LABO
EQUIPE DE RECHERCHE
EN INGENIERIE
DES CONNAISSANCES

3038 Université Lumière Lyon2
Bâtiment L
5, Avenue Pierre Mendès-France
69676 BRON Cedex FRANCE

UNIVERSITE
PIERRE & MARIE CURIE
LA SCIENCE A PARIS

Université Pierre et Marie Curie
Mention Science et Technologie du logiciel
4 PLACE JUSSIEU
75005 Paris, FRANCE

Modeling complex data in the MAP data warehouse

Internship report for Master-2

M2-STL YEAR 2005/2006

Internee :
Syed Fawad HUSSAIN

Supervisor :
Jérôme DARMONT

Acknowledgement

I would like to thank all the people who have helped me in every possible way to be able to finish this internship in time.

First and foremost, Mr. Jérôme DARMONT, director of the ‘departement d’Informatique et Statistics’, and head of research group ‘Base de Données Décisionnelle (BDD)’ at ERIC research labs, for his support and supervision in making this possible. He was there all the way through in giving me guidance, and supporting me.

To Mrs. Michele SORIA, responsible of the department ‘Science et technologie du logiciel (STL) ’, at the university Pierre et Marie Curie Paris 6.
I would also like to thank all the previous internees, who had started the MAP project and made quite an effort to put it up running.

Finally to all my teachers at Paris 6 university and the entire staff of ERIC laboratory for their warm welcome.

Syed Fawad HUSSAIN

Table of Contents

List of figures

Abstract

Healthcare presents unique challenges for the architect of a data warehouse. Medical data about a large patient population is analyzed to perform healthcare management and medical research. The MAP architecture propose a data warehouse based approach as a suitable solution for the integration of Health care for high-level sportsmen and warehouse applications as decision support tools for strategic decision making. This report proposes a cardio-vascular data mart for the MAP warehouse. Medical data such as that of cardiology contains complex data. We have also proposed a metamodel based on cardio-vascular data mart design that can be used for modeling complex data in other data marts of the MAP warehouse as well as other complex data in similar situations.

1. Introduction

With the rapid advances in the computerization of data, medical science could rest as one of its most needy beneficiary. Today computer technologies have begun to mature and have opened to us, a vast range of applications. Storing a large amount of information in a central location (databases) could open the door to maintaining complete medical information, personalized medical histories as well as other technologies that could take medical science to new heights. A large data warehouse could lead us to newer approaches of diagnosis not as common as with the keeping of paper files.

Some of the advantages of a medical data warehouse could be as follows:

1. Health care providers and insurance companies could use information networks to share electronic medical records. These data banks would help with the reduction of paperwork, in billing, identifying the most cost-effective treatment, and to fighting against false claims.
2. A person's medical information would be immediately available to doctor. Therefore in case where a new doctor sees a patient, he/she would have the patient's entire medical history at their fingertips. Included in this information could be life saving information that would be invaluable to the attending doctor.
3. The creation of such a data warehouse would also allow researchers to follow certain diseases as well as to patients' responses to their treatments. This information could be valuable to drug companies for research purposes only.
4. The creation of these databases would allow for better organization and more legibility of medical files.
5. Since elaborate security systems can be developed to monitor these medical databases, electronic records may actually be more secure than paper records.

1.1 Context

The area of data analysis developed enormously during the past years. Companies realized of the effectiveness of OLAP technology (Online Analytical Processing) in the analysis and the exploration of the data. This technology is used in the computerized decision-making systems. These systems are based on techniques of storage of data to exploit the information available therein and enable the companies at the ends to base their decisions on the analysis.

In recent years, medical professionals are witnessing an explosive growth in data collected by various organizations and institutions. At the same time, the ongoing developments of networking technologies provide doctor with the capability to access these data across the boundaries of interconnected computers.

Medicine is becoming more and more dominating in the field of the sport of high-level competition. As such, today we need more complex and complete tools to be able to cope with the. The objective is to be able to diagnose early and to improve the output of the sportsman

Clinical databases have accumulated large quantities of information about patients and their medical conditions. Relationships and patterns within this data could provide new medical knowledge. Unfortunately, few methodologies have been developed and applied to discover this hidden knowledge.

The manner most adapted to facilitate this analysis using OLAP is the multidimensional modeling of the data. Contrary to the relational models, entity/association or direct-object, the multidimensional models are adapted to make the analysis and to facilitate the decision-making in the companies.

The objective of the MAP data warehouse is to be used as a structure in the assistance of the diagnosis of player in the area of high-level sport. It is expected that it will be extended in the long term to general public. It is also used as structure of storage from a more static point of view in order to be able to be used, if necessary, to bring the proof of the last health of a patient.

Indeed, not only is it important to be able to suggest diagnosis using automated tools, but also we need to store historical data for other purposes, like legal case. Many incidents have surfaced over the years when serious or fatal injuries to sportsment had been linked to the overall state of health- an example is the heart related incident of a famous Nigerian footballer, Kanu Nwanko in the mid nineties. With the help of such data warehouses like MAP, sports physicians can keep a vigilant eye on the state of health of their top sportsmen, and such documents like the cardiogram, can be used as proof of potential dangers. It is the medico-legal aspect of the warehouse. This second aspect relates to in particular the cardiovascular module. Indeed, within the framework of intense physical activities, the cardiac system is put at hard test.

1.2 About ERIC lab:

Located at one of the best universities in France at Lyon, ERIC (*Equipe de Recherche en Ingénierie des Connaissances*) works hard to promote research in areas such as knowledge engineering especially the automated retrieving of knowledge. It was created in 1995 to develop methods and tools aiming at the field of knowledge engineering. The treated data can be texts, images, quantitative or qualitative numerical data. These automated tools would aid in retrieving, validating and organizing data from large data warehouses.

More particularly, my scope with this area has been focused on complex data. Medicine, as we know, is a complex field and the data generated by medical tests and diagnoses are also complex. Handling of data of this nature requires slight deviation from the norm to enable efficient storage, retrieval and usage.

The concept of MAP warehouse was put forward by Dr. Ferret when he felt the need of storing data for high-level sportsmen, like footballers, and its analysis. This developed a partnership between Dr. Ferret and ERIC lab of Lyon2 University, with the goal of realizing this concept into reality.

Other internees like Olivier, Georgel and Djallal, have made a good contribution in this regard in putting together many data marts for this eventual warehouse. The have provided the base on which myself and others can work and improve.

1.3 Objective

As mentioned, part of this project has been put up on prototype basis. This includes the biological, biometrical and cardio-vascular data marts. Data warehouses are huge projects that usually undergo several transformations before they evolve in their final form. Part of the reason is that with each implementation, new types of objectives are included and new shortcomings are noticed.

The overall objective of the MAP warehouse is to enable doctors, especially in the field of professional sports to be able to use this to their aid, both in terms of a reference for diagnosing patients(or sportsmen in this case) and updating their day to day status and recording their medical tests and exams. This daily reference would provide for methods to react in a proactive way by early recognition of symptoms that are likely to emerge.

The most important part of this warehouse, to date, has been the cardio-vascular data mart partly owing to its importance and also its complex nature. Naturally, it is the most difficult to implement. This single mart has already gone through 2 evolutions. The complex nature of the data that involves multiple formats and a large number of many-to-many relationships require a different approach in handling.

As part of my internship at the ERIC lab on this, I had to both improve on the current cardio-vascular mart as well as propose a meta-schema or a metamodel that would enable us to better model other marts of similar nature both in MAP and elsewhere. As I mentioned, the cardio-vascular mart has already been implemented, but is not in its present form, well enough to serve in the warehousing project.

Before starting on the real work, one has to be well versed with the basics- both the subject as well as the case of study. In this case, it meant both the technologies of data warehousing and multi-dimensional modeling, as well as the complete architecture of the MAP project.

This is not the only case where complex data would be involved in this context. Hence, another objective was to develop a metamodel that would enable us to use, in future, techniques for modeling complex data in similar situations. This adds the concept of modeling and meta modeling to the literature, I had to learn in order to be able to accomplish this goal.

This report is divided into the following sections- a set of definitions about terminologies to better understand the concept. In section 3, I have given a brief introduction to other medical warehouses. Section 4 revolves around the MAP architecture. Section 5 introduces the existing model and metamodels before proposing the new model for cardio-vascular data mart and a corresponding metamodel for handling complex data.

In section 6, I have concluded on the objectives achieved and a perspective of future extensions.

2. Definitions

2.1 Data warehousing vs. Data Marts

The data used for analytical processing is usually organized in a data warehouse. According to [1],

> "A data warehouse is a subject oriented, integrated, non-volatile, and time variant collection of data in support of management's decisions".

In other words a data warehouse is used as a foundation of a decision support system [15]. It encompasses a set of specialized versions of the domain, called data marts. A warehouse may be built entirely from scratch or may evolve from set of marts.

A **data mart (DM)** is a specialized domain of a data warehouse (DW). The key difference is that the creation of a data mart is predicated on a specific, predefined need for a certain grouping and configuration of select data.

2.2 Star schema

A star schema consists of fact tables and dimension tables. *Fact tables* contain the quantitative or factual data about a business--the information being queried. This information is often numerical, additive measurements and can consist of many columns and millions or billions of rows. *Dimension tables* are usually smaller and hold descriptive data that reflects the dimensions, or attributes, of a business. SQL queries then use joins between fact and dimension tables and constraints on the data to return selected information. Performance is an important consideration of any schema, particularly with a decision-support system in which you routinely query large amounts of data.

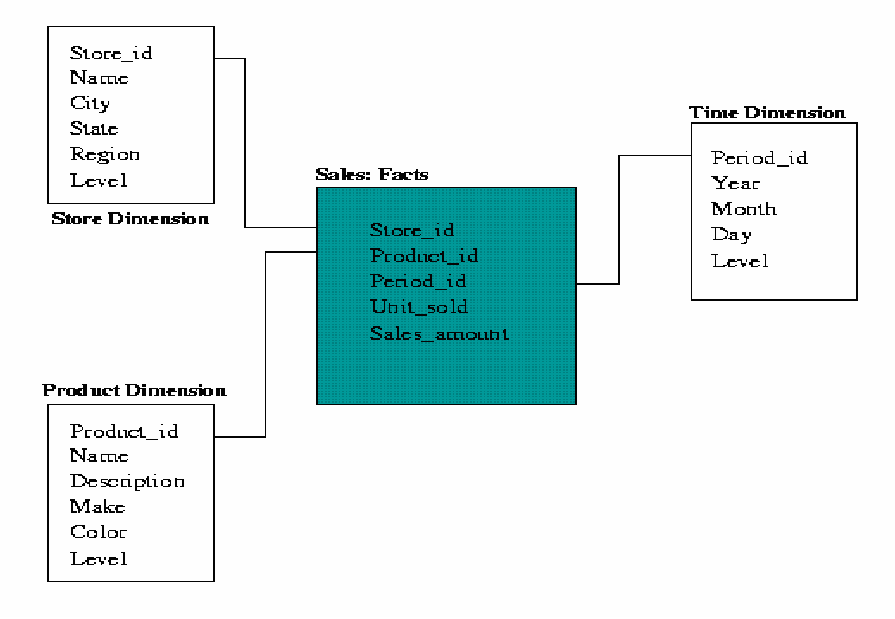

Figure 1: An example of star schema [14]

2.3 Snowflake schema

Snowflake schema is based on the approach to resolve the high cardinality problem faced in database schema design. Snowflake schemas also separate the hierarchy into its own sub-dimension table. This approach is good for that do not have very large quantities of data since otherwise, the relatively larger number of joins would “kill” the queries.

