Data Warehouse Creation for Preparing an Electricity Statistics Dashboard

[1] Shivani Khedikar, [2] Purva Kirolikar, [3] Supriya Thombre

[1, 2, 3] Department of Computer Technology, YCCE
Nagpur- 441110, Maharashtra, India

Abstract - The 21st century is making use of electricity so extensively that it has almost changed the face of the earth. To generate and harness electricity on a large scale means the development of machinery capable of doing so. An essential strategy for meeting the energy challenge is to concentrate on the generation and use of electricity. One suggested technique to assist in analysis is data warehousing and data mining. Use of the Data warehouse and Business Intelligence Systems for the betterment of the electricity related problems which are lamentably worst specially in rural areas, would enable the respective organizations to deal with the appropriate problems. It would enforce better decision -making. This paper focuses on building data warehouse that will further be followed by dashboard creation by applying data mining techniques that would enable Maharashtra State Electricity Board to analyze electricity trends and take steps accordingly to improve its performance.

Keywords - *Data Warehouse, Business Intelligence, Logical Data Model, Physical Data Model*

1. Introduction

Maharashtra State Electricity Board (MSEB) is a state -owned electricity regulation board operating within the state of Maharashtra in India. The MSEB was formed on June 20, 1960 under Section 5 of the Electricity Act, 1948. As of 1998, it was the second largest electricity generating utility in India after National Thermal Power Corporation.[1] Following figure shows the various functionalities of the MSEB.

Electric power systems are critical infrastructures for modern society. They span huge geographical areas, and comprise thousands of measurement and monitoring systems continuously collecting various data such as voltage and current, power lows, line temperatures, plus data relating to the stating of devices. The increased utilization of information and communication technologies make available a wealth of data which can be used to gain a deeper understanding of the dynamics of the process as well as an opportunity to use online data decision support.

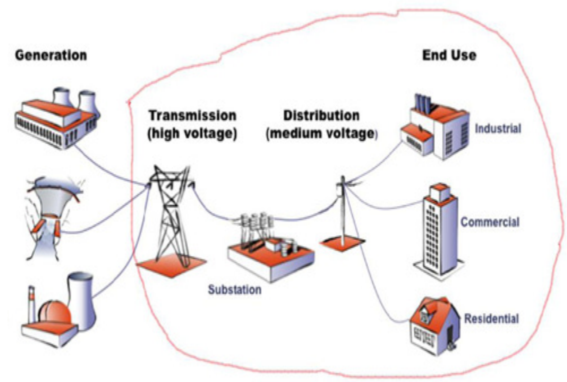


Figure 1: MSEB functionalities

When properly trained up, pattern recognition algorithms in data mining can detect deviation from the regular data, which may be useful for triggering alarms and messages (classification) that provide important information to the operator. The mining and off-line analysis of historical data can provide knowledge that can be subsequently implemented online. This knowledge might be related to fault analysis, energy consumption, analysis of distributed generation, or load pattern.[4]

Data mining is an essential step in the process of knowledge discovery in databases in which intelligent methods are applied in order to extract patterns[2].Data mining is a process that uses a variety of data analysis tools to identify hidden patterns and relationships within data.

The idea that drives us to use this lamentably awesome Data Warehousing (DW) and Business Intelligence (BI) tools in the field of Electricity and Power field is that there are three reasons that are as follows : Firstly , the population Explosion drives the use of Electricity on a large scale. Secondly, electrification of everything as a part of lavish lifestyle and thirdly, expectation inflation. Inspite of these factors, this is the field that has not use BI techniques much .

1.1 Objectives

One of the many objectives is to enable MSEB to analyze the electricity trends and take steps accordingly to improve its performance. This primary objective would solve the problem of load shedding to a large extent and at the same time it would free the people who find themselves shackled due to high power cutoff specially in rural areas. Secondly, interested to probe into details in a more interactive way and easily analyze the exceptions by slicing and dicing.

2. Problem Definition

To develop a data warehouse that would help in dashboard creation which would enable to analyze the generation, exchanges, power purchase, changing demand of electricity, the demand met, load shedding and other such important parameters on yearly, quarterly and monthly basis across the various zones in Maharashtra.

2.1 Scope

This project is all about the application of the DW and BI to the Electric Power Statistics. Today , BI has intruded into many day-to-day aspects of life. Since , this is the field still untouched by BI , we have planned to explore it more in this field for the betterment of the Maharashtra State as a whole. Dashboard Development is primarily being focused. This Dashboard will showcase the Generation volume, Exchanges, Overload Capacity, Power Purchase, State Demand and Generating power Plants parameters for the past 5 years (2009- 2013). Apart from the detailed view, the information will be available on a glance in summarized way as well. This application will help the administrative and managerial bodies of MSEB to take appropriate steps accordingly in the near future so as to free citizens from the shackles of power-cut and load shedding.

3. Literature Survey

The Data Warehouse (DW) and Business Intelligence (BI) field caters to the need of variety of applications like IPL statistics ,demographic statistics, Stocks and shares and many more. Different applications are interested in different ways of analyzing their performances . Hence, the performance evaluation of a BI tools has to be observed in terms of its impact on the performance of the applications that use it.

Use of the Data warehouse and Business Intelligence Systems for the betterment of the electricity related problems which are lamentably worst specially in rural areas, would enable the respective organizations to deal with the data quality and data definition problems. It would enforce better decision –making.

3.1 Data Warehouse

A Data Warehouse (DW) can be considered as a repository providing access to data from source systems (operational databases, mainframes, flat files, etc.) that has been extracted, transformed and loaded into a database that is optimized for analysis. The information is subject oriented, recorded over time and may be stored at various levels of summarization

A data warehouse is a copy of transactional data specifically structured for querying and analysis.

3.1.1 Goals of Data Warehouse

- Make an organisation's information easily accessible
- Present the organisation's information consistently
- Should be adaptive and resilient to change
- Should be a secure bastion that protects our information assets
- Serve as the foundation for improved decision making

3.1.2 Data Warehouse Architecture

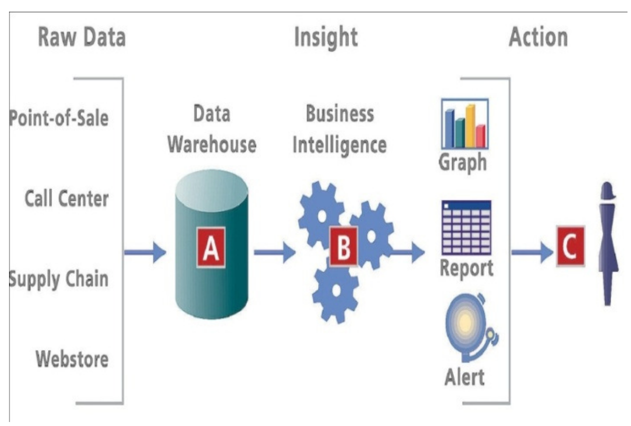


Figure 2 : Data Warehouse Architecture

Raw Data: Data from different OLTP sources is pulled and is put into the data warehouse.

Insight: Data Warehouse stores the reformatted or transformed data. Business Intelligence generates the reports, dashboards etc.

