Università di Pisa

Dipartimento di Informatica
Corso di Laurea Magistrale in Informatica per l'Economia e per l'Azienda
(Business Informatics)

TESI DI LAUREA

Design and development of a data warehouse for a company trading natural stone

Relatore:
Prof. Salvatore RUGGIERI

Candidato:
Lorenzo LA COGNATA

Tutor Aziendale:
Dott. Fabio MORSIANI

Anno Accademico 2012-2013

Abstract

This thesis describes the design and development of a decision support system for R.E.D. Graniti, a group of companies which extract and market natural stone all over the world. Initially every company of the group had its own information system, completely independent from the others. When R.E.D. Graniti decided to adopt an Enterprise Resource Planning system to increase uniformity and flexibility, the use of a data warehouse to store and analyze information became a possible powerful source of competitive advantage.

After a presentation of the case study and an analysis of the current situation of the natural stone sector, the design and development processes of both the data warehouse and the business intelligence system are described in their entirety. The phases covered by this document are the requirement analysis, the conceptual and logical design of the data warehouse, the extract, transform and load process, the business intelligence metadata construction and finally the development of reports and dashboards.

Contents

Introduction

Presentation of the problem

R.E.D. Graniti group is one of the world leaders in extracting and trading rough granite and marble blocks worldwide. The company recently considered necessary the increase of organizational flexibility and information uniformity. This need induced the management to start a development path focused on the adoption of a new global Enterprise Resource Planning (ERP) system, providing an integrated view of business processes across the different companies all over the world. R.E.D. Graniti appointed the professional services firm Deloitte for the project and the ERP software chosen was SAP.

Alongside this project it was decided to provide the management with an integrated software for enterprise reporting to assist the strategic management of the company business. The software adopted was Oracle Business Intelligence and in this document we will discuss the design and implementation of the chosen data warehouse / business intelligence solution.

The goal of the business intelligence project is to provide R.E.D. Graniti with a web-based analysis and reporting application for monitoring the business evolution, in particular sales, stocks and costs. Information will be structured in tables, graphs and other forms of data representations to give a clear and immediate view of business performances to the users, who will have the possibility of easily interact with reports to perform multidimensional analyses. The application will allow the management to autonomously produce business reports in a simple and timely manner, export them in different formats or send them via email.

This was very difficult or practically impossible to achieve with the existing information system. While it was possible to examine data and information quite easily, a great deal of effort was necessary to actually use it. Data manipulation and presentation often required additional work to be manually done using spreadsheets, especially for task like grouping, filtering and detecting anomalies and errors.

Literature review

The data warehouse design approach used and the framework for requirements analysis follow the ones presented by Albano [1] and, to a lesser degree, by Kimball et al. [3] and Oracle [7]. Information about the natural stone sector greatly draw on the *XXIV world marble and stones report 2013* by Montani

[4]. Data about Deloitte and R.E.D. Graniti come from these companies' internal documentation. Oracle documentation ([6], [9], [8] and [5]) is the reference for software characteristics and technical parts. Few [2] was of great inspiration for the development of dashboards.

Thesis contents

This thesis describes the design and development of the data warehouse for storing data about R.E.D. Graniti's sales, stocks and costs and the consequent design and development of business intelligence dashboards and reports, to be used as a decision support system by the company's management.

The first phase of the data warehouse design is the requirements analysis and can be divided in two sub-phases. First requirements are gathered and described in a natural language and then specified with a more technical tabular structure, with a view to modeling the conceptual design of the data warehouse. The second phase is the initial data mart conceptual design, derived from the analysis usually performed by users and therefore considered an analysis driven design. After that a data driven design is considered and integrated with the initial conceptual design to produce a final data mart conceptual design. The last phase consists of the data marts logical designs, finally merged in a data warehouse logical design.

Development of the data warehouse followed the design, with a particular focus on development and testing of extract, transform and load procedures. Concurrently the metadata for the business intelligence tool are created. After data was loaded into the data warehouse a set of reports and dashboards are designed and developed, to replicate the existing reporting structure in use in R.E.D. Graniti.

The case study is presented in Chapter 1, which gives information about R.E.D. Graniti and the natural stone sector as well as the requirements gathering and the structure of the existing information system. The requirement analysis and the initial conceptual design phases are covered by Chapter 2, with separate sections for each of the business processes involved. Chapter 3 is dedicated to the final conceptual design, the analysis of the source operational systems and the logical design of the data warehouse. Chapter 4 briefly presents the tools used and the Oracle development environment. A detailed description of the data warehouse structure and the PL/SQL procedures used to extract, transform and load data is given in Chapter 5. The business intelligence metadata repository construction is explained in Chapter 6. Finally Chapter 7 is dedicated to the reporting phase.

Chapter 1

The case study

This chapter describes the firms involved in the data warehouse project discussed here. The customer's business in general and its business processes interested by this project are explained in details as they were discussed in depth during the preliminary meetings with R.E.D. Graniti. Then we introduce the business requirements expressed by the senior management for every process.

1.1 Deloitte

Deloitte is the brand name for the UK private company limited by guarantee called Deloitte Touche Tohmatsu Limited (DTTL). As stated in the Global Impact Report 2013, there are currently more than 200,000 professionals in 47 independent member firm of the Deloitte network which collaborate in over 150 countries to provide audit, tax, consulting, enterprise risk and financial advisory services. These firms are organized on a country or regional basis and are owned and managed locally, but share values, methodologies, client service standards and other guidelines.

The common organizational structure can be classified in five areas:

- Audit (financial statement audits, accounting advisory, assurance services)
- Consulting (human capital, strategy & operations, technology)
- Enterprise Risk Services (governance, regulatory & risk, security, privacy & resiliency)
- Financial Advisory (corporate finance advisory, restructuring services, valuation services)
- Tax & Legal (global business tax, global employer services, legal)

Deloitte goals include enabling companies to discover revenue opportunities, navigate complex business landscapes and develop effective solutions to address challenges; the influence on these activities involve leadership, insight, expertise, and deep knowledge of the business domains. Deloitte's

approach strongly focuses on integrity and responsible business practices, quality, independence, data privacy and security.

In Italy Deloitte employs over 3,000 professionals in 18 different cities. The project described here was completed in the Deloitte eXtended Business Services (Deloitte XBS) office in Parma and involved one manager and two analysts.

1.2 The natural stone sector

Stones are natural formations of one or more minerals and have been quarried through the centuries. Hundreds of types of stone exist but the Marble Institute of America classifies natural stone into two main categories, calcareous stone, composed mainly of calcium carbonate, and siliceous stone, composed mainly of silica or quartz. Marble, travertine and limestone are some of the calcareous stone, granite, slate and sandstone are common siliceous stones. Calcareous stones are more sensitive and need more care and maintenance, while siliceous stones have a greater resistance and are more durable.

Marble and granite are two of the most famous and most used construction stones. Marble is a metamorphic rock composed of recrystallized limestone. The minerals most commonly composing marble are calcite and dolomite. Granite is a igneous rock, primarily made of quartz, potassium feldspar and sodium feldspar. It is a very hard material and usually has darker colors than marble.

Nowadays both granite and marble are extracted using the diamond wire cutting technique. This technique is based on the use of a 5 millimeters diameter steel wire with the help of silica sand and large amounts of water as a lubricant. Because of the hardness of diamonds a continuous loop of tensioned steel moving at 20 kilometers per hour is able to cut 20 centimeters of marble per hour. After drilling several holes in the block the insertion of wedges will eventually split the stone along the cut, pursuing the strategy to cut large blocks and divide them into smaller pieces.

The final output of this process is a block, usually weighting from 10 to 20 metric tons, with average dimensions of 2.5 meters length, 1.5 meters width and 1 meter height. Blocks this size are then sold to vendors equipped with appropriate sawmills to further cut a block. A block can also be split into several slabs, usually 2 to 4 centimeters thick, possibly polished and treated and then separately sold to customers. Another form to which a block can be reduced, with an added value even higher than slabs, is the tile, usually best for floors and walls coverings, though the competition of ceramic tiles is very strong.

Three quarters of the total processed material is dedicated to the building sector, especially floors and paving and special works. Other common uses for natural stone are funeral art, wall cladding and internal design, such as kitchens and bathrooms.

New records of production, interchange and consumption of natural stone were achieved in the last years, with China, India, Turkey and Brazil leading figures in this development. In the last fifty years we observed a sensible reduction of costs, partially due to improvements in technologies and transports, and high investments in research and plants. The flow of quantitative movements increased of more than five times in the last twenty five years, with a constant increase trend interrupted only in 1998, 2008 and 2009. In the same period the growth of the stone sector has been considerably higher than that of the global economic system.

In 2012 the stone industry marks a continuous productive growth, reaching a total of 123.5 million tons of raw production and 72.9 million tons of net processed production, 20% more than five years ago. Marble and other calcareous materials still represent more than half of the production, but granite and siliceous materials in general had a certain recovery, while slate maintains a secondary and declining role, as can be seen in Table 1.1.

Stone type	**Tons/1000**	**Percentage**
Calcareous	72.25	58.5%
Siliceous	45.75	37.0%
Other	5.50	4.5%
Total	123.50	100.0%

Table 1.1: Natural stone raw production by stone type in 2012

Almost two thirds of the world production comes from China, India, Turkey and Brazil, while the first 12 countries make 85% of the world production. Italy maintains the lead among the European Union, followed by Spain and Portugal.

Country	**Tons/1000**	**Percentage**
China	38.00	30.8%
India	17.50	14.2%
Turkey	11.50	9.3%
Brazil	7.50	6.1%
Italy	7.25	5.8%
Iran	7.00	5.7%
Spain	5.25	4.3%
Egypt	3.00	2.4%
Portugal	2.75	2.3%
USA	2.50	2.0%
Others	21.25	17.2%
Total	123.50	100.0%

Table 1.2: Natural stone raw production by country in 2012

The total world interchange in 2012 was 52 million tons, 4.5% more than in 2011. As Table 1.3 shows, raw materials surpassed processed materials, thanks to the good recovery of granite (+15%

with respect to the previous year) and because of the increasing demand from China and other processing countries. This is an important strategic tendency, because international transport has to expect greater loads of materials and because the theoretical increase in expenses is balanced by the contained costs of concentrating the production activities in third world countries. China, India, Brazil, Turkey and other extra-European countries increased their importance in the interchanges, reaching two thirds of export and half of import volumes, with the European Union losing positions.

Product Type	**Tons/1000**	**Percentage**
Raw calcareous	14.59	28.1%
Raw siliceous	12.58	24.2%
Subtotal Raw	27.17	52.3%
Simple finished products	3.14	6.1%
Special finished products	20.31	39.1%
Slate	1.25	2.5%
Subtotal Processed	24.70	47.7%
Total	51.87	100.0%

Table 1.3: Natural stone import-export by product type in 2012

Increase of interchanges means higher requirement in quality, hence the increasing attention to production and improvements in technology, such as the automatic resin treatment of slabs. Also constant investments in research, promotion and distribution become necessary, with Europe still leading for professional values and creativity. Another sector affected by this tendency is the infrastructures sector, especially since, given the high specific weight of natural stone, transport costs have a great impact over the final price of products. Nowadays the use of sea transport is complemented by road transport, usually required for journeys from and to harbors, with similar considerations valid for railway transport, although less used. Another point to be considered is the extraction and processing waste produced, 180 million tons in 2012, which amounts to 60-70% of the total quarried volume and which must be sent to the dump, with non-negligible effects on the environment.

Natural stone consumption grew at the same pace of production. The world consumption reached 1350 million equivalent square meters[1] in 2012, 6.7% more than the previous year. Today the use of natural stone is rather universal, as shown by the purchase rate in, for example, Barbados, Macao and Mauritius. As for production and exchanges, a strong concentration is registered, with only fifteen countries consuming more than ten million meters and China consuming more than a quarter of the total world volume. As for the per capita use of natural stone, some mid-sized countries leads the ranking, with Belgium, Switzerland, Taiwan and Saudi Arabia consuming more than one square meter per inhabitant.

[1]Equivalent square meters are referred to the conventional slab thickness of 2 centimeters

Country	Mill. eq. m^2	Percentage
China	358.0	26.5%
India	104.6	7.7%
USA	76.9	5.7%
Brazil	51.2	3.8%
Italy	49.5	3.7%
South Korea	44.4	3.3%
France	32.0	2.4%
Germany	31.5	2.3%
Taiwan	29.4	2.2%
Saudi Arabia	28.9	2.1%
Others	543.6	40.3%
Total	1350.0	100.0%

Table 1.4: Natural stone consumption by country in 2012

1.3 R.E.D. Graniti

When R.E.D. Graniti was founded by Giorgio Conti in 1965 the Apuan stone industry was focused on the extraction of marble produced in the quarries in Carrara; the company instead dealt in several varieties of granite, materials little known at the time, which then quickly became reference points for world-class architecture.

In those years some new technologies were developing, such as diamond tools, and there were innovations in polishing and in the granite processing industry in general. This lead to a general change in market trends, the morphological and chromatic characteristics of granite determined the aesthetic choices of architects and the preferred materials were produced in a limited number of countries.

The ideas of Giorgio Conti shaped R.E.D. Graniti as a complex international organization for the selection, storage and transport of marbles and granites from the quarries to the ports and then to the various markets. In the mid-seventies the group began actual trading, buying blocks for resale, and the motto "the material first" was coined. Then the key strategy involved strategic control of the raw material, trading according to exclusive contracts and acquisition of quarries throughout the world; nowadays approximately 80%-90% of all material sold are produced in the company own quarries.

The eighties were a decade of significant growth and international expansion for the Apuan stone industry, for both production and trade; in the difficult nineties the company's strategy proved to be decisive as the sales doubled in just eight years, with an increase much higher than the rest of the market.

The continuous search of new quarries and top quality materials and the substantial investments toward the strategic control of raw material lead to the expansion of R.E.D. Graniti in Brazil, South Africa, Namibia and Zimbabwe. The group has also an office in Bangalore (India) since 1976, the Vermont Quarries Corporation was set up in 1992 in the United States, in order to continue the extraction of the famous white Danby marble quarry, and in 1995 the group acquired Finska, the largest Scandinavian company working in the production of granite blocks.

The company headquarters are in Massa, Italy, where the company has a warehouse which occupies an area of approximately 60,000 square meters and where 25,000 tons of new materials arrive every month from the port in Marina di Carrara, which became the most important hub for this industry. All of this led to the creation of a network of specialized ports in various countries, including Bangalore (India), Vitoria (Brazil), Vigo (Spain), Larvik (Norway) and Durban (South Africa).

Today R.E.D. Graniti handles and trades granites and, less frequently, marbles batches in different shapes and sizes. Different qualities are commercialized, with different colors, patterns, composition and characteristics. R.E.D. Graniti's catalog offers more than eight hundred different material qualities, divided by country of origin, the most popular of which are Nero Africa, Giallo Veneziano, Rustenburg and Nero Zimbabwe.

Figure 1.1: Some of the material qualities in the company's catalog

R.E.D. Graniti S.p.A. declared 68.3 millions euro of revenues for fiscal year 2012, 8% less than the previous year, with a margin of 785 thousands euro. The companies under the direct control of the parent company work into two divisions, production and sales. These combined departments employ a work force of about 1500 people. A small portion of R.E.D. Graniti's companies also transforms blocks in slabs. Processing materials directly where they are extracted greatly contribute to reduce costs and may be an important strategic change that can be observed in the group's future.

Production companies are located in strategic areas of world stone production such as Brazil, South Africa, Zimbabwe, Namibia, United States, Finland and Norway. Over 200,000 cubic meters in blocks are extracted every year from a total of more than thirty quarries and then sold directly or, more frequently, sent to warehouses in the main sales center and sold by companies of the group's business division. Important sales centers are located in Europe (Italy, Belgium, Spain, France, Poland, Croatia), Canada, Brazil and China and also handle other varieties of stone materials, even if the company's own materials account for 90% of sales. The main companies of the group are listed in the following table.

Country	**Company**	**Division**
Brazil	R.E.D. Graniti Brasil (Ltda)	Sales
	Pedreiras do Brasil (Ltda)	Production
	R.E.D. Graniti Mineração (Ltda)	Production
Canada	R.E.D. Graniti North America Inc	Sales
China	R.E.D. Graniti Xiamen (Co Ltd)	Sales
Finland	Finska Stenindustri Ab	Production, sales
France	R.E.D. Graniti France Sarl	Sales
	Rsa Granits Sas	Sales
Germany	R.E.D.Graniti Deutschland Gmbh	Sales
India	R.E.D. Graniti Quarries and Blocks India	Sales
Italy	R.E.D. Graniti S.P.A.	Sales
Madagascar	R.E.D. Graniti Madagascar Sarl	Production
Mozambique	R.E.D. Graniti Mocambique Limitada	Sales
Namibia	R.E.D. Graniti Namibia (Pty Ltd)	Production
	Arandis Marble (Pty Ltd)	Production
	Caprivi Marble & Granite (Pty Ltd)	Production
	Damara Granite (Pty Ltd)	Production
	Aus Marble (Pty Ltd)	Production

Poland	Red Graniti Poland Sp. z o.o.	
Portugal	Beauport Transportes Maritimos (Lda)	Services
South Africa	R.E.D. Graniti South Africa (Pty Ltd)	Production
	Rossal No 136 (Pvt Ltd)	Transformation
	Kelgran Africa (Pty Ltd)	Production
	Verde Bitterfontein (Pty Ltd)	Production
Spain	R.E.D. Graniti España SA	Sales
Switzerland	Stonevolution AG	Sales
United States	Vermont Quarries Corp.	Production, transform.
	Colorado Stone Quarries Inc	Production
	Virginia Mist Granite Corp. Inc	Production
Zimbabwe	Ilford Services (Pvt Ltd)	Production
	Panhandle Mine (Pvt Ltd)	Production
	Roserocks Mine (Pvt Ltd)	Production
	Sidingo Mine (Pvt Ltd)	Production
	Southern Granites (Pvt Ltd)	Transformation

The next table lists the main stockyards controlled by the group.

Country	**Stockyard**
Belgium	Antwerp
Brazil	Suape and Vitòria
Canada	Quebec City
China	Xiamen
Croatia	Rijeka
France	St. Malo
Italy	Massa and Verona
Poland	Stettino
Serbia	Smederevo
Spain	Vigo

Figure 1.2 shows R.E.D. Graniti's stockyards in green together with companies, indicated in red. The blue dot represents headquarters in Massa.

Figure 1.2: The main companies (in red) and stockyards (in green) of R.E.D. Graniti group

1.4 Business processes analysis

Three different business processes were discussed during the preliminary meetings with the customer and were included in the Data Warehousing project discussed in this document: sales, stocks and costs. A detailed analysis of each business process is given in the rest of this section.

1.4.1 Sales process

The most important business process considered involves sales. R.E.D. Graniti primary business consists of extracting marble and granite blocks from quarries, or buying them from other companies, most of the times processing them in some way (e.g. polishing) and then selling them to customer throughout the world. A sales document is generated where date, materials, customer, storage location and other information are stored.

There are three different situations which usually lead to a sale. The first situation happens when a customer wins an important contract and needs large quantities of one or more specific materials, with a particular attention to quality and uniformity needed. The large number of quarries owned by R.E.D. Graniti make the group based in Massa one of the few in the world to be able to accommodate these kind of requests. A similar situation arises when a customer needs a small number of blocks and is willing to choose them in a stockyard or directly in a quarry. Assisted by a sales agent the customer hoses down the blocks and tests them for imperfections or defects, then the sale is negotiated and formalized.The last situation involves the group's sales agents too, since they may propose, on their own accord, business deals to long-time customers, for example suggesting new materials or special offers for materials kept in stock for a long period of time.

Collection of business requirements

The following table lists the business requirements expressed by R.E.D. Graniti management. Customer and material dimensions clearly are the most important, while sales amount, costs, margin and volume are the crucial measures.

	Sales process
1	The year to date sales amount, costs, margin and volume, and the corresponding variations versus last year, by month.
2	The total sales amount, costs, margin and volume, and the corresponding variations versus last year, by month, by customer and by material quality.
3	The top 15 customers by sales amount, or by sales volume, by year and by material quality.
4	The top 15 material qualities by sales amount, or by sales volume, by year and by customer.
5	The top 15 material countries by sales amount, or by sales volume, by year.

1.4.2 Stocks process

Due to the international sphere of R.E.D. Graniti business, marble and granite blocks are often moved from one storage location to another, in different parts of the world, to better fulfill the customer needs and to reduce the delivery time. Moreover, R.E.D. Graniti does not own any quarry in Italy, unlike in other countries, even if a large part of the customers is based in Italy. This means that the company based in Massa only buys batches, often from the other companies of the group based in the rest of the world, and then resell them to the customers.

Another reason for the need of stocking batches, possibly for several months or years, is that, with the only exception of white, black and, to a lesser degree, yellow, natural stone colors follow the trend of the moment. Some color may go out of fashion and then become fashionable again after a certain period of time, maybe in a different part of the world.

Collection of business requirements

For the reasons stated before, a particular attention is paid to the stock levels in each storage location, in terms of batches value and volume. Beside the material and the storage location, also the vendor dimension is considered important for the stocks analysis.

	Stocks process
6	The total stock amount and stock volume by month, by storage location, by material quality and by vendor.

1.4.3 Costs process

The last business process considered is about costs. Every cost paid for buying a batch, transporting it, selling it or any other operation which comes with a price, is recorded in a cost document. These costs greatly influence the price at which the batch can be sold, and therefore influence the company margin.

Collection of business requirements

Analyzing the costs by material is crucial for R.E.D. Graniti management, and the main dimensions for the analysis are the cost type and the vendor that provided the service for which the cost was paid.

	Costs process
7	The total cost and unit cost by material batch, by cost type and by vendor.

1.5 The initial information system

The preliminary meeting was attended also by the people in charge of R.E.D. Graniti's information system in Italy in recent years. Technical aspects of the transition from the old to the new information system were discussed, particularly focusing on the extraction of the necessary data and on the basic logic behind calculations and data transformations made in the existing reports.

1.5.1 AS/400

Application System/400 (AS/400), later renamed to eServer iSeries and System i, is IBM's previous generation of midrange computer systems introduced in 1988. Its object-based operating system was originally called OS/400 and was later renamed to i5/OS and IBM i. It includes DB2/400 Relational DataBase Management System (RDBMS) and support for security and multiple users.

1.5.2 Reporting

A customized business reporting software was developed for R.E.D. Graniti in Italy, accessible via a corporate intranet. Reports answered the business requirements already described in this chapter and were the base for dashboards and reports created using the new business intelligence application.

Besides the fact that only data from Italian companies using AS/400 was shown, another substantial difference was that the old reporting software was intended to be used only by senior management in Italy. No concept of internationalization or localization was present. Moreover, while reports could be distributed in print form, the reporting software was not capable of producing pixel-perfect reports, therefore the managers had to put a great deal of effort into creating highly formatted periodic balances.

1.5.3 Other requirements

Data Warehouse

Since the business intelligence system will be used by knowledge workers in different parts of the world strong requirements were made about currencies and units of measurement. R.E.D. Graniti requested that every monetary measure, such as sales amounts or costs amounts, should be expressed in four, possibly different, currencies. The first currency has to be the transaction's currency, that is the currency actually used to pay or collect the invoice. Most often this currency is Euro or United States Dollar, but can hypothetically be any existing currency. The second currency required is the official currency of the company's country, with the company being the specific entity of R.E.D. Graniti group to which the transaction is referred. As an example the company's currency for a sales document issued by *R.E.D. Graniti South Africa* will be the South African Rand. Besides these two currencies all the monetary measures must be expressed also in Euro and United States Dollars, although often equal to the transaction's or company's currency.

Reporting

For the same reasons stated in the previous section a particular focus on internationalization and localization of the business intelligence application was needed. It had to be possible to use the application in various languages, although English was the language chosen for descriptions in the data and for dashboard titles and texts. Moreover the application had to adapt to specific regions displaying locale-specific components in the desired format. Examples are date and time formats and formatting of numbers, such as decimal separators and digit grouping.

Another important requirement concerned information privacy. Except for senior management, almost exclusively based in Italy, every user of the business intelligence system should have access

only to a certain subset of data instead of the global integrated data from every R.E.D. Graniti's company. These restrictions mean that every user can have access to data from the company to which he belongs and in some cases from companies in the same country or region (e.g. all the South African and Namibian companies). This requirement translated to the use of row-level security and application roles in the business intelligence tool.

1.5.4 The final information system

The adoption of a global ERP system to have an integrated view of R.E.D. Graniti group's business results in the substitution of AS/400 with SAP. Both the system will be temporarily sources of the data warehouse, with AS/400 eventually discontinued. Details on the migration process are explained in section 3.1.

Oracle Database represents the following layer, consisting of the staging area and the data warehouse itself. The final layer of the information system, the reporting layer, consists of Oracle Business Intelligence. Details on the single components of the final information system are give in chapter 4.

Chapter 2

Analysis requirements specification and initial conceptual design of data marts

After a brief introduction to data warehousing, a detailed requirements specification is given for each business process considered. Dimensions, dimensional hierarchies, measures and the conceptual design of data marts are described.

