Easy Does it!

Sónia Patrícia F. C. B. Quaresma Gonçalves

Information Systems and Computing Department
National Statistical Institute
Portugal

sonia.quaresma@ine.pt

Abstract: When preparing information for a data warehouse, namely integrating administrative and survey data, we must perform major transformations and a thorough revision of the data structure before we can store it in the database; hence the data warehouse design importance. In this design not only the structure should be accounted for but also the transformations needed and the way all concepts relate to one another. Just as an architect needs a building model to initiate construction, for a data warehouse a conceptual model has to be developed before data can be accommodated in the databases. This is the only way to ensure it will in the end serve our purposes and provide all the required results. Understanding this pressing issue we, at the Portuguese NSI, embarked in a project with the objective of establishing a data warehouse designing process.

We identified the major structures that should be present in a data warehouse model, and at the same time the most common elements in statistical dissemination. Several conceptual modeling proposals, found in the data warehouse design literature, were evaluated and the most promising was selected for testing. The computer science and statisticians teams worked together over the model, using real statistical domains of the Portuguese NSI to test its capabilities, which caused some characteristics of the modeling process to emerge. We will discuss our findings showing some examples of data warehouse design models that we are currently using which enable:

- Earlier diagnose of architectural problems.
- Easier integration of administrative and survey data.
- Communication layer common to statisticians and computer scientists.
- Decreases the time needed to conceive a data warehouse.

As we will show you: Easy Does It!

Keywords: Improving Process Quality, Quality of Statistical Systems, Information Management in Statistical Institutes

1. Introduction

Nowadays, data repositories in most National Statistic Institutes (NSIs) are data warehouses (DWs). However, due to the way these databases appeared and developed, little attention was paid to their conceptual design. The lack of data warehouse conceptual design is motivated by the inexistence of development products that allow their specification and by communication difficulties between data warehouse developers and business intelligence teams. For NSIs in particular the statistical methods and language of the information end users is opaque to most computer science engineers, which decreases their ability to share ideas and agree on a data model.

In this paper we address in section 2 the statistical particularities and the data warehouse structure that we intend to model.

Several conceptual modeling techniques have been proposed in the literature. They will be reviewed and evaluated in section 3 keeping in mind the different approach that statistical data requires. We'll also pay special attention to the communication effort required by interdisciplinary teams (statisticians and informatics) to work with the same modeling. Some examples from the Portuguese NSI will be presented. In section 4 we present our conclusions.

2. Conceptual Data Warehouse Design

Statistical Offices always produced a lot of data and as technological solutions provided ways to store this data they quickly embraced them.

However the Data Warehouses that are today commonly used to organize large amounts of data, due to their different structure and internal organization, are necessarily different from the typical relational databases. As such, their conceptual design has to be done in different ways.

Information storage in computers has been studied since 1972 [1]. Today the three layer database architecture is commonly accepted. The first approach is the external which describes some part of the data that is relevant to a particular user, but not to all of them. As such it is not usually considered a real level because it makes no sense to model small particular realities. Instead we consider the whole picture as the first layer, which is the Conceptual Level. It describes not only the information but also the relationship between variables and data sets. It is followed by the Internal Level that takes care of the data structuring. Finally, in the Physical level we decide how data should be physically stored in the databases, depending on the physical characteristics of the machines.

To talk about the information, the relationships between the data sets, and uncover data constraints that may exist we construct models. Models are a set of concepts represented in a precise, concise and understandable way.

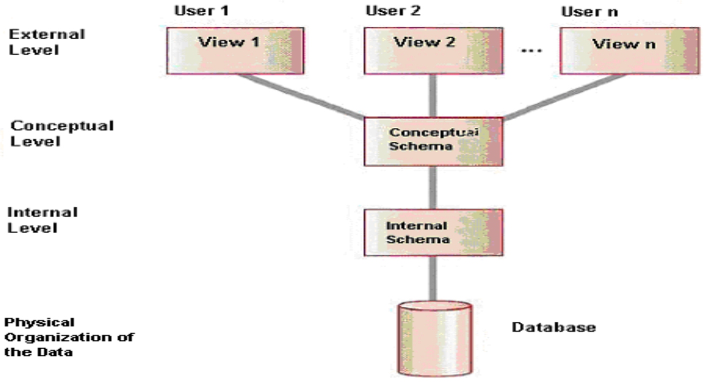

Figure 1 - Three Layer Database Architecture

Following the presented three layer database architecture several different models arise:

- Physical Models – located at the Physical Organization of the Data level. It's usually provided by the software companies and takes care of the physical structures, files, indexes, hashing, etc.
- Logic Models – for the Internal level, were first began developing in the 60s: Hierarchical Model - IMS[1] [2], Web Model [3] and already in the 70s the Relational Model [4].
- Conceptual Models – later developed: Entity/Relationship [5], Semantic [6], Functional [7] and Object Oriented [8].