Action: Act on the insight provided by BI tools by reallocating resources

IJCSN International Journal of Computer Science and Network, Volume 2, Issue 6, December 2013
ISSN (Online) : 2277-5420 www.IJCSN.org

Data Warehouse Architecture (Detailed View)

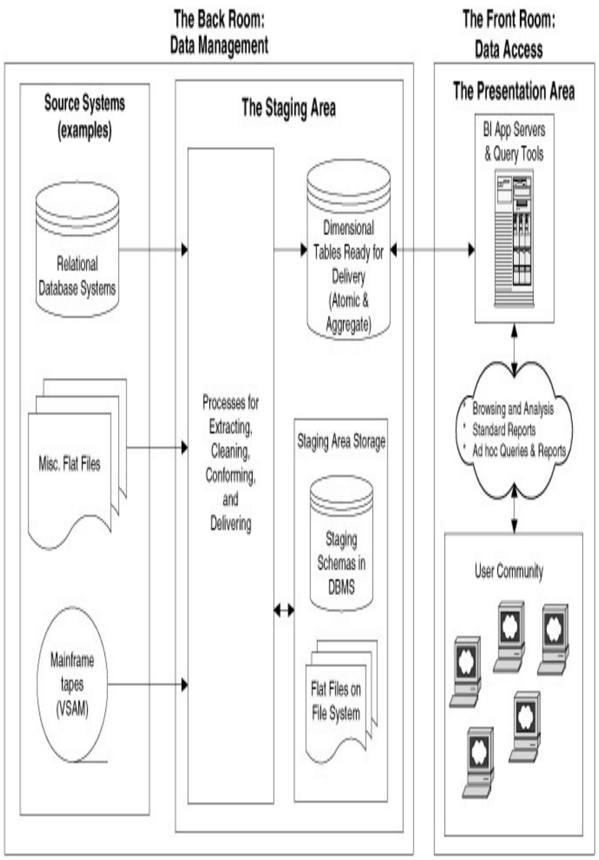


Figure 3 :Data Warehouse Architecture (Detailed View)

- *Source System***:** Provide the data, this can include databases (Oracle, SQL Server, Sybase etc) or files or spreadsheets

- *Staging Area*: In staging area data is extracted from the source **OLTP systems** prior to being published to the "warehouse". Data in the staging area is NOT visible to end users for queries, reports or analysis of any kind.

- *Presentation Layer:* The data from data warehouse can be presented to the user in different ways. Ex: Reports, documents, dashboards etc.

3.1.3 Towards designing of a Data Warehouse

To design a Data Warehouse, first we have to follow the following sequence of steps as shown in the figure below

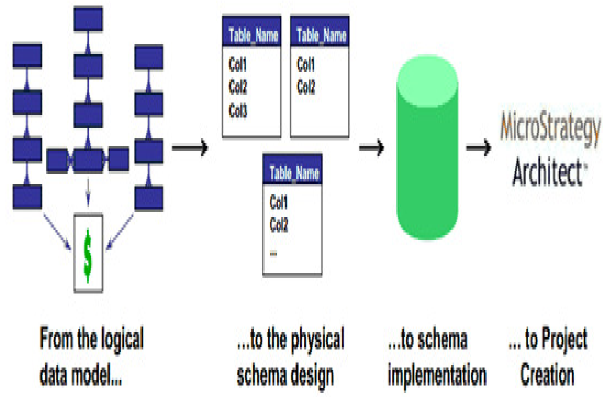


Figure 4 : To build a Data Warehouse

3.1.3.1 The Logical Data Model

In computing and more specifically systems engineering, a logical data model (LDM) is a representation of an organization's data, organized in terms of entities and relationships and is independent of any particular data management technology[7]
One technique you can use to model organization's logical information requirements is entity-relationship modeling. Entity-relationship modeling involves identifying the things of importance (entities), the properties of these things (attributes), and how they are related to one another (relationships).

The process of logical design involves arranging data into a series of logical relationships called entities and attributes. An entity represents a chunk of information. In relational databases, an entity often maps to a table. An attribute is a component of an entity that helps define the uniqueness of the entity. In relational databases, an attribute maps to a column.

While entity-relationship diagramming has traditionally been associated with highly normalized models such as OLTP applications, the technique is still useful for data warehouse design in the form of dimensional modeling. In dimensional modeling, instead of seeking to discover atomic units of information (such as entities and attributes) and all of the relationships between them, it is good to identify which information belongs to a central fact table and which information belongs to its associated dimension tables. Business subjects or fields of data should be identified, relationships between business subjects should be defined, and the attributes for each subject should be named.

IJCSN International Journal of Computer Science and Network, Volume 2, Issue 6, December 2013
ISSN (Online) : 2277-5420 www.IJCSN.org

Logical design should result in

1. A set of entities and attributes corresponding to fact tables and dimension tables
2. A model of operational data from source into subject-oriented information in target data warehouse schema.

Logical data models represent the abstract structure of a domain of information. Once validated and approved, the logical data model can become the basis of a physical data model ano rm the design of a database[7].Following figure illustrates an example of the logical data model of Dollar Sales, Quantity Sold, and Profit.

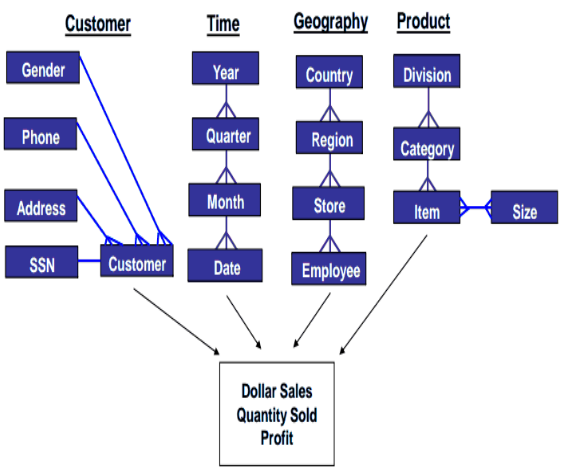


Figure 5: An example of the Logical Data Model

To understand the Logical Data Model more clearly let us take an overview of the phases that the Logical Data Model has to go through for it becoming a report. The following figure pictorially illustrates briefly the further steps after building a Logical Data Model.

As shown below, one column in the table becomes one attribute (say Day). On the other hand, one or more columns in the table become one fact (say Revenue) Both the attributes and fact are schema objects. However , when a function is applied on a fact , it becomes a metric. A metric is an Application Object. These Schema objects and Application objects together contribute to the creation of a final report.

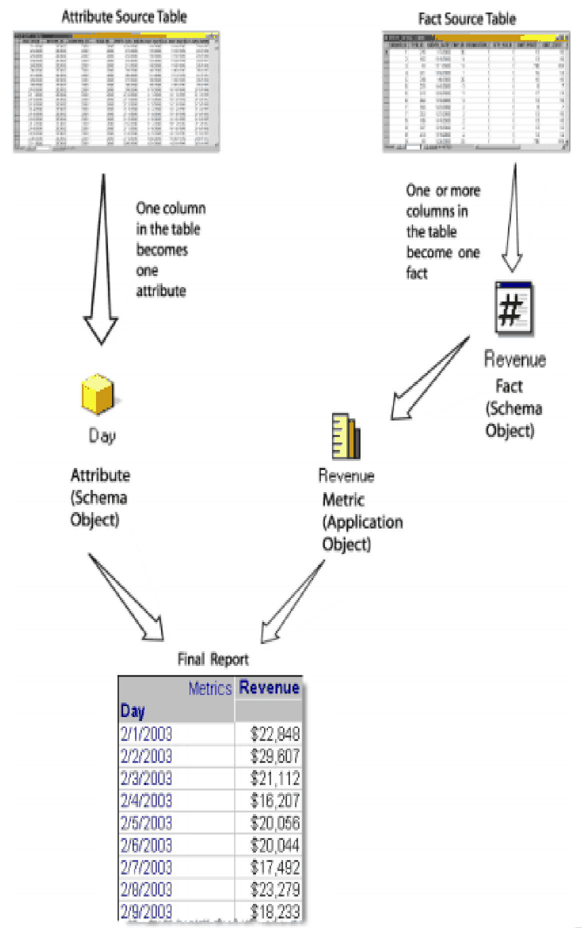


Figure 6 : From Data Warehouse to a report

Attributes: Attributes are represented by columns in a table in the data warehouse. They are the elements that further define, clarify, and or organize a business entity. They allow you to answer questions about a fact and provide a context in which to report facts. For example, if you store your sales data at a day level, a Month attribute would allow you to see the same sales data summarized at the month level.