The formalism which will be used for the conceptual modeling is the simplified version proposed in [1] of the Dimensional Fact Model (DFM) designed by Golfarelli for data warehouse modeling.

2.1 Introduction to Data Warehousing

As defined by William H. Inmon in 1990 a data warehouse is "a subject-oriented, integrated, non-volatile, and time-varying collection of data in support of management's decisions". Data is stored by business subject, meaning that related data elements are linked together. The data warehouse integrates data from multiple sources and these data are made consistent. Data is static, it is never changed or deleted, although new data is usually added periodically. Differently from operational system, the stored data do not contain the current values but the history of data values, hence a time dimension is always included in data. Data is specifically designed to be transformed into knowledge and thus support decision makers in the organization.

Business intelligence usually use data from a data warehouse. Forrester Research defines business intelligence as "a set of methodologies, processes, architectures, and technologies that transform raw data into meaningful and useful information used to enable more effective strategic, tactical, and

operational insights and decision-making" or in a narrower sense "just the top layers of the business intelligence architectural stack such as reporting, analytics and dashboards".

The foundations of a Data Warehouse are facts, single events associated with a business process. A fact is made of measures, numeric values corresponding to measurements related to the fact. As an example, the sale of a granite batch could be a fact, and the amount in Euro received for the sold batch a measure. Facts are related to dimensions, which categorize data and are used to answer business questions. A dimension is made of attributes, for example customer is a common dimension and customer country a possibile attribute of this dimension. A hierarchical relationship between different dimensional attributes may exists, the most common being the relation between year, quarter, month and day for temporal dimensions.

2.2 Sales process

In this section we describe the requirements specification and the initial conceptual design of the Sales data mart. A fact is about the sale of a certain material batch as specified in a line of a sales document.

2.2.1 Requirements specification

From the collection of business requirements the following requirements about the Sales fact are derived. Each analysis requirements is associated with the relative dimensions and measures. For brevity, the corresponding original measure is used instead of a derived measure.

Sales process

N	Requirements analysis	Dimensions	Measures
1	The year to date sales amount, costs, margin and volume, and the corresponding variations versus last year, by month.	Document Date (Month)	Sales Amount, Sales Costs, Sales Volume
2	The total sales amount, costs, margin and volume, and the corresponding variations versus last year, by month, by customer and by material quality.	Document Date (Month), Customer, Material (Quality)	Sales Amount, Sales Costs, Sales Volume
3	The top 15 customers by sales amount, or by sales volume, by year and by material quality.	Customer, Document Date (Year), Material (Quality)	Sales Amount, Sales Volume
4	The top 15 material qualities by sales amount, or by sales volume, by year and by customer.	Material (Quality), Document Date (Year), Customer	Sales Amount, Sales Volume
5	The top 15 material countries by sales amount, or by sales volume, by year.	Material (Country), Document Date (Year)	Sales Amount, Sales Volume

We describe the Sales fact, its granularity and the preliminary dimensions and measures involved.

Sales Fact

Description	Preliminary Dimensions	Preliminary Measures
A fact is about a batch sold.	Document Date, Material, Customer	Sales Amount, Sales Costs, Sales Volume

We describe the Sales fact dimensions and their granularity.

Dimensions

Name	Description	Granularity
Document Date	The date the invoice is issued	A day
Material	The material batch hierarchy	A material batch
Customer	The customer to which the document is addressed	A customer

Below we describe each dimension with its attributes and, in certain cases, their sample values.

One of the most important dimension to give facts their context is the date. Two dimensional hierarchies are defined, one going from day to year, the other including day and week. Two attributes, the day of the week and the month of the year, are outside hierarchies.

Document Date

Attribute	**Description**
Day	The day in the format YYYYMMDD.
Week	The week in the format YYYYWW.
Day Week	The day of the week (i.e. from *Monday* to *Sunday*).
Month	The month in the format YYYY MMM (e.g. 2014 April).
Month Name	The name of the month (i.e. from *January* to *December*).
Quarter	The quarter in the format YYYYQ.
Year	The year in the format YYYY.

Every fact corresponds to a specific batch, which is a block, slab or tile, to which a unique ten digit numeric code is assigned. A batch belongs to a quality, which gathers together batches with similar characteristics and usually extracted from the same quarry. Every quality is identified by a unique three digit alphanumeric code (e.g. *GVO* for *Giallo Veneziano*).

Dimensional categories are also defined for batches, with sizes ranging from 280 x 180 centimeters to 180 x 80 centimeters. However special categories are reserved for particularly small or extraordinarily big batches, for example the ones needed for large structures made of a single piece of natural stone. Every material size is defined using a three digit alphanumeric code (i.e. *C05*).

The combination of a material quality, a dimensional category and, only for slabs, a thickness and a working status, results in a material. Materials are identified using alphanumeric codes of variable lengths depending on the material type (block, slab or tile). For blocks the code includes one digit for the material country, followed by the code of the material quality and the code of the material size. For example the Brazilian material *Giallo Veneziano 280 x 180 cm* is defined by the code B GVO C00. The codification used for tiles is similar, except for the letter *T* always placed before the size code (e.g. B GVO T S00). For slabs the code includes one digit for the material country, followed by the code of the material quality, a two digits alphanumeric code for the working status, the code of the material size and two digit for the slab width. For example the material *Giallo Veneziano 260 x 125 cm Polished 4 cm* is defined by the code B GVO PO S02 04.

Finally the material country represents the top level of the material dimensional hierarchy, which also comprises material quality, material and batch. Other attributes are used to store information about the batch such as its division or working status. Another attribute stores the unit of measure,

which usually is cubic meters for granite blocks, metric tons for marble blocks and square meters for slabs.

Material

Attribute	**Description**
Batch	The material batch.
Material Quality	The commercial name given to the quality of granite or marble extracted from a quarry (e.g. *Imperial Brown*).
Material Size	The dimensional category of the batch.
Material	An intermediate level which combine the material quality with the material size.
Material Country	The country of the quarry from which the material quality is extracted.
Material Division	The division of the material batch (e.g. *Granite block*).
Material Group	The group of the material batch (e.g. *Unpolished slabs*).
Material Status	The working status of the material batch (e.g. *Leather*).
Material Type	The type of the material batch (e.g. *Block*).
Material UOM	The unit of measure used for the material quality (e.g. M^3).

Another important dimension for sales is the customer dimension. A hierarchy including customer country and region is defined, while other geographical attributes such as address, zip code and city are left outside the hierarchy because of their poor data quality. A second hierarchy involves customer group, which is used for customers belonging to a group of companies.

Customer

Attribute	**Description**
Customer	The customer.
Customer Group	The group of customers to which the customer may belong.
Customer Country	The country in which the customer is based.
Customer Region	The region of a country in which the customer is based.
Customer Address	The address of the customer.
Customer Zip	The zip code of the customer.
Customer City	The city of the customer.
Customer Tax Num	The tax number of the customer.

We describe the dimensional hierarchies, with their relative attributes and the hierarchy type.

Dimensional Hierarchies

Dimension	**Description**	**Type**
Document Date	Year → Quarter → Month → Day	Balanced
Document Date	Year → Week → Day	Balanced
Material	Material Country → Material Quality → Material → Batch	Balanced
Customer	Customer Country → Customer Region → Customer	Balanced
Customer	Customer Group → Customer	Balanced

In the following page we describe the Sales fact measures, their aggregability and their formula in case of calculated measures.

Measures

Measure	**Description**	**Aggregability**	**Calculated**
Sales Amount (Sa)	Amount received	Additive	No
Sales Costs (Sc)	Total costs incurred	Additive	No
Sales Margin (Sm)	$Sa - Sc$	Additive	Yes
Sales Margin %	$\frac{Sm}{Sa}$	Additive	Yes
Sales Volume (Sv)	Volume in M^3 or FT^3	Additive	No
Sales Amount LY(SaLy)	$AGO(Sa, Year, 1)$	Additive	Yes
Sales Costs LY (ScLy)	$AGO(Sc, Year, 1)$	Additive	Yes
Sales Margin LY (SmLy)	$AGO(Sm, Year, 1)$	Additive	Yes
Sales Margin % LY	$AGO(Sm\%, Year, 1)$	Additive	Yes
Sales Volume LY (SvLy)	$AGO(Sv, Year, 1)$	Additive	Yes
Var% Sales Amount vs LY	$\frac{Sa-SaLy}{SaLy}$	Non additive	Yes
Var% Sales Costs vs LY	$\frac{Sc-ScLy}{ScLy}$	Non additive	Yes
Var% Sales Margin vs LY	$\frac{Sm-SmLy}{SmLy}$	Non additive	Yes
Var% Sales Volume vs LY	$\frac{Sv-SvLy}{SvLy}$	Non additive	Yes
Sales Amount YTD (SaYtd)	$TODATE(Sa, Year)$	Semi additive (Date)	Yes
Sales Costs YTD (ScYtd)	$TODATE(Sc, Year)$	Semi additive (Date)	Yes
Sales Margin YTD (SmYtd)	$TODATE(Sm, Year)$	Semi additive (Date)	Yes
Sales Margin % YTD	$TODATE(Sm\%, Year)$	Semi additive (Date)	Yes
Sales Volume YTD (SvYtd)	$TODATE(Sv, Year)$	Semi additive (Date)	Yes
Sales Amount YTD LY (SaYtdLy)	$TODATE(SaLy, Year)$	Semi additive (Date)	Yes
Sales Costs YTD LY (ScYtdLy)	$TODATE(ScLy, Year)$	Semi additive (Date)	Yes
Sales Margin YTD LY (SmYtdLy)	$TODATE(SmLy, Year)$	Semi additive (Date)	Yes
Sales Margin % YTD LY	$TODATE(Sm\%Ly, Year)$	Semi additive (Date)	Yes
Sales Volume YTD LY (SvYtdLy)	$TODATE(SvLy, Year)$	Semi additive (Date)	Yes
Var% Sales Amount YTD vs LY	$\frac{SaYtd-SaYtdLy}{SaYtdLy}$	Non additive	Yes
Var% Sales Costs YTD vs LY	$\frac{ScYtd-ScYtdLy}{ScYtdLy}$	Non additive	Yes
Var% Sales Margin YTD vs LY	$\frac{SmYtd-SmYtdLy}{SmYtdLy}$	Non additive	Yes
Var% Sales Volume YTD vs LY	$\frac{SvYtd-ScYtdLy}{SvYtdLy}$	Non additive	Yes

2.2.2 Initial conceptual design of the data mart

The diagram in Figure 2.1 shows the initial conceptual design of the Sales data mart, as resulting from the requirement analysis.

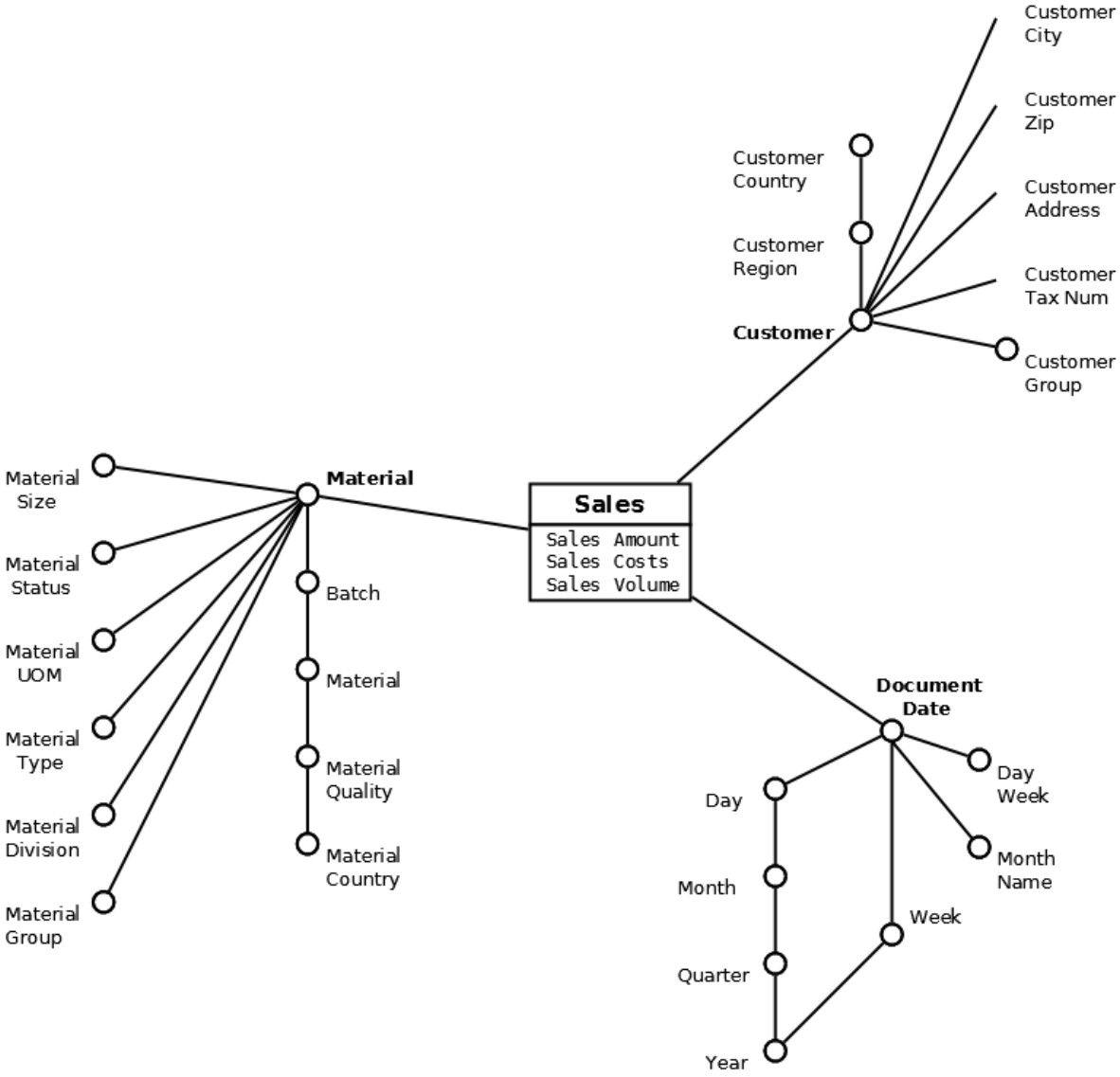


Figure 2.1: Sales initial conceptual design

2.3 Stocks process

In this section we describe the requirements specification and the initial conceptual design of the Stocks data mart. A fact is about the stock levels at the end of the month of material batches from a certain vendor, in a certain storage location.

2.3.1 Requirements specification

From the collection of business requirements the following requirements about the Stocks fact are derived. Each analysis requirements is associated with the relative dimensions and measures.

Stock process

N	Requirements analysis	Dimensions	Measures
6	The total stock amount and stock volume by month, by storage location, by material quality and by vendor.	Document Date (Month), Storage Location, Material (Quality), Vendor	Stock Amount, Stock Volume

We describe the Stocks fact, its granularity and the preliminary dimensions and measures involved.

Stocks Fact

Description	Preliminary Dimensions	Preliminary Measures
A fact is the information on the monthly values of the stocks by material batch, storage location, and vendor.	Document Date, Material, Storage Location, Vendor	Stock Amount, Stock Volume

We describe the Stocks fact dimensions and their granularity. Stocks are stored with the granularity of a month, but the granularity for the date dimension is the day. The solution adopted was to associate every stock fact with the last day of the month to which the stock was referred.

Dimensions

Name	Description	Granularity
Document Date	The date the stock is referred to	A month
Material	The material batch hierarchy	A material batch
Storage Location	The storage location in which batches are stocked	A storage location
Vendor	The vendor who provided the batches	A vendor

Below we describe, for each dimension, its attributes and, in certain cases, their sample values. The description of dimensions already presented for other facts and which have not changed is omitted.

R.E.D. Graniti group consists of several companies, which are juridically autonomous entities, each one drawing up an annual balance sheet. Every company may have one or more plants, which are facilities where the productive process take place and stocks are managed. The storage location is the

finest level of the dimension of the same name and refers to a specific plant, though it is not necessarily situated in the same physical area of the plant (e.g. a storage location in Rijeka could be related to a plant in Massa). It represent the place where batches are actually stocked and where every material movement occurs. Three different dimensional hierarchies are defined for the mentioned dimension, given the fact that a storage location can be in a different country than its plant and that a plant can hypothetically be in a different country than its company.

Storage Location

Attribute	**Description**
Storage Location	The storage location for material batches.
Storage Location Country	The country of the storage location.
Plant	The plant to whom the storage location belongs.
Plant Country	The country of the plant.
Company	The company to whom the plant belongs.
Company Country	The country of the company.

Another dimension is defined for vendors, with attributes and hierarchies very similar to the ones defined for the customer dimension.

Vendor

Attribute	**Description**
Vendor	The vendor.
Vendor Country	The country in which the vendor is based.
Vendor Region	The region of a country in which the vendor is based.
Vendor Address	The address of the customer.
Vendor Zip	The zip code of the vendor.
Vendor City	The city of the vendor.
Vendor Tax Num	The tax number of the vendor.

We describe the dimensional hierarchies, with their relative attributes and the hierarchy type. Hierarchies already presented for other facts are not repeated.

Dimensional Hierarchies

Dimension	**Description**	**Type**
Storage Location	Company Country → Company → Storage Location	Balanced
Storage Location	Plant Country → Plant → Storage Location	Balanced
Storage Location	Storage Location Country → Storage Location	Balanced
Vendor	Vendor Country → Vendor Region → Vendor	Balanced

Below we describe the Stocks fact measures, their aggregability and their formula in case of calculated measures.

Measures

Measure	**Description**	**Aggregability**	**Calculated**
Stock Amount	Value of the batches	Additive	No
Stock Volume	Volume in M^3 or FT^3	Additive	No

2.3.2 Initial conceptual design of the data mart

The diagram in Figure 2.2 shows the initial conceptual design of the Stocks data mart, as resulting from the requirement analysis.

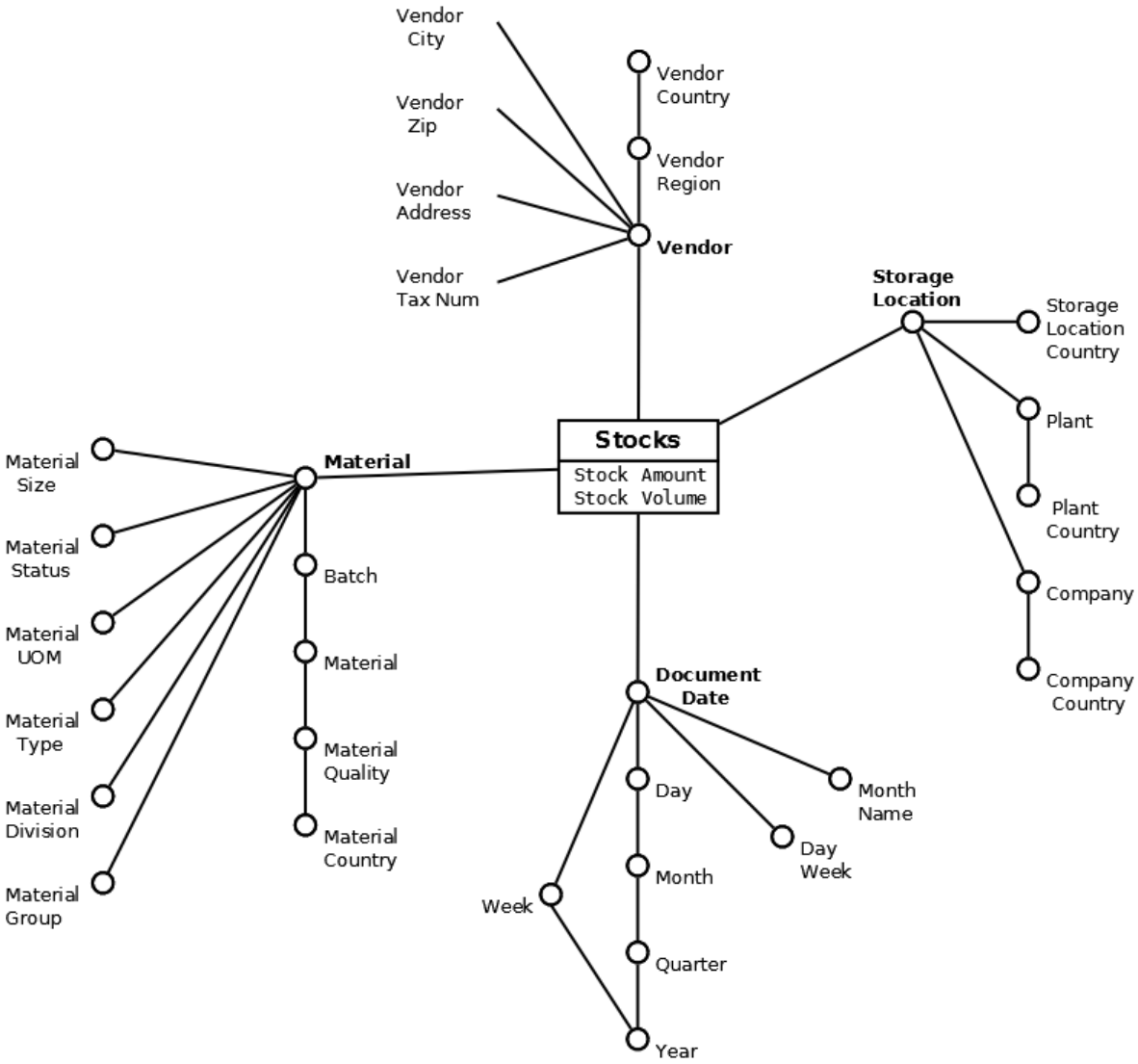


Figure 2.2: Stocks initial conceptual design

2.4 Costs process

In this section we describe the requirements specification and the initial conceptual design of the Costs data mart. A fact is about a cost paid in relation to a certain material batch, as specified in a line of a cost document.

2.4.1 Requirements specification

From the collection of business requirements the following requirements about the Costs fact are derived. Each analysis requirements is associated with the relative dimensions and measures.

Costs process

N	Requirements analysis	Dimensions	Measures
7	The total cost and unit cost by material batch, by cost type and by vendor.	Material (Batch), Cost (Cost Type), Vendor	Cost Amount, Cost Unit Amount

We describe the Costs fact, its granularity and the preliminary dimensions and measures involved.

Costs Fact

Description	Preliminary Dimensions	Preliminary Measures
A fact is about a cost paid in relation to a batch.	Document Date, Material, Vendor, Cost	Cost Amount, Cost Unit Amount

We describe the Costs fact dimensions and their granularity.

Dimensions

Name	Description	Granularity
Document Date	The date the cost was paid	A day
Material	The material batch hierarchy	A material batch
Vendor	The vendor which provided the service	A vendor
Cost	The type of cost which was paid	A cost

Below we describe, for each dimension, its attributes and, in certain cases, their sample values. The description of dimensions already presented for the other facts and which have not changed is omitted.

Every batch handled by R.E.D. Graniti is associated with a set of costs. The costs allocated to batches include costs incurred for extracting or purchasing, transporting and selling them. Two levels of costs are considered, therefore a dimensional hierarchy among them is defined.

Cost

Attribute	Description
Cost	The cost (e.g. *Loading on vessel*).
Cost Type	The cost category (e.g. *Freight Costs*).

We describe the dimensional hierarchies, with their relative attributes and the hierarchy type. Again we omit dimensional hierarchies already presented and which have not changed.

Dimensional Hierarchies

Dimension	Description	Type
Cost	Cost Type → Cost	Balanced

We describe the Costs fact measures, their aggregability and their formula in case of calculated measures.

Measures

Measure	Description	Aggregability	Calculated
Cost Amount	Amount paid	Additive	No
Cost Unit Amount	Amount paid per M^3	Non Additive	No

2.4.2 Initial conceptual design of the data mart

The diagram in Figure 2.3 shows the initial conceptual design of the Costs data mart, as resulting from the requirement analysis.