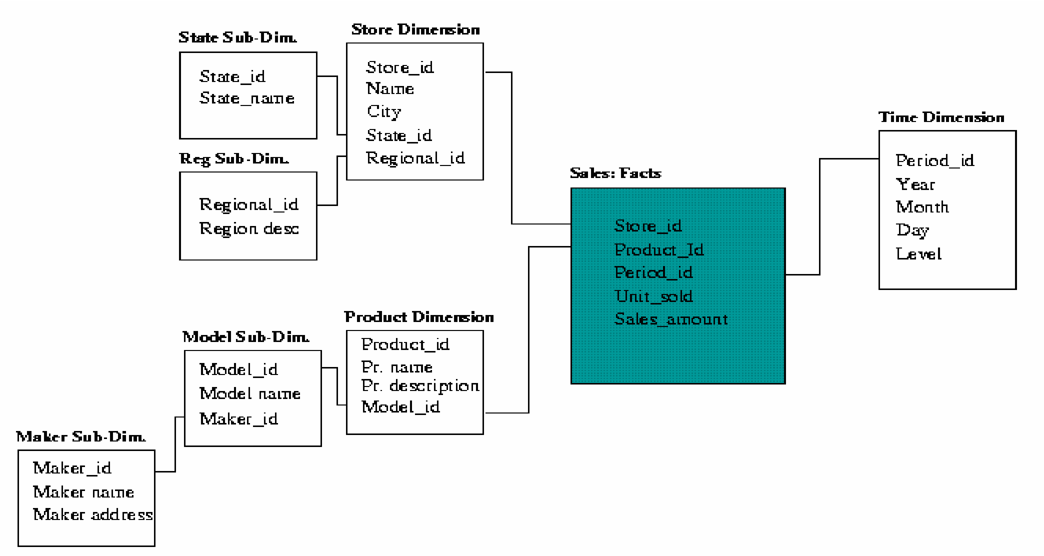

Figure 2: Snowflake schema [14]

2.4 Constellation schema

Maintaining hierarchy dimension in star schema is somewhat burdensome. The constellation approach is based on handling the hierarchical dimension problem in the data warehouse by using multiple fact tables to separate the detail and the aggregated values. Each level in a focus dimension is associated to a fact table.

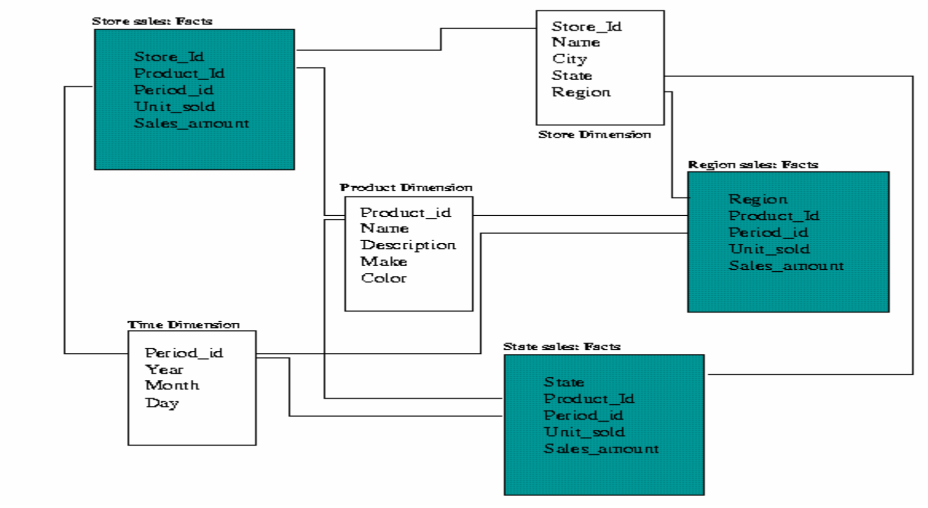

Figure 3: Schema in constellation [14]

3. State of the Art Biomedical Warehouses.

The evolution of data warehousing was developed as a means to storing enormous amounts of data, and has been primarily advocated to business needs and processes and the product development of data warehouses [19]. Data warehousing imposes itself as an attractive solution for centralizing and analyzing high quality data. Data warehousing may be considered a set of materialized views. These views may be based on autonomous, heterogeneous and distributed data sources [20]. This definition gives way to a broader perspective to look at a data warehouse, rather than just a multi-decisional support system.

In the medical research field, this technology can be used to validate assumptions and to discover trends on large amount of patient data. One important goal in bioinformatics is to integrate data from disparate sources of heterogeneous biological information. Biological data can be challenging due to the volume and complexity of the data types. Distributed search space is difficult to process there is a need for a data integration solution that facilitates search and retrieval in an efficient, manner.

There are lots of complications in the integration of heterogeneous data sources like these- like differences in data models, schemas, naming conventions, and levels of granularity used to represent data that are conceptually similar [21]. They have given 4 broad conflicts in databases:

- The heterogeneity conflict
- Semantic conflict
- Descriptive conflict
- Structural conflict

Problem 1 has a know solutions as in providing a query integrator, that can divide queries into sub queries. These sub queries can then be sent to the heterogeneous data sources (e.g. Relational data model and XML data model). The result can then be integrated and returned in the required format.

Problems 2 and 3 are more diversified and do not have any well-matured solutions [21]. Semantic conflict deals with using the same term to describe two logically different concepts, or vice versa. Descriptive conflict has to do with differences in domain scope, scale, etc. These conflicts arises more in the area of medical sciences, since certain essentially medically different concepts may be broadly described as the same e.g. scan, which can be different in different data sources like X-ray scan, ultrasound, etc.

In their approach to maintenance of integrated biological data[18], the authors argue that typical data warehouse environments are concerned about data from within a company and hence heterogeneity is low, as is autonomy since the different sources can be tailored for adjust in the data warehouse environment. They have considered this view, as comparable to a biological data warehouse. They data for a biological warehouse is obtained from remote source- like the publicly available data on the internet. The have considered two databases- the protein sequence database SWISS-PROT, as well as the classification database PROSITE. Their study deals with integration of protein sequences using protein classification.

Of most interested to our area of study is the model of data warehouse used. Since the data sources, are remotely accessed, a local copy of the data is of interest is maintained. This local copy is integrated into a view, for querying purposes. Local data is maintained for efficiency. When fresher data is required, the local copy is updated.

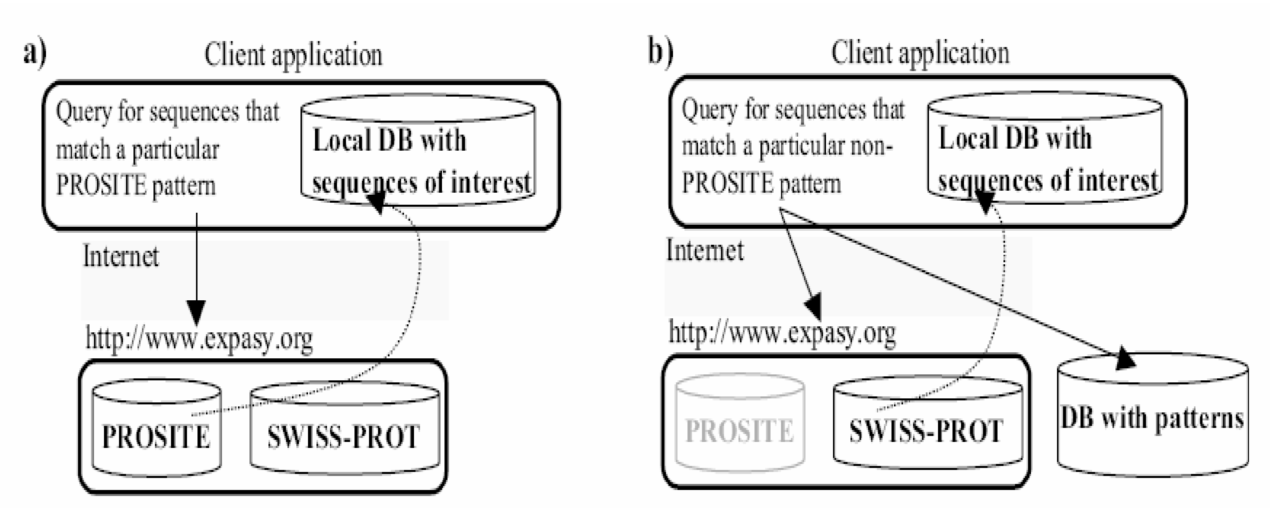
a)
Client application
Query for sequences that match a particular PROSITE pattern
Local DB with sequences of interest
Internet
http://www.expasy.org
PROSITE
SWISS-PROT
b)
Client application
Query for sequences that match a particular non-PROSITE pattern
Local DB with sequences of interest
Internet
http://www.expasy.org
PROSITE
SWISS-PROT
DB with patterns

Figure 4: Two examples of how data can be integrated from SWISS-PROT [18]

Figure 1a, is a situation where the client is only interested in protein sequences. Figure 1b, is a situation where client wants to retrieve sequences that match patterns in another database.

The architecture for the system is as show below. The application, TMID (Test bed for maintenance of integrated data), is used to retrieve data from the remote sources, and integrate it in a form available to clients. The application makes use of wrappers and integrators, and uses XML as the common data model.

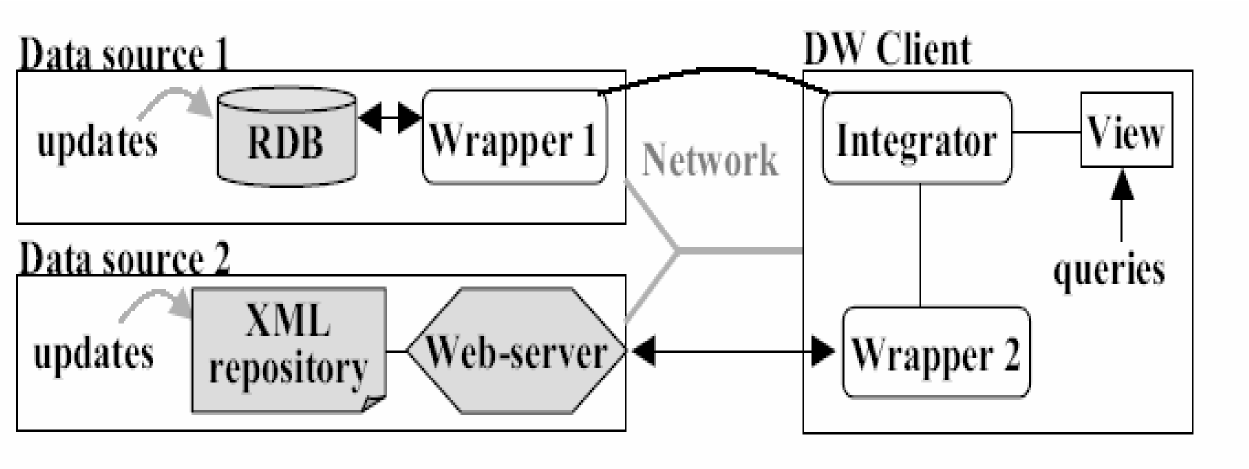

Figure 5: The TMID architecture [18]

Figure 2 shows the architecture. As can be seen, the two data sources are both heterogeneous and distributed. The first source is in the relational data model while the second in an XML repository. Wrappers are used to convert the data to a format that is more easily integrated. The integrator is the most complex component of the architecture that integrates the two data sources. The client creates a view on the integrated database. The integrator uses hash joins, and nested loop joins to integrate the data.

In an attempt to integrate vast amounts of data generated from biomedical experiments [21], a federated database system approach with a centralized mediator is used. The project is centered around 3 data bases, the SenseLab, a Human Brain Project neuroscience database at Yale, the Cell Centred Database, a database at University of California at San Diego and the CoCoDat database, built at the C. &. O. Vogt Brain Research Institute in Dusseldorf, Germany.

The authors here have taken the database federation approach, in which we create a large virtual database. This large virtual database takes into account the structures and contents of the smaller constructing units. To a client, the global view or interface is that of the federated database through a central mediator. The global query is sent to the mediator, which in turn, converts it to a set of sub queries that conform to the individual schemas of the local sources. The queries are run on the local sources, hence guaranteeing the freshness of data. The results are then recombined and integrated by the mediator. The final result is sent back to the client user as if his query was run on the global schema.

Similar efforts have been made in TAMBIS and IBM’s DiscoveryLink. TAMBIS (Transparent Access to Multiple Bioinformatics Information Sources) is based on a similar approach but tends to concentrate more on integrating sources with complementary data [22].