Attribute elements: In the Data Warehouse, Attributes are represented by columns in a table, and attribute elements are the values in the rows of the tables.

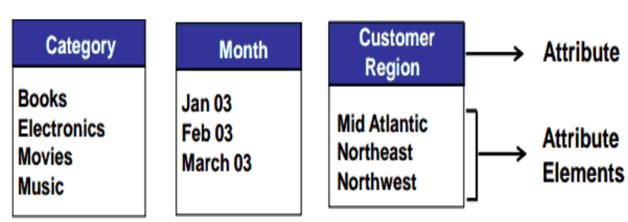


Figure 7 : Attribute and Attribute Elements

In Micro strategy environment, attributes are used to define the report, attribute elements are the report results displayed in the rows or columns of an executed report.

Metrics	Revenue
Category	
Books	$64,221
Electronics	$11,301,654
Movies	$2,778,225
Music	$2,792,334

Figure 8: Report in Microstrategy environment that is defined by the attributes

Fact : Facts can be thought of as business measurements, data, or variables, that are typically numeric and suitable for aggregation. Facts allow you to access data stored in the data warehouse and they form the basis for the majority of analyses and reports. Facts relate numeric data values to the MicroStrategy reporting environment. Facts form the basis for metrics. Sales, Inventory and Account Balance are some examples of facts.

To summarize, a Logical Data Model (LDM) can be referred to as a logical arrangement of data depicting the flow and structure of data in business environment. It is a graphical representation of facts, attributes, and hierarchies.

To build a Logical Data Model

Three things that must be taken care of while building a Logical Data Model are user requirements, existing systems/source data, and technical considerationsTo build a Logical Data Model, the following four steps should be followed:

a. Identify facts
b. Identify attributes
c. Determine the relationships between attributes
d. Define hierarchies

These steps are detailed as follows:

a. Identify facts : In the example cosidered, we observe that the Dollar Sales is the data that can be aggregated at Item level, Date Level as well as Store Level.Hence, we may consider 'Dollar Sales' as a fact. Similarly, Inventory data can be aggregated on Item level, Month Level as well as Region Level. Hence, 'Inventory' may also be identified as a fact.Similarly 'Price' is another fact in the example.

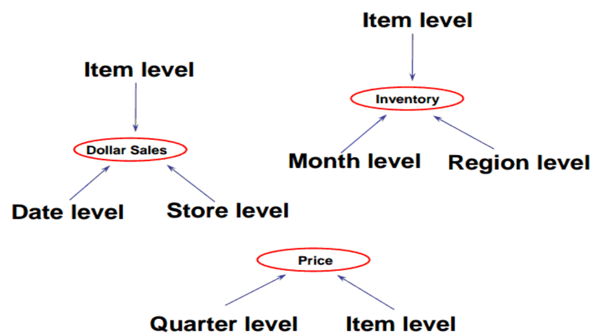


Figure 9: Identifying facts

b. Identify Attributes : In the example considered, we find that 'Item' , 'Date', and 'Store' are the elements that further define, clarify, and or organize a business entity. Hence, they can be considered as the attributes of 'Dollar Sales'. Similarly 'State' is such an element that further clarify the location of the 'Store'. Hence, 'State' can be considered as an attribute of the 'Store'

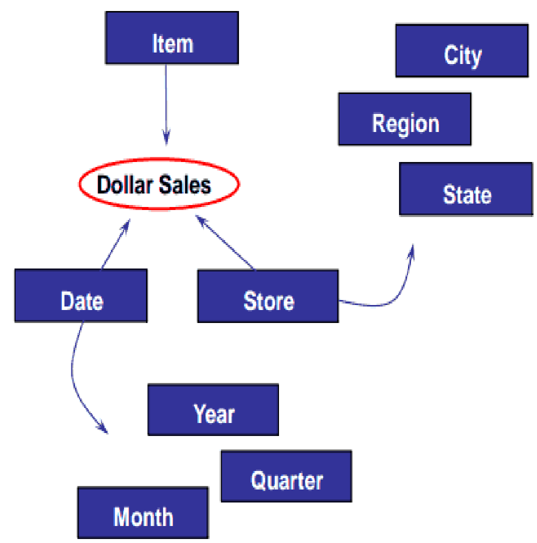


Figure 10 : Identifying the attributes

c. Determine the relationships between attributes : Keeping in view all the attributes, we observe that the attributes 'Year', 'Quarter', 'Month', and 'Date' are somehow related. As we know that, a year has many quarters (four), hence, we can say that there exists one-to-many relationship between 'Year' and 'Quarter'. Similary, there also exists one-to-many relationships between 'Quarter' and 'Month', and 'Month' and 'Date'. Also, a state may have many stores and so is the case with a city. Hence, there also exists one-to-many relationship between between 'State' and 'Store', and 'City' and 'Store'. These relationships between the different attributes are shown in the following figure.

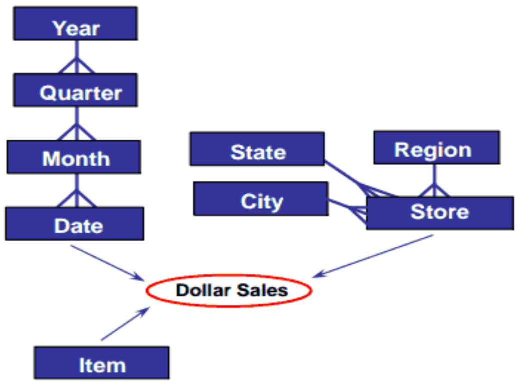


Figure 11: Determining the relationships between attributes

d. Define hierarchies : After having identified the relationships between the attributes, we also observe some hierarchies between the attributes. This hierarchy in particular, contains two hierarchies: Time hierarchy and Geography hierarchy

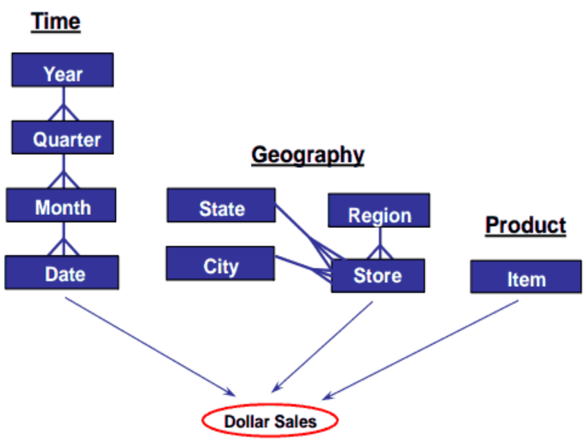


Figure 12 : Defining hierarchies

The four steps discussed above are the steps to create a basic LDM. However the LDM can further be enhanced by adding unique identifiers, cardinalities and ratios, and attribute forms.

3.1.3.2 The Physical Warehouse Schema

The second stage of building a Data Warehouse is to create the Physical Warehouse Schema. The physical warehouse schema is based on the logical data model. It is detailed graphic representation of your business data. It organizes the logical model in a method that makes sense from a database perspective[8]. The Physical Warehouse Schema describes how your data is stored in the data warehouse. While the LDM tells you what facts and attributes to create, the physical warehouse schema tells you where the underlying data for those objects is stored. The physical Warehouse Schema describes how your data is stored in the data Warehouse.[8] Two key components make up the physical Warehouse Schema-Tables and Columns. Columns and tables in the physical Warehouse Schema represent facts and attributes from the LDM. The rows in a table represent attribute elements and fact data[8] There are three types of columns: ID columns, Description columns, and Fact columns

Also there are three types of tables as follows :

a. Lookup Table
b. Relate Table
c. Fact Table

A brief overview of the tables is given below:

a. Lookup Table : A lookup table can contain information about one or more attributes.There are two types of Lookup tables.: Homogenous Lookup Tables and Heterogenous Lookup Tables.

