Business Intelligence

Project Report

Practical Summer Training

Name : Piyush Ahuja

Entry Number: 2008MT50454

Place of Training: Deloitte (Mumbai)

Training Period: 16 May 2011 – 22 July 2011

Contents

Project Overview 3

Introduction 4

Team Structure 5

Deliverable Details 6

Objectives and Requirements 7

The Data Warehouse: Components 8

Dimensional Modelling 12

Development Phases 14

Lessons Learnt 18

Summary 20

Project Overview

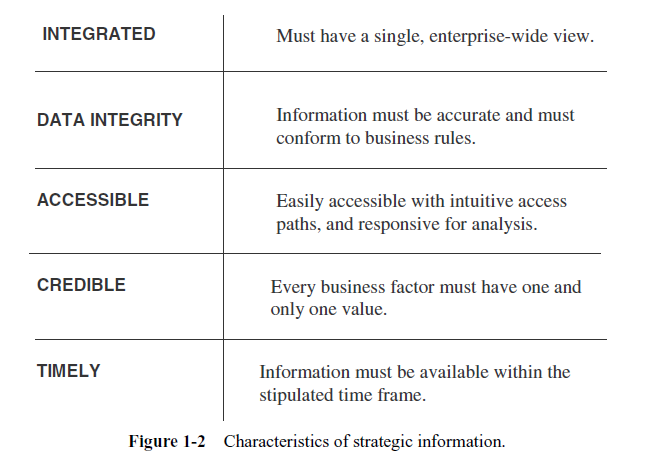
As businesses grow more complex, corporations spread globally, and competition becomes fiercer, business executives are becoming desperate for information to stay competitive and improve the bottom line. The operational computer systems do provide information to run the day-to-day operations, but what the executives need are different kinds of information that could be readily used to make strategic decisions. They want to know where to build the next warehouse, which product lines to expand, and which markets they should strengthen. The operational systems, important as they are, cannot provide strategic information.

Business intelligence refers to computer-based techniques used in identifying, extracting, and analyzing business data, such as sales revenue by products and departments, or by associated costs and incomes.

BI technologies provide historical, current and predictive views of business operations. Common functions of business intelligence technologies are online analytical processing, analytics, data mining, process mining, business performance management, benchmarking, text mining, reporting, and predictive analytics. Business intelligence aims to support better business decision-making. Thus a BI system can be called a decision support system (DSS).

We, at Deloitte, worked on improving a program called Communication Excellence by applying Business Intelligence concepts. For achieving this, we employed a special paradigm called Data Warehousing. The data warehouse is an informational environment that :

* Provides an integrated and total view of the enterprise
* Makes the enterprise’s current and historical information easily available for decision making
* Makes decision-support transactions possible without hindering operational systems
* Renders the organization’s information consistent
* Presents a flexible and interactive source of strategic information



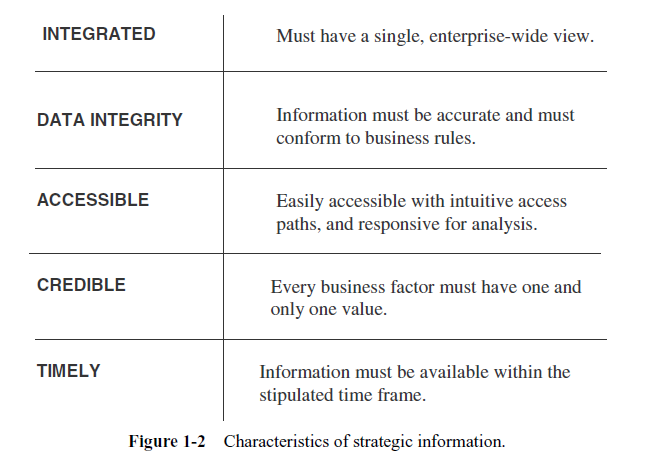
Introduction

The Communication Excellence program within Deloitte exists to improve the communication skills of the employs. The program conducts tests, trainings and workshops across all branches, locations, designations, service lines etc. The program thus takes in as input humongous amounts of data, which is entered into the OLTP databases.

The executives and managers who are responsible for keeping the program competitive need information to make proper decisions. They need information to formulate the program strategies, establish goals, set objectives, and monitor results and performance.