Architecture:

As described above, the system is organized in a hierarchical structure: At the top-most level is the Mediator. At the bottom end is the local data sources that contain the actual data on which the queries would be run. The client user interacts with the mediator. In fact, the client has no knowledge of the existence or structure of the individual local databases.

Mediator

It is a logical abstraction of the local database structure and presents an abstract integrated view. The mediator comprises the following components:

Global schema: It presents the contents of the data sources of the resources of the federation in a logical, conceptual manner. The global schema is based on the semantics of the domain rather than that of the structure of the resources.

Rules: these are rules for mapping the global schema onto the local schema of the federating units, and vice versa. These rules contain, term mapping and structural mapping.

Program

In order for the mediator to perform its tasks, it needs several programs. These are the *query parser*, which parses the query against the global schema. The next phase is the *query transformer* that disintegrates the query to be run against the local schemas. Finally the mediator needs a *result integrator* that would recombine the subsets of local results obtained against the local databases.

Wrapper

These are specific converters. A wrapper takes the transformed query from the mediator and converts it to the format of the local data model.

Yet another biological data warehouse is presented in [23]. Atlas stores and integrates biological sequences, molecular interactions, homology information, functional annotations of genes, and biological ontologies. The data in atlas is divided into 5 main parts: 'sequence', 'molecular interactions', 'gene related resources', and 'ontology'. The source data has been divided to fall in one of these categories. The database schema is based on a relational model. Atlas has both local as well as global ontologies i.e. within and between systems.

As can be seen in the diagram below, the atlas data source comprises the various sources of biological data. These are loaded into the atlas database schema, using the loader. Within the schema itself, the data base is divided into categories, so that each source data is related to a certain category. The high-end retrieval is based on SQL, API’s and other tools.

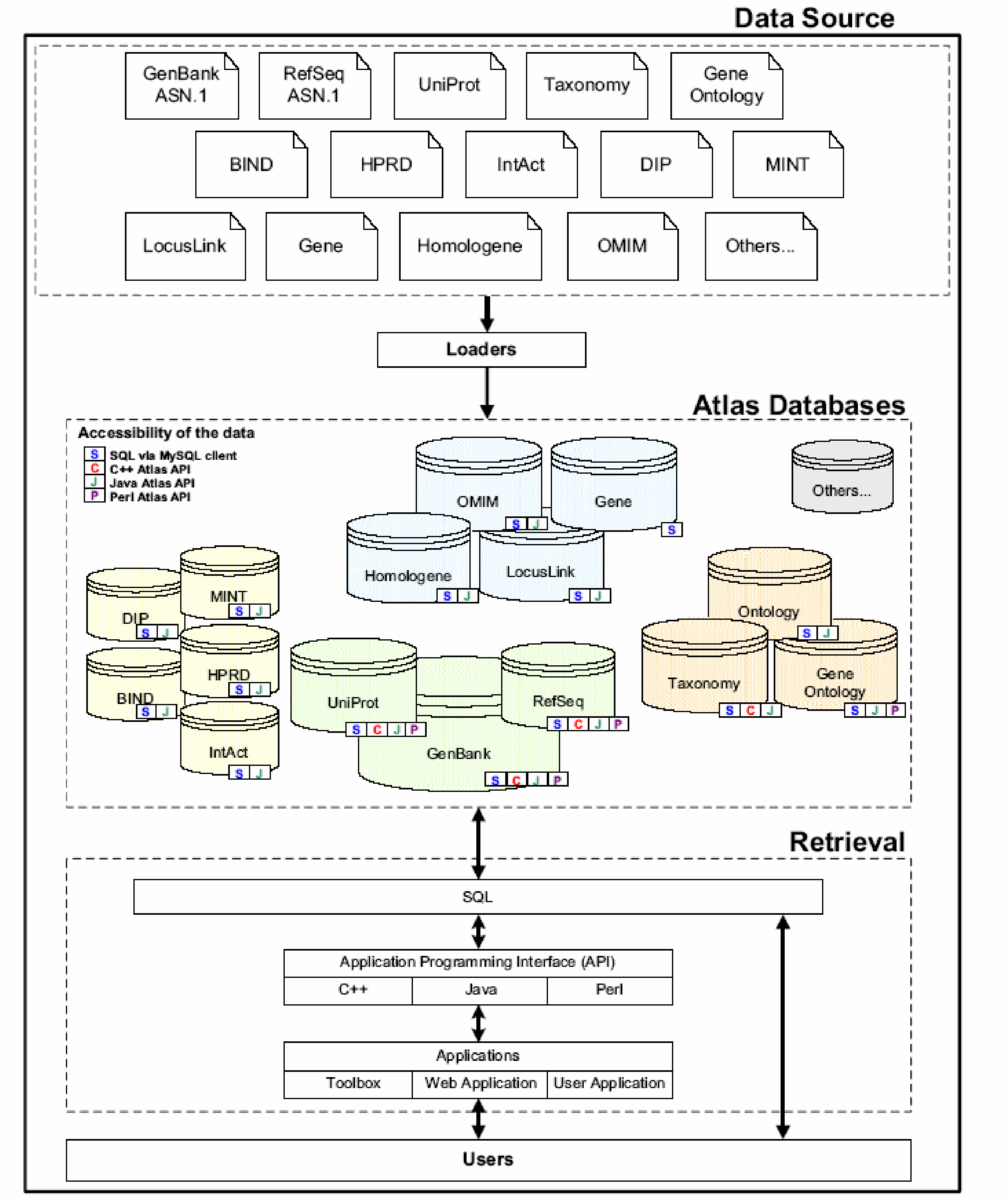

Figure 6: The Atlas data warehouse architecture [23].

Interesting to note here is that the categories or sequences are not overlapped, but are separately maintained. This can be seen in the atlas database schema in figure 7.

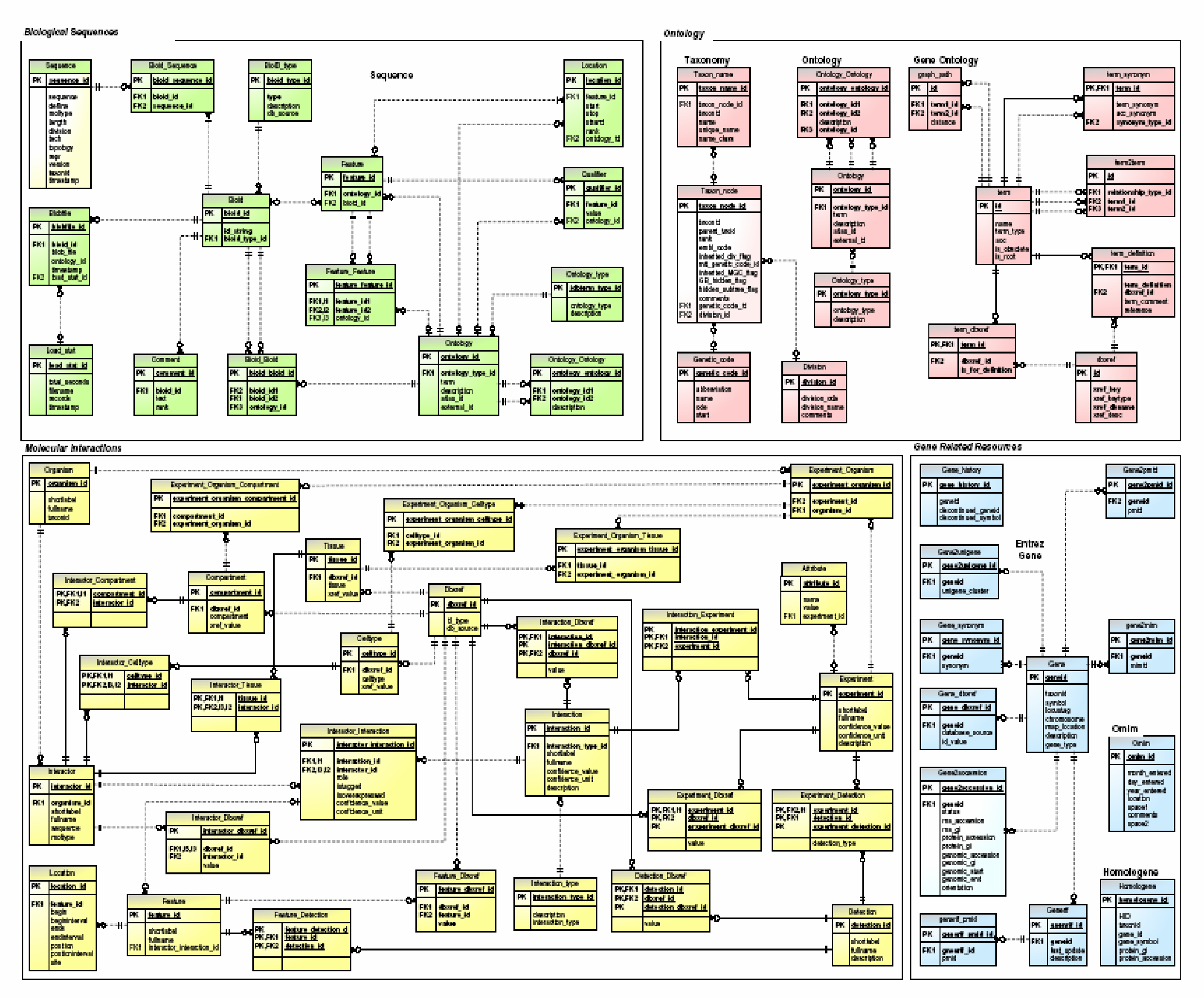

Figure 7: The atlas database schema [23]

The four functional groups, or sequences are the biological sequence, molecular interactions, gene related resources, and ontologies.

Other biomedical warehouse approaches [16, 17] have also been proposed for dealing with patient care and genome respectively.

4. Medical Data warehouse (MAP)

Based on the objectives discussed above, the laboratory of research of ERIC in collaboration with Dr. Ferret has proposed a data warehouse for medical patients. This data warehouse is basically aimed at providing personalized health care system for athletes and sportsmen.

This project is financed partly by the university Lyon 2 and partly by the Rhone-Alps area. The objective is to extend the results and empirical projections developed for the high level sportsmen with other populations and to help personalize the health care system of these sportsmen. This work is founded on the structuring, the storage and the analysis of a whole of complex medical data (qualitative, numerical, texts, images…).

4.1 The MAP design structure

There are various approaches to dealing with problems of large scale databases. The process of data warehouse design includes activities starting from identifying end-user requirements, specifying project boundary, identifying subject area data model, developing the warehouse logical data model, selecting the physical databases up to testing the fully-developed data warehouse.

The warehouse of medical data MAP is organized in the form of a collection of Data Marts. Each mart contains the specific data, concerning the medical specialty (for example biological analyses, biometric, cardiovascular, etc). These data marts are defined by a set of facts shared with other data marts. A multidimensional modeling of the biological store and biometric store were carried out by Olivier, during the initiation phase of the project [1, 6]. The data of these two modules were primarily textual or numerical.

Later during the development of the cardiovascular data mart, data that cannot be represented as simple numerical and text forms in single fact tables were encountered. A typical cardiovascular data store a set of complex information as would be shown later. These kinds of data which are more complex in nature, and may contain in addition to the textual and numerical data, images, videos and the written conclusions.

Overall, the structure adopted for the data warehouse is a bus architecture since we are starting with a couple of data marts and provisions for new successive entries. For example, the cardio-vascular data marts have been added to the two initial existing marts of biological and biometrical data. The use and advantage of using such a structure is to enable us use the commonalities between the marts e.g. each individual has a personal (individu) account that stores his /her personal details and activities. Such a table will be common amongst all the marts and being able to share it is a necessity of the warehouse.

Bus architecture also provides us with ease of integration as we would see from the figure below and the architecture of the cardio-vascular data mart in the subsequent section.