A lookup table is also called as a dimension table. A dimension is a structure, often composed of one or more hierarchies, that categorizes data. Dimensional attributes help to describe the dimensional value. They are normally descriptive, textual values. Several distinct dimensions, combined with facts, enable you to answer business questions. Commonly used dimensions are customers, products, and time.

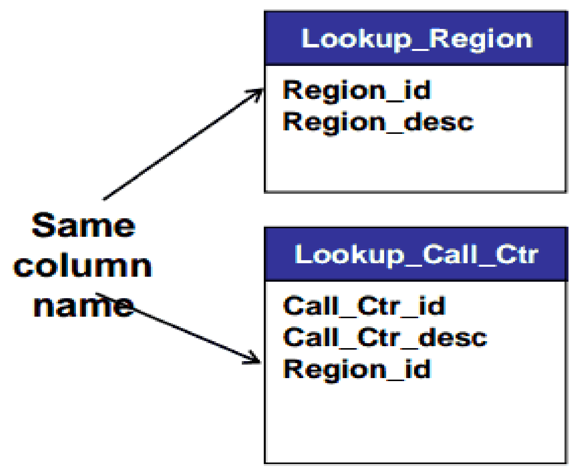


Figure 13: Homogenous Lookup Tables

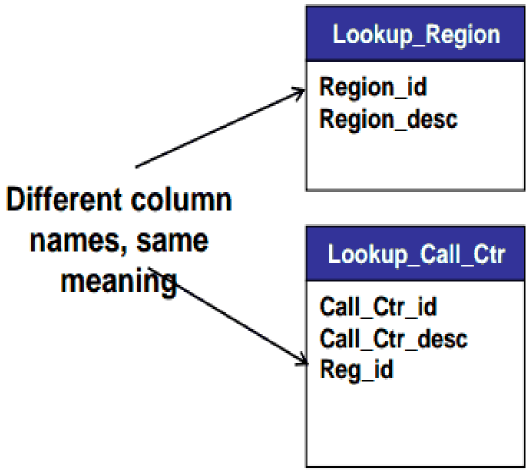


Figure 14: Heterogenous Lookup Tables

Dimension data is typically collected at the lowest level of detail and then aggregated into higher level totals that are more useful for analysis. These natural rollups or aggregations within a dimension table are called hierarchies.

b. Relate Table: A relate table shows the relationships that exist between different attributes in the schema. These can be many-to-one , one-o-one, one-to-many, or many-to-many.

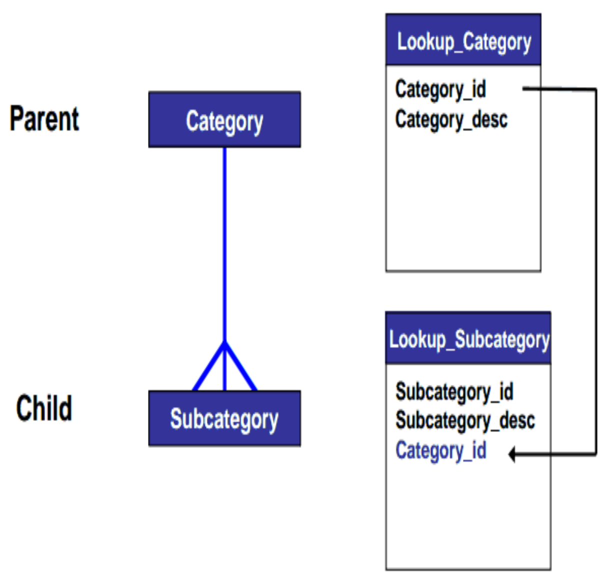


Figure 15 Combined Relate Table and Lookup Table

c. Fact Table: In data warehousing, a fact table consists of the measurements, metrics or facts of a business process. It is located at the center of a star schema or a snowflake schema surrounded by dimension tables. A fact table typically has two types of columns: those that contain facts and those that are foreign keys to dimension tables. The primary key of a fact table is usually a composite key that is made up of all of its foreign keys[10]

A fact table typically has two types of columns: those that contain numeric facts (often called measurements), and those that are foreign keys to dimension tables.Fact tables that contain aggregated facts are often called *summary tables.* Though most facts are additive, they can also be semi-additive or non-additive. Additive facts can be aggregated by simple arithmetical addition (example: sales).Non-additive facts cannot be added at all (example: averages)Semi-additive facts can be aggregated along some of the dimensions and not along others.

From a modeling standpoint, the primary key of the fact table is usually a composite key that is made up of all of its foreign keys.

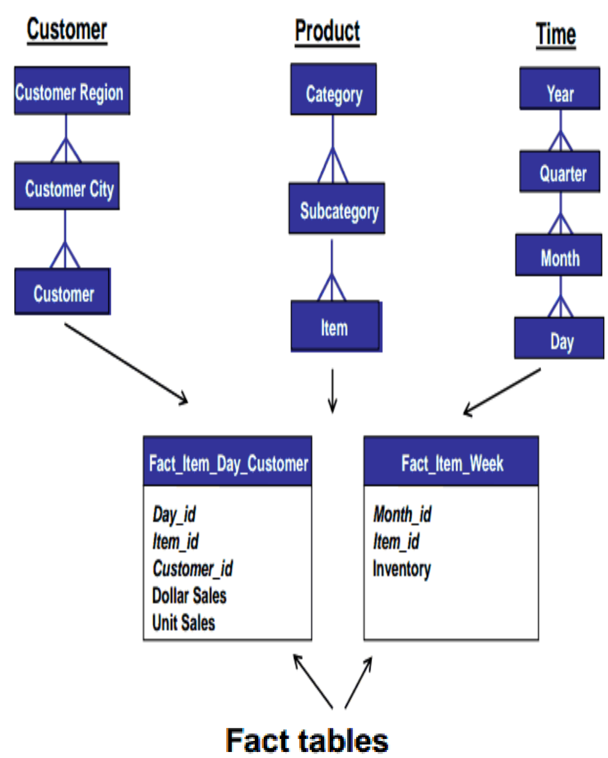


Figure 16 Fact Tables

IJCSN International Journal of Computer Science and Network, Volume 2, Issue 6, December 2013
ISSN (Online) : 2277-5420 www.IJCSN.org

3.1.3.3 Normalization

Database normalization is the process of organizing the fields and tables of a relational database to minimize redundancy and dependency. Normalization usually involves dividing large tables into smaller (and less redundant) tables and defining relationships between them. The objective is to isolate data so that additions, deletions, and modifications of a field can be made in just one table and then propagated through the rest of the database using the defined relationships[11]

First normal form (1NF) sets the very basic rules for an organized database:

- Eliminate duplicative columns from the same table
- Create separate tables for each group of related data and identify each row with a unique column or set of columns (the primary key).