Data Warehouses appeared in the late 90s and have a similar three layer architecture. For each of these layers, different models can be established [9].

- Physical Model – according to DW's providers: partitioned tables, bitmap indexes, star schemas, etc.
- Logic Model – describing the ways in which information can be stored in the DW is still the manufacturer's responsibility; the models can be multidimensional (MOLAP), relational (ROLAP) or hybrids (HOLAP).
- Conceptual Model - describes the concepts; different analysis perspectives and how the concepts relate to each other.

The levels at which a DW should be modeled have not been subject to the same effort and attention than relational databases, which explains why they are not as developed. In spite of the recent interest DWs have raised in academia, they developed uniquely as technological products from the industry to solve the increasing problems of the organizations in analyzing ever growing databases. As a by-product, DWs conception and structuring were not a concern until these databases were already being used and not always with the expected results.

Nowadays, because of the DW's historical evolution, modeling still focuses on the physical level. Even the logic level is strongly attached to the physical model because each modeling methodology is closely related with the database engine, and as so, very dependent upon the manufacturers. However, to accomplish a specific data analysis the data warehouse has to be modeled at a conceptual level, only concerned with the concepts involved and aware but not dependent of the physical constraints.

Keeping this in mind, and preceding the evaluation of some modeling techniques proposals, we studied the elements that are commonly present in any data warehouse, but also some particularities that have to be accounted for to ensure the effectiveness of the resulting analysis tool in the statistical domain.

[1] Developed in 1968 to organize and store the information for the Apollo program

2.1 Data Warehouse Elements

Data Warehouse solutions aim at simplifying data analysis, so every situation requires a different approach that considers all the problem angles and data involved and is able to provide the necessary answers. The construction of an accurate DW model is vital to ensure the correct representation of facts and to make possible all the desired points of view during the analysis.

Every model is centred on events relevant for the description and comprehension of the situation. Any event can be regarded from several different perspectives. In DWs these are called dimensions. Underlying each dimension several classifications may exist. That's what enables multidimensional data analysis.

We will take as an example the tourism expenditures project. Whenever persons that do not reside in Portugal enter the country we count them as tourists. Even if we don't know anything else about this entrance other than that it took place, we still have a fact, and one that can be counted: the number of tourists that entered Portugal. We call this a **measure**. However if we don't know anything else about it we cannot distinguish it from all the other tourist arrivals. If we associate this fact with a date, such as the 1st January 2006, we are giving the fact a temporal **dimension**. If we add to this some geographic information about the Portuguese region the tourist is going to visit, we already have a bi-dimensional fact.

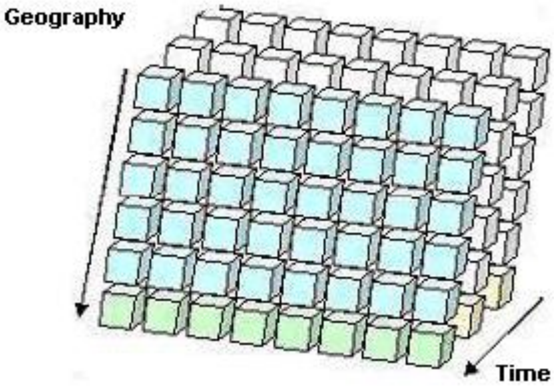

Figure 2 - Analysis Multidimensionality

We can go on adding information like the visitor's age, sex or nationality and so increasing the analysis multidimensionality. These dimensions are **classifications** that build in themselves a **hierarchy**. Ascending or descending through the hierarchy aggregates or disaggregates the facts.

Continuing with our example the temporal dimension is easily seen as a hierarchical tree.