In this regard, a Decision-support system to get strategic information *out of* the database is needed, as opposed to OLTP systems that are designed to put the data *into* the database.

Strategic information is not for running the day-to-day operations of the program. It is not intended to produce report card, display a score, find out the communication training hours or requirement etc. Strategic information is far more important for the continued health and survival of the program.



Thus there was a requirement for building a Decision Support System, rom the raw data of Communication Excellence Program, aimed at improving the efficiency and effectiveness of the program by providing vital strategic information. Such a system should support faster and better decision-making, incorporating various computing techniques in identifying, extracting and analyzing Business Data.

Team Structure

Our Project was headed by Mr. Jignesh Barai. He used give direction and guidance in the team meetings on a frequent basis. My teammates and me reported daily to Ms Nikita Dhameja, who used to help us in our daily work and conveyed the progress to the project head.

Deliverable Details

The following table gives a list of deliverable which I supplied to my company as part of different stages in the project.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Deliverable Name** | **Del. Date** | **%age Completion** | **Status** |
| 1 | Theoretical Data Model | Jun 15, 2011 | 100% | Done |
| 2 | Technical Specifications of Tables | Jun 17, 2011 | 100% | Done |
| 3 | Unit Test Document | Jun 27, 2011 | 100% | Done |
| 4 | Combined ETL Code | Jul 6, 2011 | 100% | Done |
| 5 | Final Code | Jul 19, 2011 | 100% | Done |
| 6 | Presentation | Jul 21, 2011 | 100% | Done |

Objectives and Requirements

Let us examine the desirable features and processing requirements of this new type of

system environment. Let us also consider the advantages of this type of system environment

designed for strategic information.

**A New Type of System Environment**

The desired features of the new type of system environment are:

* Database designed for analytical tasks
* Data from multiple applications
* Easy to use and conducive to long interactive sessions by users
* Read-intensive data usage
* Direct interaction with the system by the users without IT assistance
* Content updated periodically and stable
* Content to include current and historical data
* Ability for users to run queries and get results online
* Ability for users to initiate reports

**Processing Requirements in the New Environment**

Most of the processing in the new environment for strategic information will have to be

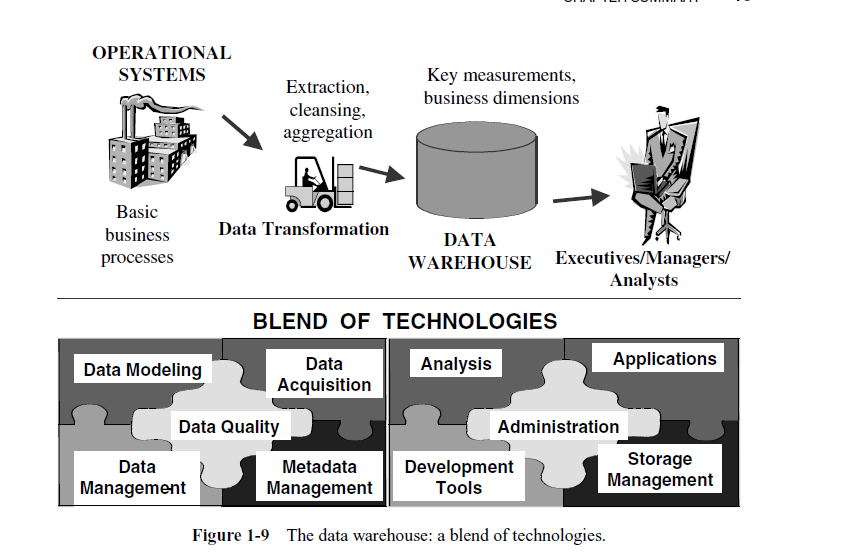
Analytical. There are four levels of analytical processing requirements:

