *In Proceedings of 27rd Int. Conf. on Conceptual Modeling*.

Song, I., Khare, R., & Dai, B. (2007). SAMSTAR: A Semi-Automated Lexical Method for Generating STAR Schemas from an ER Diagram In I. Song, T. B. Pedersen (Eds.), *Proceedings of ACM 10th International Workshop on Data Warehousing and OLAP;* pp 9-16, Lisbon, Portugal: ACM Press.

Vrdoljak, B., Banek, M., & Rizzi, S. (2003). Designing Web Warehouses from XML Schemas. In In Y. Kambayashi, M. K. Mohania, W. Wöß (Eds.), *Proceedings of 5th International Conference on Data Warehousing and Knowledge Discovery; Vol. 2737, Lecture Notes of Computer Science* (pp 89-98). Prague, Czech Republic: Springer.

Winter, R., & Strauch, B. (2003). A Method for Demand-Driven Information Requirements Analysis in DW Projects. *In Proceedings of 36th Annual Hawaii International Conference on System Sciences*; pp 231-239. Hawaii, USA: IEEE.

Alberto Abelló has a MSc and a PhD in computer science from the Universitat Politècnica de Catalunya (Polytechnical University of Catalonia). He is associate professor at the Facultat d'Informàtica de Barcelona (Computer Science School of Barcelona). He is a member of the GESSI research group (Grup de recerca en Enginyeria del Software per als Sistemes d?Informació) at the same university, specializing in software engineering, databases and information systems. His research interests are database design, data warehousing, OLAP tools, ontologies and reasoning. He is the author of articles and papers in national and international conferences and journals on these subjects.

Oscar Romero has a MSc in computer science from the Universitat Politècnica de Catalunya (Polytechnical University of Catalonia). Currently, he is a PhD student at the same university and an assistant professor at the Escola Tècnica Superior d'Enginyeria Industrial i Aeronàutica de Terrassa (Industrial and Aeronautical Engineering School of Terrassa). He is a member of the GESSI research group (Grup de recerca en Enginyeria del Software per als Sistemes d'Informació) at the same university, specializing in software engineering, databases and information systems. His research interests are database design, data warehousing, OLAP tools, ontologies and reasoning. He is the author of articles and papers in national and international conferences on these subjects.