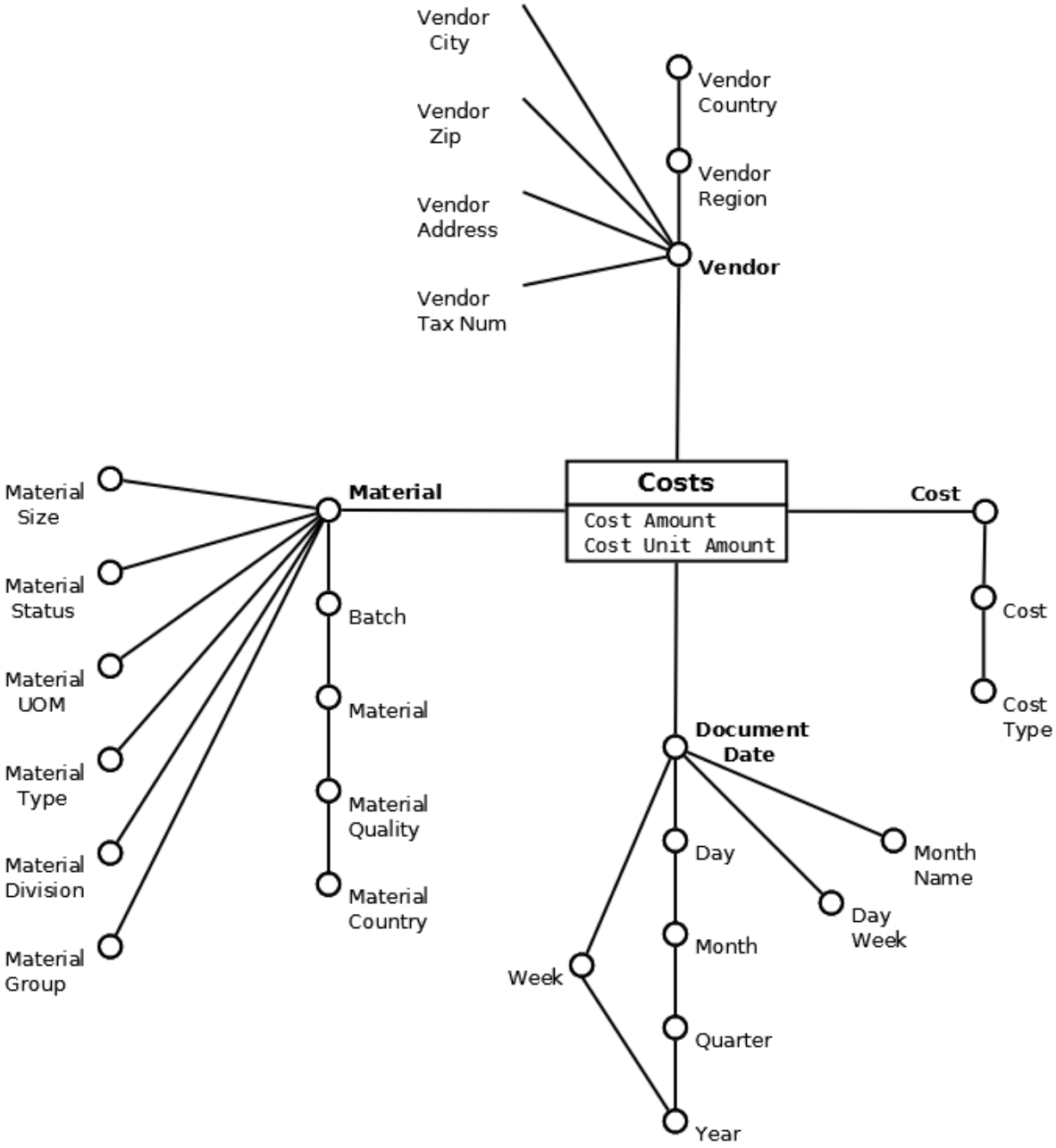


Figure 2.3: Costs initial conceptual design

2.5 Summary tables of dimensions and measures

The following table summaries the dimensions and measures discovered in the previous analysis, to show which ones are shared among two or more facts.

	Fact Dimensions		
Dimension	**Sales**	**Stocks**	**Costs**
Document Date	X	X	X
Material	X	X	X
Customer	X		
Storage Location	X	X	
Vendor	X	X	X
Cost			X

Fact Measures

Dimension	**Sales**	**Stocks**	**Costs**
Sales Amount	X		
Sales Costs	X		
Sales Margin	X		
Sales Margin %	X		
Sales Volume	X		
Sales Amount LY	X		
Sales Costs LY	X		
Sales Margin LY	X		
Sales Margin % LY	X		
Sales Volume LY	X		
Var% Sales Amount vs LY	X		
Var% Sales Costs vs LY	X		
Var% Sales Margin vs LY	X		
Var% Sales Volume vs LY	X		
Sales Amount YTD	X		
Sales Costs YTD	X		
Sales Margin YTD	X		
Sales Margin % YTD	X		
Sales Volume YTD	X		
Sales Amount YTD LY	X		
Sales Costs YTD LY	X		
Sales Margin YTD LY	X		
Sales Margin % YTD LY	X		
Sales Volume YTD LY	X		
Var% Sales Amount YTD vs LY	X		
Var% Sales Costs YTD vs LY	X		
Var% Sales Margin YTD vs LY	X		
Var% Sales Volume YTD vs LY	X		
Stock Amount		X	
Stock Volume		X	
Cost Amount			X
Cost Unit Amount			X

Chapter 3

Final conceptual design and logical design of the data marts

In this chapter we describe the operational database which will be the source systems for the data warehouse, with the main tables and attributes used. Then the final conceptual design is presented, which expands the initial conceptual design with dimension and measures considered useful after the analysis of the source systems. Finally, logical design of the single data marts, followed by the final logical design of the entire data warehouse are described.

3.1 The source systems

Up until the start of the project described in the present document, different companies of R.E.D. Graniti's group used different information systems to collect, create and process data. A few examples are AS/400 which was used in Italy and ACCPAC which was the choice for South African companies. The project approved jointly by R.E.D. Graniti and Deloitte includes the redefinition of business processes and implementation of SAP ERP Central Component 6 (SAP ECC 6). Besides the present data managed by SAP, a part of the historical data will also be loaded into the data warehouse. It was decided that the only historical data to be loaded will be the data stored in Italy's AS/400 system starting from year 2001.

The new integrated system will be gradually implemented in all the group's companies over a span of several months according to a precise schedule, progressively replacing the single management softwares. Given that Italian companies will not be the first to go live with the new system, AS/400 will continue to be used by R.E.D. Graniti S.P.A. in Italy while other companies (e.g. Southern Granites in Zimbabwe) may have already migrated to SAP. During this period the data warehouse will be fed by both source system. After a few months SAP will also go live for Italy, then the use of AS/400

will be discontinued and data from the old system will no more be extracted or loaded into the data warehouse.

3.1.1 AS/400

AS/400 is part of the initial information system used in Italy and it was already described in section 1.5. Every night a series of files is generated from AS/400 data and transferred to the business intelligence server via File Transfer Protocol. The following table lists the main files which are used for extracting data:

File	Description	Dimension
Agent00f	Agents	Agent
Blocc001	Blocks	Batch
Client00f	Customers	Customer
Costi001	Cost types	Costs
Eudetven	Statistical file	Sales / Material
Forni00f	Vendors	Vendor
Movim001	Warehouse handling	Stocks
Quali00f	Qualities	Material Quality
Quanz00f	Qualities and countries	Material Quality
Tabcs00f	Warehouse costs	Costs
Tabna00f	Countries	Country

3.1.2 SAP

SAP is an Enterprise Resource Planning software produced by the German multinational corporation SAP AG. ERP softwares are suites of integrated applications used to manage business data, have an integrated view of business processes and facilitate information flows.

R.E.D. Graniti SAP's architecture will consist of various interconnected modules which communicate to build a complex information system. The selected modules are not the entire SAP offer but cover the group needs in terms of business process and requirements. The SAP modules in scope are:

- FI (Financial Accounting)
- CO (Controlling)
- AA (Asset Accounting)
- SD (Sales and Distribution)
- MM (Materials Management)

- PP (Production Planning)
- PS (Project System)
- QM (Quality Management)
- PM (Plant Maintenance)

The following table lists the main SAP tables which will be used for extracting data:

Table	**Description**	**Dimension**
KNA1	Customer Master (General Section)	Customer
KNB1	Customer Master (Company Code)	Customer
LFA1	Vendor Master (General Section)	Vendor
LFB1	Vendor Master (Company Code)	Vendor
MARA	Material Master (General section)	Material
MAKT	Material Descriptions	Material
TWEWT	Descriptions for External Material Groups	Material Quality
T134T	Material Type Descriptions	Material Type
T023T	Material Group Descriptions	Material Group
T141T	Descriptions of Material Status from MM/PP View	Material Status
MCHA	Material Master: Batches	Batch
TSPA	Organizational Unit: Sales Divisions	Division
TCURC	Currency Codes	Currency
TCURT	Currency Code Names	Currency
T006A	Assign Internal to Language-Dependent Unit	Unit of Measure
TWKO	Organizational Unit: Sales Organizations	Sales Organization
T001	Company	Company
T001W	Plants / Branches	Plant
T001L	Plant / Storage location	Storage Location
T005	Countries	Country
T005T	Country Names	Country
T005S	Taxes: Region	Region
VBRK	Billing Document: Header Data	Sales
VBRP	Billing Document: Item Data	Sales
MCHB	Batch Stocks	Stocks

3.2 Final conceptual design

Some dimensions and measures were not included in the initial conceptual design because not mentioned in the customer's requirements. The main reason behind this is the much higher informative potential at disposal using SAP instead of the old systems, which increases the number of useful dimensions and measures. Another reason is that some dimension is needed to fulfill other requirements made by R.E.D. Graniti's management, such as the localization requirement.

3.2.1 Sales process

The final description of the Sales fact is the following. Storage Location and Vendor dimensions, already described for different facts, were added. Completely new dimensions added in this phase are Sales Organization, Agent, Currency and Unit of Measure. New measures include Sales Units, Sales Weight and Sales Depreciation.

Sales Fact

Description	**Dimensions**	**Measures**
A fact is about a batch sold.	Document Date, Material, Customer, Storage Location, Vendor, Sales Organization, Agent, Currency, Unit of Measure	Sales Amount, Sales Costs, Sales Volume, Sales Units, Sales Weight, Sales Depreciation

We describe the Sales fact dimensions and their granularity.

Dimensions

Name	Description	Granularity
Document Date	The date the invoice is issued	A day
Material	The material batch hierarchy	A material batch
Customer	The customer to which the document is addressed	A customer
Storage Location	The storage location from which the goods were shipped	A storage location
Vendor	The vendor which provided the batch sold	A vendor
Sales Organization	The sales organization used for the transaction	A sales organization
Agent	The sales agent who get a commission on the sale	An agent
Currency	The currency used to settle the invoice	A currency
Unit of Measure	The unit used for quantities used in the document	A unit of measure

Below we describe only the new dimensions with their attributes and sample values.

Every company may have one or more sales organizations, which are operating units responsible for the sales process of goods and services to customers.

Sales Organization

Attribute	Description
Sales Organization	The sales organization.

R.E.D. Graniti takes advantage of sales agents, which are subjects who act on behalf of the company to procure sales. Agents may have a regular contract with the group or may work under an independent contractor agreement. Every agent then receives a commission for the service rendered which can be fixed or proportional to the sales volume procured.

Agent

Attribute	Description
Agent	The sales agent.

The currency dimension is essential because of the localization requirements specified in section 1.5.3.

Currency

Attribute	Description
Currency	The currency code in ISO 4217 format.

A unit of measure dimension is needed since companies in the United States do not use the metric system (e.g. feet are used instead of meters) and because, among other things, granites prices are based on their volume, while weight is the main quantity indicator used for marbles.

Unit of Measure

Attribute	Description
UOM	The unit of measure code.

We describe the dimensional hierarchies, with their relative attributes and the hierarchy type.

Dimensional Hierarchies

Dimension	Description	Type
Document Date	Year → Quarter → Month → Day	Balanced
Document Date	Year → Week → Day	Balanced
Material	Material Country → Material Quality → Material → Batch	Balanced
Customer	Customer Country → Customer Region → Customer	Balanced
Customer	Customer Group → Customer	Balanced
Storage Location	Company Country → Company → Storage Location	Balanced
Storage Location	Plant Country → Plant → Storage Location	Balanced
Storage Location	Storage Location Country → Storage Location	Balanced
Vendor	Vendor Country → Vendor Region → Vendor	Balanced
Sales Organization	Sales Organization Company → Sales Organization	Balanced

In the following page we describe the Sales fact measures, their aggregability and their formula in case of calculated measures.

Measures

Measure	**Description**	**Aggregability**	**Calculated**
Sales Amount (Sa)	Amount received	Additive	No
Sales Costs (Sc)	Total costs incurred	Additive	No
Sales Margin (Sm)	$Sa - Sc$	Additive	Yes
Sales Margin %	$\frac{Sm}{Sa}$	Additive	Yes
Sales Volume (Sv)	Volume in M^3 or FT^3	Additive	No
Sales Amount LY(SaLy)	$AGO(Sa, Year, 1)$	Additive	Yes
Sales Costs LY (ScLy)	$AGO(Sc, Year, 1)$	Additive	Yes
Sales Margin LY (SmLy)	$AGO(Sm, Year, 1)$	Additive	Yes
Sales Margin % LY	$AGO(Sm\%, Year, 1)$	Additive	Yes
Sales Volume LY (SvLy)	$AGO(Sv, Year, 1)$	Additive	Yes
Var% Sales Amount vs LY	$\frac{Sa-SaLy}{SaLy}$	Non additive	Yes
Var% Sales Costs vs LY	$\frac{Sc-ScLy}{ScLy}$	Non additive	Yes
Var% Sales Margin vs LY	$\frac{Sm-SmLy}{SmLy}$	Non additive	Yes
Var% Sales Volume vs LY	$\frac{Sv-SvLy}{SvLy}$	Non additive	Yes
Sales Amount YTD (SaYtd)	$TODATE(Sa, Year)$	Semi additive (Date)	Yes
Sales Costs YTD (ScYtd)	$TODATE(Sc, Year)$	Semi additive (Date)	Yes
Sales Margin YTD (SmYtd)	$TODATE(Sm, Year)$	Semi additive (Date)	Yes
Sales Margin % YTD	$TODATE(Sm\%, Year)$	Semi additive (Date)	Yes
Sales Volume YTD (SvYtd)	$TODATE(Sv, Year)$	Semi additive (Date)	Yes
Sales Amount YTD LY (SaYtdLy)	$TODATE(SaLy, Year)$	Semi additive (Date)	Yes
Sales Costs YTD LY (ScYtdLy)	$TODATE(ScLy, Year)$	Semi additive (Date)	Yes
Sales Margin YTD LY (SmYtdLy)	$TODATE(SmLy, Year)$	Semi additive (Date)	Yes
Sales Margin % YTD LY	$TODATE(Sm\%Ly, Year)$	Semi additive (Date)	Yes
Sales Volume YTD LY (SvYtdLy)	$TODATE(SvLy, Year)$	Semi additive (Date)	Yes
Var% Sales Amount YTD vs LY	$\frac{SaYtd-SaYtdLy}{SaYtdLy}$	Non additive	Yes
Var% Sales Costs YTD vs LY	$\frac{ScYtd-ScYtdLy}{ScYtdLy}$	Non additive	Yes
Var% Sales Margin YTD vs LY	$\frac{SmYtd-SmYtdLy}{SmYtdLy}$	Non additive	Yes
Var% Sales Volume YTD vs LY	$\frac{SvYtd-ScYtdLy}{SvYtdLy}$	Non additive	Yes
Sales Units	Number of batches sold	Additive	No
Sales Weight	Weight in Kg	Additive	No
Sales Depreciation	In case of old batches	Additive	No

The diagram in Figure 3.1 shows the final conceptual design of the Sales data mart.

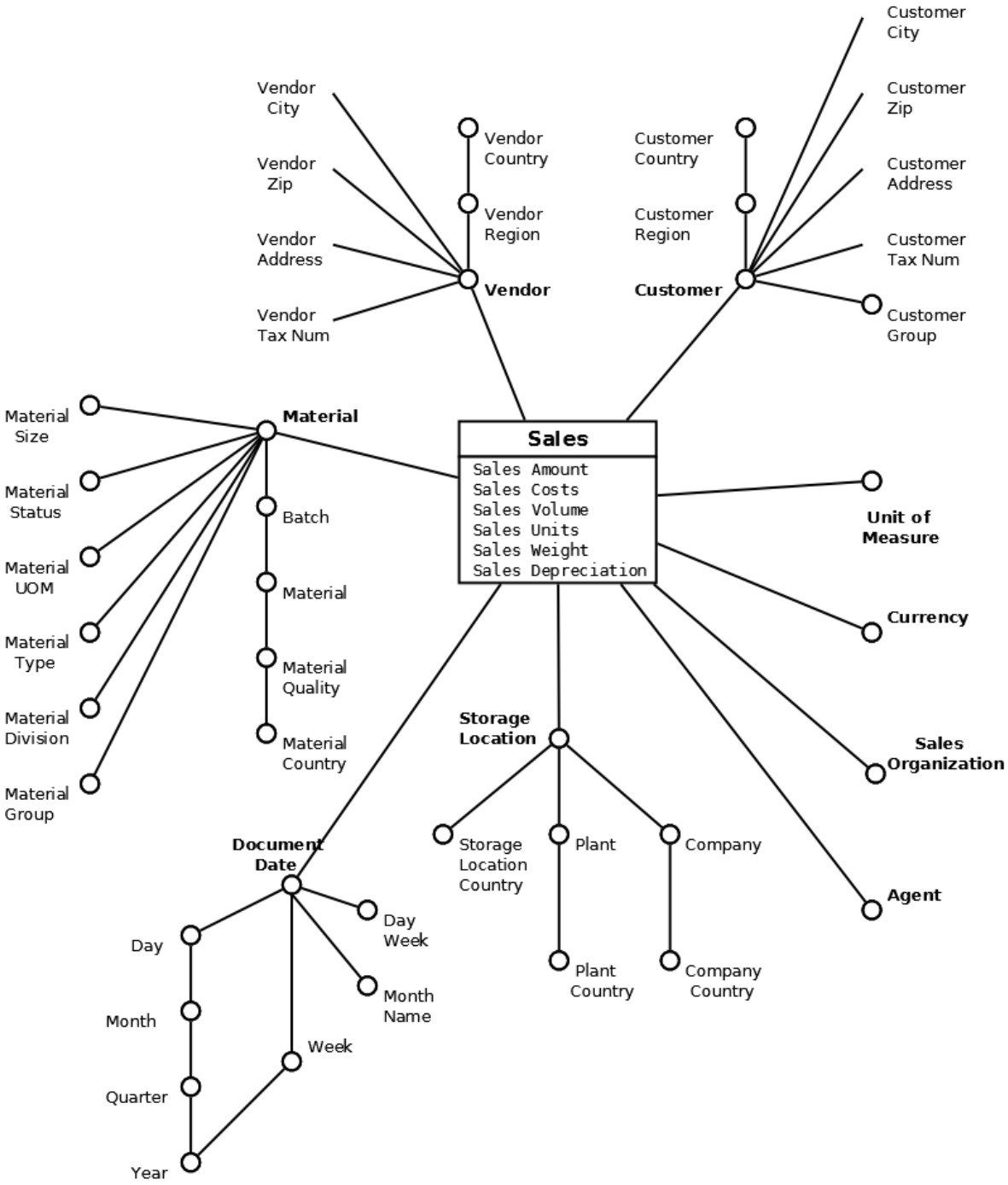


Figure 3.1: Sales final conceptual design

3.2.2 Stocks process

The final description of the Stocks fact is the following. Currency and Unit of Measure dimensions, and Stock Units, Stock Weight and Stock Depreciation measures were added.

Stocks Fact

Description	**Dimensions**	**Measures**
A fact is the information on the monthly values of the stocks by material batch, storage location, and vendor.	Document Date, Material, Storage Location, Vendor Currency, Unit of Measure	Stock Amount, Stock Volume, Stock Units, Stock Weight, Stock Depreciation

We describe the Stocks fact dimensions and their granularity. Descriptions of single dimensions are not repeated since all of the fact's dimensions have already been presented.

Dimensions

Name	**Description**	**Granularity**
Document Date	The date the stock is referred to	A month
Material	The material batch hierarchy	A material batch
Storage Location	The storage location in which batches are stocked	A storage location
Vendor	The vendor who provided the batches	A vendor
Currency	The currency used to express the stock value	A currency
Unit of Measure	The unit used to express the stock volume	A unit of measure

We describe the Stocks fact measures, their aggregability and their formula in case of calculated measures.

Measures

Measure	**Description**	**Aggregability**	**Calculated**
Stock Amount	Value of the batches	Additive	No
Stock Volume	Volume in M^3 or FT^3	Additive	No
Stock Units	Batches in stock	Additive	No
Stock Weight	Weight in Kg	Additive	No
Stock Depreciation	In case of old batches	Additive	No

The diagram in Figure 3.2 shows the final conceptual design of the Stocks data mart.

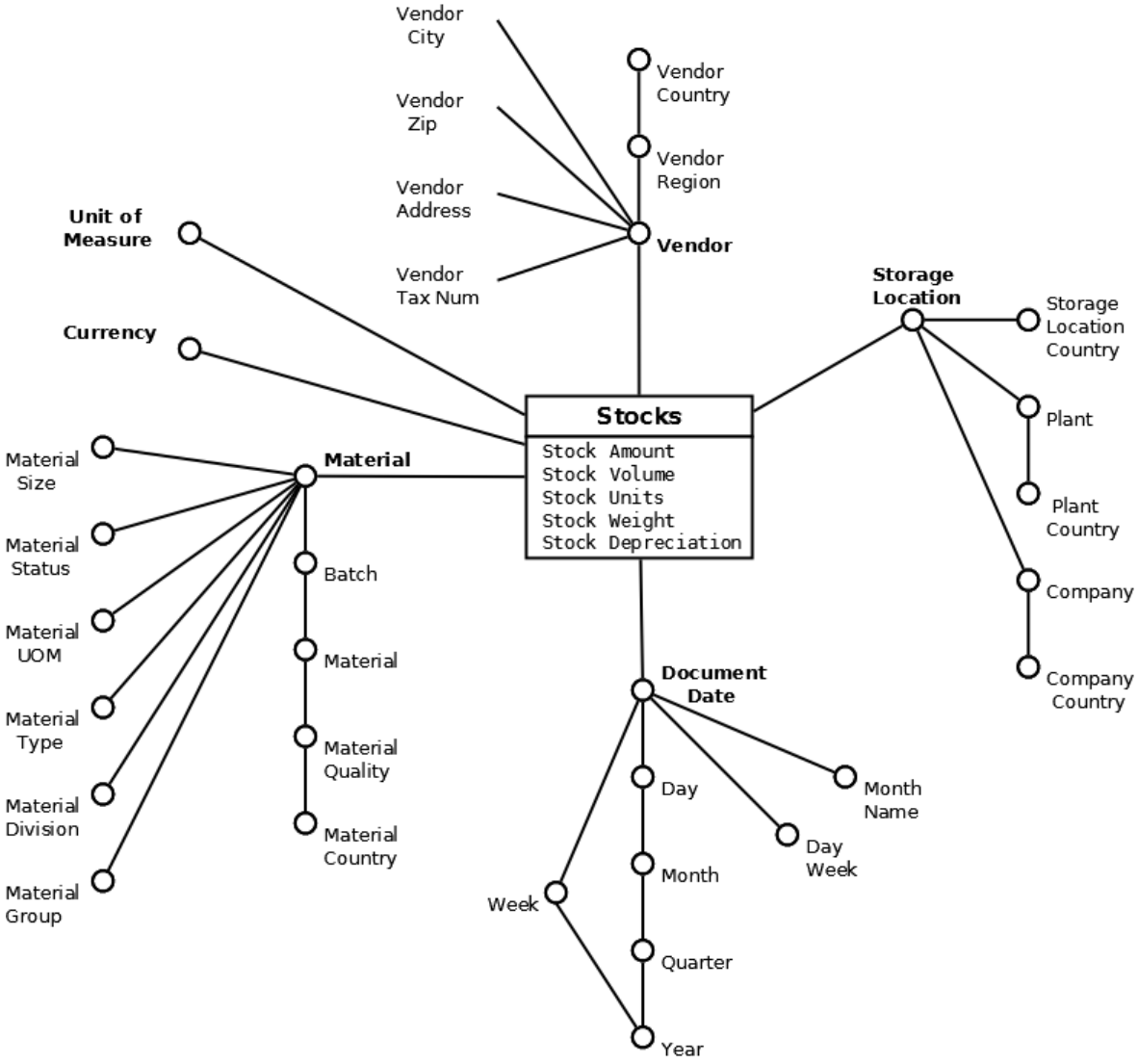


Figure 3.2: Stocks final conceptual design

3.2.3 Costs process

The final description of the Costs fact is the following. Only the Currency dimension was added.

Costs Fact

Description	**Preliminary Dimensions**	**Preliminary Measures**
A fact is about a cost paid in relation to a batch.	Document Date, Material, Vendor, Cost, Currency	Cost Amount, Cost Unit Amount

We describe the Costs fact dimensions and their granularity. Descriptions of single dimensions are not repeated since all of Costs fact's dimensions have already been presented.

Dimensions

Name	**Description**	**Granularity**
Document Date	The date the cost was paid	A day
Material	The material batch hierarchy	A material batch
Vendor	The vendor which provided the service	A vendor
Cost	The type of cost which was paid	A cost
Currency	The currency used to pay the cost	A currency

We describe the Costs fact measures, their aggregability and their formula in case of calculated measures.

Measures

Measure	**Description**	**Aggregability**	**Calculated**
Cost Amount	Amount paid	Additive	No
Cost Unit Amount	Amount paid per M^3	Non Additive	No

The diagram in Figure 3.3 shows the final conceptual design of the Costs data mart.