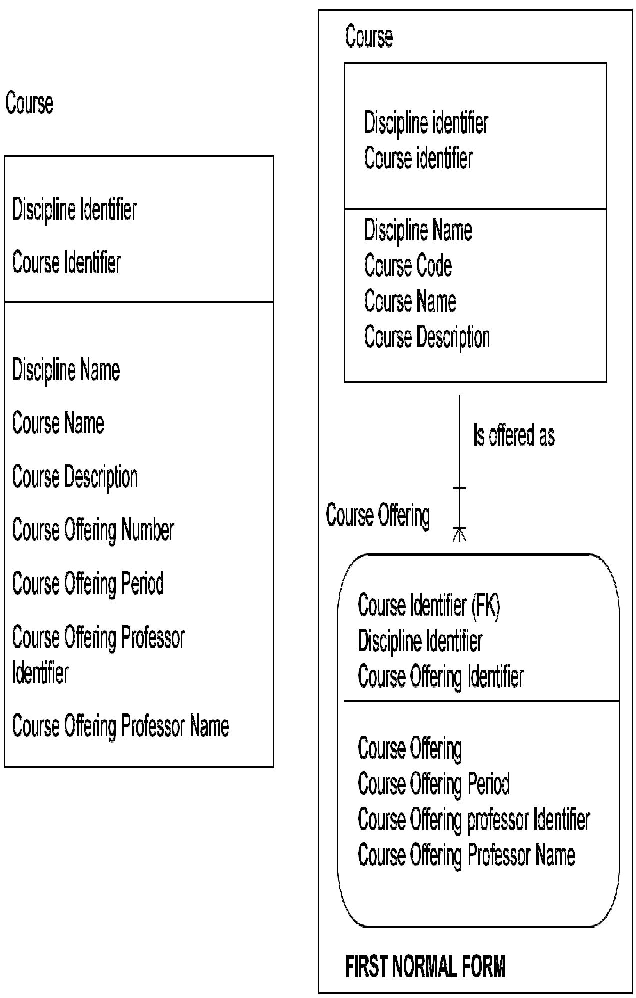


Figure 17: First Normal Form (1NF)

Second normal form (2NF) further addresses the concept of removing duplicative data:

- Meet all the requirements of the first normal form.
- Remove subsets of data that apply to multiple rows of a table and place them in separate tables.
- Create relationships between these new tables and their predecessors through the use of foreign keys.

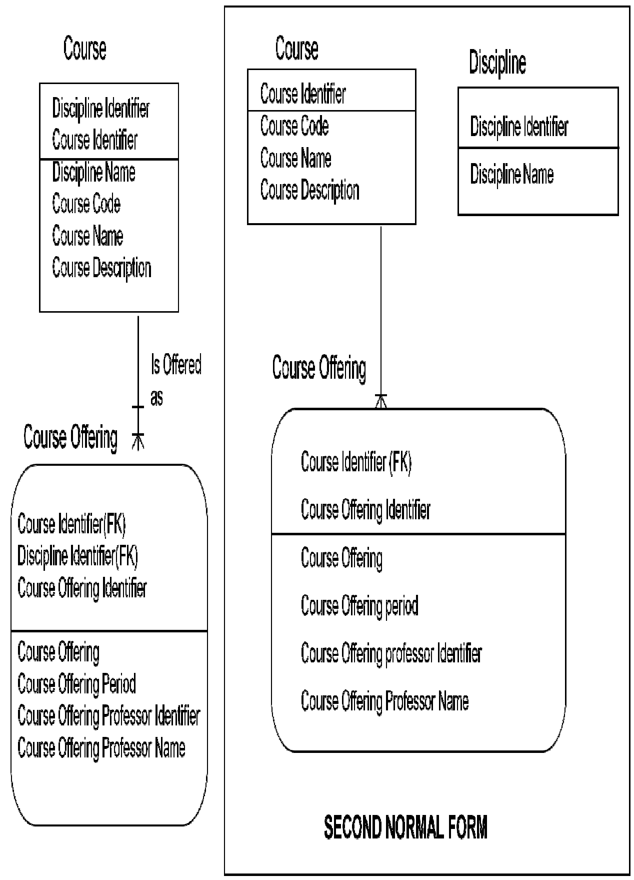


Figure 18 : Second Normal Form (2NF)

Third normal form (3NF) goes one large step further:

- Meet all the requirements of the second normal form.
- Remove columns that are not dependent upon the primary key.

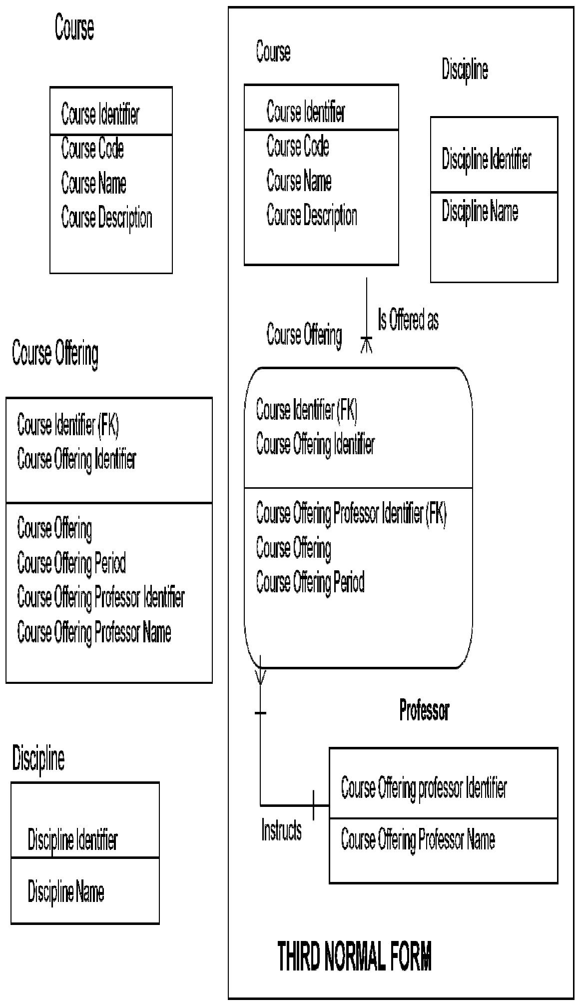


Figure 19 : Third Normal Form (3NF)

Based upon the concept of normalization, there are three types of schemas:

i. Completely Normalized Schema
ii. Moderately Denormalized Schema
iii. Completely Denormalized Schema
Following figure shows an example of a Completely Normalized Schema

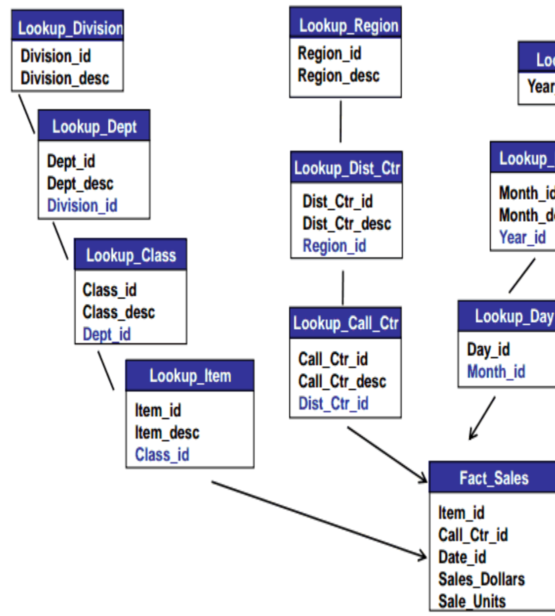
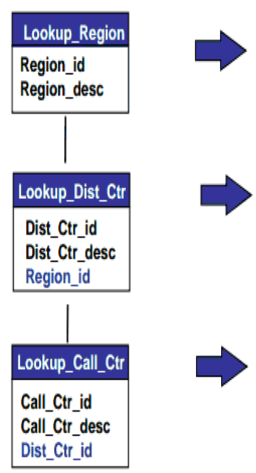


Figure 20 : Completely Normalized Schema

Following are the Completely Normalized Schema Tables:



REGION_ID	REGION_DESC
1	Northeast
2	South

DIST_CTR_ID	DIST_CTR_DESC	REGION_ID
10	Mid-Atlantic	1
20	New England	1
30	Carolinas	2
40	Deep South	2