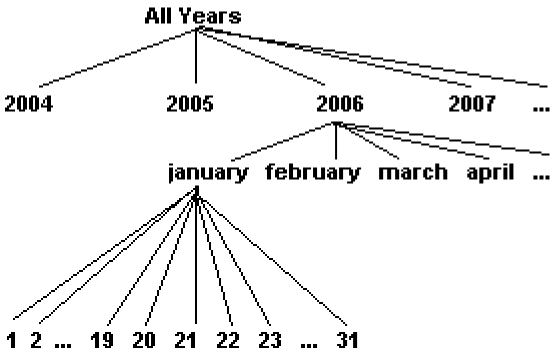

Figure 3 - Temporal Classification

The classification's several **levels** form the hierarchy. The lowest level also called the **leaf** level is the one that connects with the fact. The classification **member** that aggregates all the other is unique and constitutes the **top** level of the tree, and we usually have several intermediate levels. It is obvious at this point that when modelling a DW at a conceptual level we have to be able to talk about all these concepts as different entities because a fact is intrinsically different from a dimension, and a fact's measure is not the same as a dimension member. Usually the fact's measures can be summed. For example, to know all the tourists in January in Portugal we had to sum all the entrances, so we have to model this measure and relate it with the relevant classifications. In statistical universes **derived measures** are common so we have to be able to show the formula used to build them. The information about the fact is important but the metadata about the dimension also has to be represented. Every NSI has several geographic hierarchies that it wishes to use simultaneously. For example when we're talking about the Portuguese regions visited by the tourists we can have a hierarchy from country to parish whose intermediate levels could be the nomenclature of territorial units for statistics II and III (NTUS) or administrative regions and districts. In this case we have two classifications that can be defined in only one dimension, the geographical dimension.

Top Level	Intermediate Levels		Leaf Level
Country - Portugal	Ntus II – Lisboa	NtusIII – Lisboa	City – Lisboa
Country - Portugal	District - Lisboa		City – Lisboa

Table 1 - Geographic Classifications

The example shows two different classifications, with a different number of levels in their hierarchical tree, but with the same top level and the same leaf level. This is very important because it means the facts will connect with the classifications at the same level, City, and also that the maximum aggregation, Country, will return the same tourist's counting for both classifications. Sometimes the classification members have more information attached to them but this data is not useful to discriminate between the members. An example would be the phone number area code. Each city has an area code but some cities are so close that they share the same number. This would be a **dimensional attribute** of the city level in the geographic dimension. As we may want to store this information for further contacts with the local tourist information service we have to represent it in our DW model.

Finally sometimes is useful to represent all the classification members, mainly when the project is still developing and we want to make sure we're not forgetting any details. For example the purpose of the visit could be holydays, work, family visit or other. It's important to represent the 4 elements because in the future "other" can be detailed in several more classes like attend a conference or participate in sports events and other. This second "other" it's not the same as the first, and if we had collected some information according with the first dimensional design before changing it we should build a sublevel rather than just adding new members.

Having enumerated the components we need to be able to represent in a good DW model of typical NSI data, we should keep in mind the importance of the model's legibility. This kind of model is usually developed between the statisticians and the informatics teams. The differences between all the described concepts should be clear for all the elements involved. To facilitate this, both the syntax and semantics of the representation on the different elements should be well understood by everyone. In the next section while evaluating all the other characteristics in the several DW models that have been presented in the literature, we shall pay a special attention to this more subjective feature that is the model's legibility.

3. Conceptual Modelling Proposals

Different DW modelling techniques have been proposed in the state-of-the-art literature involving more or less of the elements we described in the previous section. Although a detailed description of all of them [10] is not our aim in this paper we shall present a table of the most prominent among them and we'll engage on a brief discussion of their merits.

3.1 Global Evaluation

The following table presents five modeling techniques that were selected from a set of eleven [10] because of their highest legibility and easier representation of different elements.

Model	Facts			Dimensions			
	Measures	Formula	Several Facts	Hierarchies	Levels	Members	Attributes
DFM	Y	Y	Y	Y	Y	N	Y
ADAPT	Y	Y	Y	Y	Y	Y	Y
MD	Y	N	Y	Y	Y	N	Y
MERM	Y	Y	Y	Y	Y	N	Y
CWD	Y	Partially	Y	N	Y	N	Y

Table 2 - Modeling Techniques Comparison

Of all the five modelling techniques Multidimensional Databases (MD) and Conceptual Warehouse Design (CWD) are the weakest. The representation of several hierarchical classifications in a dimension is not possible using MD [12]. The dimensional members and the formulas are not contemplated by either of the two models, although CWD [11] allows us to represent functional dependence among the measures, we cannot use in the formula any element that is not going to be a measure which is not satisfactory.

The Dimensional Fact Model [13] (DFM) and Multidimensional Entity Relationship Model [14] (MERM) are more complete modelling techniques but present the same handicap with the dimensional members that they are not able to represent.

Finally the Application Design for Analytical Processing Technologies [15] (ADAPT) is the only technique to design a DW model that contains an object for all the elements we will need to model. So it was selected to further testing with some of the particular problems we currently deal with at statistical offices. Before presenting the actual tests we will briefly discuss the ADAPT objects in his later format [16].

3.2 Elements of the Model

The primordial element of any DW model should be the fact it's measures and their calculation formulas. It is also important to be able to show several different facts simultaneously, so that we can understand how they relate to each other.