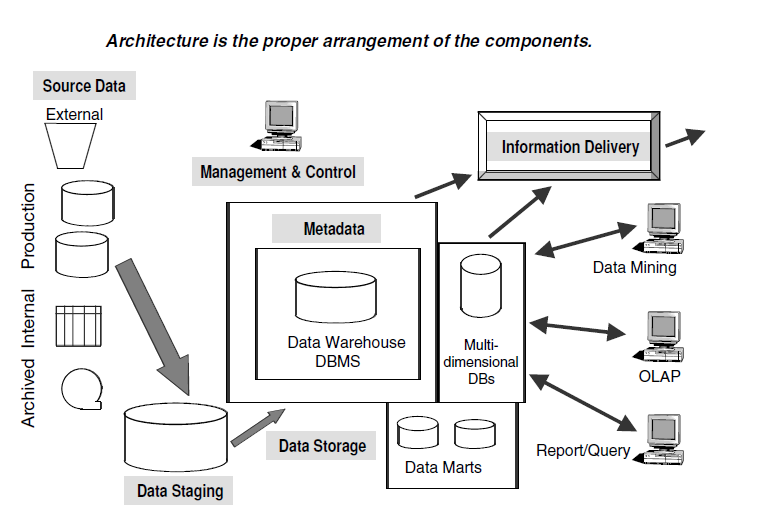
* Running of simple queries and reports against current and historical data
* Ability to perform “what if” analysis is many different ways
* Ability to query, step back, analyze, and then continue the process to any desired

length

* Spot historical trends and apply them for future results

The Data Warehouse: Components





Our database maintained its functions in three layers: staging, integration, and access

**1. Data Staging Component**

After we have extracted data from various operational systems and from external sources, we had to prepare the data for storing in the data warehouse. The extracted data coming from several disparate sources needed to be changed, converted, and made ready in a format that is suitable to be stored for querying and analysis.

Three major functions needed to be performed for getting the data ready. We had to extract the data, transform the data, and then load the data into the data warehouse storage.

These three major functions of extraction, transformation, and preparation for loading take place in a staging area. The data staging component consisted of a workbench for these functions. Data staging provided a place and an area with a set of functions to clean, change, combine, convert, deduplicate, and prepare source data for storage and use in the data warehouse.

We needed a separate place or component to perform the data preparation and we could not move the data from the various sources into the data warehouse storage itself and then prepare the data because when we implement an operational system, we are likely to pick up data from different sources, move the data into the new operational system database, and run data conversions. Why can’t this method work for a data warehouse? The essential difference here was this: in a data warehouse we pulled in data from many source operational systems. Data in a data warehouse is subject-oriented and cuts across operational applications. A separate staging area, therefore, was a necessity for preparing data for the data warehouse.

I will now briefly discuss the three major functions that take place in the staging area:

**Data Extraction.**This function has to deal with numerous data sources. We had to employ the appropriate technique for each data source. Source data may be from different source machines in diverse data formats. Part of the source data was in relational database systems. Some data was on other legacy network and hierarchical data models. Many data sources was still in flat files. We wanted to include data from spreadsheets and local departmental data sets. Data extraction became quite complex.

Tools were available on the market for data extraction. We considered using outside tools suitable for certain data sources.

We extracted the source into a separate physical environment from which moving the data into the data warehouse

would be easier. In the separate environment, we could extract the source data into a group of flat files, or a data-staging relational database, or a combination of both.

**Data Transformation.**In every system implementation, data conversion is an important function. For example, when we implement an operational, we have to initially populate your database with data from the prior system records.

Data for a data warehouse comes from many disparate sources. If data extraction for a data warehouse posed great challenges, data transformation presented even greater challenges. Another factor in the data warehouse is that the data feed was not just an initial load. We had continue to pick up the ongoing changes from the source systems. Any transformation tasks we set up for the initial load were adapted for the ongoing revisions as well.

We performed a number of individual tasks as part of data transformation. First, we clean the data extracted from each source. Cleaning involved correction of misspellings, included resolution of conflicts between state codes and zip codes in the source data, dealt with providing default values for missing data elements, or elimination of duplicates when you bring in the same data from multiple source systems.

Standardization of data elements formed a large part of data transformation. We standardized the data types and field lengths for same data elements retrieved from the various sources. Semantic standardization was another major task. We resolved synonyms and homonyms. When two or more terms from different source systems meant the same thing, we resolved the synonyms. When a single term meant many different things in different source systems, we resolve the homonym.

Data transformation involved many forms of combining pieces of data from the different sources. We combined data from single source record or related data elements from many source records. On the other hand, data transformation also involved purging source data that is not useful and separating out source records into new combinations. Sorting and merging of data takes place on a large scale in the data staging area.