CALL_CTR_ID	CALL_CTR_DESC	DIST_CTR_ID
101	Boston	20
102	Greenwich	20
103	Providence	20
104	Baltimore	10
105	Philadelphia	10
106	Charlotte	30
107	Durham	30
108	Greenville	30
109	Atlanta	40
110	Fayetteville	40

Figure 21 : Completely Normalized Schema Tables

Following figure shows an example of moderately normalized schema and moderately denormalized schema tables

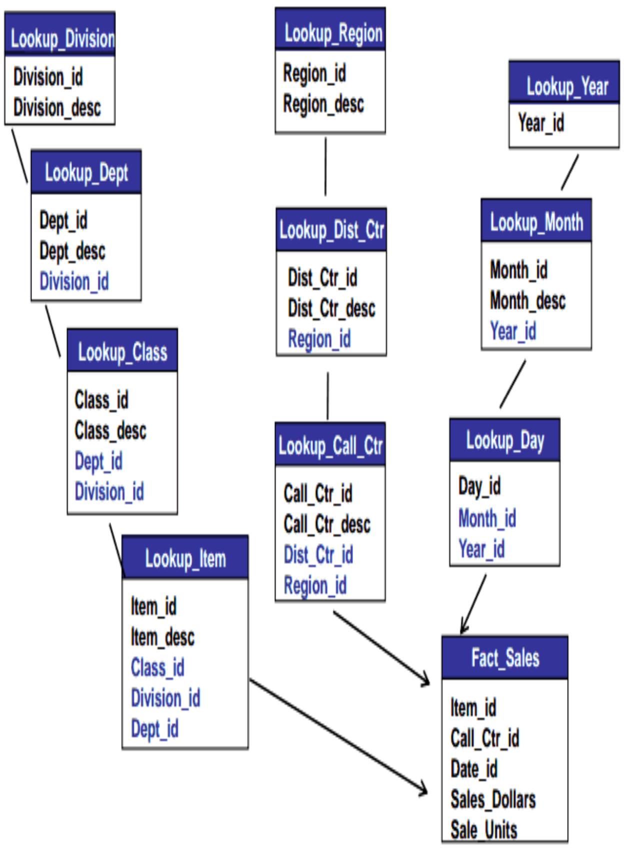
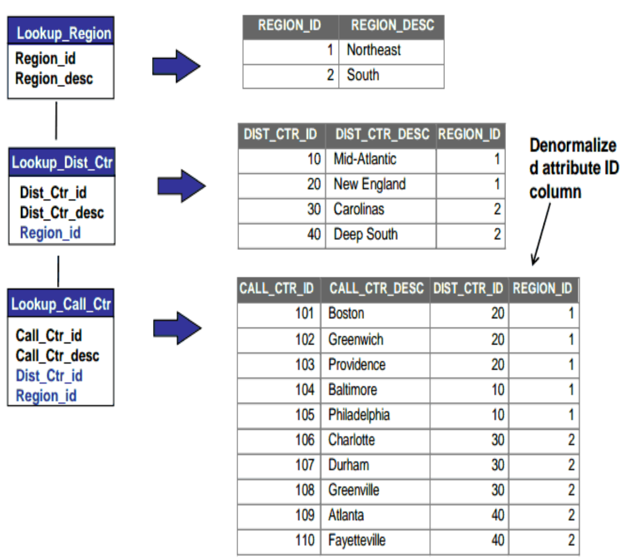


Figure 22 : Moderately Denormalized Schema



REGION_ID	REGION_DESC
1	Northeast
2	South

DIST_CTR_ID	DIST_CTR_DESC	REGION_ID
10	Mid-Atlantic	1
20	New England	1
30	Carolinas	2
40	Deep South	2

CALL_CTR_ID	CALL_CTR_DESC	DIST_CTR_ID	REGION_ID
101	Boston	20	1
102	Greenwich	20	1
103	Providence	20	1
104	Baltimore	10	1
105	Philadelphia	10	1
106	Charlotte	30	2
107	Durham	30	2
108	Greenville	30	2
109	Atlanta	40	2
110	Fayetteville	40	2

Figure 23 : Moderately denormalized schema tables

Following figure shows an example of a completely denormalized schema:

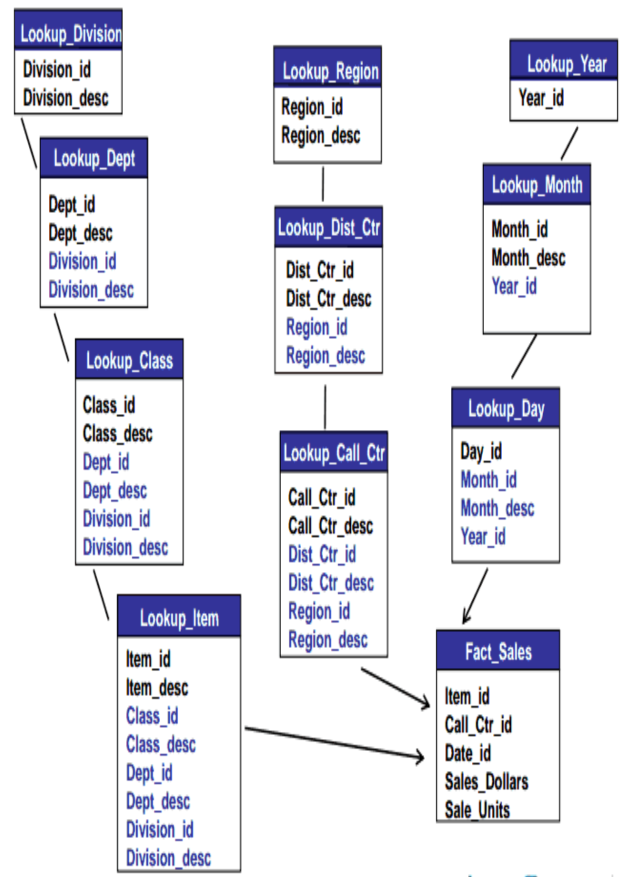


Figure 24 : Completely denormalised schema

After having applied proper normalization techniques, we get a schema (probably a star schema or a consolidated schema). Following is an example of consolidated star schema built from the above data.

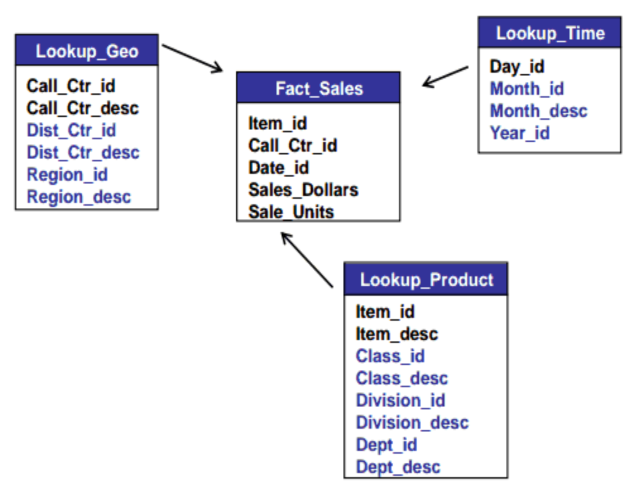


Figure 25 : Consolidated Star schema

3.1.4 Design of Data Warehouse Systems

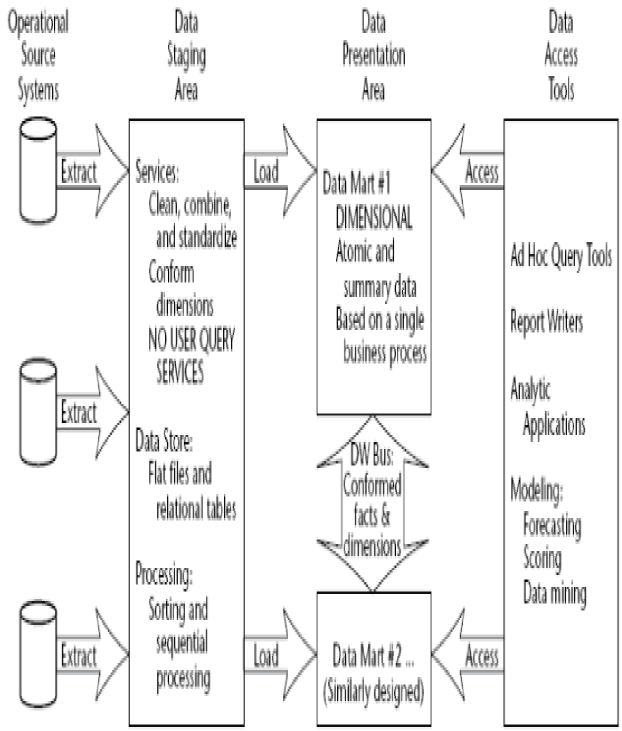


Figure 26: Basic elements of a Data Warehouse Systems

Following are the steps to design a Data Warehouse:

- *Extraction* : Data from operational databases, flat files, external systems is obtained
- *Staging Area* **:** It involves large, sequential bulk operations.
- *Cleansing* **:** Data is checked for missing parts and erroneous values . Also the default values are provided and out of range values are marked.
- *Transformation :* Data is transformed to decision-oriented format .Also the data from several sources is merged, and optimized for querying
- *Aggregation* **:** It ensures whether transactional level details are needed in the DW.
- *Loading into the DW :* It loads large bulk of data rather than SQL INSERTS. However , it requires Fast indexing (and pre-aggregation)
- *Initial Load :* This is First time load of Dimension and Fact tables. Since all DW tables are blank, all statements are insert or bulk insert. It is scheduled once during life time of DW.
- *Incremental Load :* This is recurring Step based on user defined requirement of schedule. Since DW tables have existing data after initial load or incremental load most of the statements are insert and update.
- *Process flow :* Initial Load → Incremental load

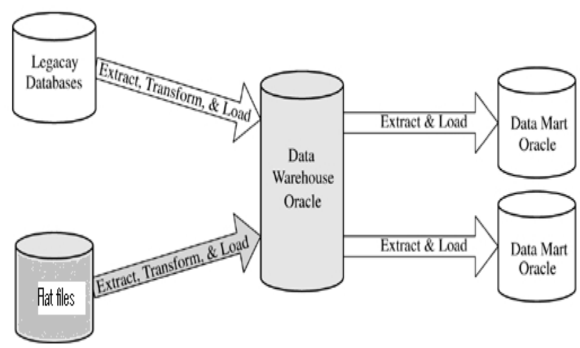


Figure 27 : Building of a Data Warehouse

3.2 Business Intelligence

Business intelligence systems combine operational data with analytical tools to present complex and competitive information to planners and decision makers. The objective is to improve the timeliness and quality of inputs to the decision process. Business Intelligence is used to understand the capabilities available in the firm; the state of the art, trends, and future directions in the markets, the technologies, and the regulatory environment in which the firm competes; and the actions of competitors and the implications of these actions. The emergence of the data warehouse as a repository, advances in data cleansing, increased capabilities of hardware and software, and the emergence of the web architecture all combine to create a richer business intelligence environment than was available previously[5]

Business Intelligence is the process of analyzing data, most often contained in a Data Warehouse and extracting useful insight from it.

BI systems combine data gathering, data storage, and knowledge management with analytical tools to present complex internal and competitive information to planners and decision makers.[5]

End users use BI system to make tactical and strategic decisions about their business.

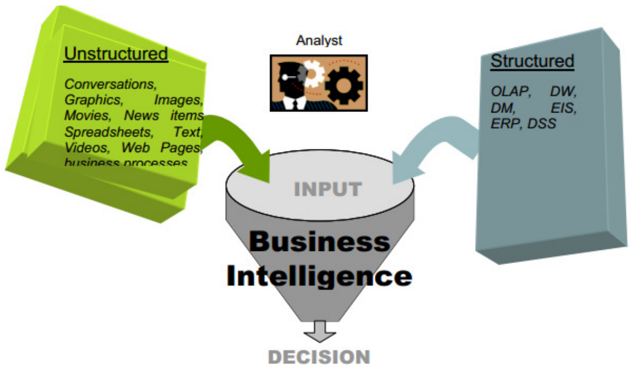


Figure 28: Inputs to Business Intelligence Systems

3.2.1 Architecture for Structured Data

Typical BI architecture for structured data centers on a data warehouse. The data are extracted from operational systems and distributed using Internet browser technologies . The specific data needed for BI are downloaded to a data mart used by planners and executives. Outputs are acquired from routine push of data from the data mart and from response to inquiries from Web users and OLAP analysts. The outputs can take several forms including exception reports, routine reports, and responses to specific request. The outputs are sent whenever parameters are outside pre-specified bounds[6]

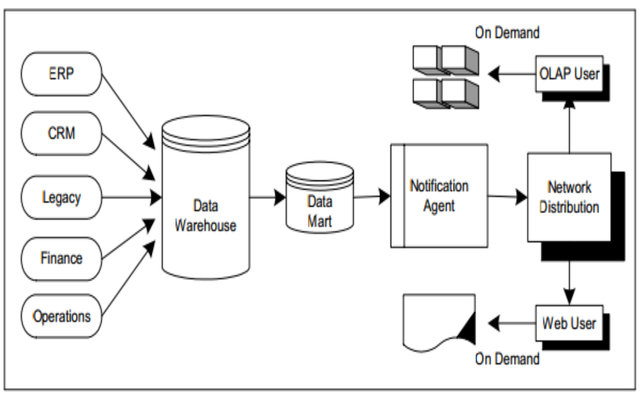


Figure 29: Typical BI Architecture for Structured Data

3.3 The connection between Data Warehousing and Business Intelligence

The Data Warehousing Institute defines business intelligence as:

"The process, technologies, and tools needed to turn data into information, information into knowledge, and knowledge into plans that drive profitable business action. Business intelligence encompasses data warehousing, business analytic tools, and content/ knowledge management. "

The fact that the leading authority on BI calls itself the Data Warehousing Institute highlights the vital role that data warehouses play. Unfortunately, the warehouse adds its value behind the scenes. Its job is to provide data to the high-profile tools and applications with which users interact. This background role can hide its significance, especially since BI solution vendors frequently play down the importance of the data warehouse[12]

Technically, it is not necessary to build a data warehouse in order to create a BI environment. As a result, there are many substandard solutions on the market that avoid the use of data warehouses. Those advocating these solutions often suggest that the absence of a data warehouse is a good thing. They are following the old marketing adage, “If you can’t fix it, feature it.” What countless BI pioneers have discovered, however, is that taking the short cut around data warehousing will put you on a path that leads to lost time and money.

3.4 Data Warehouse based BI solutions provide

Once a data warehouse is in place and populated with valuable data, good things begin to happen. Examples of the many ways in which data warehouse-based BI systems deliver value to their users include:

- *The generation of scheduled reports*. Moving the creation of reports to a BI system increases consistency and accuracy and often reduces cost. A greater number of more useful reports result from the power and capability of BI tools. The creation of reports directly by end users is easier to accomplish in a BI environment.
- *Packaged analytical applications* A growing number of outstanding analytical software applications are coming onto the market. These packages provide predefined reports and metrics that business units can use to measure their performance.
- *Ad hoc reporting and analysis*. Since the data warehouse eliminates the need for BI tools to compete with transaction processing, users can analyze data faster and generate reports more easily. The tools that come with BI systems also tend to vastly improve the analysis function.
- *Dynamic presentation through dashboards*. A growing number of managers want access to an interactive display of up-to-date critical management data. Sophisticated displays that show real time information in creative, highly graphical form are often called dashboards. The name comes from the similarity to the instrument panel on an automobile.
- *Drill down capability*. The leading BI systems all allow users to drill down into the details underlying the summaries in reports and dashboards. The presence of a

data warehouse makes it practical to use this capability as much as needed. Without one, places the burden on a user's access to application data.