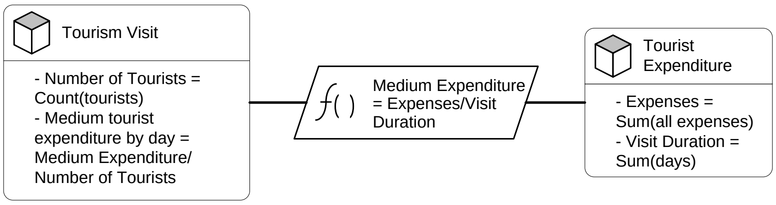
Tourism Visit
- Number of Tourists = Count(tourists)
- Medium tourist expenditure by day = Medium Expenditure/ Number of Tourists
f( )
Medium Expenditure = Expenses/Visit Duration
Tourist Expenditure
- Expenses = Sum(all expenses)
- Visit Duration = Sum(days)

Figure 4 - Facts, Measures and their Formulas

A fact is represented with a distinctive cube sign in the upper left. We can give the fact a name and write in the lower box all its measures with their respective formula. Figure 4 shows two facts at a different aggregation level so that to navigate through one to the other we also use a formula upon the less aggregated fact measures.

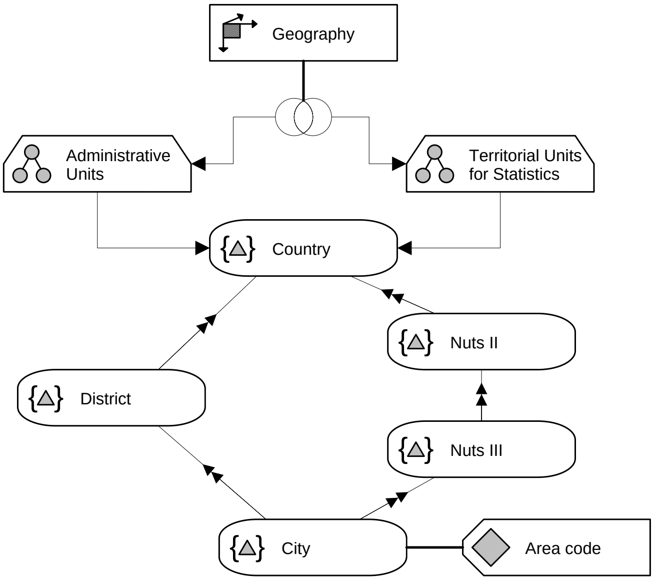

Figure 5 - Geographic Dimension

Following the given example we find that the Geography dimension has 2 hierarchical classifications: Administrative units and Territorial Units for Statistics. The symbol ⚭ means that they totally overlap, i.e., all the cities that belong to one also belong to the other. We could also represent partial overlapping ⌒⌒ or fully exclusive ⊗ elements.

In each hierarchy we represented the different levels and we can show that the top and the lower level are common to both classifications. Between the levels we indicated a strict precedence (line with two arrows), meaning that every member of a lower level aggregates to the upper level. We could also have loose precedence if some members did not aggregate.

Another particularity of this dimension is the asymmetry between the classifications that can be easily represented and that should be accounted for in the DW to enable all the possible data crossings and prevent aggregation errors.

We still had an area code connected to the city. It was a dimensional attribute and not a member as it could be the same across several cities. We represent the dimensional attributes with still another icon, and connect it with the appropriate dimension level.

The dimensional members' representation in the model can be particularly important when we are discussing all possible answers to a question and at which level of the classification should they be. In the given example of the visit purpose, we had in the first case "holydays", "work", "family" or "other", and this "other" would further subdivide in "attend a conference", "participate in sports events" and "other".

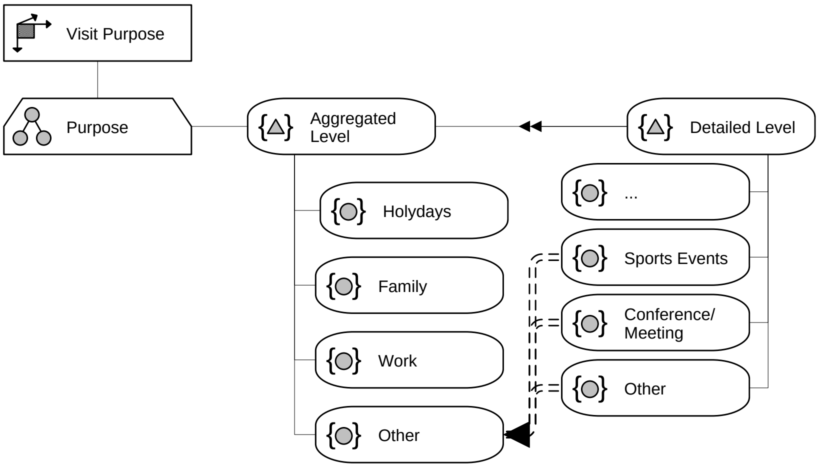

Figure 6 - Dimensional Members

We should represent both levels and their members, at least those that aggregate to a specific member of the aggregated level. In this case to prevent misinterpretations of the "other" element it would be advisable to change the name of one of them.