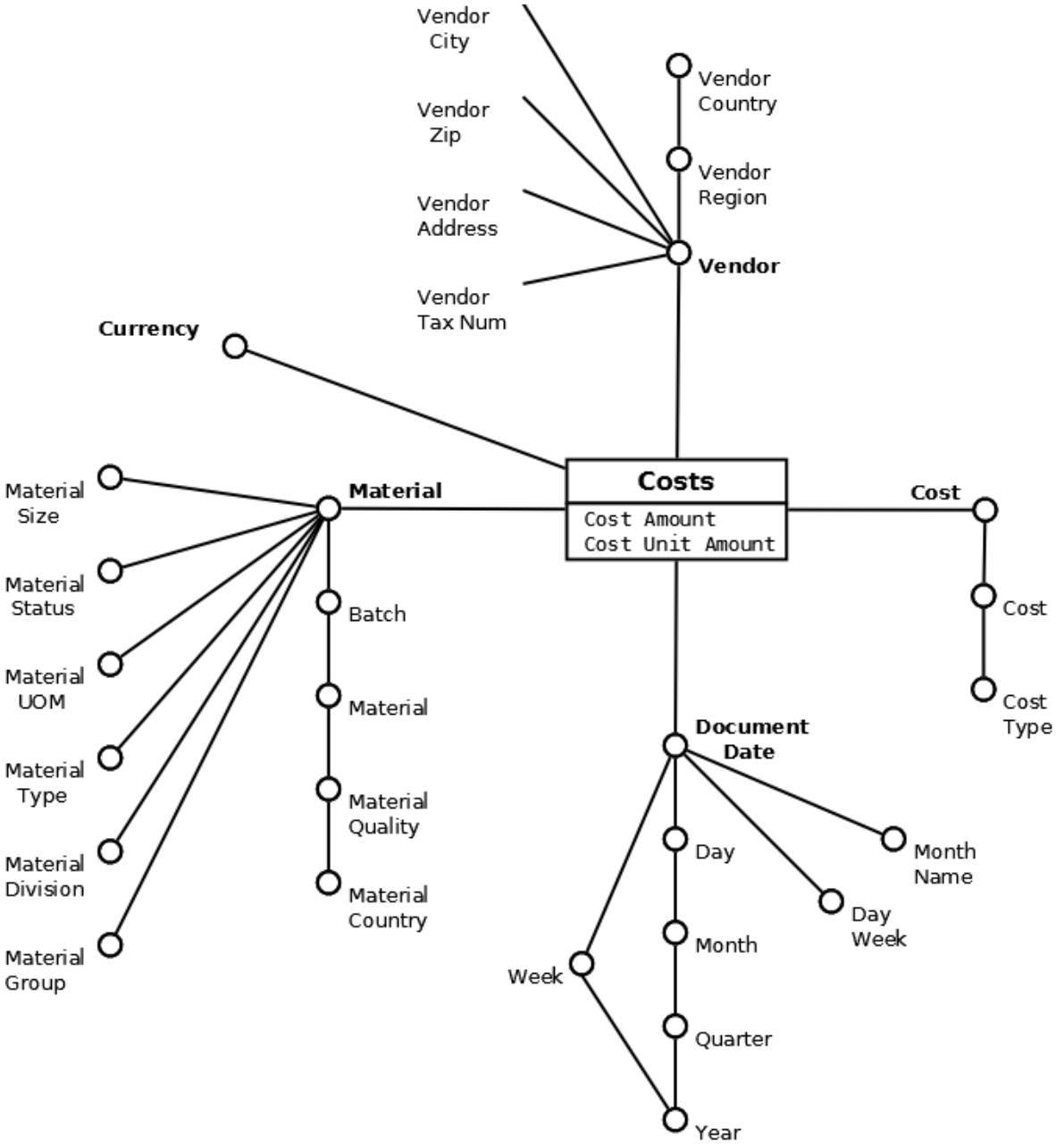


Figure 3.3: Costs final conceptual design

3.2.4 Summary tables of dimensions and measures

The following table summaries all the dimensions and measures, to show which ones are shared among two or more facts.

Fact Dimensions

Dimension	**Sales**	**Stocks**	**Costs**
Document Date	X	X	X
Material	X	X	X
Customer	X		
Storage Location	X	X	
Vendor	X	X	X
Cost			X
Sales Organization	X		
Agent	X		
Currency	X	X	X
Unit of Measure	X	X	

Fact Measures

Dimension	**Sales**	**Stocks**	**Costs**
Sales Amount	X		
Sales Costs	X		
Sales Margin	X		
Sales Margin %	X		
Sales Volume	X		
Sales Amount LY	X		
Sales Costs LY	X		
Sales Margin LY	X		
Sales Margin % LY	X		
Sales Volume LY	X		
Var% Sales Amount vs LY	X		
Var% Sales Costs vs LY	X		
Var% Sales Margin vs LY	X		
Var% Sales Volume vs LY	X		

Dimension	Sales	Stocks	Costs
Sales Amount YTD	X		
Sales Costs YTD	X		
Sales Margin YTD	X		
Sales Margin % YTD	X		
Sales Volume YTD	X		
Sales Amount YTD LY	X		
Sales Costs YTD LY	X		
Sales Margin YTD LY	X		
Sales Margin % YTD LY	X		
Sales Volume YTD LY	X		
Var% Sales Amount YTD vs LY	X		
Var% Sales Costs YTD vs LY	X		
Var% Sales Margin YTD vs LY	X		
Var% Sales Volume YTD vs LY	X		
Stock Amount		X	
Stock Volume		X	
Cost Amount			X
Cost Unit Amount			X
Sales Units	X		
Sales Weight	X		
Sales Depreciation	X		
Stock Units		X	
Stock Weight		X	
Stock Depreciation		X	

3.3 Logical design of the data marts

The logical design of the Sales data mart is shown in Figure 3.4. None of the calculated measures is present in the logical design, since values of derived measures are not stored in the data warehouse. Instead, derived logical columns were directly created in the Business Intelligence tool, as described in Chapter 6. This was possible because of the Business Intelligence tool transformation capabilities and because most of the functions to be applied are time-related. Therefore doing these transformations directly in the Business Intelligence layer was easier, given the possibility to take advantage of concepts

like dimensions, hierarchies etc.

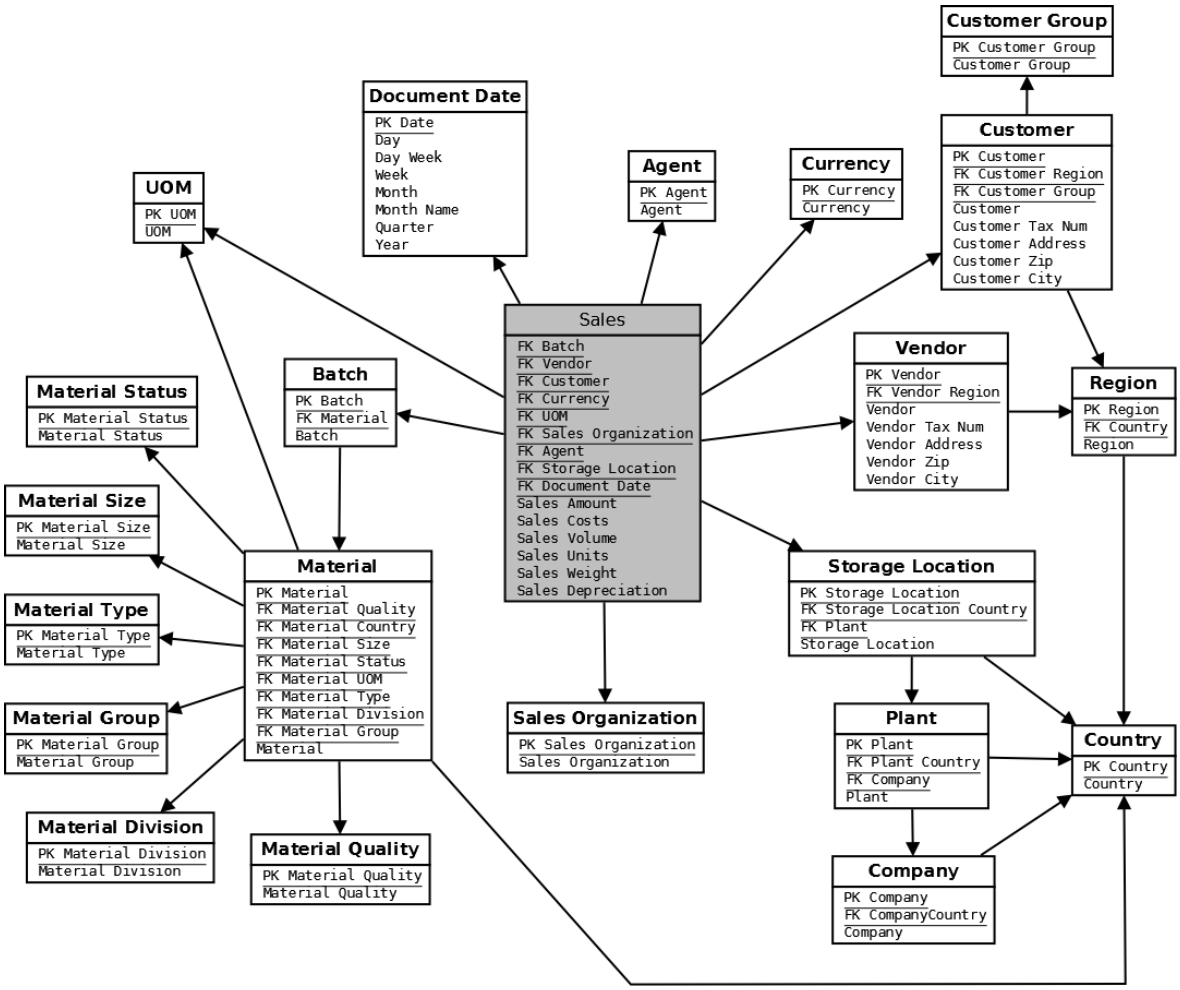


Figure 3.4: Sales logic design

The document date table's source is neither AS/400 nor SAP. Given the particular nature of the table, data is inserted just once in the table using a specific PL/SQL procedure. It was decided that the primary key for the table would not be a surrogate key, like for other tables, but the date in *YYYYMMDD* format. The *Day* attribute is the only one which is not a number or a string, but it is instead of *DATE* data type. The reason behind this choice is that a value with this data type can be presented by the business intelligence tool in different formats according to the localization preferences specified by every user. Therefore dates will be shown with *DD/MM/YYYY* format to Italian users and *MM/DD/YYYY* format to users from United States.

The logical design of the Stocks data mart is shown in Figure 3.5.

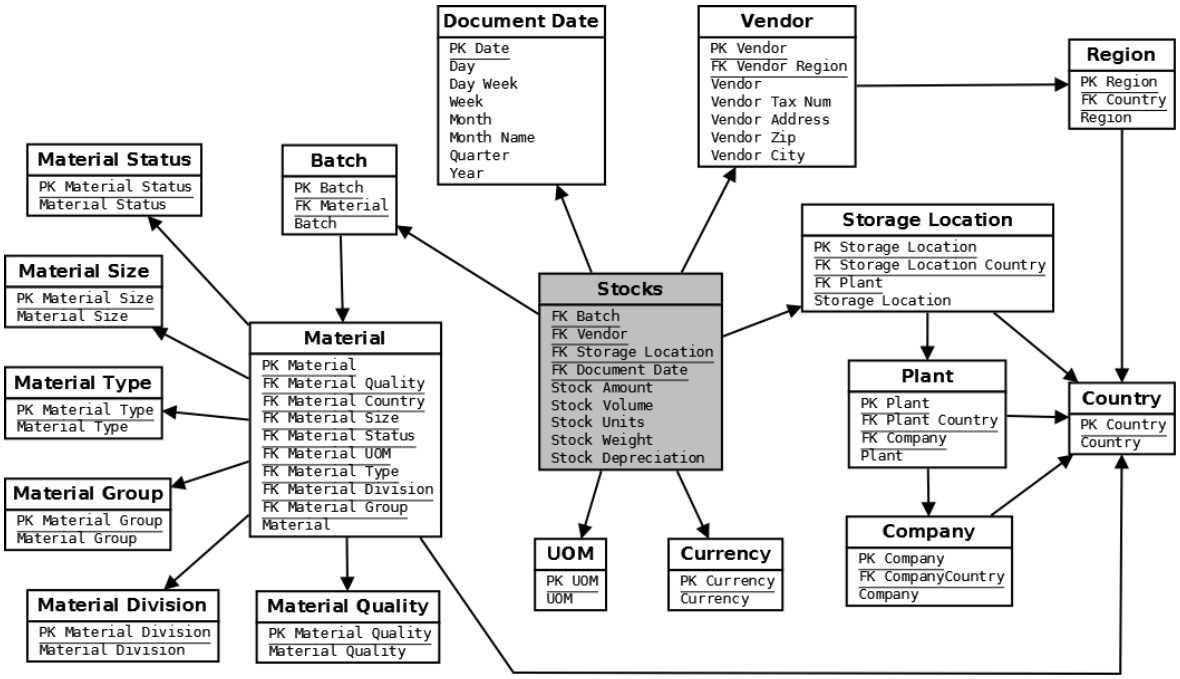


Figure 3.5: Stocks logic design

The logical design of the Costs data mart is shown in Figure 3.6.

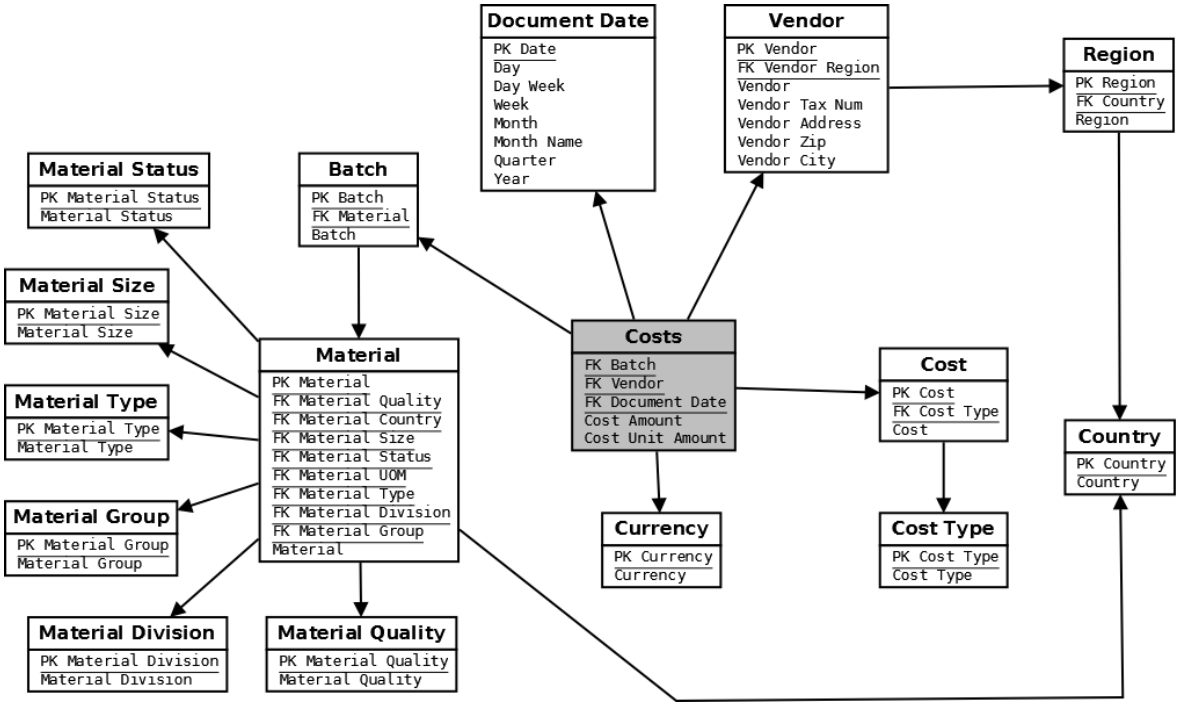


Figure 3.6: Costs logic design

3.4 Logical design of the data warehouse

The logical design of the data warehouse is shown in Figure 3.7.

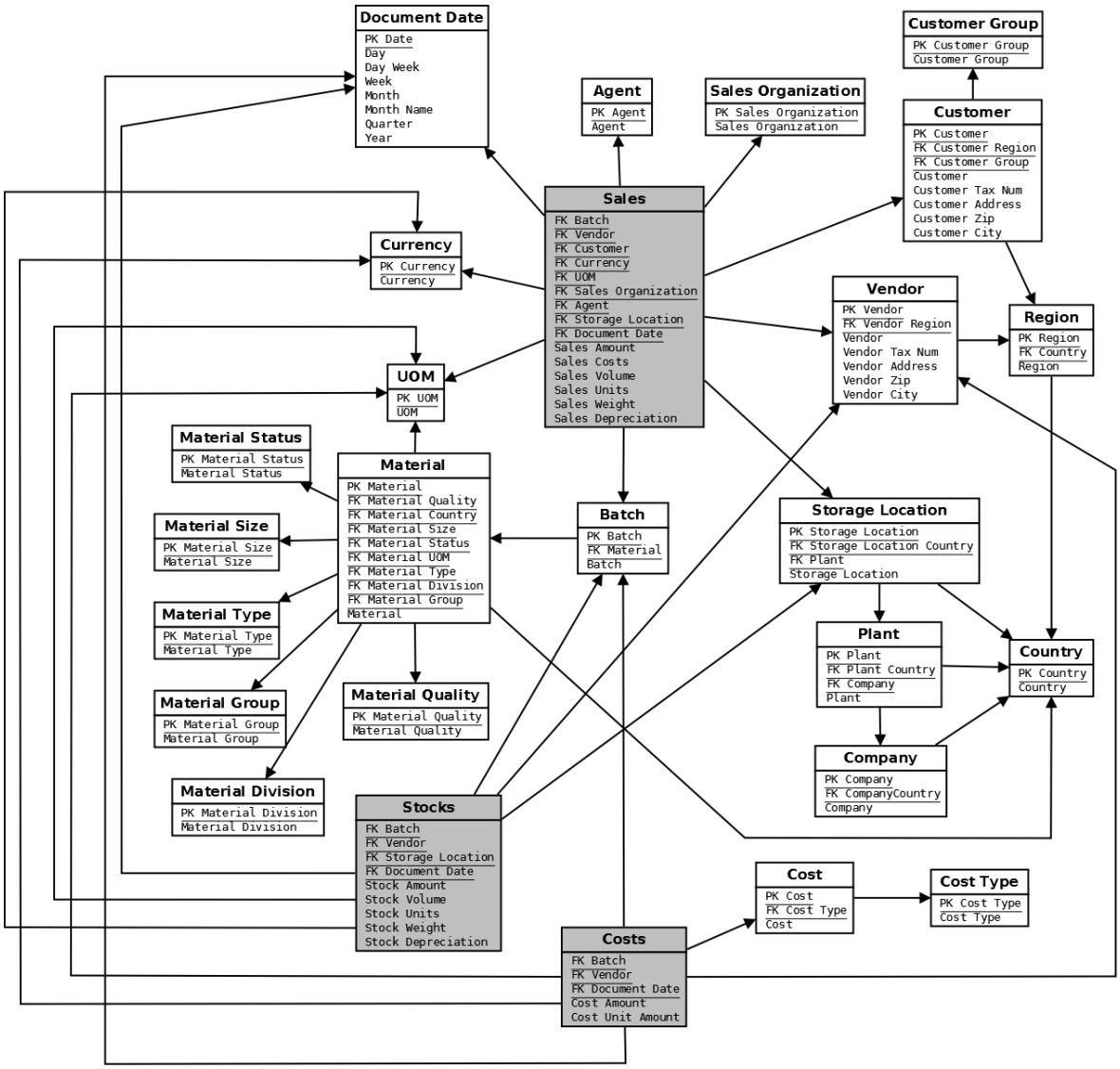


Figure 3.7: Data warehouse logic design

Chapter 4

Development environment and tools

In this chapter we describe the Data Base Management System (DBMS) and business intelligence tools used for the project, along with the implementation choices made.

4.1 Comments on the Oracle environment choice

The business intelligence market offers several valid tools to retrieve, analyze and report data. As shown in Figure 4.1 Oracle is one of the main competitors and its complete and architecturally unified BI system, offering reporting abilities, ad hoc analysis, OLAP, dashboards, and scorecards, is among the leaders in both ability to execute and completeness of vision according to Gartner's 2014 Magic Quadrant for Business Intelligence and Analytics Platforms.

Deloitte XBS is a longtime Oracle diamond partner and it has a successful history using Oracle DB Server and Oracle BI together to provide accurate and consistent insight to its customers to assist them with decision making and planning. Furthermore Oracle Business Intelligence is suitable for R.E.D. Graniti's requirements on internationalization and localization explained in section 1.5.3.

Figure 4.1: Gartner's 2014 BI Magic Quadrant.

4.2 Oracle Database Server 11g R2 Standard Edition One

The Oracle Database is an object-relational database management system produced by Oracle Corporation. The system include one or more instances of the application and data storage. An instance includes memory structures and operating system processes.

Data is logically stored in tablespaces and physically stored in datafiles. Tablespaces contain memory segments and these segments comprise one or more extents, which are formed of contiguous data blocks. Also partitioning is possible, dividing the database or only parts of it into distinct partitions, to increase performance.

Each instance uses a shared-memory area called System Global Area (SGA) to store data and other information in RAM, and a memory-area called Program Global Area (SGA) to store information for server's processes.

Some of the other features are concurrency management, a scheduler subsystem and the ability to execute PL/SQL functions and stored procedures.

4.3 Oracle SQL Developer and Data Modeler

Oracle SQL Developer is an integrated development environment giving database developers a simple way to connect to a target database schema and manage database objects, run SQL statements and scripts, develop and run PL/SQL code, export data and many other basic tasks. SQL Developer was used to manage the Oracle Database, run queries, develop and schedule Extract, Transform and Load procedures, while Data Modeler was used for database design.

Oracle Data Modeler is a graphical data modeling tool for capturing and modeling metadata. A database design consists of one logical model, one or more relational models based on the logical model and one or more physical models based on each relational model. The logical model offers an implementation-independent view of information and allows the creation of entities, logical views, attributes etc. The relational model describes the database database in terms of SQL tables, columns, and joins between tables. The physical model describes the database in terms of Oracle Database objects. Data Modeler can generate Data Definition Language (DDL) scripts for the database and also import metadata using direct connections, guaranteeing a perfect synchronization between the data model and the database.

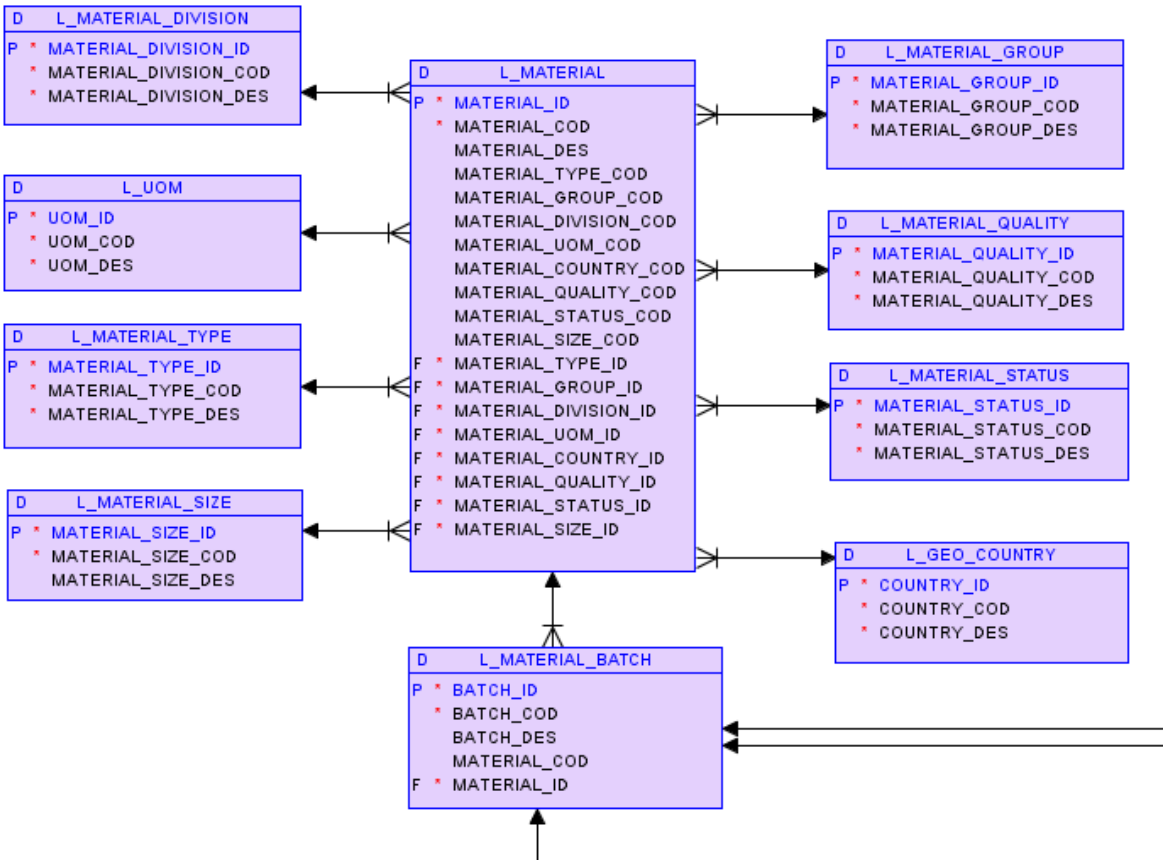


Figure 4.2: A small portion of the Data Modeler database design.