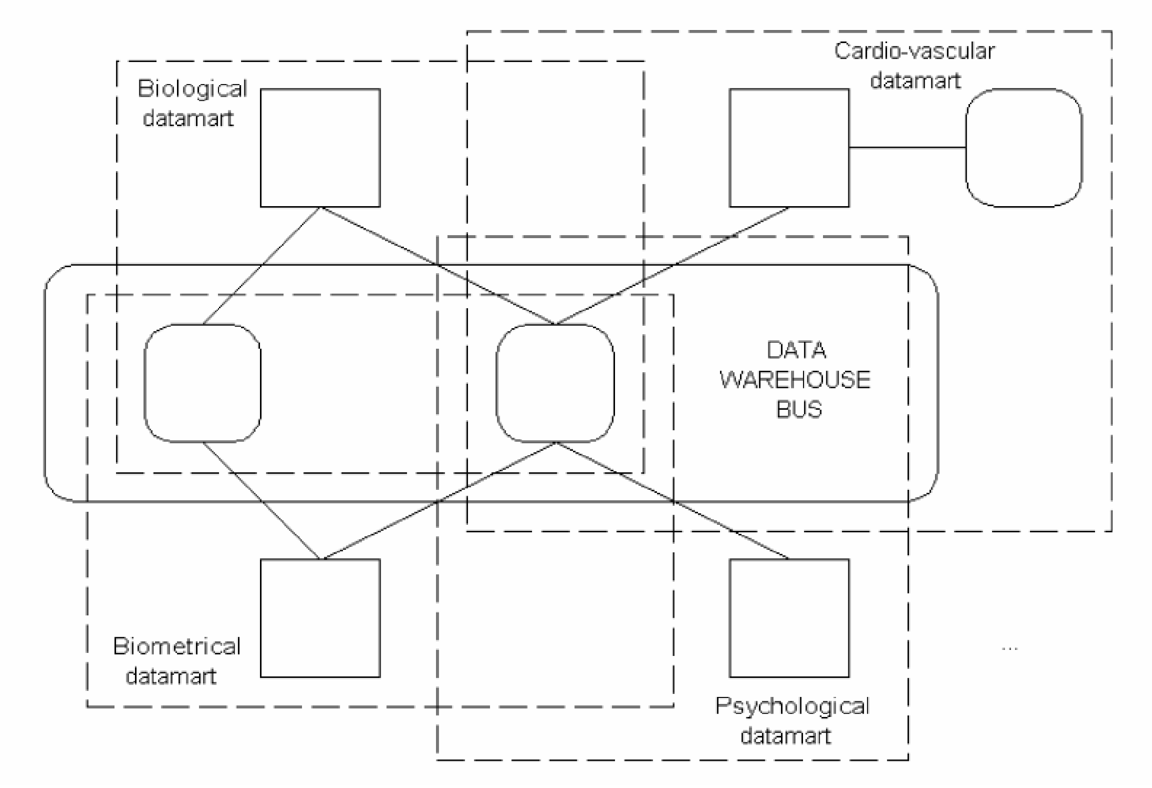

Figure 8: The MAP architecture [5, 6]

The rounded rectangle shows the bus architecture that allows us to integrate the other data marts into existing ones. The biological and the biometrical data marts are both up and running. For reference to their design, please refer to [1].

4.2 The Cardio-Vascular Data Mart

As can be seen from the architecture of the MAP model, there exists several data marts that would together contribute to the creation of the MAP warehouse. Previously, other data marts like biological data mart have already been implemented. Another internee, Georgel, had tried to implement the cardio vascular model. This is the first attempt to integrate the cardio vascular model into the warehouse.

He had started with an initial model and revised it successively to better represent the relationships. While this model is far better than the original version, there exist several drawbacks/shortcomings that could be improved to this model in order to enhance its performance and its capabilities in dealing with a wider range of queries.

The important idea here is in the table Compte_Rendu, which acts like a fact table and a dimension table. If we look at the Resultat_Cardio_Num table, then we have a 1:M relationship between the two tables and here Compte_Rendu acts like a dimension table. On the other hand, considering the table Individu, the table Compte_Rendu is on the many side and acts as a fact or transaction table.

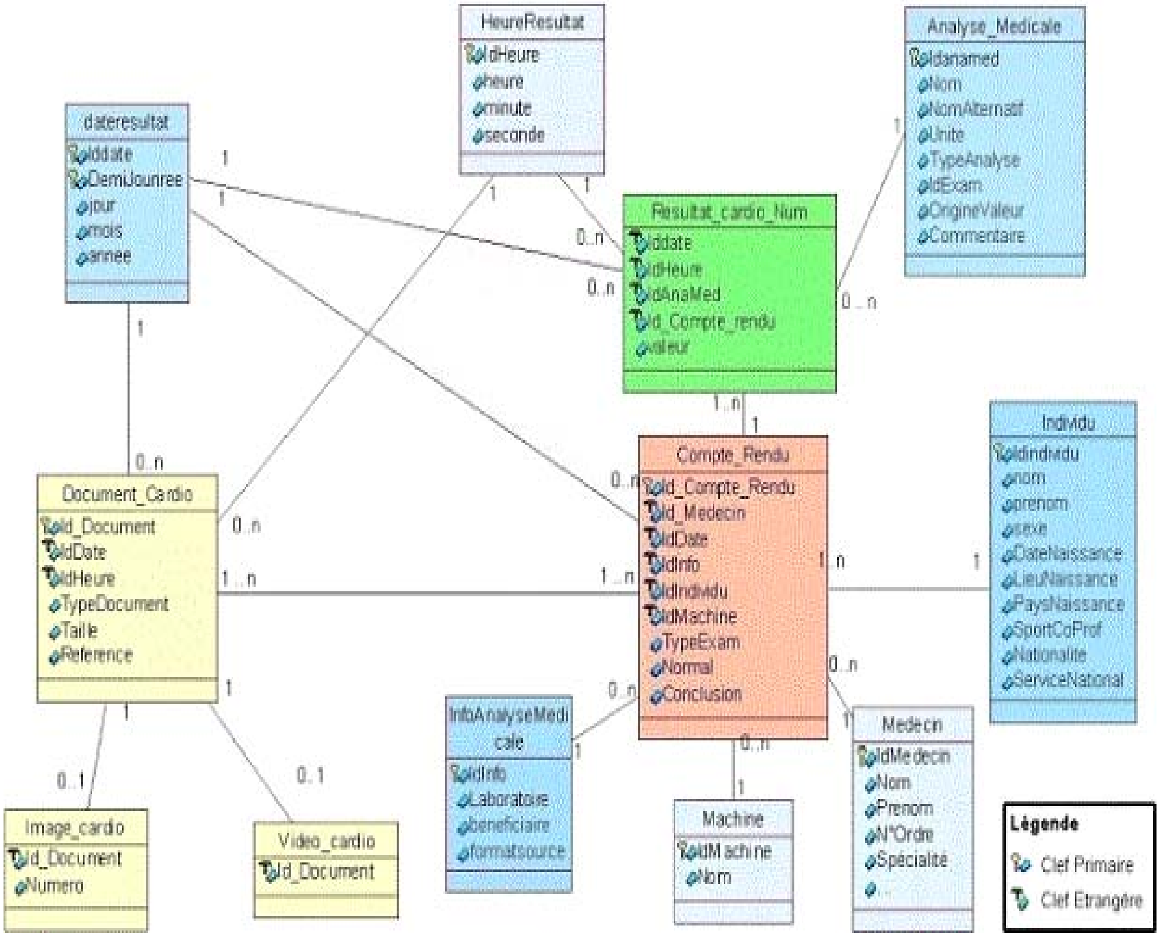

Figure 9: The existing cardio-vascular data mart [2]

Even though this is a very good effort to model the data mart, there are some aspects which would require a redesign. First of all, as discussed earlier in our definition of star schema and snowflake schema, the question of hierarchy is one of the most important one, especially in modeling warehouses. It reflects in the performance of queries that would require hierarchical splitting of the data. To this effect, the hierarchy of time is missing in this diagram. As noted in the diagram, there can be several files, videos or images that will relate to cardio vascular exams, and there can be successive exams at short time intervals.

Yet another point is the breaking down of the time dimension. The tables DateResultat and heureResultat are both dimension but with no corresponding relationship. This in effect could mean that if we would ever like to see records, with both date and time constraints, there would be no easy solution. Consider the case where a certain lab assistant responsible for conducting cardio-vascular tests, who works in a shift at certain dates and at certain hours, had actually conducted erroneous tests. We would like to correct the values or at least ignore those results when mining our data. With breaking down these dimensions, we are cutting the easy traversal path.

Another model, which aims to resolve some of these issues, has been proposed by Djallal [3]. In his approach where he has tried to improve on this schema as well as to go on and propose a metamodel for the other data marts of the MAP warehouse, he has used the hierarchical approach of a snowflake schema.

First of all the many to many relationship between the Compte_Rendu and the Document_Cardio tables was resolved by introducing a new intermediate table known as Groupe_Doc that would contain the documents relating to an exam. This is a conceptual model and such conflicts are resolved during implementation or at a much deeper level. However, the basic way to resolve a many to many relationship is to introduce a primary key to the new table, and the foreign keys of the original tables- in his case these are missing. Consider the figure below: how can we find the corresponding documents relating to an exam?

On the hierarchical dimension of time, the finer dimensions of time have been omitted. In the previous schema, we had the time dimension in minutes and seconds which seems to be missing in the newer version. But yet more important is the fact that the time dimension, which was linked to the Document_cardio, as well as Compte_Rendu and Resultat_Cardio_Num, is missing. This leads us to a situation where we cannot traverse documents on the basis of time dimension.

In the newer version there is no link between the date related tables and the documents being stored (document cardio) which was present previously and is also a requirement (since we can have new/revised prescriptions based on previously stored documents).

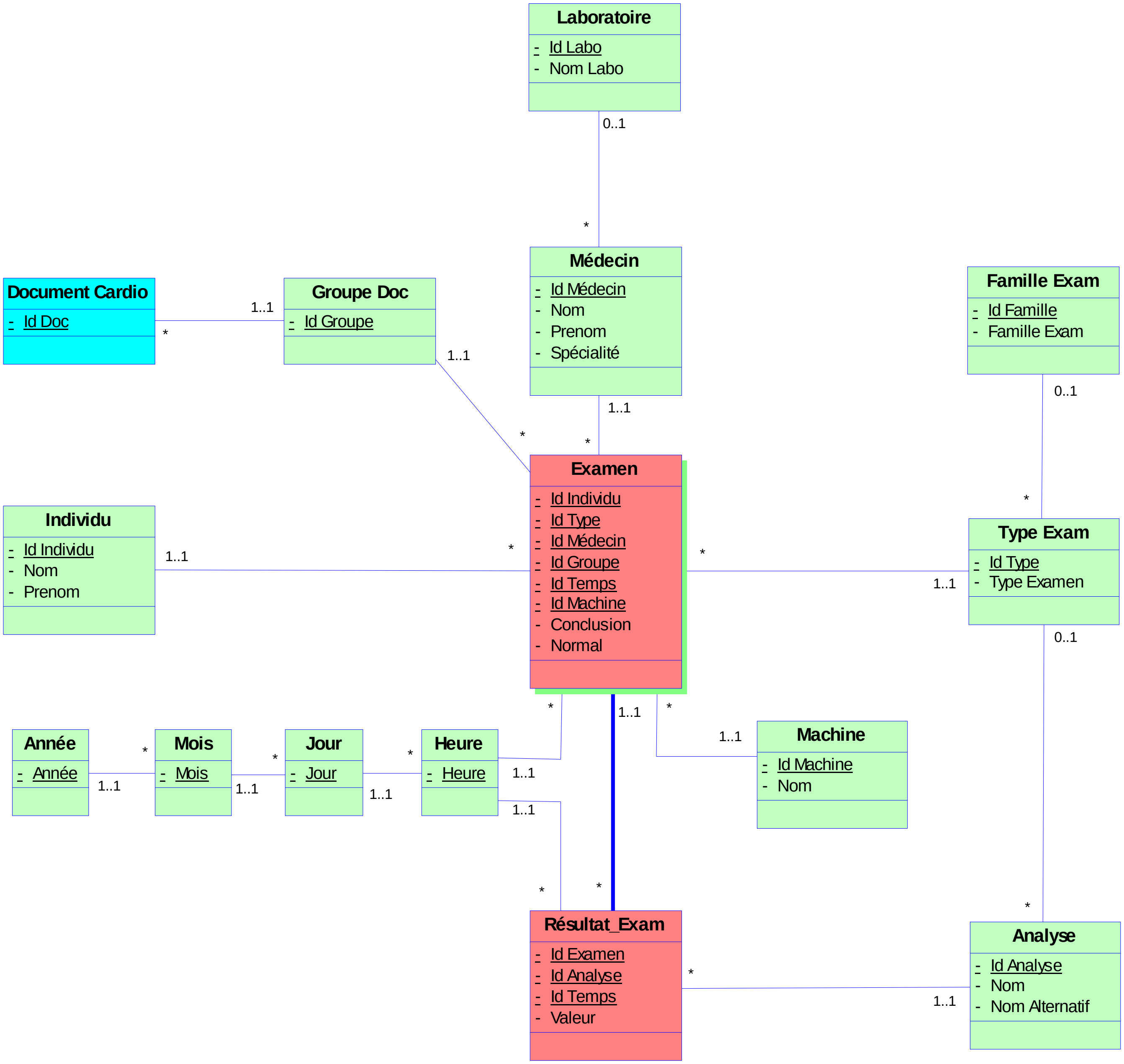

Figure 10: The new schema for the cardio-vascular data mart [3].