These examples illustrate several of the situations that should be modelled when preparing the data warehouse to prevent architectural problems in the database design but also to explicitly state what should be a classification or not.

In the next section we will still present some examples from several domains to show the suitability of this modelling technique and also the importance of modelling all the details.

3.3 Some Examples

We will present several examples from: international trade, buildings construction, and agriculture. The situations involved and the problems solved are distinct in the different cases.

3.3.1 Classifications

The first examples from the external trade relate to the economic classification of the traded products and the countries involved in the deal. At an economic level we want to be able to group several transactions so that we can talk about the value in euros of a flow, import or export, over a period of time of some specific kind of product. However there are several different classifications that serve distinct purposes. The identification of each one of them from the beginning is important to ensure that each fact, each transaction, is classified according to all the required economic classifications. Sometimes this involves an effort from the methodology department to build correspondence tables across the classifications; this supposes a long and tiresome work that should start as early as possible.

The identification of all classifications requires the business team to make an inventory not only of all classifications used during data collection but also of all the ones that would be desirable to use for dissemination purposes.

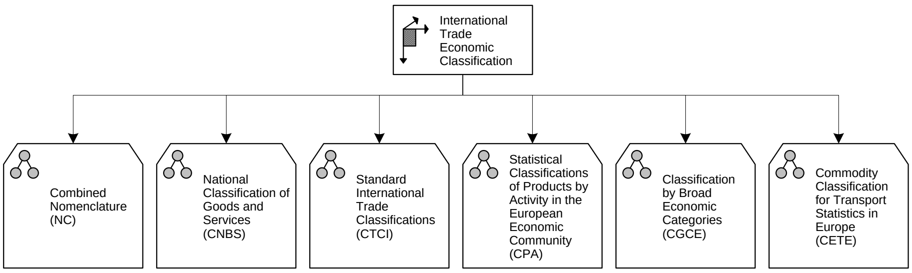
International Trade Economic Classification
Combined Nomenclature (NC)
National Classification of Goods and Services (CNBS)
Standard International Trade Classifications (CTCI)
Statistical Classifications of Products by Activity in the European Economic Community (CPA)
Classification by Broad Economic Categories (CGCE)
Commodity Classification for Transport Statistics in Europe (CETE)

Figure 7 - International Trade Economic Classifications

The data warehouse team can help build this list providing a graphic image of the results, alerting for the need of correspondence tables and being the moderator between the data collection and data dissemination teams. Usually these teams are not one and the same and as their objectives regarding the data use differ they focus too much on their own needs and have trouble considering their colleagues' requests. The DW team as a third party has no requirements towards the data and this turns them into the most suitable and detached referees, often in cooperation with the methodology team, in the difficult cases that may arise.

3.3.2 Hierarchies

A different problem emerged, in the Portuguese NSI, in the international trade project when building a country classification of the

Portuguese trading partners. Most Portuguese goods and services trade occurs within the European Economic Community but that is not always the case. The desired country classification will not only include the country itself, like Spain, Norway, Nigeria, Mozambique or Canada, but also the economic group to which they belong; UE, EFTA, OPEP, PALOP or OCDE. Sometimes it is also important to have the geographic continent associated to the country as some countries may not belong to any particular economic group. Other classification could simply be used to know if the deal was done with another country in the European Union or not.

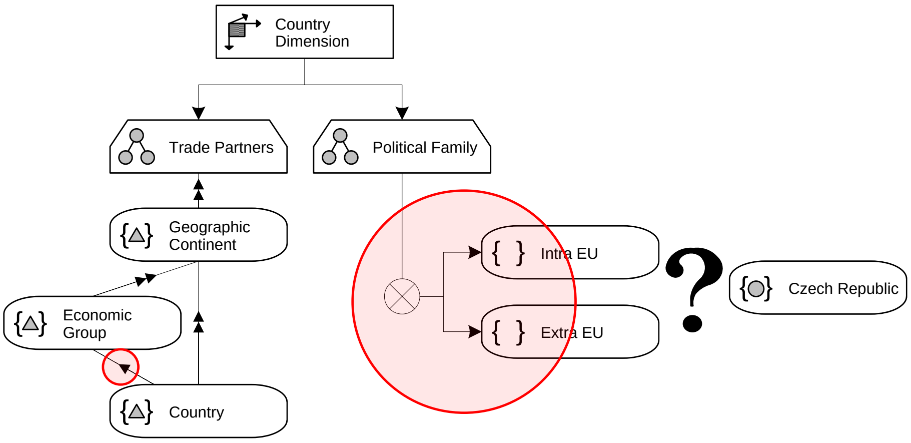

Figure 8 - Country Dimension

Both classifications present problems. In the first case we have to deal with the fact that some countries do not belong to an economic group. To this effect we should from the first moment represent this unbalanced hierarchy. The DW team is thus alert to the fact that they have to build a ragged dimension.