4.4 Oracle Business Intelligence Standard Edition One

Oracle Business Intelligence is the central component of the Oracle Business Intelligence Foundation Suite, an architecturally integrated business intelligence foundation, in which metadata for source systems, business logic, and presentation access is centrally managed in a Common Enterprise Information. Because all data sources, metrics and calculations are managed in a centralized model, the consistency and accuracy of information is guaranteed.

The Oracle BI metadata repository provides an abstraction layer between the business intelligence application seen by the users and the complex underlying data sources. The Oracle BI Server processes user Logical SQL queries, interprets them and then transforms them in the corresponding physical queries against the appropriate data sources. In this case a unique data source, the Oracle Database, is necessary, and is accessed through the native Oracle Call Interface (OCI) APIs. Details on the construction of the metadata repository are given in Chapter 6.

Oracle Business Intelligence provide ad hoc query and analysis, interactive dashboards, scorecards, multidimensional OLAP, and all popular forms of reporting, analysis and data visualizations. Its business-oriented visualization capabilities include a wide range of interactive chart types, maps, and tabular formats, from which information can be accessed, navigated, and analyzed.

Chapter 5

Extract, transform and load procedures

In this chapter we describe the ETL (Extract, Transform, Load) process, the PL/SQL procedures used in this stage, the problems identified during the implementation of these procedures and the choices and solutions adopted to solve them.

5.1 The ETL process

The extract, transform and load is the process of extracting data from one or more operational systems and load it into the data warehouse. Between the extraction and the loading, various transformations may occur, the most common being filtering and cleaning data, deriving calculated values, joining and integrating data from multiple sources, applying business rules and generating surrogate-key values.

The ETL process assumed a significant role in the project presented here. The integration and consolidation of data from different applications (i.e. AS/400 and SAP) was one of the most challenging aspects, therefore the transform phase was quite complex compared to the extract and load phases.

A selection of the columns to be loaded in the data warehouse was made together with the customer; other transformations included translating coded values (e.g. 1 or 0 for flag values) and joining data from multiple sources using lookups. Deriving calculated values was postponed and done directly with the business intelligence tool.

The ETL procedures implementation followed, and also influenced, the conceptual and logical design of the data warehouse and was done using PL/SQL procedures.

5.2 Reasons for not using an ETL tool

In the preliminary stage of the project the use of Oracle Warehouse Builder for the ETL process was taken into consideration. The main advantages of using an ETL tool are the reliability and ease of use, at the cost of lesser flexibility. The choice made was to manually implement PL/SQL procedures for various reasons: the little effort required by the transformation stage, the versatility possible for the data integration and the availability of reusable PL/SQL procedures implemented in previous similar projects.

5.3 Staging Area

A staging area is used as an intermediate storage area between the data sources and the data warehouse for data processing during the ETL process. It is a temporary area, implemented in the form of database tables, and it is erased before every ETL process run. The primary function of the staging area in this project is to consolidate and integrate data from both the source systems.

5.4 Naming convention

Every object in the data warehouse must follow a rigid naming convention. Names are uppercase and are separated by the underscore character.

Every table's name in the staging area begins with the prefix *T_*, dimension tables' names begin with the prefix *L_* and fact tables' names begin with *F_*. External tables, used for reading data from source text files, have names beginning with the prefix *X_*, other auxiliary tables used for the transform phase have names beginning with *Z_*.

For every dimension attribute, like *Material Size* in the table of the same name, two different attributes are actually created in the data warehouse: one for the code and one for the description. A third attribute is used for the primary key. Surrogate keys' names end with the suffix *_ID*, attributes corresponding to codes in the source systems end with the suffix *_COD*, description attributes end with *_DES*. Attributes of type *DATE* end with the suffix *_DATE*. Attributes on which a foreign key is set have the same name of the referenced candidate key.

Sequence names are composed of the prefix *SEQ_* followed by the name of the relative table. Index names are composed of the table name followed by the suffix *_PK* for indexes on the primary key or *_AK* for unique indexes on other attributes. Foreign keys' names end with the suffix *_FK* followed by a sequential number.

5.5 Packages structure

PL/SQL procedures are split in two packages, *LOAD_SA* for the loading of the staging area and *LOAD_DW* for the loading of the data warehouse. Another package, *LOAD_SA_DW*, consists of the procedure for the sequential run of the staging area loading procedures, followed by the data warehouse loading procedures.

The package *LOAD_SA* includes one procedure for each staging area table and the package *LOAD_DW* include one procedure for each dimension table and for each fact table. Every procedure's name begins with the prefix *LOAD_* followed by the name of the relative table.

An Oracle Job was created and scheduled to run every morning at 06:00. The job action is set to run the procedure in the *LOAD_SA_DW* package, which perform a complete load of the data warehouse and which needs about ten minutes to complete.

5.6 Extract phase

Data from both source systems is read from comma separated value files. Oracle external tables are defined for each of these files, allowing to read data as if they were tables in the database. A log file and a bad file are used for each table to store information about each *SELECT* statement ran on the table and about the records which were discarded because of some error. External tables are read in the *LOAD_SA* package procedures. An example of a SQL create statement for an external table is the following:

```
CREATE TABLE "DWPROD"."X_COMPANY" (
        "BUKRS" VARCHAR2(4 CHAR),
        "BUTXT" VARCHAR2(25 CHAR),
        "LAND1" VARCHAR2(3 CHAR),
        "WAERS" VARCHAR2(5 CHAR)
)
ORGANIZATION EXTERNAL (
        TYPE ORACLE_LOADER
        DEFAULT DIRECTORY "EXT_DATA"
        ACCESS PARAMETERS (
                RECORDS DELIMITED BY NEWLINE
                CHARACTERSET WE8MSWIN1252
                STRING SIZES ARE IN CHARACTERS
                LOGFILE EXT_LOG   :'X_COMPANY.LOG'
                BADFILE EXT_LOG   :'X_COMPANY.BAD'
                FIELDS TERMINATED BY ';'
                MISSING FIELD VALUES ARE NULL
        )
        LOCATION ("EXT_DATA":'COMPANY.csv')
```

```
)
REJECT LIMIT UNLIMITED ;
```

A full extraction of data from AS/400 is performed at each run. This is possible because of the small data volumes and because AS/400 will serve as a source system only for a limited period of time. A full extraction of data from SAP is done only for dimension tables. Incremental extraction is used for SAP fact tables, that is only records created in the current or previous year are extracted each time. Data is extracted from both source systems every day during the night.

5.7 Transform phase

In the *LOAD_SA* package all the dimension tables procedures follow the same structure. The table is truncated, then records from SAP external tables are read and inserted in the staging area and finally records from AS/400 external tables are inserted, but only in case these records were not already present in SAP source files. This check is possible because of the *OCOD* attribute, with the attribute's name being the abbreviation for old code. The pseudo-code of one of these procedures, *LOAD_T_CUSTOMER*, is now presented as an example.

```
TRUNCATE TABLE T_CUSTOMER;

-- INSERT SAP
INSERT INTO T_CUSTOMER
SELECT ...
FROM X_CUSTOMER;

COMMIT;

-- UPDATE COD
UPDATE L_CUSTOMER TARGET
SET CUSTOMER_COD = (
        SELECT NVL(T.CUSTOMER_COD, L.CUSTOMER_COD)
        FROM L_CUSTOMER L
    LEFT JOIN T_CUSTOMER T ON L.CUSTOMER_COD = T.CUSTOMER_OCOD
    WHERE TARGET.CUSTOMER_ID = L.CUSTOMER_ID
);

COMMIT;

-- INSERT AS/400
INSERT INTO T_CUSTOMER
SELECT ...
FROM X_AS_CUSTOMER
```

```
WHERE CUSTOMER_COD NOT IN (
        SELECT CUSTOMER_OCOD
        FROM T_CUSTOMER
);

COMMIT;
```

When records are inserted in the SAP system an additional field with the value of the old code, the code that record had on AS/400 system, must be filled. After inserting records from SAP source files in the staging area, an UPDATE statement updates the value of the *COD* attributes in the data warehouse dimension table. The records updated will be the ones just read from SAP but only present in AS/400 until that moment. To clarify this step suppose a customer whose name is *Marble S.p.A.* existed in AS/400 and hence it is already stored in L_CUSTOMER with *CUSTOMER_COD* equal to *ABCDE*. Its *CUSTOMER_OCOD* will be equal to the code since there is no need of a reference to an older code for records loaded from AS/400. Then *Marble S.p.A.* is inserted in the SAP database, with the field containing the old code filled with *ABCDE*. Suppose SAP assigns to the newly created record the code *00001*. The next time the procedure *LOAD_T_CUSTOMER* will run the record for *Marble S.p.A.* with *CUSTOMER_COD* equal to *00001* and *CUSTOMER_OCOD* equal to *ABCDE* will be inserted in the staging area. The UPDATE statement mentioned before will update the *CUSTOMER_COD* for *Marble S.p.A.* in *L_CUSTOMER* data warehouse table from *ABCDE* to *00001*, with the *CUSTOMER_OCOD* value remaining *00001*.

The last phase consists in loading into the staging area records from AS/400 source files, but only in case that same record has not already been loaded from SAP. This mechanism guarantees that only one record for each customer, like in the case of *Marble S.p.A.*, will be stored in the data warehouse. If that records exists in SAP the values of its attributes will be the ones stored in SAP, whether or not that record existed in AS/400. Clearly if that record do not exists in SAP but do exists in AS/400 the values of its attributes will be the ones stored in AS/400.

Other transformations which not involve integration from different data sources are implemented in the *SELECT* clauses of the *INSERT* statements above. Values of string attributes are always trimmed and some concatenation between strings is made. In some case a *DECODE* operation is needed, for example for currencies where *EURO* and *LIRA*, values used for Euro and Italian Lira in AS/400, have to be transformed to *EUR* and *ITL*, since it was decided to use ISO 4217 format for currency codes in the data warehouse.

Flag attributes are denoted in the AS/400 source files using the character *X*, therefore in these cases every *X* is transformed to 1 and every empty string is transformed to 0. Every code of a foreign key attribute is transformed using the *NVL* function, assigning a default value (e.g. *000*) in case the foreign key is *NULL*.

In case one of the auxiliary tables is needed, a *LEFT JOIN* is appended to the *FROM* statement. This is the case of the table *Z_CONVERSIONS*, which is needed to convert the two digit country code used in AS/400 to the more informative and standard ISO alphanumeric country codes. The table *Z_CONVERSIONS* contains two columns, one for the old code and one for the new code. The *ON* statement of the *LEFT JOIN* compare the code contained in the external table with the old codes in the auxiliary table, then the new code retrieved from the *JOIN* with the auxiliary table is used.

Procedures for fact tables are very similar but there is no need of updating or filtering on codes, since there is no overlapping between AS/400 and SAP facts.

5.8 Load phase

The load phase is performed by the *LOAD_DW* package, where the only data manipulation done is the assignment of ID to foreign key attributes. Procedures regarding dimension tables and procedures regarding fact tables follow a different structure. The pseudo-code for dimension tables is the following:

```
MERGE INTO L_CUSTOMER l
USING (
        SELECT ...
        FROM T_CUSTOMER c
        LEFT JOIN L_REGION r ON (c.REGION_COD = r.REGION_COD)
) t
ON (l.CUSTOMER_COD = t.CUSTOMER_COD)

WHEN NOT MATCHED THEN
        INSERT VALUES (
                ...
        )

WHEN MATCHED THEN
        UPDATE SET ...
        WHERE
                l.CUSTOMER_DES <> t.CUSTOMER_DES
                OR ...
;

COMMIT;
```

A *MERGE* operation is performed on each data warehouse table. Records from the staging area are selected and their codes are compared with the existing codes in the data warehouse table. In case the code does not match any of the codes already present, the record is inserted in the data warehouse. In case there is a matching, meaning the record is already present in the data warehouse,

an *UPDATE* operation is performed, but only if one of the attributes values has changed.

The assignment of ID values is done with the *SELECT* statement in the *USING* clause of the *MERGE* operation. The staging area referencing table is joined with the data warehouse referenced table using the code attribute (e.g. REGION_COD for the relationship between the customer and its region), then the ID value from the lookup table, *REGION_ID* in this case, is selected and then used in the *INSERT* or in the *UPDATE* operations.

In case the *LEFT JOIN* returns a NULL value, meaning the referenced code is not found in the lookup table, the ID value is set to 0. In fact, every dimension table in the data warehouse has a default record, created with a standalone procedure *SET_DEFAULT_VALUES*, with ID equal to *0*, codes attributes equal to string of zeros and descriptions equal to *Not Defined*. This mechanism guarantees that records with wrong references do not cause errors but instead are assigned to a special record for that dimension clearly distinguishable when analyzing reports and dashboards.

The loading of fact tables is slightly different, the pseudo-code of the *LOAD_F_COST_BATCH* is:

```
SELECT NVL(MIN(CREATE_DT),20991231) INTO v_Min_Date
FROM T_COST_BATCH
WHERE SOURCE = 'SAP';

-- DELETE SAP
DELETE FROM F_COST_BATCH
WHERE
        CREATE_DT >= v_Min_Date
        AND SOURCE = 'SAP';

-- INSERT SAP
INSERT INTO F_COST_BATCH
SELECT ...
FROM T_COST_BATCH t
LEFT JOIN L_VENDOR v ON (v.VENDOR_COD = t.VENDOR_COD)
...
WHERE SOURCE = 'SAP';

COMMIT;

-- DELETE AS
DELETE FROM F_COST_BATCH
WHERE SOURCE = 'AS';

-- INSERT AS

INSERT INTO F_COST_BATCH
SELECT ...
```

```
FROM T_COST_BATCH t
LEFT JOIN L_VENDOR v ON (v.VENDOR_OCOD = t.VENDOR_COD)
...
WHERE SOURCE = 'AS';

COMMIT;
```

Two *DELETE* and *INSERT* operations are performed instead of a *MERGE* for each fact table. The first *SELECT* stores in the variable *v_Min_Date* the minimum creation date of records extracted from SAP. This would normally be the first of January of the previous year, but this way the procedure supports also changes in the time span selected for the incremental extraction from SAP. Records created before this date will not be deleted or changed.

Then next step is to delete from the data warehouse records created after this date and originating from SAP and reload them by inserting SAP records in the staging area's fact table. For records from AS/400 the steps are similar, except for the fact that all records are deleted and reinserted.

The assignment of ID values is done with the *SELECT* statement, similarly to the assignment of ID values for dimension tables, only with a subtle but yet fundamental difference. The staging area referencing fact table is joined with the data warehouse referenced dimension table using the code attribute (e.g. VENDOR_COD for the relationship between the cost and its vendor). SAP facts contain references to the actual codes of dimension objects, so the *COD* value of the fact is joined with the *COD* value of the dimension. Instead since AS/400 facts contain references to the old codes of dimension objects, the *COD* value of the fact is joined with the *OCOD* value of the dimension.

The ID value from the lookup table, *VENDOR_ID* in this case, is then used in the *SELECT* clause. Again, in case the *LEFT JOIN* returns a NULL value, meaning the referenced code is not found in the lookup table, the ID value is set to 0, assigning the fact to the default record for that dimension.

5.9 Slowly Changing Dimensions

In data warehousing projects it is necessary to choose the strategy to deal with dimension attributes which can change over time. The most common methodologies to managed slowly changing dimensions are called *Type 1*, *Type 2* and *Type 3*.

Type 1 consists of overwriting the history. For every dimensional attribute only the most recent value is stored and the previous values are not preserved. This is the simplest solution but it is not suitable when the history of values is important. *Type 2* instead preserves the history. The structure of the dimension does not need to change, both old and new values are stored in the data warehouse in multiple records. Each of these record has a different surrogate key but all of them will have the same natural key. Less common is the *Type 3* strategy, where one or more version of history is preserved.

Additional columns are used to store a limited part of history, with another column storing the date of the change between values.

R.E.D. Graniti's choice was the *Type 1* methodology, meaning the history is not preserved and attribute values are overwritten. This strategy was chosen because only a few dimension attributes are supposed to change over time (e.g. customer addresses, plant names etc.) and none need its history to be preserved.

5.10 Data warehouse refresh frequency

At the present time the data warehouse data is refreshed daily, but there is the possibility to schedule the refresh procedures twice a day or more frequently in the future.

Every night at 05:00 the source data files are scheduled to be transferred from both source systems to the data warehouse server and a backup of the entire data warehouse is stored. At 06:00 the procedure in the package *LOAD_SA_DW* is scheduled to run, loading the staging area tables and then refreshing the data warehouse.

Chapter 6

Business Intelligence metadata

In this chapter we describe the application used for reporting, Oracle Business Intelligence, and the design of the metadata used by this software.

The Oracle BI metadata repository provides an abstraction layer between the business intelligence application seen by the users and the complex underlying data sources. The Oracle BI Server processes user Logical SQL queries, interprets them and then transforms them in the corresponding physical queries against the appropriate data sources. In this case a unique data source, the Oracle Database, is necessary, and is accessed through the native Oracle Call Interface (OCI) APIs.

The tool used to create and manage the Oracle BI repository is called Oracle BI Administration. As shown in Figure 6.1, this tool connect to the repository through the Oracle BI Server, but can also directly connect to the repository in offline mode. When opened in offline mode a repository can be modified only while it is not loaded into the Oracle BI Server. If it is opened in offline mode while it is loaded into the Oracle BI Server, the repository opens in read-only mode.

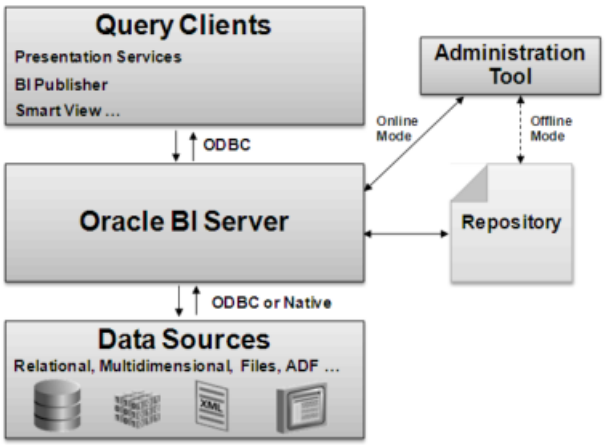


Figure 6.1: Oracle BI Server Architecture

A repository has three layers: physical layer, business model and presentation layer.

6.1 Physical layer

This layer defines the physical data sources and allow the import of tables, cubes and flat files from them. The separation between physical model and logical model improve the dimension conformance and the isolation from the physical sources.

Physical tables and joins in the Physical layer are automatically created while importing metadata from data sources. Having only the Oracle Database as data source, the creation of the physical layer was very straightforward and required a very modest amount of time. The resulting model is shown in Figure 6.2, while in Figure 6.3 we show the view where joins between tables are displayed.

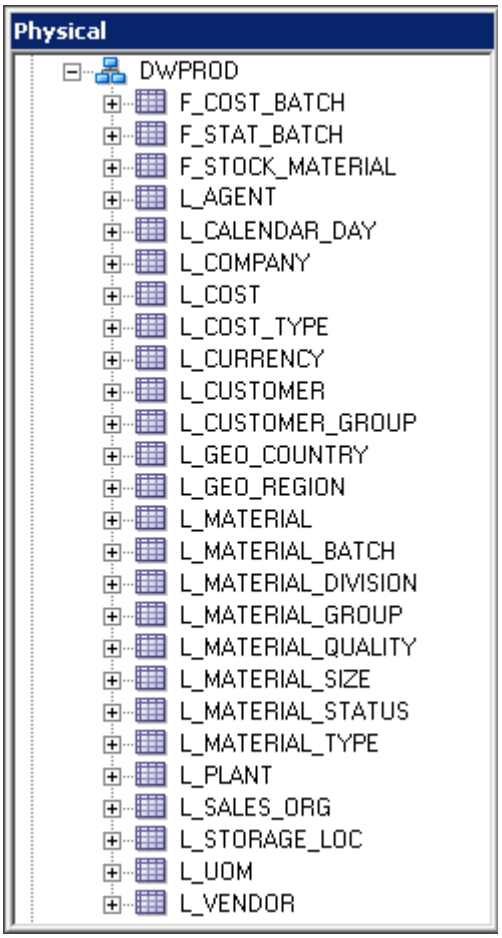


Figure 6.2: Physical model layer of the repository

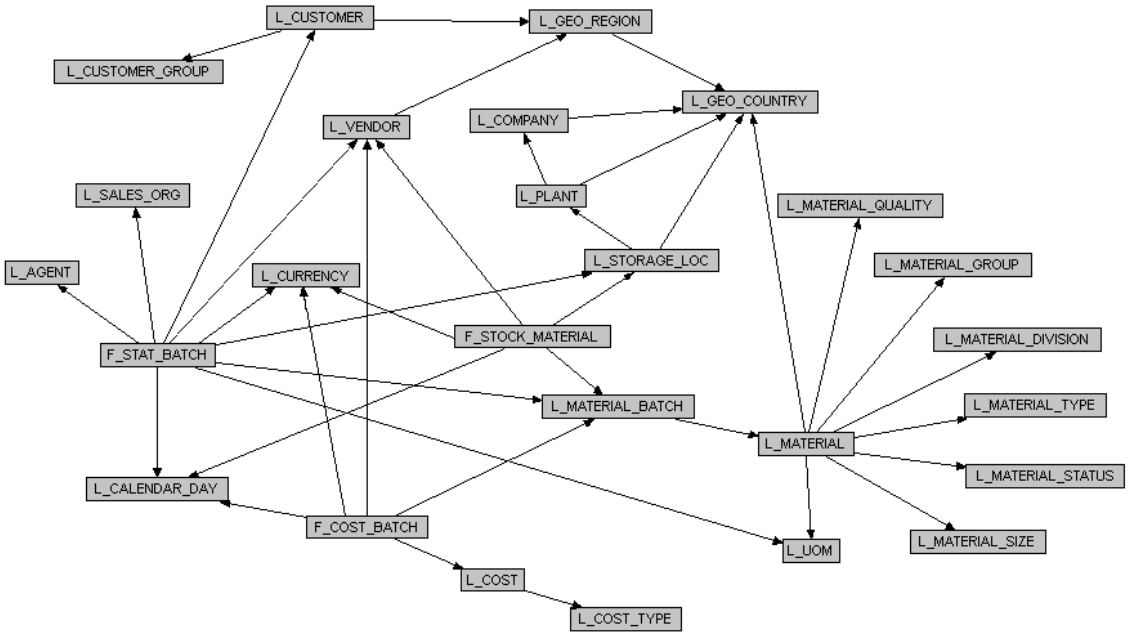


Figure 6.3: Joins view of the physical model

6.2 Business Model layer

This layer defines the business model of the data, hiding the complexity of the source data models. Mapping between the business model and the physical layer are defined, meaning each logical object can map to one or more columns in the physical layer. When the Oracle BI Server evaluates at run time Logical SQL queries issued by the Presentation layer, it determines, using these mappings, the best set of tables and files for generating the appropriates physical queries.

One or more business models can be defined for each repository, containing mappings from logical to physical tables. Logical tables are created by dragging and dropping physical tables from the physical layer to one business model. Logical columns and logical joins are automatically created and can be subsequently reviewed and modified. Figure 6.4 in the next page shows a logical table as example.

The customer physical table was dragged and dropped to the business model layer and renamed to *Dim - Customer*. Attributes, which were automatically created, are renamed using a business terminology agreed with R.E.D. Graniti.

Customer Group (Cod) and Customer Group (Des) are attribute dragged from the Customer Group physical table, meaning that when a logical query contains one of them, a physical join with the source table *L_CUSTOMER_GROUP* will be made using the logical foreign key Customer Group (Id). The same applies to Customer Region (Cod) and Customer Region (Des). *L_GEO_REGION* is also the source logical folder for Customer Country (Id) attribute, while Customer Country (Cod) and Customer Country (Des) are mapped to the *L_GEO_COUNTRY* physical table. The *Sources* folder contains all the physical tables used for the mappings of the logical table.