When the data transformation function ends, we have a collection of integrated data that is cleaned, standardized, and summarized. We have data ready to load into each data set in your data warehouse.

**Data Loading*.*** Two distinct groups of tasks formed the data loading function. When we

completed the design and construction of the data warehouse and went live for the first time,

we did the initial loading of the data into the data warehouse storage. The initial load

moved large volumes of data using up substantial amounts of time. As the data warehouse

starts functioning, we continue to extract the changes to the source data, transform the

data revisions, and feed the incremental data revisions on an ongoing basis.

**2. Data Storage Component**

The data storage for the data warehouse was a separate repository. The operational systems

of our program supported the day-to-day operations. These were online transaction processing

applications. The data repositories for the operational systems contained only

the current data. Also, these data repositories contained the data structured in highly normalized

formats for fast and efficient processing. In contrast, in the data repository for a data

warehouse, we need ed to keep large volumes of historical data for analysis. Further, we

had to keep the data in the data warehouse in structures suitable for analysis, and not for

quick retrieval of individual pieces of information. Therefore, the data storage for the data

warehouse was kept separate from the data storage for operational systems.

In our databases supporting operational systems, the updates to data happen as transactions

occur. These transactions hit the databases in a random fashion. How and when

the transactions change the data in the databases is not completely within our control.

The data in the operational databases could change from moment to moment. When our

analysts use the data in the data warehouse for analysis, they need to know that the data is

stable and that it represents snapshots at specified periods. As they are working with the

data, the data storage must not be in a state of continual updating. For this reason, the data

warehouses were “read-only” data repositories.

The database in our data warehouse was open. We used tools from multiple vendors. It employed relational database management systems.

**3. Information Delivery Component**

In order to provide information to the wide community of data warehouse users, the information

delivery component was to include different methods of information delivery. Ad hoc reports are predefined reports

primarily meant for novice and casual users. Provision for complex queries, multidimensional

(MD) analysis, and statistical analysis catered to the needs of the business analysts

and power users. Information fed into Executive Information Systems is meant

for senior executives and high-level managers.

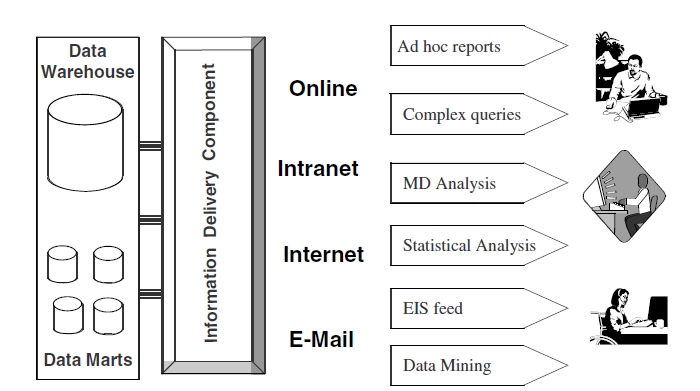
In our data warehouse, we included several information delivery mechanisms.

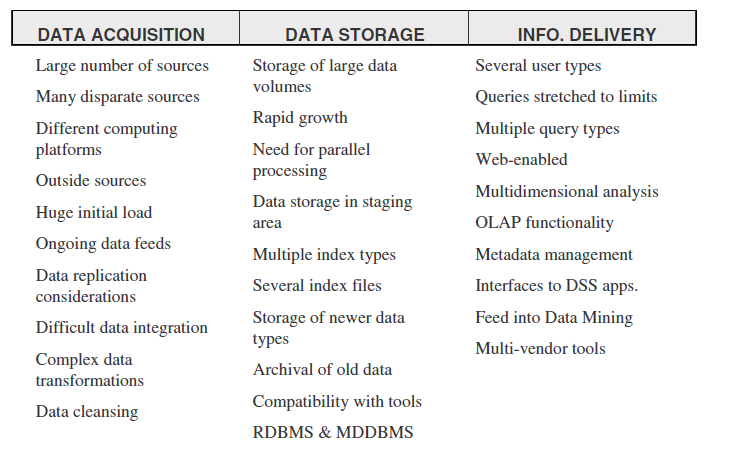
We provided for online queries and reports. The users will enter their requests

online and will receive the results online. We set up delivery of scheduled reports

through e-mail or you may make adequate use of your organization’s intranet for information

delivery.