Yet another problem could arise from the fact that the information about the lab has now been fixed as a dimension of a doctor table rather than a relationship with an examination.

On the other hand, the table compte_Rendu has been rightly renamed as Examen which forms a link with another table Resultat_Exam. Both of these tables are fact tables as well as dimension tables- a good representation that would help in modeling of complex data.

Before proposing a model that would take into account the plus points of these two schemas and try to improve of the shortcomings, I would like to make mention here of complex data so we can better understand and appreciate the newer model.

5. Proposed Model and Metamodel

In this section, I have tried to propose a new model for the cardio-vascular data mart to improve on some of the shortcomings as mentioned earlier. Besides, this newer data mart model is going to be used for proposing a new Meta model that would enable us create more data marts for the MAP framework.

The Meta model is proposed to enable model complex data as they occur in our biomedical data warehouse. Before we can do so, however, I consider it appropriate to define few definitions that would help in better understanding of the proposed model and well as its context.

5.1 Complex data

A data can be classified as either simple data or complex. According to Darmont et al [26], one or more of the following observations concludes a complex data.

1. *multiformat*, i.e., represented in various formats (databases, texts, images, sounds, videos...)
2. *multistructure*, i.e., diversely structured (relational databases, XML documents repository...)
3. multisource, i.e., originating from several different sources (distributed databases, the Web...)
4. *multimodal*, i.e., described through several channels or points of view (radiographies and audio diagnosis of a physician, data expressed in different scales or languages...);
5. *multiversion*, i.e., changing in terms of definition or value (temporal databases, periodical surveys...).

The authors also present the complexity in an axis format

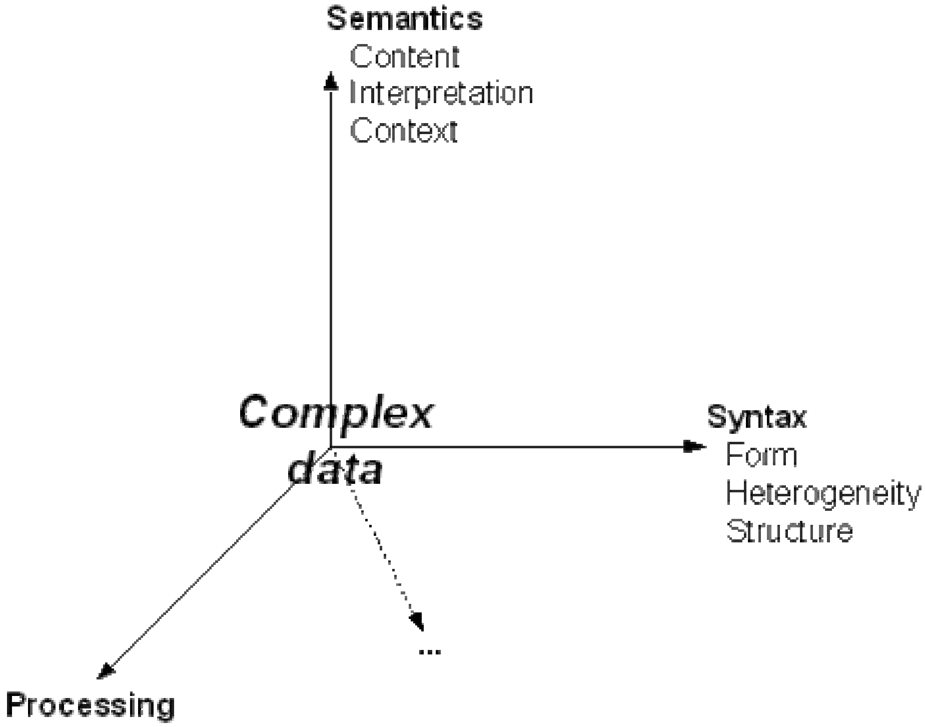

Figure 11: Representing complex data [26]

However, this is not an exhaustive list, and different data can be said to complex in different situations that need to hand the data.

Complex data can also be characterized in the following way [11].

- A lack of unique, natural identification
- A large number of many-to-many relationships.
- Access using traversals

5.2 Modeling and MetaModelling.

A model is an abstract representation of a real world phenomenon. Each model contributes to a “concept” in that domain. MetaModelling is the construction of a set of these "concepts" within a certain domain. The metamodel groups the characteristics of the models [7, 8]. This model is said to conform to its metamodel like a program conforms to the grammar of the programming language in which it is written.

5.2.1 Fact and Dimension Tables.

Data modeling is the hardest and most important activity in the RDBMS world. If we get the data model wrong, our application might not do what users need, it might be unreliable, and it might fill up the database with garbage.

Fact Table: The centralized table in a star schema is called as FACT table. A fact table typically has two types of columns: those that contain facts and those that are foreign keys to dimension tables. The primary key of a fact table is usually a composite key that is made up of all of its foreign keys.

A fact table contains either detail-level facts or facts that have been aggregated. Fact tables that contain aggregated facts are often called summary tables. A fact table usually contains facts with the same level of aggregation.

Dimension Tables: A dimension is a structure, often composed of one or more hierarchies, that categorizes data. Dimensional attributes help to describe the dimensional value. They are normally descriptive, textual values. Several distinct dimensions, combined with facts, enable you to answer business questions.

Hierarchies: Hierarchies are logical structures that use ordered levels as a means of organizing data. A hierarchy can be used to define data aggregation. For example, in a time dimension, a hierarchy might aggregate data from the month level to the quarter level to the year level.

5.3 Proposed Cardiovascular Data Mart Schema

In the proposed model for the cardio-vascular data mart, Shortcomings of the previous model have been eliminated as well as the overall has been improved. In this respect we have taken into account both the star schema and the snowflake.

As a first step, the fact table is the Exam table. Since this table stores the exams as well as conclusions of the physician, we have referred to it as our fact table. One way is to use the star schema, and all other facts should go into this one. In which case the results, the documents, etc would all go into this table. This would not only make it inefficient because it does not go with the natural flow , but the ‘real’ fact or the report of the physician also depends on other ‘peripheral facts’ that would also be stored in the table. Hence, this is not a good scheme.

To cater for the above, we need to have a set of fact tables. In our case, and also as pointed out in earlier schemas, the set of fact tables are categorized as Exam table, the Results table and the Document_Cardio table. These three fact tables are not only related to other dimension tables but also to themselves. For example the Exam and Document_Cardio tables have many-to-many relationship.

The earlier schema had not resolved properly this relationship. To resolve a many-to-many relationship, we introduce a gerund table that contains as foreign keys, the primary keys of both the participating tables to form a composite primary key for itself.

Another thing that I have retained from the previous model [2] is the fact that Lab is associated to the exam that is conducted in it rather than the doctor. For example consider the fact that a doctor refers to another lab with better facilities for a certain test which his own lab might not be able to conduct.

But the major improvement in my opinion is the time dimension. As can be seen in a typical star schema, a fact table is gotten rid of repeating groups into dimensions to get it from 2nd Normal form to 3rd Normal form. These dimensions are hence in the 2nd Normal form. In these cases, maintaining the hierarchy is cumbersome as opposed to a snowfake.

Since time is a determining factor in our peripheral fact tables ‘Results’ as well as in ‘Document_Cardio’, I have maintained both the hierarchical as well as the denormalized version of this dimension. A data warehouse, as opposed to a simple relational data model for an OLAP, contains huge amounts of data. The queries themselves are more suited to decisional purposes and are quite complex. Adding every join in our query can directly affects our performance of the query.

OLAP systems on the other hand are more suited to fast responses, maintaining of full hierarchies and less complex and voluminous queries. In this regard Denormalization is not suited. Since MAP is a personalized health care system, it requires queries of both nature. OLAP system for quick references by physician as well as a Decision support system for mining historical records. By maintaining both the collapsed and the hierarchical dimensions, we could get the best of both worlds.

Besides this, the time dimension has also been linked with Document_Cardio as in the earlier version [2] . Therefore the 3 fact tables are:

1. The “Exam” Table: This table stores data about the tests that have prescribed by the physician and also the Conclusion on the tests. It is also related to other fact tables and other dimension tables, which together contributes to it. It is because of this fact that we would later identify it as our “Central Fact table”.

2. The “Document_Cardio” table is used to store references to documents of medical examination. It also stores measures about their size and type. It is also linked to the Exam table in a M:N relationship.

3. The third fact table is that “Exam_Results” table. It stores the results of the exams prescribed. It is also related to the Analysis dimension table, from where we can categorize results.

The other Tables are dimension tables:

Time: stores the time dimension of exams and their documents as well as results. It contains the time hierarchy as a collapsed dimension and is also linked to the time hierarchical tables.

- Day
- Month
- Year

The medical Analysis: This dimension is broken into hierarchy as

- Analysis
- Exam_type
- Exam_family.

The exam type is connected to the Exam central fact table to store the type of the exam. The Analysis is related to the Results to store the results of the different analysis on the exam.

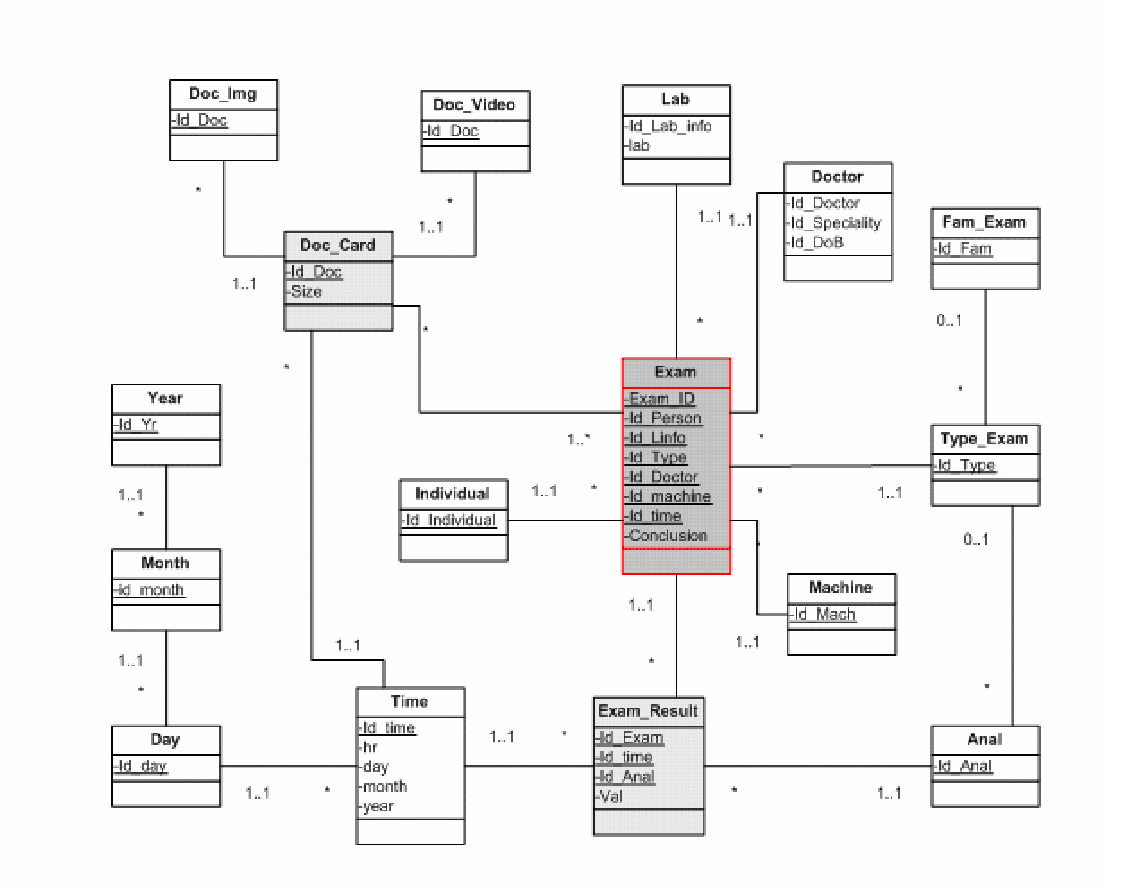

Figure 12: New Cardio-Vascular Data Mart.