- *Support for regulations.* Sarbanes-Oxley and related regulations create demands that transaction systems cannot always support. Well-designed BI solutions can ensure that the necessary data is retained in the data warehouse for as long as is required by law.
- *The creation of metadata*. Data warehouses sit between source applications and BI tools, creating an ideal opportunity to predigest some of the data. Metadata is defined as "data about data." It can include something as simple as an average. Data warehouses can be used to create and store a great deal of metadata of potentially great value.
- *Support for operational processes*. The creation of a sound BI infrastructure is often the best way to meet certain ongoing business needs. The most common example is to facilitate the consolidation of financial results within complex organizations, especially those whose divisions use different software systems. Meeting regulatory reporting requirements is another common situation.
- *Data mining*. The outstanding software tools that can sift through mountains of data and uncover hidden insights work best on a data warehouse.
- *Security*. A data warehouse makes it much easier to provide secure access to those that have a legitimate need to specific data and to exclude others.

This long list of benefits is what makes BI based on a data warehousing an essential management tool for businesses that have reached a certain level of complexity.

3. Developing the High Level Design

3.1 Logical Data Model

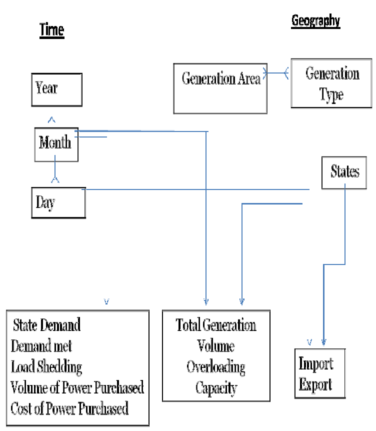


Figure 30: Logical Data Model

3.2 Physical Data Model

A physical data model (or database design) is a representation of a data design which takes into account the facilities and constraints of a given database management system.

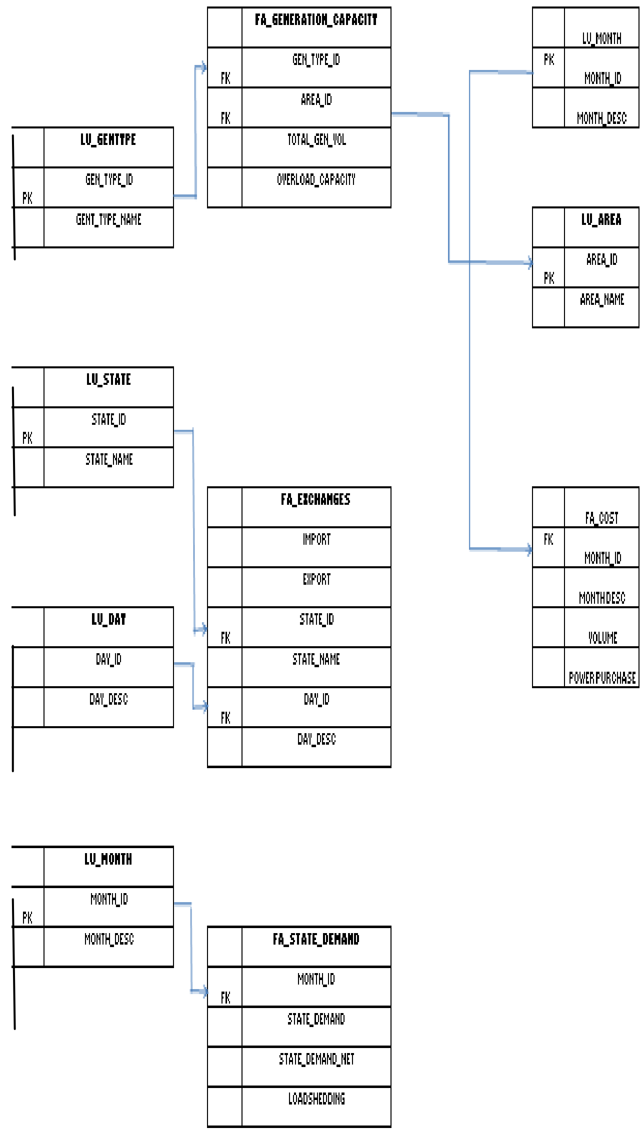


Figure 31 : Physical Data Model

After having created the LDM and PDM, we can create tables accordingly feed the corresponding values in the respective tables. With this have successfully created the database that has to followed with the creation of project source and a DSN connection for developing a dashboard.

4. Conclusion

The value of a data warehouse increases over time.

Benefits of Data Warehouse

- Hard savings come from things like discovering lost discounts in payables or that sales people are offering discounts beyond approved limits.
- Real-time consolidation of financial data becomes practical.
- Debates cease over which source of data is correct.
- The IT costs and staff dedicated to reporting are greatly reduced.

Other reasons why the value of a data warehouse increases over time include:

- Use of BI becomes more widespread as users discover its value.
- Users become more skilled at extracting useful information with experience.
- Historical data becomes more valuable as the amount available increases.
- Additional dashboards, pre-built analytics, and reports become available from vendors.
- Metadata is added over time increasing the usefulness of the underlying data.
- Software tools that build and access data continually improve.
- Additional data sources can be added to the warehouse

With such an overwhelming list of advantages, it is easy to wonder why every organization does not already have a data warehouse. The only reason is that before the availability of prebuilt warehouses, custom creation was an expensive, time-consuming, and expert-intensive process. Thousands of organizations, including the majority of the most successful businesses in the world, have made the investment to create data warehouses. Their pioneering work has made it much easier for those starting today.

The case for obtaining a BI solution based on a data warehouse has become compelling, even for businesses struggling with layoffs and drastic cost cutting. Without one it is very hard to determine how to rebuild a business model around current levels of demand. Trying to manage a complex business in a highly challenging economic environment without a BI solution based on a data warehouse is thus fraught with risk.

References

[1] Maharashtra State Electricity Board , From Wikipedia, the free encyclopedia http://en.wikipedia.org/wiki/Maharashtra_State_Electricity_Board.
[2] Golfarelli & Rizzi," *Chapter 1 Introduction to Data Warehousing* ",Data Warehouse Design : Modern Principles and Methodologies.
[3] Ali Asheibi,David A. Stirling, Sarath Perera, D A. Robinson, " *Power Quality Data Analysis Using Unsupervised Data Mining" ,* Australasian Universities Power Engineering Conference (AUPEC 2004) 26-29 September 2004, Brisbane, Australia.
[4] Can ANIL, " *Benchmarking of Data Mining Techniques as Applied to Power System Analysis",*Department of Information Technology.
[5] Abel Damtew, *"Designing a predictive model for Heart Disease Detection using data mining techniques",* A Thesis Submitted to the School of Graduate Studies of Addis Ababa University in Partial Fulfillment of the Requirements for the Degree of Master of Science in Health Informatics.
[6] Solomon Negash, *" Business Intelligence*", Communications of the Association for Information Systems (Volume13, 2004) 177-195.
[7] Logical Data Model, From Wikipedia the free encyclopedia http://en.wikipedia.org/wiki/Logical_data_model.
[8] http://microstrategy-tutorials.blogspot.in/2010/01/physical-warehouse-chema.html.
[10] http://en.wikipedia.org/wiki/Fact_table.
[11] Joseph Guerra, David Andrews " *Why You Need a Data Warehouse"*, Andrews Consulting Group.