Dimensional Modelling

**Design Decisions**

Before we proceeded with designing the dimensional data model, let us quickly review some

of the design decisions we had to make:

**Choosing the process.** Selecting the subjects from the information packages for the

first set of logical structures to be designed.

**Choosing the grain.** Determining the level of detail for the data in the data structures.

**Identifying and conforming the dimensions.** Choosing the dimensions

(such as service line, designation, time, etc.) to be included in the first set of structures and

making sure that each particular data element in every business dimension is conformed

to one another.

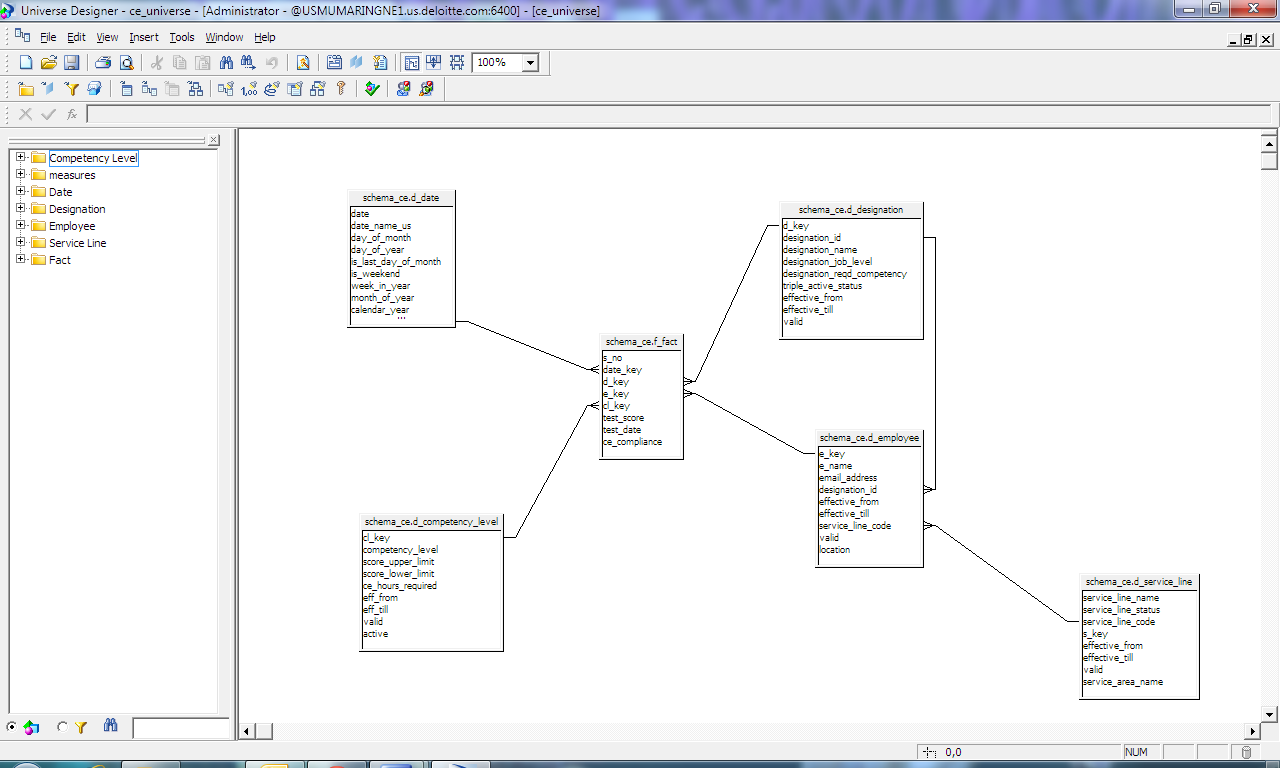
**Choosing the facts.** Selecting the metrics or units of measurements (such as test scores, training hours etc.) to be included in the first set of structures.