In the second case the depiction of the Intra and Extra EU as being fully exclusive sets of countries, makes us immediately aware we don’t know where to list some countries, like the Czech Republic, that only recently joined EU. If we are analyzing today’s data it should be in the Intra list, but if we want to query data from 2000 it belongs to the Extra EU group. The situation could be solved adding a dimensional attribute that would be the successive enlargements dates in which the countries joined; this would ensure that no country would be on both lists in any specific moment.

3.3.3 Uncover Dimensions

Another curious case emerged in the construction sector, while representing the data we had some discussion around the number of apartments in the buildings. It is highly desirable to know the total habitable units in a parish, but it is also important to know if lately we have been building large blocks of flats or houses for only one or two families.

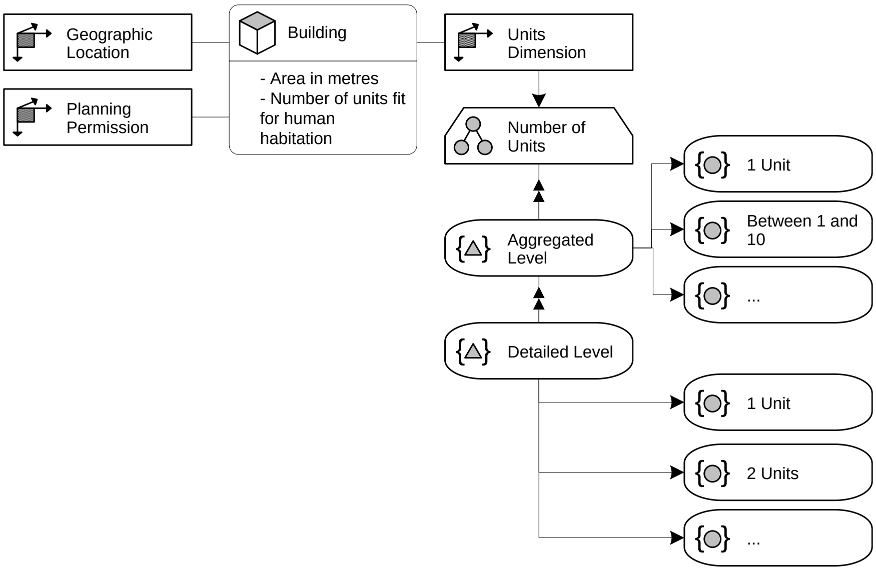

Figure 9 - Construction Model

In this situation the number of units should not only be a measure that can be summed up, but also a classification so that we can distinguish between houses and blocks of flats. This is often the case with population models where we have the person's age or with health models where we register the patient's weight, height and other discriminatory measures that can position an individual in a specific risk group.

In the construction sector a more interesting detail is related with the planning permission. An official number is associated with ever planning permission to construct or demolish a building. In this case, as previously in the Intra and Extra EU, both classification members are fully exclusive. More important than that, they will affect the measure: total number of livable units in the country, in different ways.

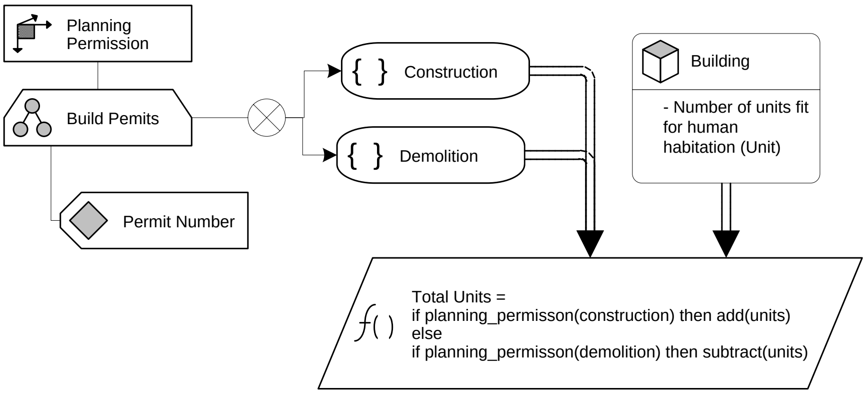
Planning Permission
Build Pemits
Permit Number
Construction
Demolition
Building
- Number of units fit for human habitation (Unit)
Total Units =
if planning_permisson(construction) then add(units)
else
if planning_permisson(demolition) then subtract(units)

Figure 10 - Planning Permission Model

It is important to show that the total units number formula changes depending on the planning permission classification. If this was not detected in the design phase probably the resulting system would only add all the units causing errors.