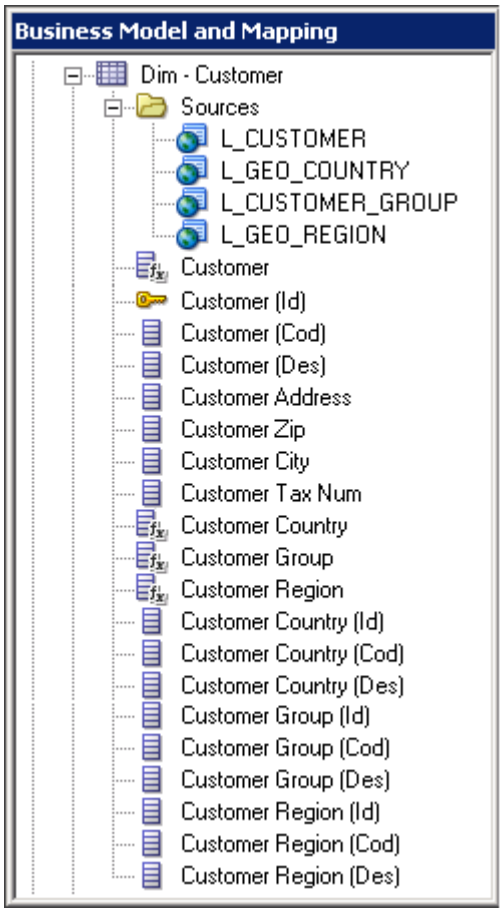


Figure 6.4: Customer logical table in the business model layer of the repository

The attributes *Customer*, *Customer Country*, *Customer Group* and *Customer Region*, represented by a different icon, are calculated logical column and were added manually. Each of them does not map to a particular physical column like the other logical columns, but is derived from existing columns using an expression. As an example the expression for *Customer* is:

```
"Dim - Customer"."Customer (Des)" || ' ' || "Dim - Customer"."Customer (Cod)"
```

This solution was adopted because using the *Customer* calculated column in the reporting allows future changes in the structure of names (e.g. use the Code followed by a dash and then the Description) and is more robust against changes in the data sources.

For dimensions like Customer, the logical table can be associated with a dimension object, where logical columns are organized into the structure of the hierarchy, a structure used by Oracle BI Server to drill into and across dimensions. Oracle BI offers three different types of logical dimensions. Dimensions with level-based hierarchies are the most common and are also called structure hierarchies. Dimensions with parent-child hierarchies, also called value hierarchies, contains members of the same type, differently from level-based hierarchies where each level contains a different type of member. Time dimensions are a special type of level-based dimension, providing additional functionalities for modeling time series data.

The level-based hierarchy created for Customer is shown in Figure 6.5. The first level is the Grand Total, a special level representing the grand total for the dimension. In this case the Grand Total level has two child levels, Customer Country and Customer Group. Each customer is associated with one and only one Customer Region, which in turn is associated with only one Customer Country. Moreover a customer belongs to one and only one Customer Group.

For each level, except the Grand Total level, a logical level key is defined as the primary key, as the yellow key icon clearly illustrates. Another level key is used as the display key, displayed when a user clicks to drill down. Here the Id attribute for the level is always used as the primary key and the calculated logical column is used as the display key.

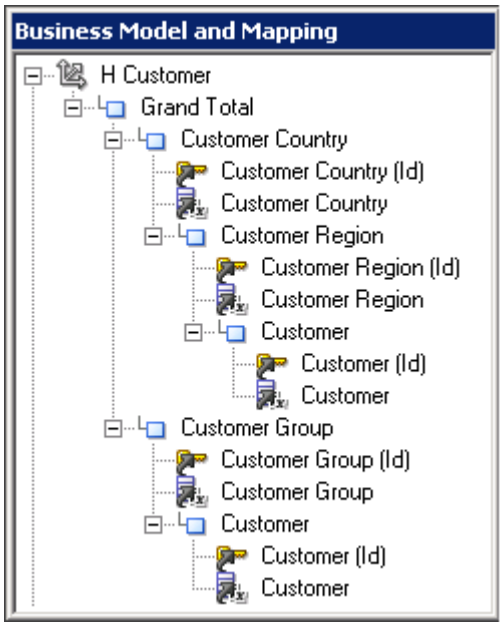


Figure 6.5: Customer hierarchy in the business model layer of the repository

A different approach is used for fact tables. The source table is only the fact table itself and for every measure an aggregation rule is set. Additive measures, as indicated in the analysis requirements specification in chapter 2, are set to use a *Sum* aggregation rule. An aggregation rule based on dimensions is used instead for semi-additive measures.

Calculated logical columns are used for creating *Sales Net Margin* as the difference between *Sales Amount* and *Sales Total Costs*, *Sales Net Margin %* as *Sales Net Margin* over *Sales Net Amount* and the time-based measures.

LY (last year) measures are created using the AGO function. As an example, *Sales Amount LY* is created using the expression:

```
AGO("Fact - Statistics Batch"."Sales Amount","H Calendar Document Date"."Year",1)
```

YTD (year to date) measures are created using the TODATE function. The expression for *Sales Amount YTD* is the following:

```
TODATE("Fact - Statistics Batch"."Sales Amount","H Calendar Document Date"."Year")
```

The other calculated logical columns are created using combinations of the mentioned functions and calculated columns (e.g. for *YTD LY* measures) or with simple mathematical operations like the subtractions and divisions used for calculating *Var %* measures.

Figure 6.6 shows the resulting logical table for the Sales fact table.

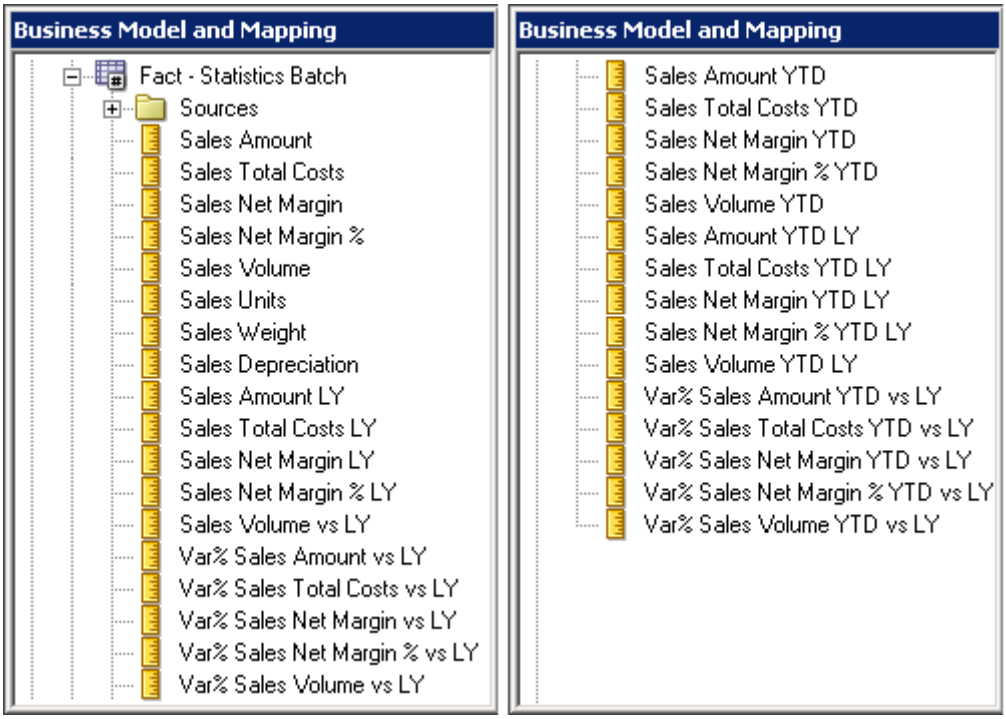


Figure 6.6: Sales fact table in the business model layer of the repository

6.3 Presentation layer

This layer defines views of business models to users. This additional layer of abstraction allow the definition of application roles and the use of customized and secure role-based views. Presentation layer views are called subject areas and are what users actually see in the analysis creation web page.

Every dimension is split into two different presentation tables, one for the attributes intended to be used for creating analyses and dashboards and one with details attributes, which are not normally used. Figure 6.7 shows the complete Presentation layer of the project and the details of the presentation tables used for the Vendor dimension.

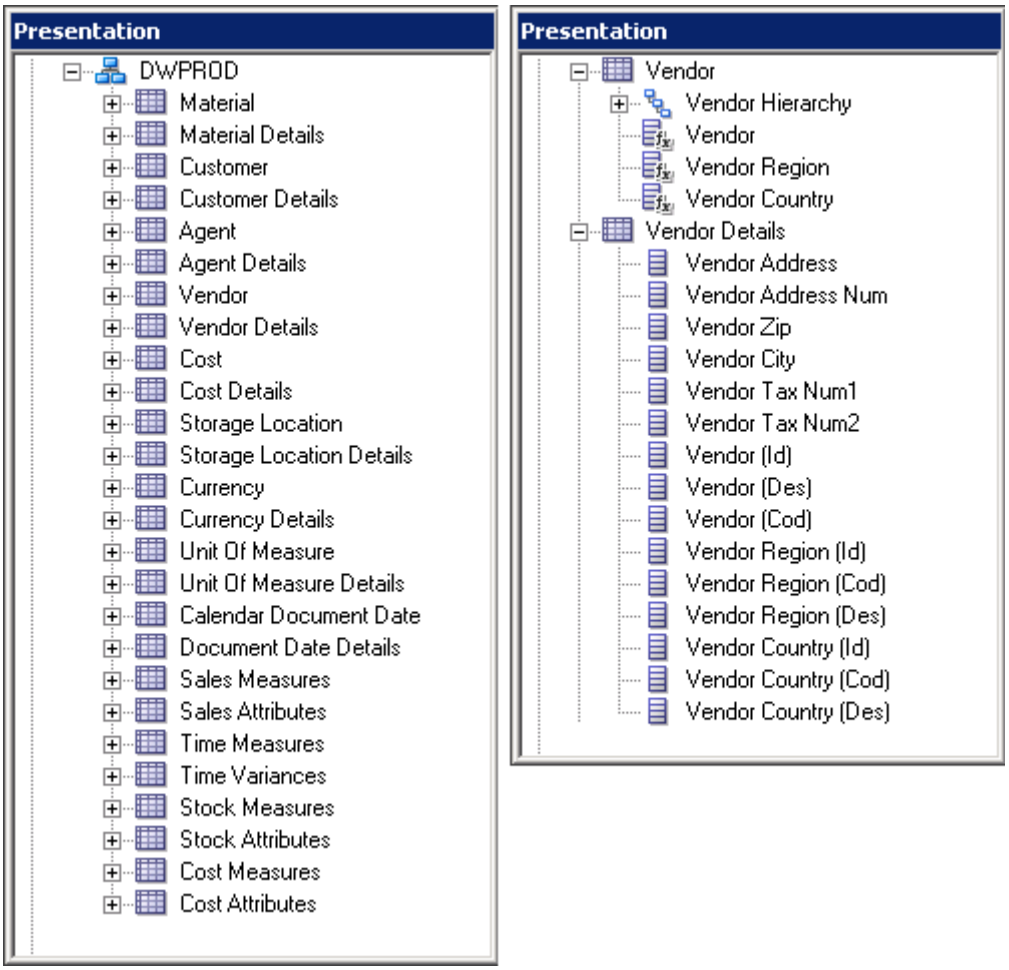


Figure 6.7: Presentation layer of the repository

Presentation tables containing details are set as children of the corresponding presentation table (e.g. *Vendor Details* is set as a child of *Vendor*). For fact tables with time-based measures two child presentation tables are defined, one called *Time Measures* for the pure time-based measures (e.g. last year, year to date and year to date LY measures) and one called *Time Variances* containing only *Var %* measures.

The final result, the subject area seen by the user, is shown in Figure 6.8.

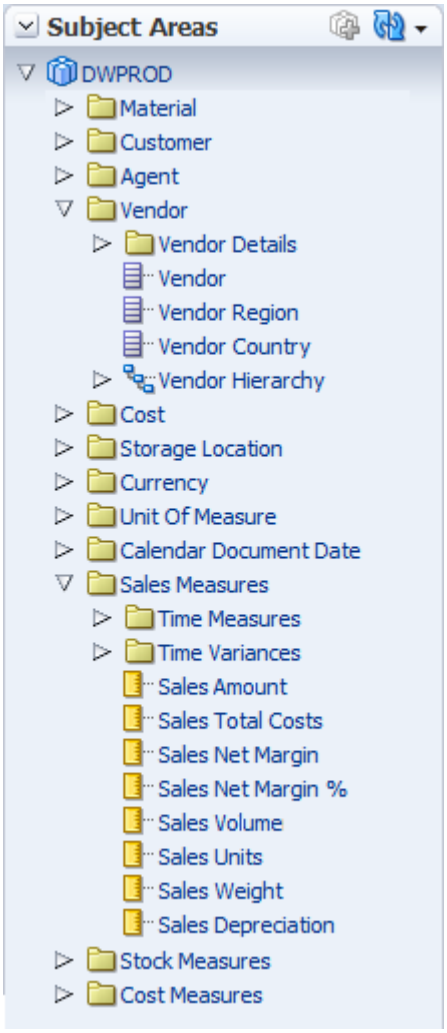


Figure 6.8: Presentation tables as seen by the users

6.4 Logical and physical queries

Oracle Business Intelligence applies several transformations to reduce a graphical report to a SQL query over the data warehouse. Starting from the Presentation layer objects present in the report, Oracle BI Presentation services issue a Logical SQL query, in terms of objects of the Business Model layer. This query is totally independent of the source physical database. Finally Oracle BI Server translates the logical query to physical SQL, in terms of objects of the Physical layer, which is sent to the appropriate source database to retrieve data.

As an example we describe the consecutive forms that the various reports embedded in the *Top 15 Customers by Sales Amount* dashboard page illustrated in the next chapter take. The first section of the page is the dashboard prompt section shown in FigureDashboardPrompt. Display attributes, like a date or a description, can have a *Descriptor ID* column, usually the corresponding primary key attribute. This column is used to uniquely identify display values, for example in *WHERE* and *GROUP BY* clauses. The logical query is given below.

```
SELECT
        DESCRIPTOR_IDOF("Calendar Date"."Date") saw_0,
        "Calendar Date"."Date" saw_1
FROM "DWPROD"
WHERE DESCRIPTOR_IDOF("Calendar Date"."Date") IN (20140101, 20141231)
ORDER BY saw_0, saw_1
```

The physical query to which the logical query is translated is the following. The generic business model *DWPROD* is substituted by the appropriate database table, *L_CALENDAR_DAY* in this case. The function *DESCRIPTOR_IDOF* takes the display attribute, in this case a date in a format dependent on the user's localization settings, and returns its Descriptor ID, that is the date primary key in *YYYYMMDD* format, which is consistent with sorting and locale-independent.

```
SELECT
        T2851.DAY_ID as c1,
        T2851.DAY_DATE as c2
FROM L_CALENDAR_DAY T2851
WHERE T2851.DAY_ID in (20140101, 20141231)
ORDER BY c1, c2
```

The logical query issued for the report in Figure 7.2 is more complex because of the *Top 15* selection step. Customers are filtered using a subquery which selects the Descriptor ID of the top customers by sales amount with the use of two functions, *RANK* and *FILTER*.

```
SELECT
        "Customer"."Customer" s_1,
        DESCRIPTOR_IDOF("Customer"."Customer") s_2,
        "Sales Measures"."Sales Net Margin %" s_3,
        "Sales Measures"."Sales Net Amt" s_4,
        "Sales Measures"."Sales Net Margin" s_5
FROM "DWPROD"
WHERE
        DESCRIPTOR_IDOF("Calendar Date"."Date") IN (20140101,20141231)
        AND DESCRIPTOR_IDOF("Customer"."Customer") IN (
                SELECT DESCRIPTOR_IDOF("Customer"."Customer")
                FROM "DWPROD"
                WHERE
                        DESCRIPTOR_IDOF("Calendar Date"."Date") IN (20140101,20141231)
                        AND RANK(FILTER(
                                "Sales Measures"."Sales Net Amt"
                                USING DESCRIPTOR_IDOF("Calendar Date"."Date") IN
                                    (20140101,20141231)
                        )) <= 15
)
ORDER BY s_4 DESC, s_1 ASC
```

Unlike other vendors, such as MicroStrategy, Oracle Business Intelligence does not use multi-pass SQL, in which data is written into temporary tables and more than one query is used to obtain the report results. Instead it uses single-pass SQL, meaning that extremely complex SQL is encapsulated in a single SQL statement, without the use of temporary tables. Therefore the corresponding physical query in this case is very difficult to read. *OBISUBWITH0* calculates the rank of customers and filters them. A difference with respect to the previous physical query example is that more than one physical table is now needed, hence the join statements in the *WHERE* clause. *SAWITH0* aggregates customers and calculates the needed simple measures with the *SUM* function. The final *SELECT* statement is used to calculate derived measures from the values previously obtained, in this case the percentage sales margin and the sales margin are computed.

```
WITH OBISUBWITH0 AS (
SELECT D1.c1 as c1
FROM (
        SELECT
                D1.c2 as c1,
                CASE WHEN D1.c1 IS NOT NULL THEN
                        RANK() OVER (ORDER BY D1.c1 DESC)
                END AS c2
        FROM (
                SELECT
                        SUM(T11623.SALES_NET_EUR) as c1,
                        T2949.CUSTOMER_ID as c2
                FROM
                        L_CUSTOMER T2949,
                        L_CALENDAR_DAY T2851,
                        F_BILL_DOC_ITEM T11623
                WHERE
                        T2851.DAY_ID = T11623.ASSIGNMENT_DT
                        AND T2949.CUSTOMER_ID = T11623.CUSTOMER_ID
                        AND T2851.DAY_ID IN (20140101,20141231)
                GROUP BY T2949.CUSTOMER_ID
         ) D1
) D1
WHERE D1.c2 <= 15
),

SAWITH0 AS (
SELECT
        SUM(T11623.SALES_COST_EUR) as c1,
        SUM(T11623.SALES_NET_EUR) as c2,
        CONCAT(CONCAT(CONCAT(T2949.CUSTOMER_DES, ' ('), T2949.CUSTOMER_COD), ')') as c3,
        T2949.CUSTOMER_ID as c4
```

```
FROM
        L_CUSTOMER T2949,
        L_CALENDAR_DAY T2851,
        F_BILL_DOC_ITEM T11623
WHERE
        T2851.DAY_ID = T11623.ASSIGNMENT_DT
        AND T2949.CUSTOMER_ID = T11623.CUSTOMER_ID
        AND T2851.DAY_ID IN (20140101,20141231)
        AND T2949.CUSTOMER_ID IN(
                SELECT DISTINCT D1.c1 as c1
                FROM OBISUBWITH0 D1
        )
GROUP BY
        T2949.CUSTOMER_ID,
        CONCAT(CONCAT(CONCAT(T2949.CUSTOMER_DES, ' ('), T2949.CUSTOMER_COD), ')')
)

SELECT DISTINCT
        D1.c3 as c2,
        D1.c4 as c3,
        ROUND((D1.c2 - D1.c1) * 100 / NULLIF( D1.c2, 0) , 2) as c4,
        D1.c2 as c5,
        D1.c2 - D1.c1 as c6
FROM SAWITH0 D1
ORDER BY c5 DESC, c2
```

The last report in the aforementioned dashboard page is even more complex, hence only sales amount measures will be included in the queries text and some parts of the physical query will be omitted. The logical query contains three additional calculations for each measures, obtained using the *REPORT_SUM* function, which is an aggregation function that the Oracle BI Server uses calculate the total.

```
SELECT
        "Customer"."Customer" s_1,
        "Material"."Material Quality" s_2,
        DESCRIPTOR_IDOF("Customer"."Customer") s_3,
        DESCRIPTOR_IDOF("Material"."Material Quality") s_4,
        "Sales Measures"."Sales Net Amt" s_6,
        "Time Measures"."Sales Net Amt LY" s_11,
        REPORT_SUM("Sales Measures"."Sales Net Amt" BY ) s_30,
        REPORT_SUM("Sales Measures"."Sales Net Amt" BY DESCRIPTOR_IDOF("Customer"."Customer
            ")) s_31,
        REPORT_SUM("Sales Measures"."Sales Net Amt" BY DESCRIPTOR_IDOF("Material"."Material
             Quality")) s_32,
```

```
        ...
        REPORT_SUM("Time Measures"."Sales Net Amt LY" BY ) s_39,
        REPORT_SUM("Time Measures"."Sales Net Amt LY" BY DESCRIPTOR_IDOF("Customer"."
            Customer")) s_40,
        REPORT_SUM("Time Measures"."Sales Net Amt LY" BY DESCRIPTOR_IDOF("Material"."
            Material Quality")) s_41,
        ...
FROM "DWPROD"
WHERE DESCRIPTOR_IDOF("Calendar Date"."Date") IN (20140101,20141231)
ORDER BY s_1 ASC, s_32 DESC, s_41 DESC, s_2 ASC
```

The physical query begins with a large number of calculations on the date table, necessary because of the time-based measures *LY*. The result of the table is shown in Table 6.1.

```
WITH OBICOMMON0 AS (
SELECT
        T2851.DAY_ID as c5,
        T2851.YEAR_ID as c6,
        ROW_NUMBER() OVER (PARTITION BY T2851.YEAR_ID ORDER BY T2851.YEAR_ID DESC) as c7,
        T2851.QUARTER_ID as c8,
        ROW_NUMBER() OVER (PARTITION BY T2851.YEAR_ID, T2851.QUARTER_ID ORDER BY T2851.
            YEAR_ID DESC, T2851.QUARTER_ID DESC) as c9,
        T2851.MONTH_ID as c10,
        ROW_NUMBER() OVER (PARTITION BY T2851.QUARTER_ID, T2851.MONTH_ID ORDER BY T2851.
            QUARTER_ID DESC, T2851.MONTH_ID DESC) as c11,
        ROW_NUMBER() OVER (PARTITION BY T2851.MONTH_ID, T2851.DAY_ID ORDER BY T2851.
            MONTH_ID DESC, T2851.DAY_ID DESC) as c12
FROM L_CALENDAR_DAY T2851
),

SAWITH0 AS (
        SELECT
                CASE WHEN
                        CASE D1.c7 WHEN 1 THEN D1.c5 ELSE NULL END
                        IS NOT NULL THEN RANK() OVER (
                                ORDER BY CASE D1.c7 WHEN 1 THEN D1.c5 ELSE NULL END ASC )
                END as c1,
                ...
```

D3					D5				
c1	**c2**	**c3**	**c4**	**c5**	**c1**	**c2**	**c3**	**c4**	**c5**
Current Year					Last Year				
Year	Quarter	Month	Day	Date	Year	Quarter	Month	Day	Date
16	1	1	1	20140101	16	1	1	1	20130101
16	1	1	2	20140102	16	1	1	2	20130102
16	1	1	3	20140103	16	1	1	3	20130103
...	...	...	...	...	...	...	...	...	...
16	1	2	1	20140201	16	1	2	1	20130201
...	...	...	...	...	...	...	...	...	...
16	1	3	1	20140301	16	1	3	1	20130301
...	...	...	...	...	...	...	...	...	...
16	2	1	1	20140401	16	2	1	1	20130401

Table 6.1: Results of calculations on the date table for LY measures

In *SAWITH5* aggregate value of measures for the previous year is computed, using the results of the two mentioned *WITH* clauses.

```
SAWITH5 AS (
SELECT
        SUM(T11623.SALES_NET_EUR) as c2,
        ...
        CONCAT(CONCAT(CONCAT(T2949.CUSTOMER_DES, ' ('), T2949.CUSTOMER_COD), ')') as c4,
        CONCAT(CONCAT(CONCAT(T3098.MATERIAL_QUALITY_DES, ' ('), T3098.MATERIAL_QUALITY_COD)
            , ')') as c5,
        T2949.CUSTOMER_ID as c6,
        T3054.MATERIAL_QUALITY_ID as c7
FROM
        L_CUSTOMER T2949,
        L_MATERIAL_QUALITY T3098,
        L_MATERIAL T3054,
        F_BILL_DOC_ITEM T11623,
        SAWITH2 D5,
        SAWITH4 D3
WHERE
        T2949.CUSTOMER_ID = T11623.CUSTOMER_ID
        AND T3054.MATERIAL_ID = T11623.MATERIAL_ID
        AND T3054.MATERIAL_QUALITY_ID = T3098.MATERIAL_QUALITY_ID
        AND T11623.ASSIGNMENT_DT = D5.c5
        AND D3.c1 = D5.c1
        AND D3.c2 = D5.c2
        AND D3.c3 = D5.c3
        AND D3.c4 = D5.c4
        AND D3.c5 IN (20140101,20141231)
GROUP BY
```

```
        T2949.CUSTOMER_ID,
        T3054.MATERIAL_QUALITY_ID,
        CONCAT(CONCAT(CONCAT(T2949.CUSTOMER_DES, ' ('), T2949.CUSTOMER_COD), ')'),
        CONCAT(CONCAT(CONCAT(T3098.MATERIAL_QUALITY_DES, ' ('), T3098.MATERIAL_QUALITY_COD)
            , ')')
),
```

The same query is run in *SAWITH6* but for the current year. This means that the previous *WITH* clauses are not used and the join is done only with the *L_CALENDAR_DAY* table.

```
SAWITH6 AS (
SELECT ...
FROM
        L_CUSTOMER T2949,
        L_MATERIAL_QUALITY T3098,
        L_MATERIAL T3054,
        L_CALENDAR_DAY T2851,
        F_BILL_DOC_ITEM T11623
WHERE ...
        AND T11623.ASSIGNMENT_DT = T2851.DAY_ID
        AND T2851.DAY_ID IN (20140101,20141231)
GROUP BY ...
),
```

Eventually the *SELECT* statement joins values from the current and the previous year *WITH* clauses, filtering out rows (i.e. pairs Customer - Material Quality) which have no values neither for the current nor for the previous year.

```
SELECT
        D1.c1 as c1,
        ...
        D1.c15 as c15
FROM (
SELECT
        CASE WHEN D1.c4 IS NOT NULL THEN D1.c4 WHEN D2.c4 IS NOT NULL THEN D2.c4 END as c2,
        CASE WHEN D1.c5 IS NOT NULL THEN D1.c5 WHEN D2.c5 IS NOT NULL THEN D2.c5 END as C3,
        CASE WHEN D1.c6 IS NOT NULL THEN D1.c6 WHEN D2.c6 IS NOT NULL THEN D2.c6 END as C4,
        CASE WHEN D1.c7 IS NOT NULL THEN D1.c7 WHEN D2.c7 IS NOT NULL THEN D2.c7 END as C5,
        D2.c2 as c7,
        ...
        D1.c2 as c12,
        ...
        ROW_NUMBER() OVER (
                PARTITION BY
                        CASE
                                WHEN D1.c4 IS NOT NULL THEN D1.c4
```

```
                        WHEN D2.c4 IS NOT NULL THEN D2.c4
                END,
                CASE
                        WHEN D1.c5 IS NOT NULL THEN D1.c5
                        WHEN D2.c5 IS NOT NULL THEN D2.c5
                END,
                CASE
                        WHEN D1.c6 IS NOT NULL THEN D1.c6
                        WHEN D2.c6 IS NOT NULL THEN D2.c6
                END,
                CASE
                        WHEN D1.c7 IS NOT NULL THEN D1.c7
                        WHEN D2.c7 IS NOT NULL THEN D2.c7
                END
        ) AS C46
FROM SAWITH5 D1 FULL OUTER JOIN SAWITH6 D2
                ON D1.c6 = D2.c6 AND D1.c7 = D2.c7
) D1
WHERE ( D1.c46 = 1 )
ORDER BY c4, c5, c2, c3
```

Chapter 7

Reporting

In this chapter we describe the reports and dashboards realized, with sample images where the actual data has been masked for privacy reasons.