Other dimension tables are the same as in an RDBM system. They are just dimensions that provide details of an attribute or measure in the fact table. Since they do not have any hierarchy, they are self-explanatory in themselves.

In figure 12 above, the grayed tables represent the fact tables. The red bordered table is used to represent the “central fact” table while the black outlined tables represent the “peripheral fact tables”.

5.4 Existing Metamodel for Complex data

In his report [3], Djallal has tried to give us a metamodel for modeling multidimensional complex data. The model is an object oriented model that would enable us to construct the different data marts of the MAP data warehouse.

The UML based metamodel allows us to represent the generic aspects of a multidimensional data (fact tables, dimensions, etc) and also the concept proposed by him i.e. a relationship between two fact tables, which could not be represented in other such metamodels.

The concept of a link/relationship between two fact tables to model a multidimensional data model was introduced by the author, and also a categorization of dimension tables into temporal, multimedia and classical dimensions.

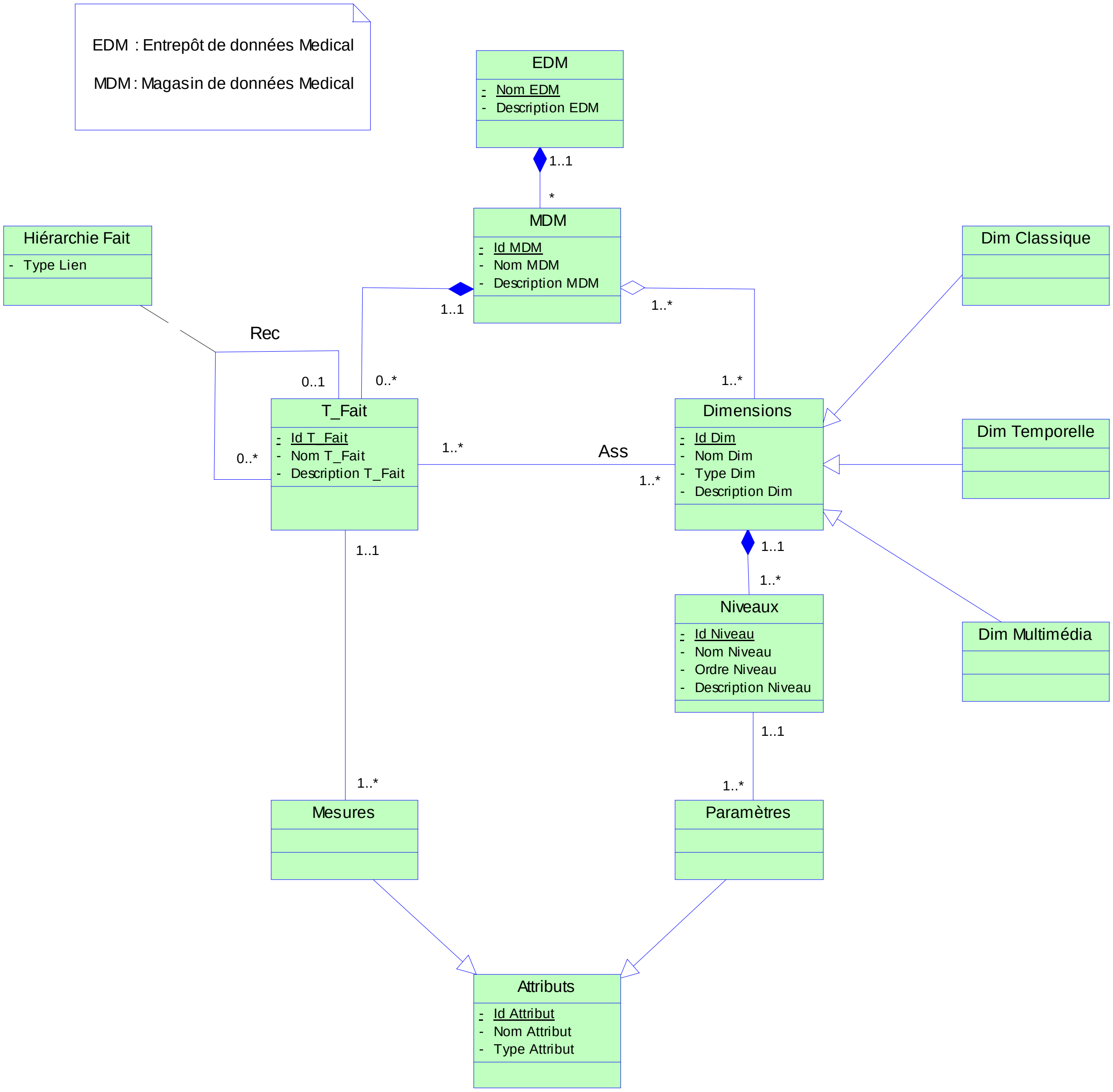

Figure 13: Metamodel proposed by [3]

As can be seen, the Metamodel allows for the representation of many links between fact tables. Also a hierarchy is maintained to know the level of these fact tables. This is too generic and quite vague. It tells us that we can have any number of fact tables and introduce hierarchies in between them.

Another concept that is too vague is for the dimension tables. These can be classic, temporal and Multimedia. A concept of level is associated with a dimension, but there is no representation that mentions dimensions can have relationships amongst themselves. It is anybody’s guess to represent the aggregation of “Levels” concept. A metamodel typically represents a strict set of rules for modeling real time problems.

Without any recursive relationship within dimensions and a “level” concept, that means no dimension can exist without being associated with a fact table. That would eliminate the concept of hierarchies in dimensions (snowflake or 3NF).

Also the concept of generalization/specification in the dimensions does not explain anything. It does not give any detail that is associated with the three types of dimensions. What is the difference between a dimension that inherits from temporal or multimedia? The Meta model doesn’t explain what the difference is except that they have been differentiated.

Another metamodel that exists is the Common Warehouse Model (CWM) by Object Management Group (OMG). This standard includes a set or combination of Metamodels to be used for modeling a warehouse. Hence, a single metamodel for multidimensional databases only represents a generic view of the concept. The idea is that since it is a combination of models of the same standard, it has to be used with other models of the same standard to enable us to come up with a complete model for multidimensional data.

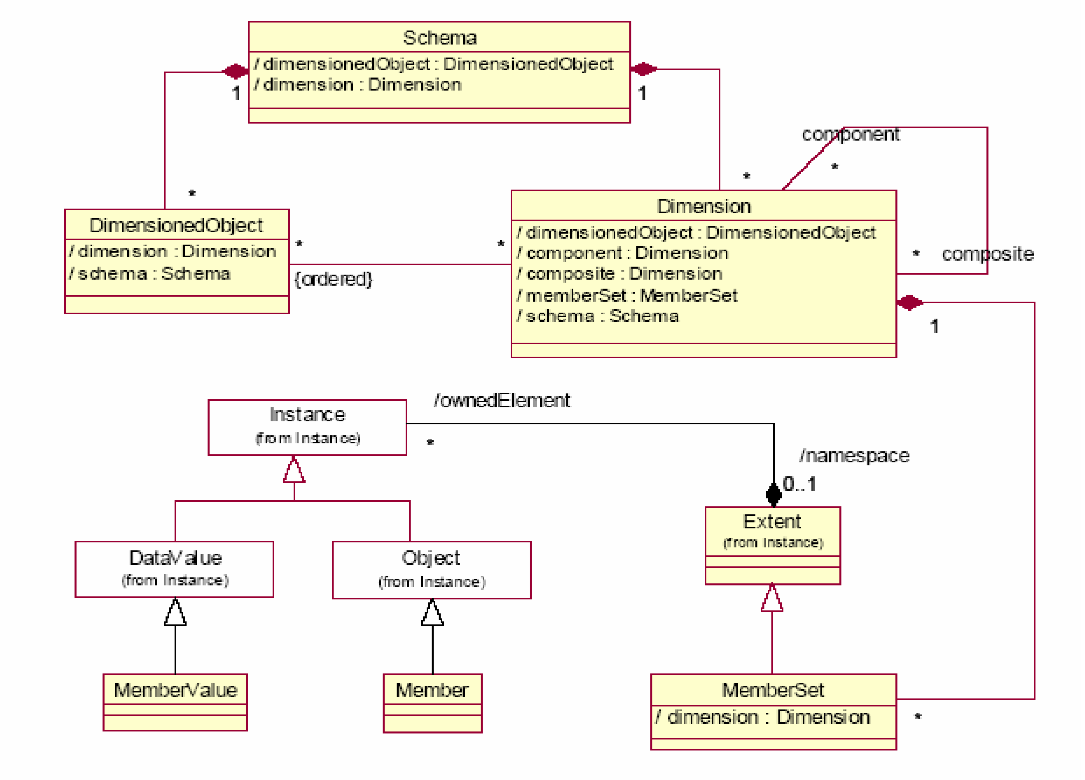

Figure 14: The CWM proposed by OMG [9].

Metamodel proposed by CWM also doesn't allow the representation of the proposed concept of a "central fact" table associated with "peripheral fact" tables. Hence the need for a new metamodel to enable us represent the schema as proposed in the cardio vascular data mart of MAP.

Other Metamodels such as COMET and YAMM, which are UML based metamodel for data warehouse are not adapted to our problem in question. The COMET [25] metamodel is based on temporal data warehouse, enables us to keep track of changes both on the instance level and on the schema level. YAMM [10] on the other hand, deals with introducing more semantics in relations like generalization and composition.

A metamodel has been proposed by *Darmont et al* [4]. It is one of the closest to the one proposed but it wasn't really designed for handling complex data with fact tables having many to many relationships.

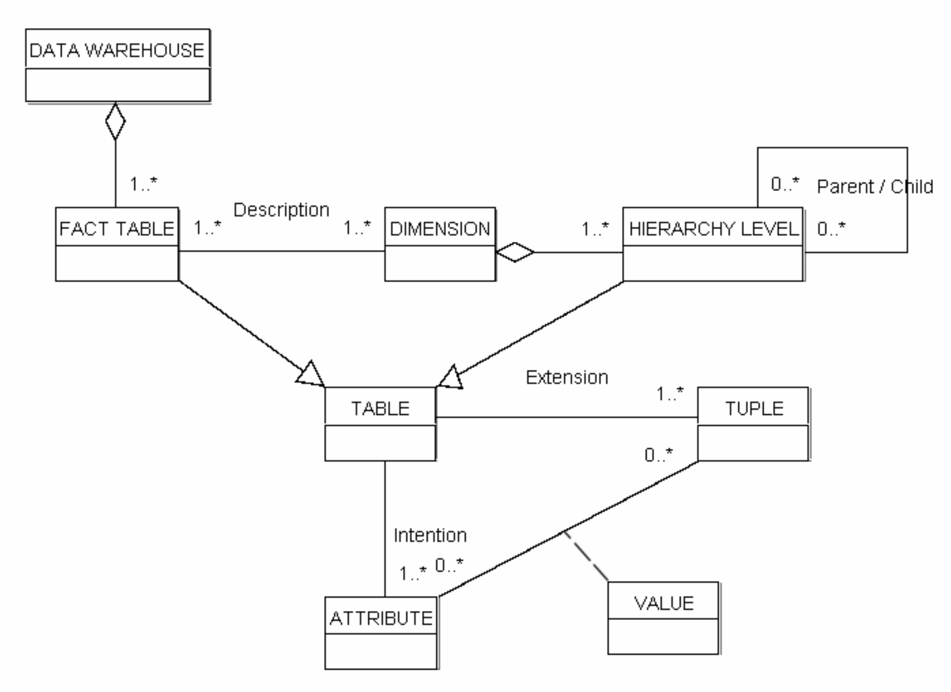

Figure 15: DWEB data warehouse Metaschema [4].