**Choosing the duration of the database.** Determining how far back in time you should go for historical data.

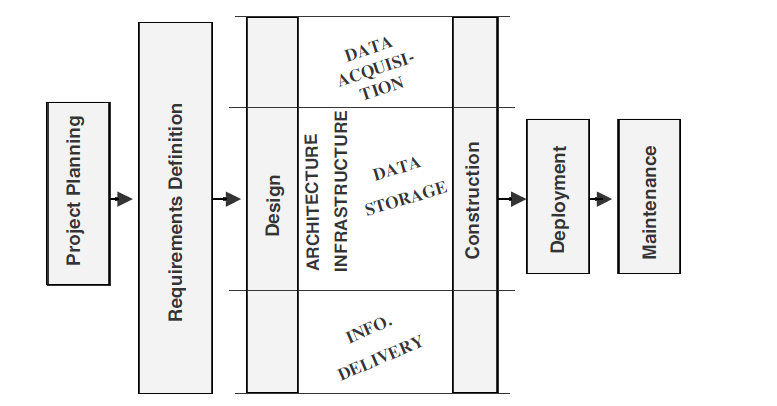
We applied a dimensional approach, where the transeaction data partitioned into either "facts", generally numeric transaction data, or "dimensions", the reference information that gives context to the facts .

There were several benefits of using this approach:

* Most Suitable for Query Processing
* Optimizes Navigation
* Easy for Users to Understand
* Faster and easier implementation of manageable pieces
* Favorable return on investment and proof of concept
* Less risk of failure
* Inherently incremental
* Allowed project team to learn and grow



Development Phases



**Steps followed**

1. **Requirement Gathering**

**Task Description**

The first thing that our project team engaged in was gathering requirements from end users. The primary goal of this phase was to identify what constitutes as a success for this particular phase of the data warehouse project. In particular, end user reporting / analysis requirements were identified, and the project team was to spend the remaining period of time trying to satisfy these requirements.

Associated with the identification of user requirements was a more concrete definition of other details such as hardware sizing information, training requirements, data source identification, and most importantly, a concrete project plan indicating the finishing date of the data warehousing project.

We also looked at some sample reports that the end users will need.

**Deliverables:**

* A list of reports / cubes to be delivered to the end users by the end of this current phase.
* A updated project plan that clearly identifies resource loads and milestone delivery dates.

**2. Physical Environment Setup**

**Task Description**

Once the requirements were somewhat clear, it was necessary to set up the physical servers and databases. At a minimum, it was necessary to set up a development environment and a production environment.

It was not enough to simply have different physical environments set up. The different processes (such as ETL, OLAP Cube, and reporting) also needed to be set up properly for each environment.

We saw it was best for the different environments to use distinct application and database servers. In other words, the development environment would have its own application server and database servers, and the production environment will have its own set of application and database servers.

Having different environments is very important for the following reasons:

* All changes can be tested and QA'd first without affecting the production environment.
* Development and QA can occur during the time users are accessing the data warehouse.
* When there is any question about the data, having separate environment(s) will allow the data warehousing team to examine the data without impacting the production environment.

**Time Requirement**

Getting the servers and databases ready took less than 1 week.

**Deliverables**

* Hardware / Software setup document for all of the environments, including hardware specifications, and scripts / settings for the software.

**3. Data Modeling – Conceptual Model -> Logical Model -> Physical Model**

**Task Description**

This was a very important step in the data warehousing project. In fact, foundation of the data warehousing system was the data model. Our good data model allowed the data warehousing system to grow easily, as well as allowing for good performance.

The logical data model was built based on user requirements, and then it was translated into the physical data model. Part of the data modeling exercise was the identification of data sources.

**Deliverables**

* Identification of data sources.
* Logical data model.
* Physical data model.

**4. ETL**

**Task Description**

The **ETL** (Extraction, Transformation, Loading) process took the longest to develop, and easily took up to 50% of the data warehouse implementation cycle or. The reason for this was that it takes time to get the source data, understand the necessary columns, understand the rules, and understand the logical and physical data models.

**Deliverables**

* Data Mapping Document
* ETL Script / ETL Package in the ETL tool
* Query Optimization

Having long-running SQL queries not only consumed system resources that made the server and application run slowly, but also may lead to table locking and data corruption issues. So, query optimization was a very important task. Here we applied some scripts to make our algorithms more efficient.