Three different dashboards were created, with several pages in each one. Every page includes on the left side a section with the dashboard prompts shown in Figure 7.1.

At the top of the section there is a prompt for the date, with radio buttons for the most frequently used options (*Current Year*, *Current Month* and *Previous 12 Months*) and the possibility to select different dates by clicking on the *Next* button. The rest of the section contains various choice list prompts for the most important attributes of Storage Location, Material, Customer, Vendor and Agent dimensions.

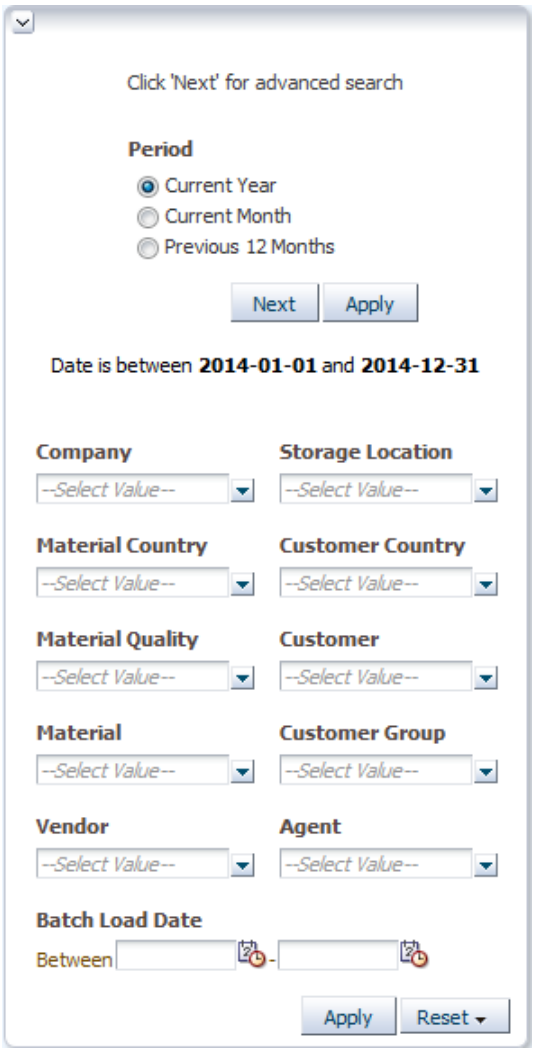
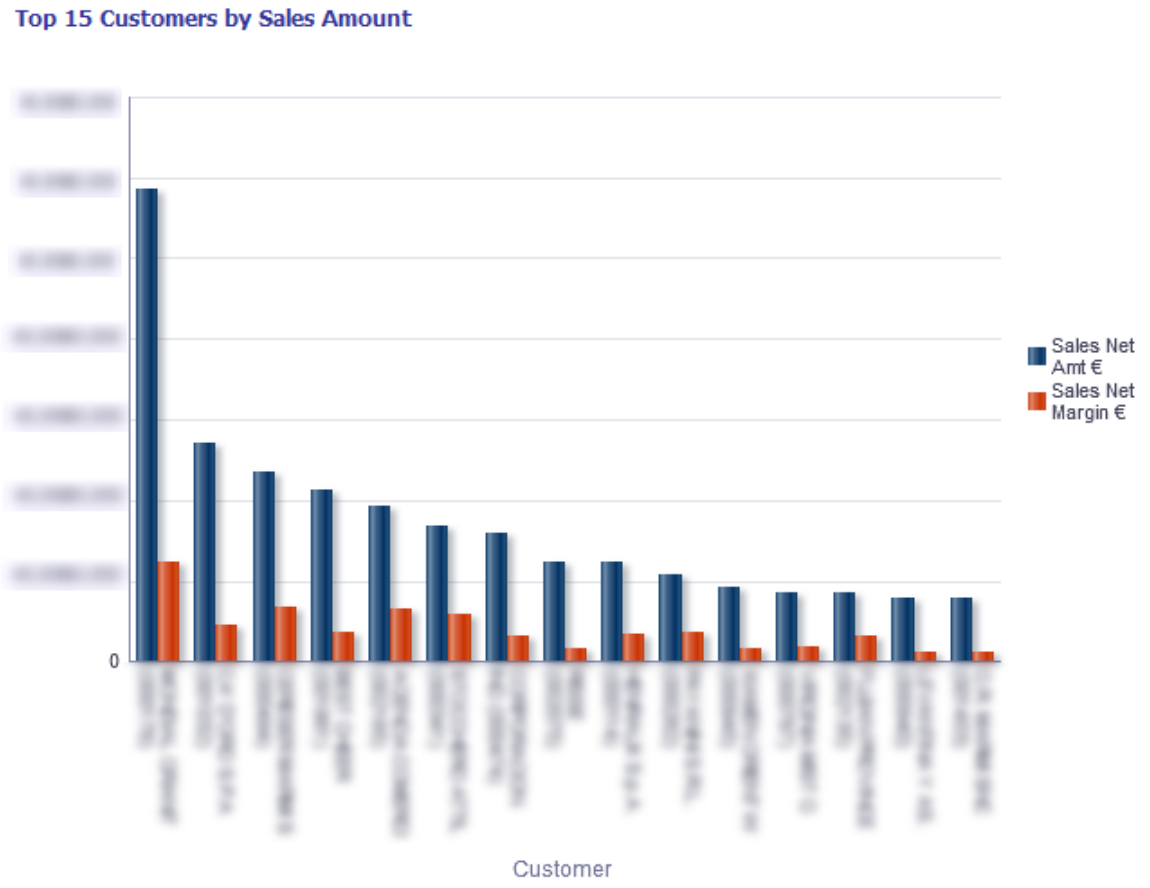


Figure 7.1: Dashboard prompts section

7.1 Company Dashboard

The first dashboard, called *Company Dashboard*, will replace reports already used by R.E.D. Graniti for periodic balances. This dashboard shows top customers and top material qualities according to several measures. *Top 15 Customers by Sales Amount*, as can be seen in Figure 7.2, is the landing page of the dashboard. Like the majority of the dashboard pages it contains a graph on the top, a histogram in this case, to present critical information at a glance, and a table below with all the details and measures. Similar pages were created to monitor top customers by sales volume, top and bottom customers by sales amount variation versus last year.

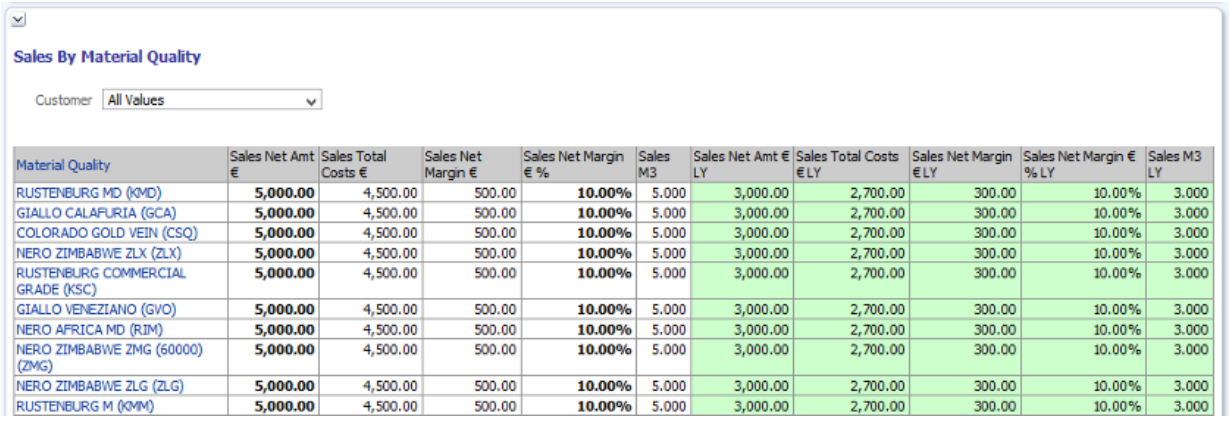


Customer	Sales Net Amt €	Sales Net Margin €	Sales Net Margin € %
CUSTOMER 1 (COD001)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 2 (COD002)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 3 (COD003)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 4 (COD004)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 5 (COD005)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 6 (COD006)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 7 (COD007)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 8 (COD008)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 9 (COD009)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 10 (COD010)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 11 (COD011)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 12 (COD012)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 13 (COD013)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 14 (COD014)	**10,000.00**	1,000.00	**10.00%**
CUSTOMER 15 (COD015)	**10,000.00**	1,000.00	**10.00%**

Figure 7.2: Top 15 Customers by Sales Amount page

To give the users the possibility of digging deep in the details, another section was added underneath each of the already mentioned analyses. This section, called *Sales by Material Quality*, shows the detail of the material qualities bought by a given customer using the *Master-Detail Linking* mechanism. The detail table listens to events occurring in the master table and automatically change the value of its table prompt. In this case clicking on a customer in the first table set the table prompt of

the second table, which is set by default to *All Values*, to that customer. Figure 7.3 shows an example detail table.



Sales By Material Quality

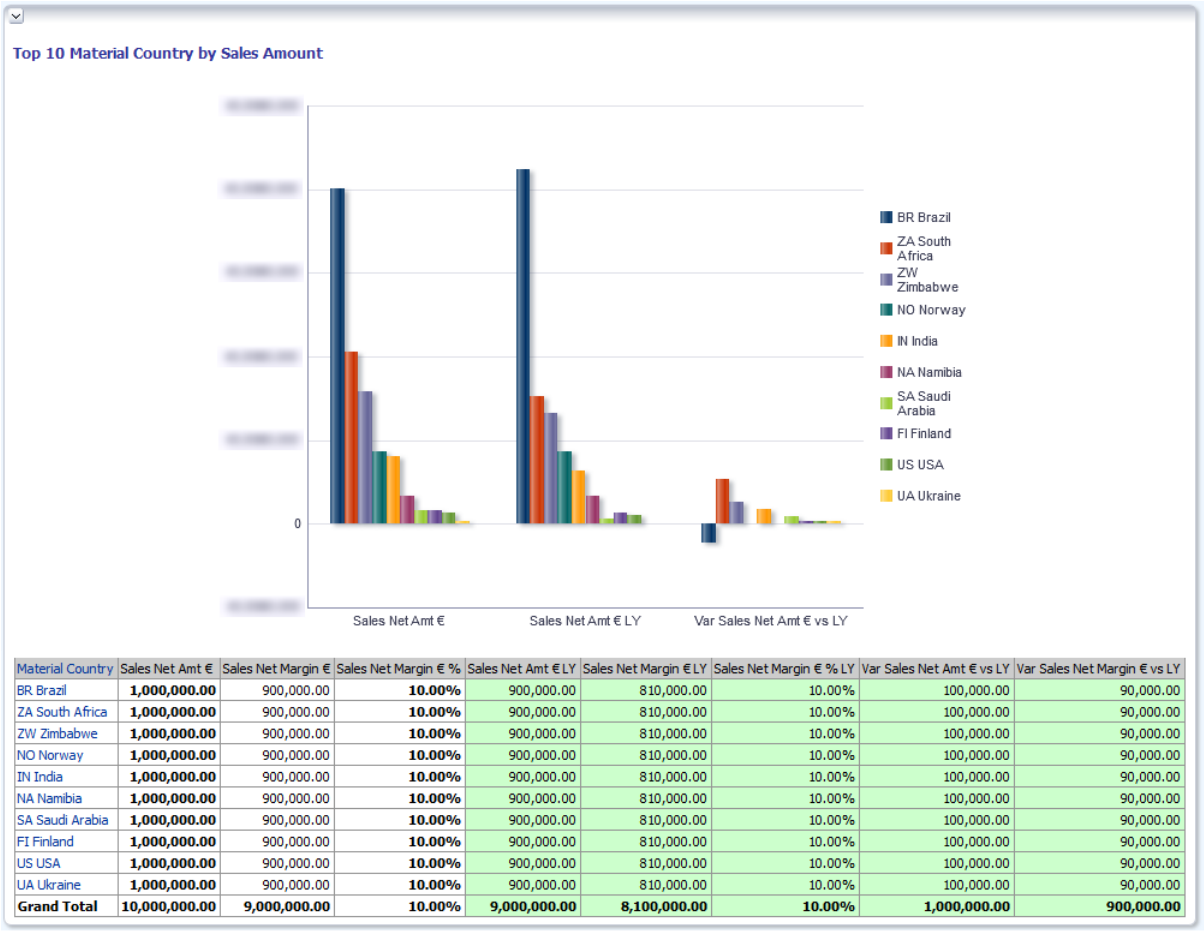
Customer: All Values

Material Quality	Sales Net Amt €	Sales Total Costs €	Sales Net Margin €	Sales Net Margin € %	Sales M3	Sales Net Amt € LY	Sales Total Costs € LY	Sales Net Margin € LY	Sales Net Margin € % LY	Sales M3 LY
RUSTENBURG MD (KMD)	**5,000.00**	4,500.00	500.00	**10.00%**	5.000	3,000.00	2,700.00	300.00	10.00%	3.000
GIALLO CALAFURIA (GCA)	**5,000.00**	4,500.00	500.00	**10.00%**	5.000	3,000.00	2,700.00	300.00	10.00%	3.000
COLORADO GOLD VEIN (CSQ)	**5,000.00**	4,500.00	500.00	**10.00%**	5.000	3,000.00	2,700.00	300.00	10.00%	3.000
NERO ZIMBABWE ZLX (ZLX)	**5,000.00**	4,500.00	500.00	**10.00%**	5.000	3,000.00	2,700.00	300.00	10.00%	3.000
RUSTENBURG COMMERCIAL GRADE (KSC)	**5,000.00**	4,500.00	500.00	**10.00%**	5.000	3,000.00	2,700.00	300.00	10.00%	3.000
GIALLO VENEZIANO (GVO)	**5,000.00**	4,500.00	500.00	**10.00%**	5.000	3,000.00	2,700.00	300.00	10.00%	3.000
NERO AFRICA MD (RIM)	**5,000.00**	4,500.00	500.00	**10.00%**	5.000	3,000.00	2,700.00	300.00	10.00%	3.000
NERO ZIMBABWE ZMG (60000) (ZMG)	**5,000.00**	4,500.00	500.00	**10.00%**	5.000	3,000.00	2,700.00	300.00	10.00%	3.000
NERO ZIMBABWE ZLG (ZLG)	**5,000.00**	4,500.00	500.00	**10.00%**	5.000	3,000.00	2,700.00	300.00	10.00%	3.000
RUSTENBURG M (KMM)	**5,000.00**	4,500.00	500.00	**10.00%**	5.000	3,000.00	2,700.00	300.00	10.00%	3.000

Figure 7.3: Sales by Material Quality section

Similar pages were created to analyze material qualities. In these pages the top section shows a histogram and a table with the top material qualities by sales amount or the top material qualities by sales volume, while the bottom section, linked to the first with the *Master-Detail Linking*, specifies the customers who bought a given material quality.

A different perspective is given in the page *Top 10 Material Country by Sales Amount* as can be seen in Figure 7.4. Here the countries of the quarries from which R.E.D. Graniti extracts the different material qualities are compared, with a focus on the variation between the period currently selected and the previous year.



Material Country	Sales Net Amt €	Sales Net Margin €	Sales Net Margin € %	Sales Net Amt € LY	Sales Net Margin € LY	Sales Net Margin € % LY	Var Sales Net Amt € vs LY	Var Sales Net Margin € vs LY
BR Brazil	**1,000,000.00**	900,000.00	**10.00%**	900,000.00	810,000.00	10.00%	100,000.00	90,000.00
ZA South Africa	**1,000,000.00**	900,000.00	**10.00%**	900,000.00	810,000.00	10.00%	100,000.00	90,000.00
ZW Zimbabwe	**1,000,000.00**	900,000.00	**10.00%**	900,000.00	810,000.00	10.00%	100,000.00	90,000.00
NO Norway	**1,000,000.00**	900,000.00	**10.00%**	900,000.00	810,000.00	10.00%	100,000.00	90,000.00
IN India	**1,000,000.00**	900,000.00	**10.00%**	900,000.00	810,000.00	10.00%	100,000.00	90,000.00
NA Namibia	**1,000,000.00**	900,000.00	**10.00%**	900,000.00	810,000.00	10.00%	100,000.00	90,000.00
SA Saudi Arabia	**1,000,000.00**	900,000.00	**10.00%**	900,000.00	810,000.00	10.00%	100,000.00	90,000.00
FI Finland	**1,000,000.00**	900,000.00	**10.00%**	900,000.00	810,000.00	10.00%	100,000.00	90,000.00
US USA	**1,000,000.00**	900,000.00	**10.00%**	900,000.00	810,000.00	10.00%	100,000.00	90,000.00
UA Ukraine	**1,000,000.00**	900,000.00	**10.00%**	900,000.00	810,000.00	10.00%	100,000.00	90,000.00
Grand Total	**10,000,000.00**	**9,000,000.00**	**10.00%**	**9,000,000.00**	**8,100,000.00**	**10.00%**	**1,000,000.00**	**900,000.00**

Figure 7.4: Top 10 Material Country by Sales Amount page

The last part of the *Company Dashboard* is used for performance comparisons between months. Figure 7.5 shows the *Sales Amount Comparison by Month* page, where each row correspond to a month and where the measure analyzed is the sales amount with respect to the same month of the previous year. Similar pages are included in the dashboard for analyzing sales volumes instead of sales amounts and year to date measures (e.g. *Sales Amount Progression by Month* and *Sales Volume Progression by Month* pages).

Month	Sales Net Amt €	Sales Net Amt € LY	Var Sales Net Amt € vs LY	Var% Sales Net Amt € vs LY
2002 January	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 February	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 March	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 April	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 May	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 June	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 July	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 August	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 September	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 October	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 November	**1,000,000.00**	900,000.00	100,000.00	10.00%
2002 December	**1,000,000.00**	900,000.00	100,000.00	10.00%
Grand Total	**12,000,000.00**	**10,800,000.00**	**1,200,000.00**	**10.00%**

Figure 7.5: Sales Amount Comparison by Month page

7.2 Sales Details

The *Sales Details* dashboard is intended to give both a synthetic and an analytic picture of sales. The *Start Page*, besides the usual dashboard prompts section, contains the sections shown in Figure 7.6 and 7.7.

The first section takes advantage of an element called *Performance Tiles* to give a view as synthetic as possible over the four main measures (*Net Amount*, *Total Costs*, *Net Margin* and *M3*) for the current month and for year to date period. Conditional formatting is used to color each tile differently

according to the variation of that measure with respect to the previous year.

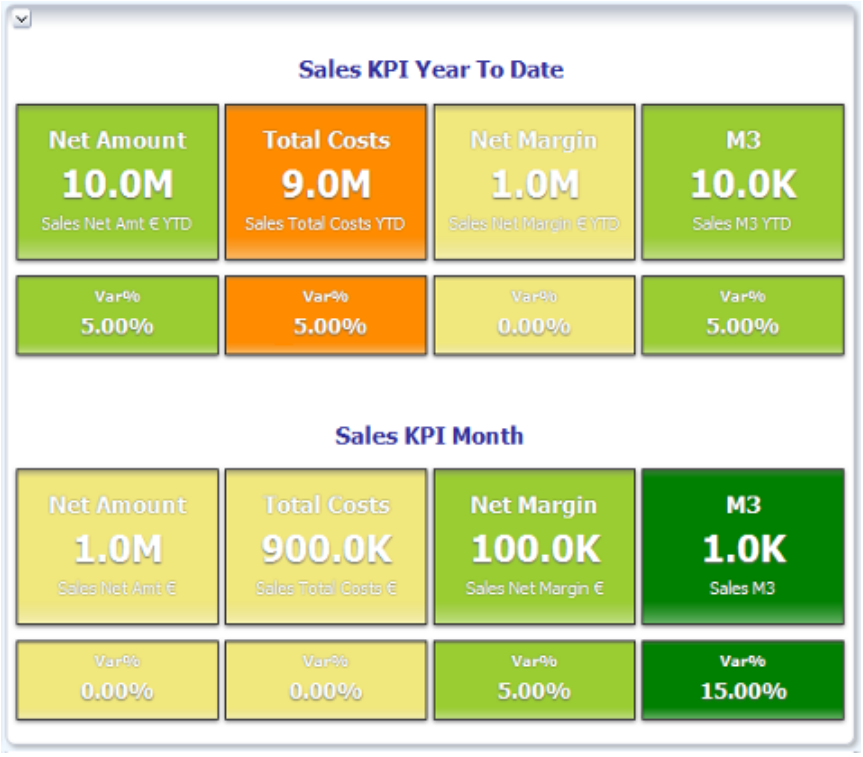


Figure 7.6: Performance Tiles section of the Start Page

The second section uses instead the *Column Selector* element of Oracle Business Intelligence to hold a large amount of measures in a single histogram. The measure shown with the blue bars, red bars or lines can be changed with a drop-down menu. The eight different measures displayed in the performance tiles section and the corresponding values for the previous year can be chosen for the bars, while the eight different percentage variation shown in the performance tiles section can be chosen for the graph lines.