3.3.4 Time Series Integration

Changes in the formulas usually occur related to a classification or when we are building a time series where the data collected changed along the timeline. This happens in the last example presented about the structure of agricultural holdings.

Farm structure surveys data is collected every couple of years and one of its major groups of questions relates to land uses. A possible land use is permanent crops such as olive, vineyards or fruit and berry plantations. In this latest category, fruit and berry plantations, some changes occurred between the 2003's and the 2005's surveys.

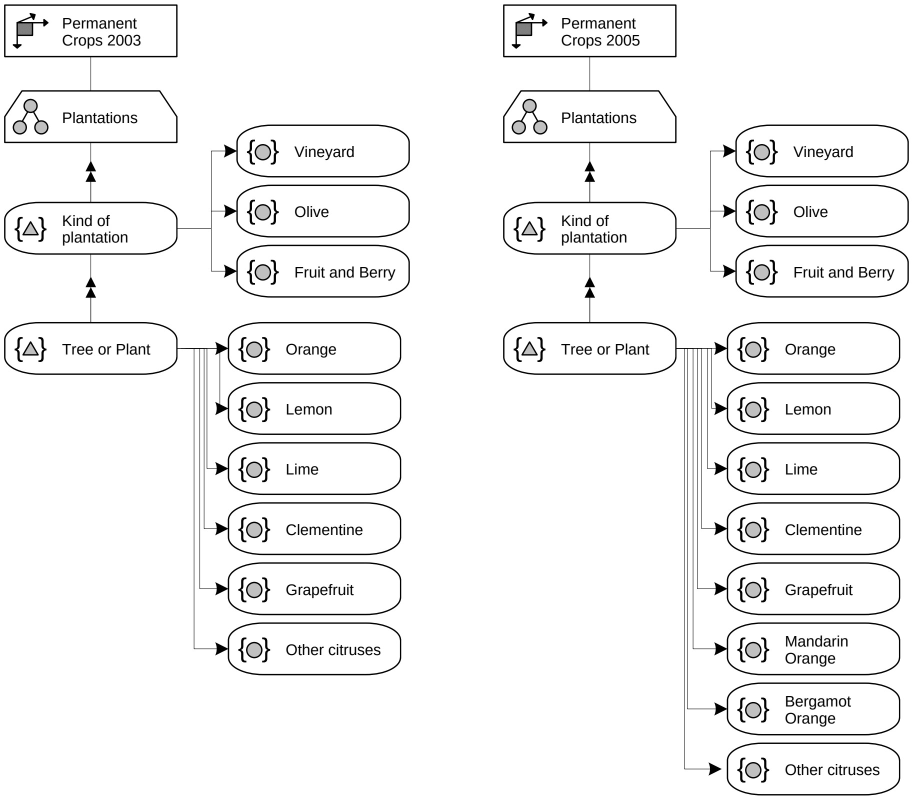

Figure 11 - Permanent Crops in Agricultural Time Series

In 2003 we had some juicy acidic fruits: orange, lemon, lime, clementine, grapefruit and other citruses. Nowadays mandarin and bergamot orange joined the list, thus causing that 2003's "other citruses" are not comparable to 2005's "other citruses". However the sum of all citruses is, so what we should do is design two new levels; an upper level for all citruses, and a lower level for miscellaneous citrus.