Some of the query optimizing strategy we followed:

* + **Use Index**  
    Using an index is the first strategy one should use to speed up a query.
* **Aggregate Table**  
  Pre-populating tables at higher levels so less amount of data need to be parsed.
* **Vertical Partitioning**  
  Partition the table by columns. This strategy decreases the amount of data a SQL query needs to process.
* **Horizontal Partitioning**  
  Partition the table by time. This strategy decreases the amount of data a SQL query needs to process.
* **Denormalization**  
  The process of denormalization combines multiple tables into a single table. This speeds up query performance because fewer table joins are needed.
* **Server Tuning**  
  Each server has its own parameters, and often tuning server parameters so that it can fully take advantage of the hardware resources can significantly speed up query performance.

**5.Front End Design**

**Task Description**

Regardless of the strength of the OLAP engine and the integrity of the data, if the users cannot visualize the reports, the data warehouse brings zero value to them. Hence front end development was an important part of a data warehousing initiative.

The most important thing is that the reports needed to be delivered over the web, so the only thing that the user needs is the standard browser. It is no longer desirable nor feasible to have the IT department doing program installations on end users desktops just so that they can view reports

The front-end options ranged from an internal front-end development using scripting languages such as ASP, PHP, or Perl, to off-the-shelf products such as Seagate Crystal Reports, to the more higher-level products such as Actuate

Another area we were concerned with was the complexity of the reporting tool. For example, How regular reports needed to be published, the specific formatting requirements, the GUI interface so that each user can customize her reports etc

**Deliverables**

* Front End Deployment Documentation

**6. Report Development**

**Task Description**

Report specification typically came directly from the requirements phase. To the end user, the only direct touchpoint he or she has with the data warehousing system is the reports they see. So, report development, although not as time consuming as some of the other steps such as ETL and data modeling, nevertheless played a very important role in determining the success of our project. Factors that we considered in report development:

* **User customization**: Do users need to be able to select their own metrics? And how do users need to be able to filter the information? The report development process needed to take these factors into consideration so that users can get the information they need in the shortest amount of time possible.
* **Report delivery**: What report delivery methods are needed? In addition to delivering the report to the web front end, other possibilities include delivery via email, via text messaging, or in some form of spreadsheet.
* **Access privileges**: Special attention needs to be paid to who has what access to what information. A report can show 8 metrics covering the entire Communication Excellence program to the company CEO, while the same report may only show 5 of the metrics covering only a single member of the program.

**7. Performance Tuning**

**Task Description**

There were three major areas where we undertook performance tuning:

* ETL - Given that the data load was a very time-consuming process and that data warehousing-related batch jobs were typically of lower priority,that meant that the window for data loading was not very long. A data warehousing system that has its ETL process finishing right on-time is going to have a lot of problems simply because often the jobs do not get started on-time due to factors that is beyond the control of the data warehousing team. As a result, it was always an excellent idea for our group to tune the ETL process as much as possible.
* Query Processing
* Report Delivery - It was possible that end users are experiencing significant delays in receiving their reports due to factors other than the query performance. For example, network traffic, server setup, and even the way that the front-end was built sometimes play significant roles

**8. Quality Assurance**

**Task Description**

Once a first prototype of our data warehouse was completed we presented to our supervisors to check that we met their expectations, and later we incorporated their advices in our project.