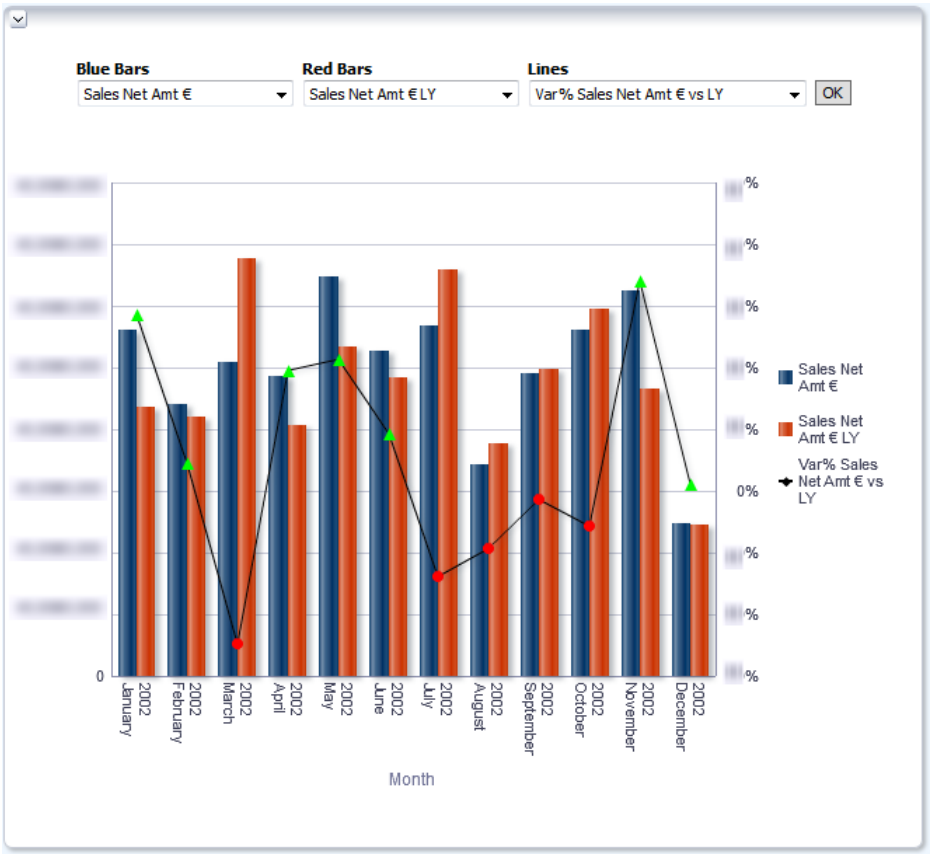
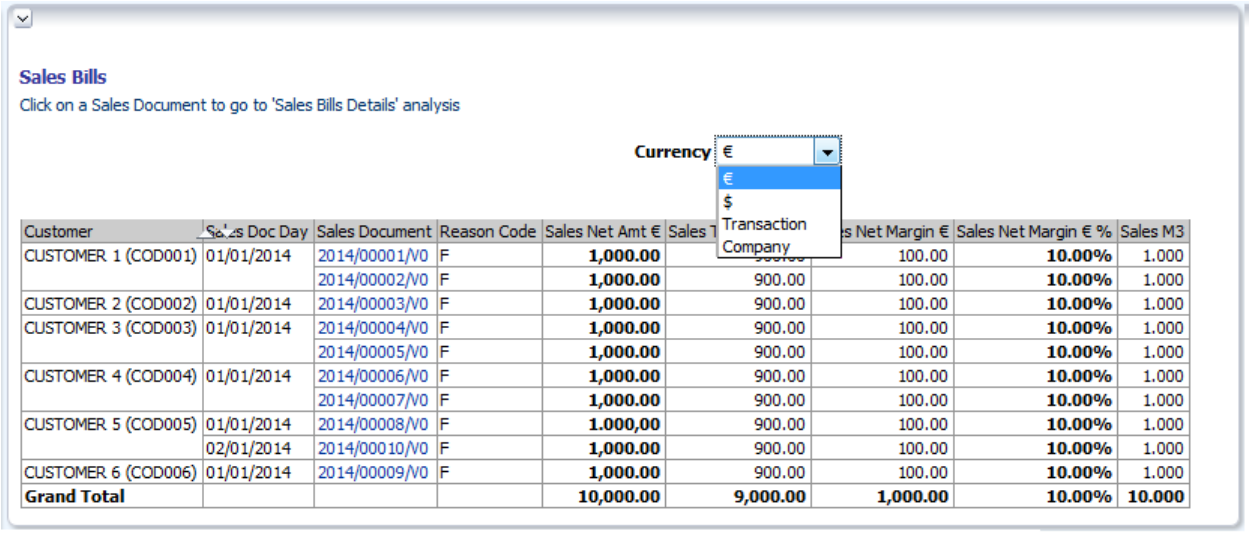


Figure 7.7: Histogram section of the Start Page

Another page in the *Sales Details* dashboard is called *Sales Bills* and gives every possible detail about sales, starting from the sales documents made in a certain period, yesterday by default.



Sales Bills

Click on a Sales Document to go to 'Sales Bills Details' analysis

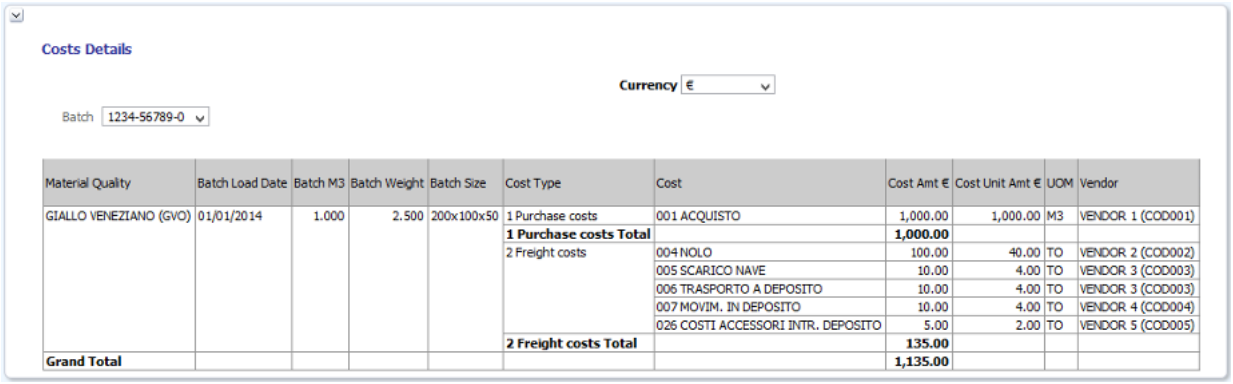
Currency: € (€, $, Transaction Company)

Customer	Sales Doc Day	Sales Document	Reason Code	Sales Net Amt €	Sales T	s Net Margin €	Sales Net Margin € %	Sales M3
CUSTOMER 1 (COD001)	01/01/2014	2014/00001/V0	F	**1,000.00**		100.00	**10.00%**	1.000
		2014/00002/V0	F	**1,000.00**	900.00	100.00	**10.00%**	1.000
CUSTOMER 2 (COD002)	01/01/2014	2014/00003/V0	F	**1,000.00**	900.00	100.00	**10.00%**	1.000
CUSTOMER 3 (COD003)	01/01/2014	2014/00004/V0	F	**1,000.00**	900.00	100.00	**10.00%**	1.000
		2014/00005/V0	F	**1,000.00**	900.00	100.00	**10.00%**	1.000
CUSTOMER 4 (COD004)	01/01/2014	2014/00006/V0	F	**1,000.00**	900.00	100.00	**10.00%**	1.000
		2014/00007/V0	F	**1,000.00**	900.00	100.00	**10.00%**	1.000
CUSTOMER 5 (COD005)	01/01/2014	2014/00008/V0	F	**1,000.00**	900.00	100.00	**10.00%**	1.000
	02/01/2014	2014/00010/V0	F	**1,000.00**	900.00	100.00	**10.00%**	1.000
CUSTOMER 6 (COD006)	01/01/2014	2014/00009/V0	F	**1,000.00**	900.00	100.00	**10.00%**	1.000
Grand Total				**10,000.00**	**9,000.00**	**1,000.00**	**10.00%**	**10.000**

Figure 7.8: Sales Bills page

With a mechanism called *Action Links*, clicking on a certain sales document opens a new analysis, filtered by the element selected. In this case the analysis opened shows the details of the selected document in terms of the specific batches sold. Here again is possible to click on a batch code to launch

another detail analysis with the costs incurred for extracting or buying that batch, transporting it and selling it. This analysis, called *Costs Details*, is shown in Figure 7.9.



Costs Details

Currency €

Batch 1234-56789-0

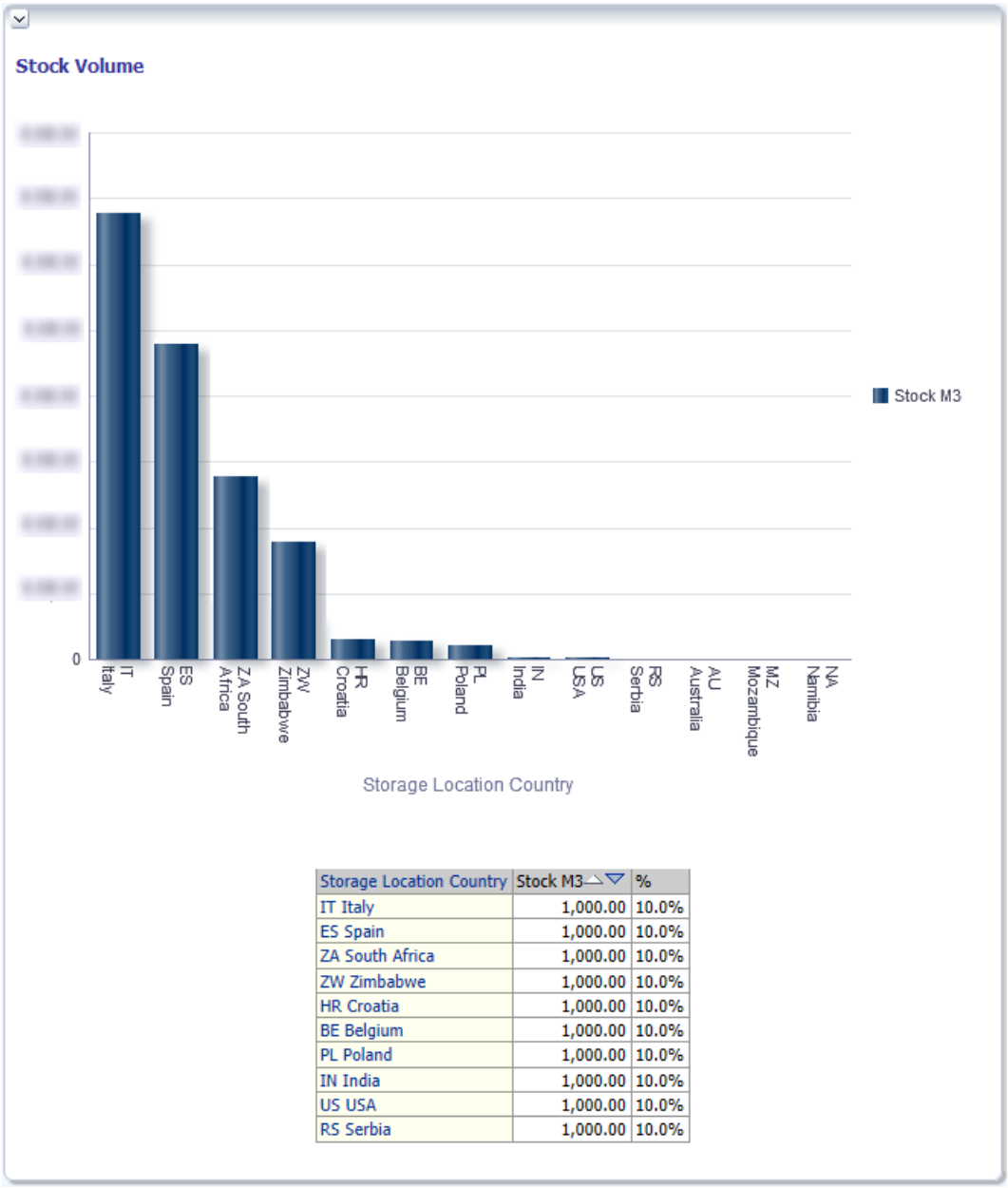
Material Quality	Batch Load Date	Batch M3	Batch Weight	Batch Size	Cost Type	Cost	Cost Amt €	Cost Unit Amt €	UOM	Vendor
GIALLO VENEZIANO (GVO)	01/01/2014	1.000	2.500	200x100x50	1 Purchase costs	001 ACQUISTO	1,000.00	1,000.00	M3	VENDOR 1 (COD001)
					1 Purchase costs Total		**1,000.00**			
					2 Freight costs	004 NOLO	100.00	40.00	TO	VENDOR 2 (COD002)
						005 SCARICO NAVE	10.00	4.00	TO	VENDOR 3 (COD003)
						006 TRASPORTO A DEPOSITO	10.00	4.00	TO	VENDOR 3 (COD003)
						007 MOVIM. IN DEPOSITO	10.00	4.00	TO	VENDOR 4 (COD004)
						026 COSTI ACCESSORI INTR. DEPOSITO	5.00	2.00	TO	VENDOR 5 (COD005)
					2 Freight costs Total		**135.00**			
Grand Total							**1,135.00**			

Figure 7.9: Costs details of a batch

Some other pages in this dashboard give details about sales documents by customer or by material quality, with similar presentations and the same navigation path, from the sales document to the single batch sold and its costs detailed, hence we will not present them in detail.

7.3 Stocks Details

A third dashboard is used to analyze stock levels according to various dimensions. An example page, *Stocks Volume*, is illustrated in Figure 7.10 and the grouping attribute used here is the country of the storage location, although users have the possibility to drill down to the storage location level of the hierarchy. The usual format of a synthetic histogram followed by a detailed table is used in the mentioned page and throughout the dashboard. Similar pages were created using also *Stock Amount* measures and material quality and vendor dimensions.



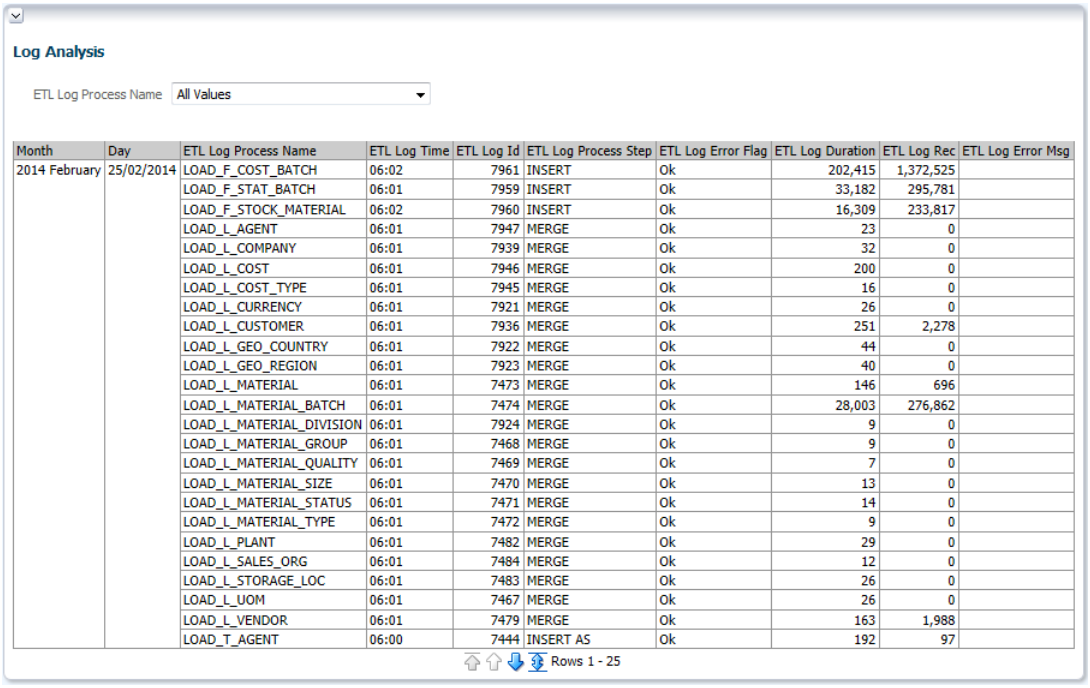
Storage Location Country	Stock M3	%
IT Italy	1,000.00	10.0%
ES Spain	1,000.00	10.0%
ZA South Africa	1,000.00	10.0%
ZW Zimbabwe	1,000.00	10.0%
HR Croatia	1,000.00	10.0%
BE Belgium	1,000.00	10.0%
PL Poland	1,000.00	10.0%
IN India	1,000.00	10.0%
US USA	1,000.00	10.0%
RS Serbia	1,000.00	10.0%

Figure 7.10: Stocks Volume page

7.4 Other reports

Other auxiliary reports, not necessarily embedded into dashboards, were created. An example is the *Log Analysis* in Figure 7.11. At the end of every ETL procedure a record is inserted in a dedicated table of the database, with information regarding the name of the procedure, its outcome and the error message in case the procedure failed for some reason, the number of records inserted or updated and the time in milliseconds it took.

This table is the only source for the aforementioned analysis and this data is also used to send an automatic email every morning containing the final outcome of the entire ETL process.



Log Analysis

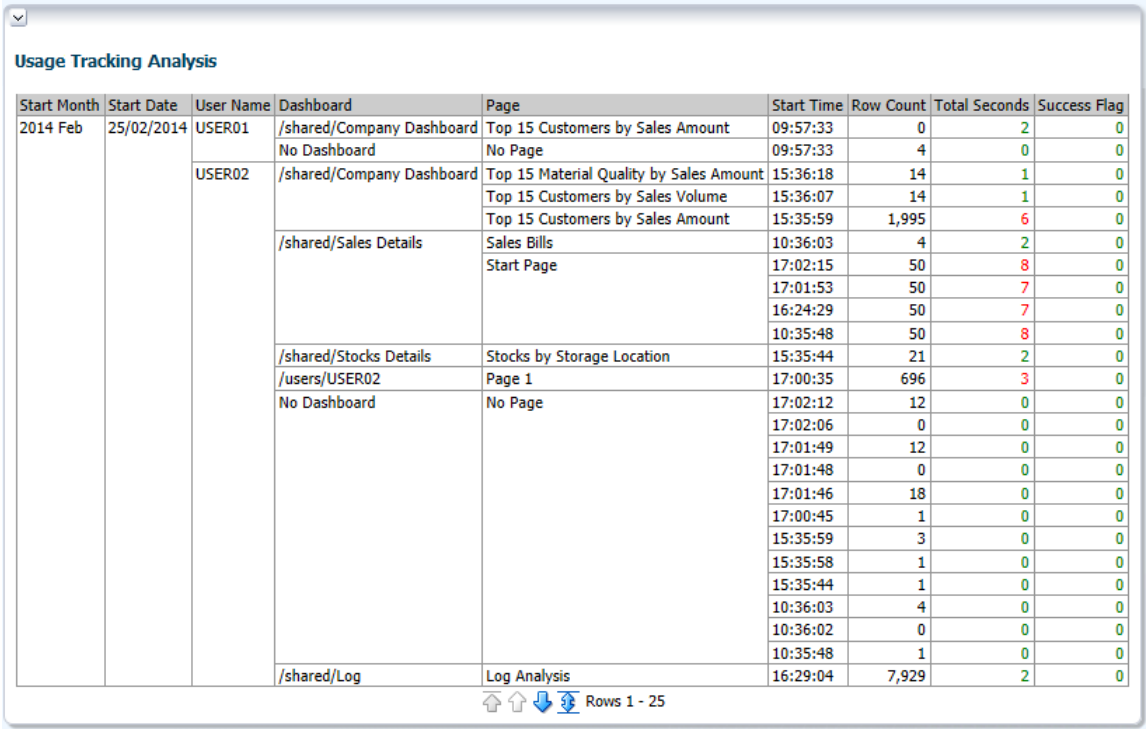
ETL Log Process Name: All Values

Month	Day	ETL Log Process Name	ETL Log Time	ETL Log Id	ETL Log Process Step	ETL Log Error Flag	ETL Log Duration	ETL Log Rec	ETL Log Error Msg
2014 February	25/02/2014	LOAD_F_COST_BATCH	06:02	7961	INSERT	Ok	202,415	1,372,525	
		LOAD_F_STAT_BATCH	06:01	7959	INSERT	Ok	33,182	295,781	
		LOAD_F_STOCK_MATERIAL	06:02	7960	INSERT	Ok	16,309	233,817	
		LOAD_L_AGENT	06:01	7947	MERGE	Ok	23	0	
		LOAD_L_COMPANY	06:01	7939	MERGE	Ok	32	0	
		LOAD_L_COST	06:01	7946	MERGE	Ok	200	0	
		LOAD_L_COST_TYPE	06:01	7945	MERGE	Ok	16	0	
		LOAD_L_CURRENCY	06:01	7921	MERGE	Ok	26	0	
		LOAD_L_CUSTOMER	06:01	7936	MERGE	Ok	251	2,278	
		LOAD_L_GEO_COUNTRY	06:01	7922	MERGE	Ok	44	0	
		LOAD_L_GEO_REGION	06:01	7923	MERGE	Ok	40	0	
		LOAD_L_MATERIAL	06:01	7473	MERGE	Ok	146	696	
		LOAD_L_MATERIAL_BATCH	06:01	7474	MERGE	Ok	28,003	276,862	
		LOAD_L_MATERIAL_DIVISION	06:01	7924	MERGE	Ok	9	0	
		LOAD_L_MATERIAL_GROUP	06:01	7468	MERGE	Ok	9	0	
		LOAD_L_MATERIAL_QUALITY	06:01	7469	MERGE	Ok	7	0	
		LOAD_L_MATERIAL_SIZE	06:01	7470	MERGE	Ok	13	0	
		LOAD_L_MATERIAL_STATUS	06:01	7471	MERGE	Ok	14	0	
		LOAD_L_MATERIAL_TYPE	06:01	7472	MERGE	Ok	9	0	
		LOAD_L_PLANT	06:01	7482	MERGE	Ok	29	0	
		LOAD_L_SALES_ORG	06:01	7484	MERGE	Ok	12	0	
		LOAD_L_STORAGE_LOC	06:01	7483	MERGE	Ok	26	0	
		LOAD_L_UOM	06:01	7467	MERGE	Ok	26	0	
		LOAD_L_VENDOR	06:01	7479	MERGE	Ok	163	1,988	
		LOAD_T_AGENT	06:00	7444	INSERT AS	Ok	192	97	

Rows 1 - 25

Figure 7.11: Log Analysis

Another report, or analysis using Oracle Business Intelligence terminology, is used to track user activities on the Business Intelligence Application. In fact it is possible to let the Oracle BI Server store every query issued because of a user request in an appropriate table. Data from this table is then presented as in Figure 7.12, with information about dashboards and pages visited, time of the activity, number of rows returned by the query and time elapsed and so on.



Usage Tracking Analysis

Start Month	Start Date	User Name	Dashboard	Page	Start Time	Row Count	Total Seconds	Success Flag
2014 Feb	25/02/2014	USER01	/shared/Company Dashboard	Top 15 Customers by Sales Amount	09:57:33	0	2	0
			No Dashboard	No Page	09:57:33	4	0	0
		USER02	/shared/Company Dashboard	Top 15 Material Quality by Sales Amount	15:36:18	14	1	0
				Top 15 Customers by Sales Volume	15:36:07	14	1	0
				Top 15 Customers by Sales Amount	15:35:59	1,995	6	0
			/shared/Sales Details	Sales Bills	10:36:03	4	2	0
				Start Page	17:02:15	50	8	0
					17:01:53	50	7	0
					16:24:29	50	7	0
					10:35:48	50	8	0
			/shared/Stocks Details	Stocks by Storage Location	15:35:44	21	2	0
			/users/USER02	Page 1	17:00:35	696	3	0
			No Dashboard	No Page	17:02:12	12	0	0
					17:02:06	0	0	0
					17:01:49	12	0	0
					17:01:48	0	0	0
					17:01:46	18	0	0
					17:00:45	1	0	0
					15:35:59	3	0	0
					15:35:58	1	0	0
					15:35:44	1	0	0
					10:36:03	4	0	0
					10:36:02	0	0	0
					10:35:48	1	0	0
			/shared/Log	Log Analysis	16:29:04	7,929	2	0

Rows 1 - 25

Figure 7.12: Usage Tracking Analysis

Chapter 8

Conclusions

We described the design and development process of the data warehouse and business intelligence system. The design was appropriate and accurate, since no major changes were needed to the requirements analysis during the course of the project and only minor corrections regarding the column mapping between the data sources and the data warehouse were made. The chosen source table for stocks revealed itself to be wrong, because of its coarse granularity (i.e. Material instead of Batch), and was later replaced by the table storing warehouse handling, from which monthly stock levels were reconstructed. Some business rules were not initially mentioned in the requirements gathering phase and emerged at a later stage, but this did not cause any significant problem to the extract, transform and load process, which required only minor modifications.

Promptly correcting these small flaws was at the base of the small effort required by the corrections themselves. This was possible because of the quick and incremental development of the data warehouse and the reporting, which went at the same speed. At the moment of the SAP go-live the developed decision support system was already able to accomodate data from this new source.

A few open points are still left at the moment of this writing, but mainly concern the particular structure given to some of the reports. Future developments instead are numerous. R.E.D. Graniti management has already shown interest in adding the Orders business process to the data warehouse, to analyze, for example, non-billed orders and batch reservations. Moreover a new level may be added to the Material hierarchy, to group similar material qualities, such as *African Red* for grouping African Red Dark, African Red Medium and African Red Light qualities. Also the integration of the Business Intelligence with data coming from Hyperion Financial Management (HFM) is expected in the near future.

Bibliography

[1] Antonio Albano. Decision support databases essentials, December 2013.

[2] Stephen Few. Information dashboard design: the effective visual communication of data, 2006.

[3] Ralph Kimball, Margy Ross, Warren Thornthwaite, Joy Mundy, and Bob Becker. The data warehouse lifecycle toolkit, 2008.

[4] Carlo Montani. XXIV world marble and stones report 2013, November 2013.

[5] Oracle. Security guide for Oracle Business Intelligence Enterprise Edition, April 2012.

[6] Oracle. Metadata repository builder's guide for Oracle Business Intelligence Enterprise Edition, February 2013.

[7] Oracle. Oracle Database data warehousing guide, July 2013.

[8] Oracle. System administrator's guide for Oracle Business Intelligence Enterprise Edition, February 2013.

[9] Oracle. User's guide for Oracle Business Intelligence Enterprise Edition, February 2013.