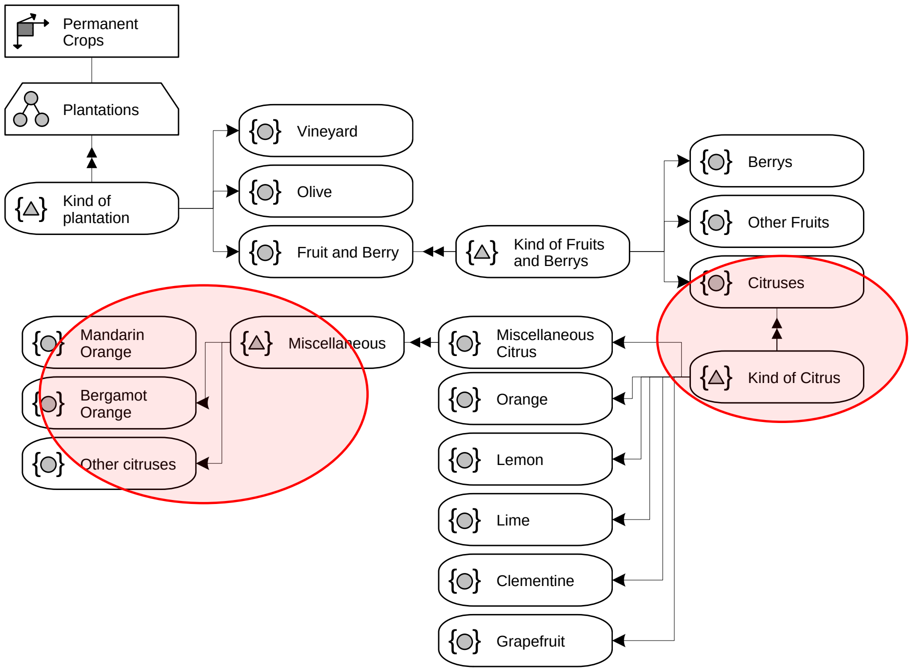

Figure 12 - Permanent Crops Time Series

The 2003 "other citruses" is in this case renamed to miscellaneous citrus and in 2005 this new value is calculated summing the mandarin and bergamot oranges with the residual group, other citruses.

This examples show the error detection that is possible in early stages if we design conceptual models before building our data warehouses. In every case the most important thing is to accurately represent the concepts and data relationships so that we can anticipate and solve the problems.

4. Conclusions

Since we adopted this conceptual design technique, almost a year ago, the time usually spent in the conceptual phase of a data warehouse project has reduced to only one third. Not merely that but also the need to revise and alter the structure has diminished and the future integration of other classifications has been simplified.

Conceptually modelling our data warehouse allows us to predict which data manipulations will be needed to integrate data collected for different purposes, specifically administrative and survey data.

Accepting and working with a well defined modelling technique also eases the communication between the statistician's and the informatics' teams, building a bridge of understanding. However, one of the most positive features of the adoption of any conceptual design technique is making it possible for the statistician's and the informatics' teams to work together towards a common goal. The obligation of developing an information conceptual model collectively, forces the working teams to give serious and careful thought to the involved concepts and their relationships in order to justify before others their design choices. This methodology has long term benefits in the soundness of the chosen structures and improving collaboration and coordination.

5. References

[1] ANSI (1975) Interim Report of the ANSI/X3/SPARC Study Group on Data Base Management Systems. *ACM SIGFIDET*, 7, 3-139.

[2] IBM (1969) http://www-03.ibm.com/servers/eserver/zseries/timeline/1960s.html accessed on 16/02/2006.

[3] CODASYL DATA BASE TASK GROUP (1971). Report, ACM, New York.

[4] Codd E. F. (1970). A Relational Model of Data for Large Shared Data Banks, *in Communications of the ACM*,Vol. 13, No. 6, pp. 377-87.

[5] Chen P. P. (1976). The Entity-Relationship Model - Toward a Unified View of Data. *ACM Transactions on Database Systems (TODS)*, Vo. 1 No. 1, pp. 9-36

[6] Hartson H. R., Hsiao D. K. (1976). Full protection specifications in the semantic model for database protection languages. *ACM/CSC-ER*. pp 90-95

[7] Shipman D. W. (1981). The functional data model and the data languages DAPLEX. *ACM Transactions on Database Systems (TODS)*. Vol.6, No. 1, pp. 140-173.

[8] Atkinson M., DeWitt D, Maier D. Bancilhon F, Dittrich K & Zdonik S. (1992). The object-oriented database system manifesto. Building an object-oriented database system: the story of 02, *Morgan Kaufmann Series In Data Management Systems*. USA.

[9] Abelló A., Samos J & Saltor F. (2001). A Data Warehouse Multidimensional Data Models Classification. *Spanish Research Program PRONTIC*. Universitad Politècnica de Catalunya.

[10] Gonçalves S (2005). Modelação Conceptual de Data Warehouses. Universidade Nova de Lisboa.

[11] Husemann B., Lechtenborger J. & Vossen G. (2000). Conceptual DataWarehouse Design, *Proceedings of the International Workshop on Design and Management of Data Warehouses (DMDW'2000), Stockholm, Sweden*.

[12] Cabibbo L. & Torlone R. (1998). A Logical Approach to Multidimensional Databases. *Lecture Notes in Computer Science*, Vol. 1377.

[13] Golfarelli M., Maio D. & Rizzi S. (1998). The Dimensional Fact Model: A Conceptual Model for Data Warehouses. *In International Journal of Cooperative Information Systems*, vol. 7, number 2-3.

[14] Sapia C., Blaschka M., Höfling G & Dinter B. (1998). Extending the E/R Model for the Multidimensional Paaradigm. *ER Workshops*.

[15] Bulos D. (1996). OLAP Database Design: A New Dimension. *DatabaseProgramming and Design*.

[16] Bulos D. & Forsman S. (2002). Getting Started with ADAPT: OLAP Database Design. *Symmetry Corporation*.