It specifies one or more fact tables that are each described by several dimensions. Each dimension may also describe several fact tables. These can be classified as shared dimensions. Each dimension may be containing one or several hierarchies made of different levels. However, if the dimension is not a hierarchy there is only one level. Both fact tables and dimension hierarchy levels are relational tables.

The model was indeed used for Benchmarking techniques and that explains its restrictions for not allowing the modeling of multiple fact tables with many to many dimensions as is our requirement.

5.5 New Metamodel for modeling Complex data (As in MAP)

In this section, I have tried to express a new Metamodel that could be used to model our other data mart models in the MAP project as well as for a general Metamodel for modeling complex data. The approach here is to use the cardio-vascular data mart model as a sample of complex data and try to devise a generic model that could be used in modeling other similar complex data. In this methodology, the new Metamodel is also tested on another complex medical data.

Little literature exists at present that would suggest a Metamodel for multi-dimensional modeling [3, 9, 10]. Besides, to the best of my effort, I haven't found any of these or other research that deals with modeling multi dimensional complex data. Probably [3], was an introduction that was specific in this study on complex data.

As we will shortly see, our Metamodel presents some concepts that are more adapted to modeling complex data rather than simple multidimensional modeling. The basic idea here is to present a 'set of fact tables'. These set of tables present the 'actual facts' as well as 'derived' or 'would-be' facts. Furthermore, the fact tables are categorized as a central fact table and a not mandatory (0 or many) set of peripheral fact tables.

The relationship amongst these fact tables can also be a set of one-to-many and many-to-many. The peripheral tables can associate with themselves as well as with the Central fact table. Both of these fact tables can be related to many dimensional tables one-to-many or many-to-many relationship as well.

At present we have confined the Central fact table to only one. This is in accordance to the star schema design in data warehousing, where we have a central fact table and a set of dimension tables. However, peripheral fact tables which are logically related to the Central fact table (e.g. the peripheral table Exam_results helps in providing the 'would-be' fact Conclusion in the Exam table), can me more than one. This allows a certain degree of flexibility in our model to divide a set of contributing facts into each peripheral table. It is not a schema in constellation in a specific sense but the principles are the same.

Peripheral tables themselves are fact tables and hence may be linked to one or more dimensional tables with a one-to-many as well as many-to-many relationship. A peripheral table is associated (aggregated) in a single data mart. A data mart may contain more than one peripheral table. See figure 8 below.

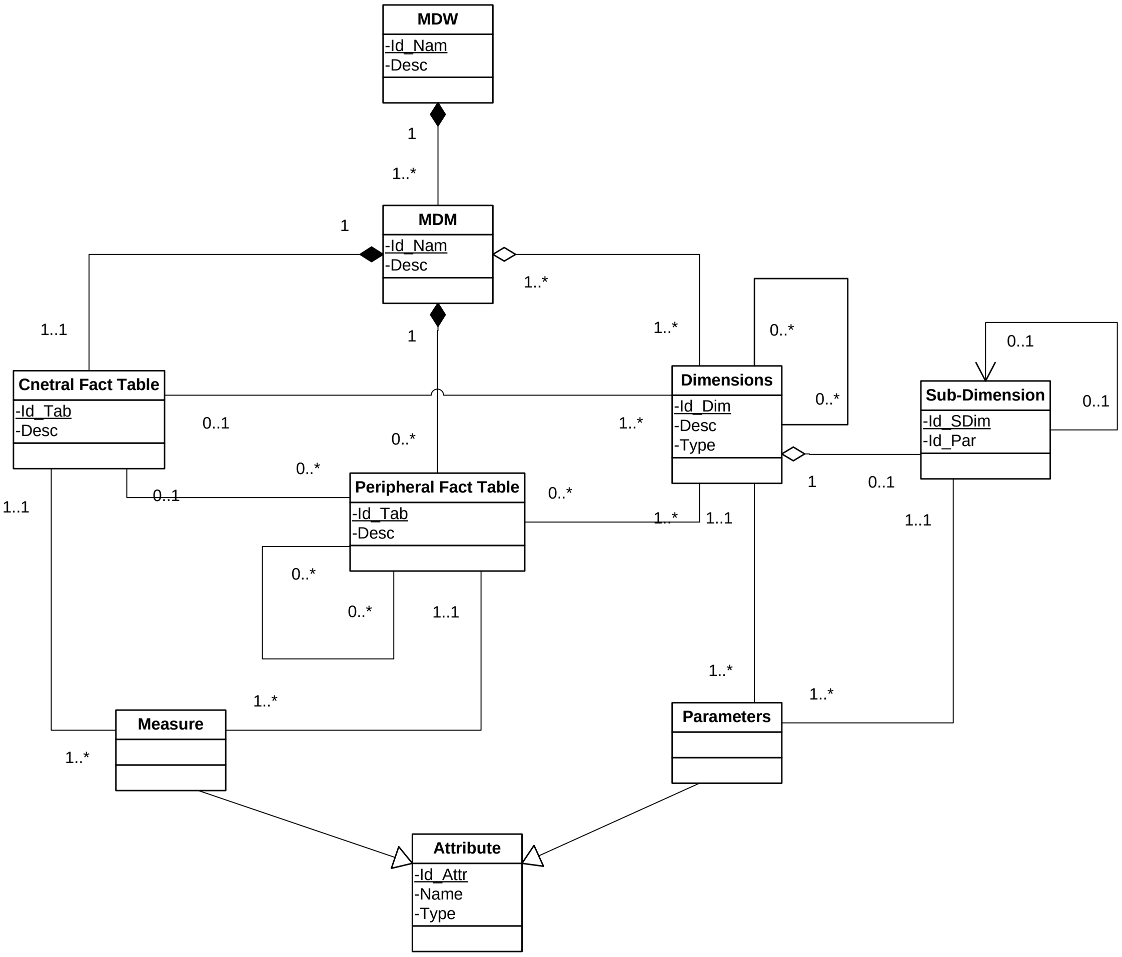

Figure 16: The new Metamodel for Complex data.

The Dimension tables are used to store the details of attributes and measure in the fact tables. A dimension table may be associated with zero or more dimension tables. The relationship between two dimension tables can be many-to-many. One dimension table must be associated with at least one (or more) fact tables. Dimension tables can also be aggregated into more than one data mart. This symbolizes the bus architecture being used and hence dimensions can be part of the bus architecture being shared by different data marts.

Another concept being used here is the Sub-Dimension. A sub dimension has a dependency association with a dimension table. Sub-dimension table can also be associated with another sub-dimension table. These tables (sub-dimension) are used to express hierarchy in the dimension. The sub dimensions for a leveled hierarchy of the dimension as a snowflake. However, a sub dimension lower down the hierarchy must be associated to a sub-dimension which is upper in level but in the same hierarchy. The sub-dimension at the top most level is then associated to the dimension table.

The concept above is derived from the idea of maintaining both the snowflake as well as the denormalized version of a possible one or more dimension in the model. The relationship between a dimension and its sub-dimensions has been shown as a dependency. A sub-dimension cannot exist without the depending dimension. This should not be confused with the idea of simply storing a hierarchical dimension. That is reflected in the Metamodel as a recursive association on Dimension.

The measure and attributes are used to define the fact and dimension classes in the model. Both of them have a set of attributes that would enable us to define our Central fact and Peripheral fact class, as well as the dimensions and sub dimensions.

The entities in the metamodel can be defined as follows:

Data warehouse:
The ware house will be defined by the MDW class which will contain the other classes in it. It can be defined as a set of 2 tuples (Nm, Des) where

- Nm is the name of the warehouse
- Des is the description of the warehouse.

Data Mart
A data mart is part of the warehouse that deals with the specific domain of the warehouse. The marts are relatively independent that manage their own data and may share a set of common tables in the bus architecture. The mart is defined by a set of tuples as (Nm, Cft, Pft, Dim, Ass, Des) where

- Nm is the name of the mart
- Cft is the Central fact table
- Pft is the set of peripheral fact tables
- Dim is the set of dimensions
- Ass is the set of associations between fact tables (central and peripheral) as well as between fact and dimension tables
- Des is the description of the mart.

Central Fact table
This is the main fact table that comprises the mart. Fact tables are the centre of star schema in a warehouse environment. It is defined as (Nm, Ms, Ass, des) where

- Nm is the name of the central fact table
- Ms is the set of measures
- Des is the description of the central fact table

Peripheral fact tables

This is the table that comprises the fact but gather around the central fact table to give a more complete picture. It can be define on similar basis as the central fact table as (Nm, Ms, Ass) where

- Nm is the name of the table
- Ms is the set of measures
- Ass is the set of associations between the peripheral fact table and other fact tables.

Dimension tables

Dimension tables contain a set of parameters that contain the same thematic nature. A dimension table is defined by (Nm, Pm, Ass)

- Nm is the name of the dimension
- Pm is the set of parameters
- Ass is the set of associations that would relate the dimensions.

Sub-Dimension tables

Some of the dimension tables form sub dimensions that show the dimension at a lower granularity. Such tables form the central theme in selective denormalization as well as maintaining dimension hierarchy. They are defined as (Nm, Pt, Pm, Ass) where

- Nm is the name of the sub dimension
- Pt is the name of the parent dimension
- Pm is the set of parameters
- Ass is the set of associations that relate the sub dimensions.

For example in the cardio vascular data mart, time dimension has both a denormalized table as well as a breakdown into hierarchy.

5.6 Case Study of our Metamodel

Here is an example of a detailed case study of a clinical database schema. The clinic data is the schema of a of a clinic as part of a larger collection in a warehouse that could be used for risk analysis.

The patient table contains data about a patient identified by a SSN number. Data is also stored about the clinic and the clinician. The schema represents a complex clinical data with many M:N relationships. This is an “easier” complex data in that it doesn’t store multidimensional data and there are relatively fewer many to many relationships. Moreover, as compared to cardio-vascular data

However, as a natural flow of activities in a clinic, the fact table is the Diagnosis of patients. This is coupled with the Prescription and Therapy table that explains “related” facts.