Lessons Learnt

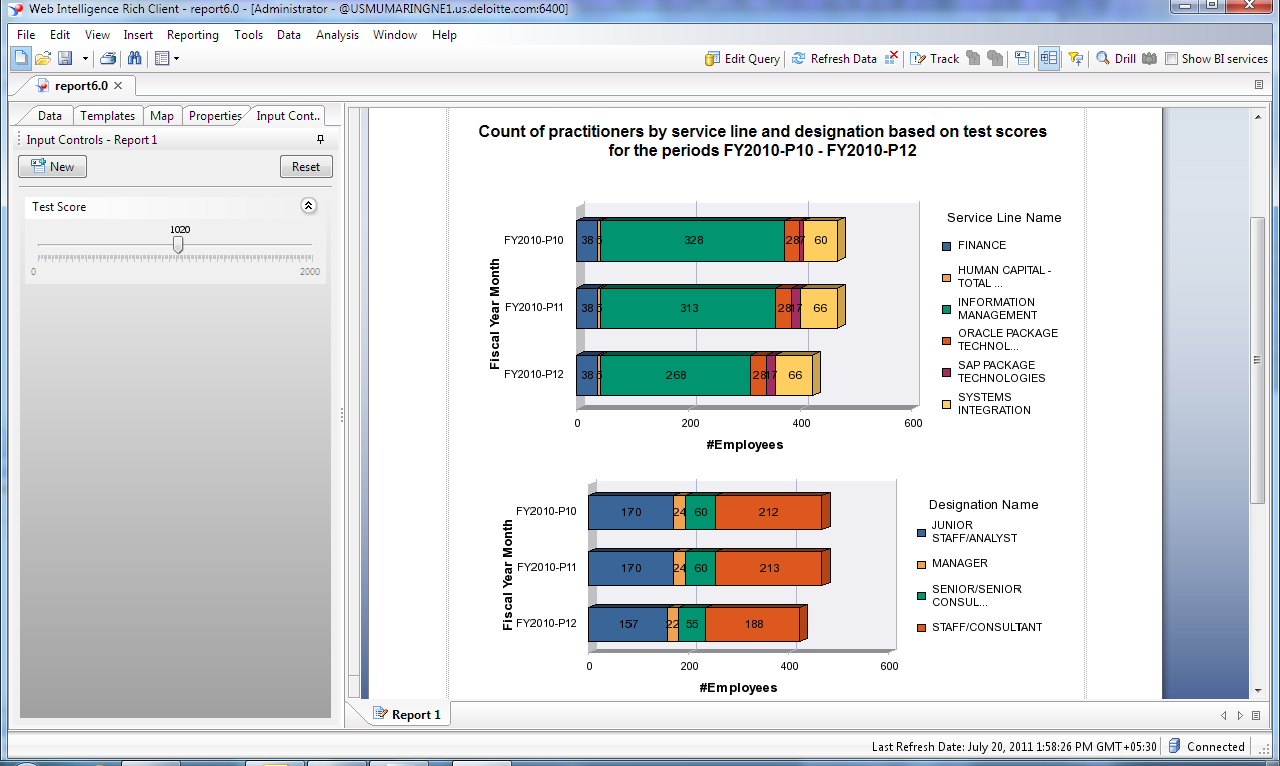
* Initially we made the ETL process more complicated than necessary. In ETL design, the primary goal should be to optimize load speed without sacrificing on quality. There was a tendency in the team to cover all possible future uses, whether they are practical or not. We slowly learnt how to identify what is practical and whats not, and how to set objectives that are achievable in the two months we had.
* There was a tendency to finish the script work quickly because it was not so interesting. This could have proved suicidal to the project because end users will usually tolerate less formatting, longer time to run reports, less functionality (slicing and dicing), or fewer delivered reports; one thing that they will not tolerate is wrong information.
* To save on time and resources, our teams initially decided to use only a single database and a single server for the different environments. Environment separation is achieved by either a directory structure or setting up distinct instances of the database. This turned out to be problematic for the following reasons:

1. Sometimes it is possible that the server needs to be rebooted for the development environment. Having a separate development environment prevented the production environment from being impacted by this.

2. There might have be interference when having different database environments on a single box. For example, having multiple long queries running on the test database could affect the performance on the employee database.

* Even after the project was completed, the exact definitions of the report had to be communicated to the users. Otherwise, user interpretation of the report could have been erroneous.

**Example of a report generated**:



**Tools used**

* MYSQL 5.0
* SAP Business Objects Data Services Designer
* SAP Business Objects Universe Designer
* SAP Business Objects Web Rich Client
* Info View

**Reference Material**

* Data Warehousing Fundamentals by P.Ponniah
* OCA Oracle Database 11g: SQL Fundamentals 1 by J.Watson

Summary

The project has been a great learning experience for me. I learnt the essentials of planning for a data warehouse and found out how to distinguish between data warehouse projects and OLTP system projects. I also gained knowledge of how to adapt the life cycle approach for a data warehouse project. Apart from the technical nitty-gritty’s, we discussed project team organization, roles, and responsibilities .

According to a 2009 Gartner paper, following developments are predicted in the business intelligence market:   
1. More than 35 percent of the top 5,000 global companies will regularly fail to make insightful decisions about significant changes in their business and markets because of lack of information, processes, and tools, through 2012.  
2. By 2012, business units will control at least 40 percent of the total budget for business intelligence.

Our project helped not only to improve the Communication Excellence program of Deloitte, but also, introduce us to key aspects in this field.