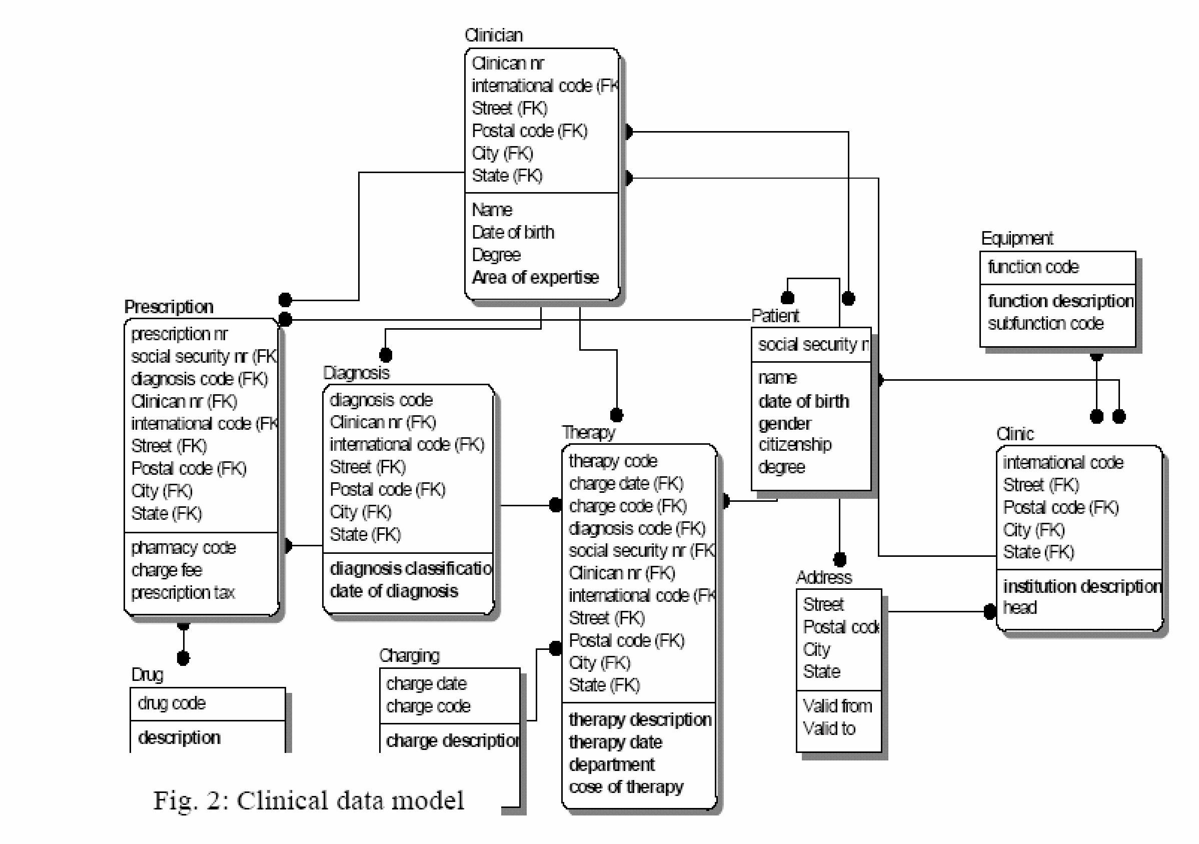

Fig. 2: Clinical data model

Figure 17: Clinical data model [24]

A patient is treated in a clinic that has an associated clinician. A clinician may give a prescription. A prescription has a diagnosis and a diagnosis may result in one or more therapies. A therapy may be changed from time to time or may be based on time intervals.

Considering out metamodel, the basic fact table or the "central" fact table is diagnosis. This is associated with a set of related or "peripheral" fact tables- Prescription and Therapy. A clinician is a dimension for the Prescription table and patient is also a dimension for the prescription table and therapy tables. These dimensions are also related in a many-to-many relationship.

6. Conclusion

The work presented in this report concerns the multidimensional modeling of the complex data. A data warehouse approach for storing medical data has lots of modeling and integration problems. Our objective is to be able to generate more data marts for the project MAP (Medicine d'Anticipation Personalisé), which is already composed of several modules.

To meet this aim, we proposed an approach for modeling and implementation of the medical warehouse while basing ourselves on a metamodel which we proposed. MetaModelling is a technique that would enable us to be able to generate marts of the MAP architecture. We have modeled the most complex module of project MAP, the cardiovascular module and used this as a basis for our metamodel.

Integrating the medical data such as the cardiovascular data in a multidimensional structure has given rise to a lot of problems. Hence, we felt the need to propose new concepts which extend the existing models towards a new type of metamodel. This new addition to the previous model would enable us to handle complex data with multi-format data and a large number of many to many relationships.

In this context, we proposed a metamodel by generalizing the multidimensional model of the cardiovascular module. This metamodel takes into account a main fact table and a set of 'peripheral' fact tables that would revolve round the central fact table. This would enable us handle more complex relationships in data.

GEDM is a prototype version developed at ERIC that allows previous metamodels to be used for generation new marts for the MAP data warehouse. In our case, we hope to adopt this, with little modification if necessary to enable us develop and integrate the MAP project.

Future trends could revolve round, generalizing this model to include all types of complex data. Moreover, other metamodel characteristics like features from temporal metamodels could be added to make one generalized metamodel for all warehouses.

7. Bibliography

[1] Olivier Emerson, *Conception de l'entrepot de données MAP,* ERIC lab, 2003.

[2] Jean-Noel Georgel, *Le module Cardiovascular de l'rntrepot de données MAP*, ERIC lab , 2004.

[3] Midouni sid Ahmed Djallal, *modélisation multidimensionelle des données complexe, application aux données medicale,* ERIC lab, 2005.

[4] J. Darmont, F. Bentayeb, O. Boussaid, "Benchmarking Data Warehouses", International Journal of Business Intelligence and Data Mining, 2007.

[5] J. Darmont and Emerson Olivier, A complex data warehouse for personalized, anticipative medicine, ERIC University of lyon 2.

[6] J. Darmont, E. Olivier, "Biomedical Data Warehouses", Encyclopaedia of Healthcare Information Systems, Idea Group Publishing, 2007.

[7] What is meta-modeling? http://www.metamodel.com/staticpages/index.php?page=20021010231056977

[8] Booch, G., Rumbaugh, J., Jacobson, I. (1999). The Unified Modeling Language User Guide. Redwood City, CA: Addison Wesley Longman Publishing Co., Inc.

[9] OMG, *Common Warehouse Metamodel (CWM) Specification*, March 2003, Version 1.1.

[10] Abelló A., YAM (Yet Another Multidimensional Model): *A Multidimensional Conceptual Model*, PhD Thesis, Universitat Politècnica de Catalunya. Barcelona, April 2002.

[11] www.service-architecture.com

[12] http://www.stanford.edu/dept/itss/docs/oracle/10g/server.101/b10736/logical.htm

[13] www.LearnDatamodeling.com

[14] http://cimic.rutgers.edu/~gusadi/design/

[15] Inmon WH. Building the Data Warehouse. Second Edition. John Wiley & Sons. 1996.

[16] Saad, K. (2004). Information-based Medicine: A New Era in Patient Care. (2004). ACM 7th International Workshop on Data Warehousing and OLAP (DOLAP 04), Washington, USA. 58.

[17] Sun, Y.M., Huang, H.D., Horng, J.T., Huang, S.L., & Tsou, A.P. (2004). RgS-

Miner: A Biological Data Warehousing, Analyzing and Mining System for Identifying Transcriptional Regulatory Sites in Human Genome. 15th International Database and Expert Systems Applications Conference (DEXA 04), Zaragoza, Spain. LNCS. 3180, 751-760.

[18] Henry Engstrom and Kjarton Asthorsson. A data warehouse approach to maintenance f or Integrated Biological data.

[19] W.H. Inmon and C. Kelley. Rdb/VMS: Developing the Data Warehouse. QED Publishing Group, 1993.

[20] M. Jarke and Y. Vassiliou. Data warehouse quality: A review of the DWQ project. In Proceedings of the 2nd Conference on Information Quality, Massachusetts Institute of Technology, Cambridge, USA, 1997.

[21] S. Yadlapalli, A. Silberschatz , G. Hepherd ,P. Miller and L. Marenco. Integration of Heterogeneous Bio-Medical Databases: A Federated Approach using Semantic Schemas, Yale University. March 2006.

[22] Baker P, Brass A, Bechhofer S, Goble C, Paton N, and Stevens R. TAMBIS: Transparent Access to Multiple Bioinformatics Information Sources. ISMB 1998.

[23] Sohrab P Shah, Yong Huang, Tao Xu, Macaire MS Yuen, John Ling and BF Francis Ouellette. Atlas – a data warehouse for integrative bioinformatics. BMC Bioinformatics, 2005.

[24] Navena stolba and A Min Tjoa, The relevance of data warehousing and data mining in the field of evidence-based medicine to support healthcare decision making.

[25] Johann Eder, Christian Koncilia, and Tadeusz Morzy, The COMET metamodel for temporal data warehouse.

[26] J. Darmont, O. Boussaid, J. Ralaivao, K. Aouiche, "An Architecture Framework for Complex Data Warehouses", 7th International Conference on Enterprise Information Systems (ICEIS 05), Miami, USA, May 2005, 370-373.

8. Appendix 1

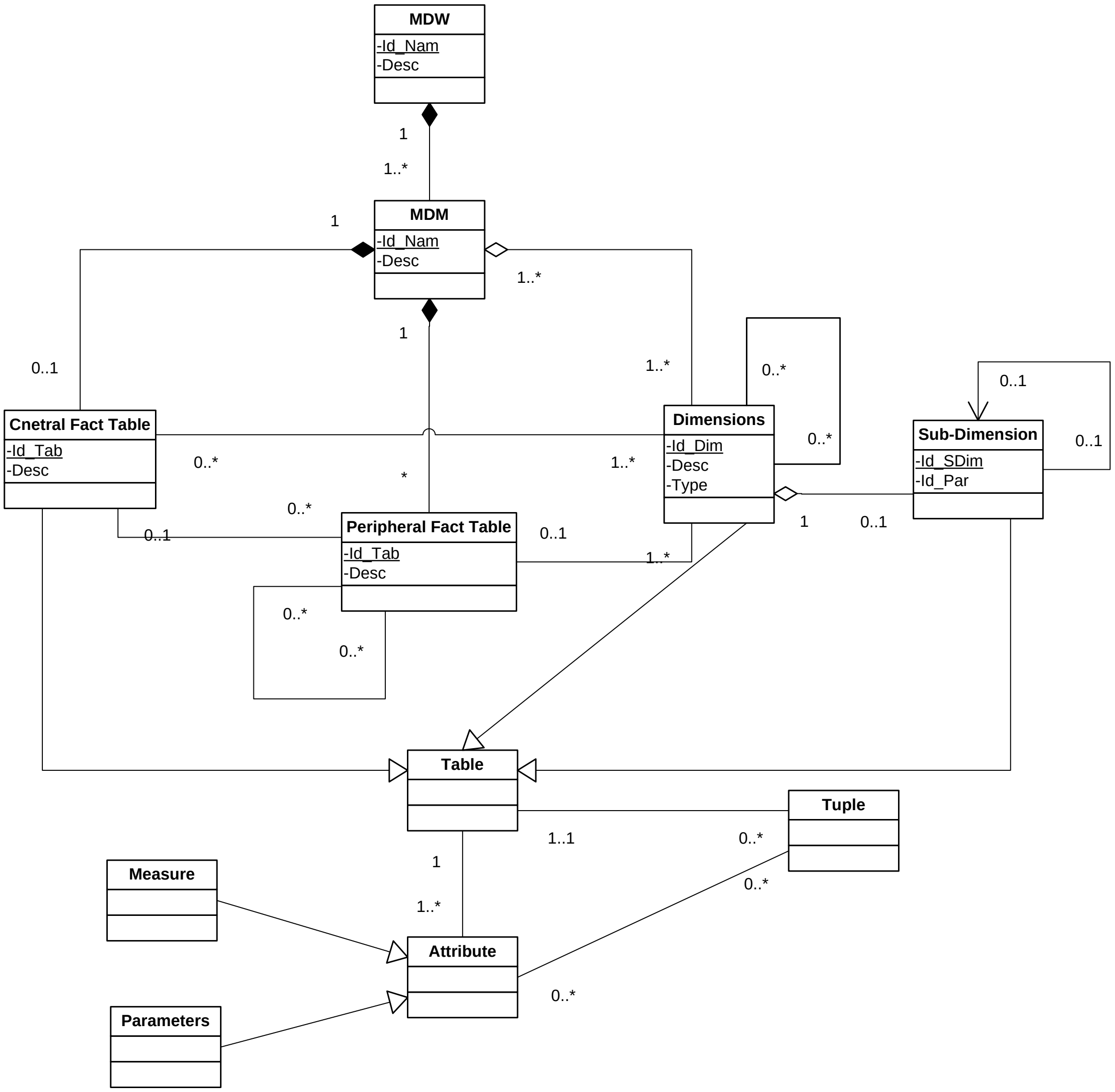

Figure 18: Alternative Metamodel based on [4]

As suggested by *Darmont et al* [4], the lower half of the metamodel demonstrates the representation of a database table in UML. Both the fact and Dimensions are tables. A table has 1 or more attributes. These attributes can be Measures, which is used for aggregation purposes, as in fact tables or Parameters in dimensions that are a set of semantically related attributes.

There may be 0 or more tuples, or rows, in a table. The association between a tuple and an attribute is the location of a field or value in the table.