Chapter-2

Review of Literature

2.1 DATA WAREHOUSE DESIGN PATTERN

The web Data incorporation identified two main ways to join together heterogeneous data into a data warehouse. First come within build of related to federated databases, that are spread and heterogeneous web databases constituted from data sources of various natures: UML, HTML or XML documents, databases, and so on, and providing users with an integrated view of the data. The casual architecture for a federated database is layered in three levels:

- Presentation components allowing formulating queries in the federated database language.
- Arbitration mediators in charge of collecting queries issued by users in the presentation components and translating them in the proper language of each data source.
- Adaptation components allowing communication between data sources and mediators.

The second possible approach consists in capturing the common characteristics of the different data types we need to integrate. In order to
propose a unified data model, we took interest in how data is structured, stored, and indexed in textual and multimedia databases.

Indexing strategies in textual databases include reversed lists of significant terms with their frequency of appearance in each document, signatures obtained by hashing keywords, and relative frequency matrix of words present in a set of documents. Multimedia databases may adopt the following characteristics to index images: signatures derived from (manually captured) keywords describe Designing the image, color, texture or brightness distributions, etc. The XML mapping emerged as a data exchange standard language, its storage in databases became a research issue. Several approaches have been adopted to map an XML document into a database propose algorithms that exploit a UML schema to map a DTD (Document Type Definition) into a relational schema. Another approach consists in representing an XML document as a labeled, oriented graph where vertices are data types, edges are classes or objects, and leaves are data. In these two approaches, a relational or object-relational database system was used to store the XML documents.

The Design Pattern is the delivery of accurate, useful information to the appropriate decision makers with necessary timeframe to support effective decision-making.

According to Larson (2009) Data warehouse is a system that retrieves and consolidates data periodically from the source systems into a dimensional or
normalized data store. It usually keeps years of history and is queried for Design Pattern or other analytical activities. It is typically updated in batches, not every time a transaction happens in the source system Rainardi (2008).

The Data Mart is a subset of data warehouse and is defined as body of historical data in electronic repository that does not participate in the daily operations of the organization. Instead, this data is used to create Design Pattern. The data in the data mart usually applies to a specific area of organization. (Larson, 2009) Fact Table is the primary table in a dimensional model where the numerical performance measurements of the business are stored. We try to store the measurement data resulting from a business process in a single data mart.

The Dimension Table is an integral companion to a fact table. The dimension tables contain the textual descriptors of the business. In a well-Designed dimensional model, dimension tables have many columns or attributes. These attributes describe the rows in the dimension table. The Dimension tables tend to be relatively shallow in terms of the number of rows (often far fewer than 1 million rows) but are wide with many large columns. Dimension tables are the entry points into the fact table. The dimensions implement the user interface to the data warehouse Online analytic processing (OLAP) database is a technology for storing, managing, and querying data specifically Designed to support Design Pattern uses.
The Extract, Transformation, and Load (ETL) system is a set of processes that clean, transform, combining Design, duplicate, archive, conform, and structure data for use in the data warehouse.

### 2.2 DATA WAREHOUSE CONCEPTS

The Data warehousing is the process of collecting data to be stored in a managed database in which the data are subject-oriented and integrated, time variant, and nonvolatile for the support of decision making (Inmon, 1993). Data from the different operations of a corporation are reconciled and stored in a central repository (a data warehouse) from where analysts extract information that enables better decision making (Cho and Ngai, 2003).

The Data can then be aggregated or parsed, and sliced and diced as needed in order to provide information (Fox, 2004). There are two main authors that are known in the world of data warehouse Design, their approaches to some area of the data warehousing are different; William Inmon and Ralph Kimball. The approach by Inmon is top down Design while that of Kimball is bottom up Design. Most of the practitioners of Data warehouse subscribe to either of the two approaches.

According to Inmon (1993), a Data Warehouse is a subject-oriented, integrated, time-variant, non-volatile collection of data used in support of decision making processes. “Subject Oriented” means that a data warehouse focuses on the high-level entities of the business and the data are organized
according to subject “Integrated” means that the data are stored in consistent formats, naming conventions, in measurement of variables, encoding structures, physical attributes of data, or domain constraints. For example, whereas an organization may have four or five unique coding schemes for ethnicity, in a data warehouse there is only one coding scheme (Chan, 1999). “Time-variant” means warehouses provide access to a greater volume of more detailed information over a longer period and that the data are associated with a point in time (Chan, 1999; O’Leary, 1999) such as month, quarter, or year.

The Warehouse data are non-volatile so that data enter the database are hardly ever, if ever, changed one time they are entered into the warehouse. The data in the warehouse are read-only updates or refresh of the data occur on a periodic, incremental or full refresh basis (Zeng et. al., 2003) Finally, “nonvolatile” means that the data do not change (Chan, 1999).

According to Kimball (2002), Data warehouse is the multinational of all data marts within the enterprise. Information is always stored in the dimensional model. Kimball views data warehousing as a constituency of data marts. The Data marts are focused on delivering business objectives for departments in the organization. And the data warehouse is a conformed dimension of the data marts.

According to Kimball a data mart as a subset of data warehouse. The data warehouse is the sum of all the data marts, each representing a business process in organization by a means of a star schema, or a family of star
schemas of different granularity. The main difference between the approach of Kimball and that of Inmon (1993) is that Kimball’s conformed dimensions are de-normalized, whereas Inmon uses a highly normalized central database model.

According to Inmon’s data marts stores a second copy of the data from the centralized data warehouse tables, whereas the dimensions of Kimball used in the data marts, are not copies of the conformed dimensions, but the dimension table themselves.

Kimball refers to the set of conformed dimensions as the data warehouse bus. There is no right or wrong between these two ideas, as they represent different data warehousing philosophies. In reality, the data warehouse in most enterprises is closer to Ralph Kimball's idea. This is because most data warehouses started out as a departmental effort, and hence they originated as a data mart. Only when more data marts are built later do they evolve into a data warehouse.

Basaran (2005) reveals some of the DW characteristics to include the following.

- It is subject-oriented.
- It is non-volatile.
- It allows for integration of various application systems. It supports information processing by consolidating historical data.
Data is stored in a format that is structured for querying and analysis.

Data is summarized. DWs usually do not keep as much detail as transaction-oriented systems.

2.2.1 STATEMENTS DESIGN PATTERN

In the literature we find lots of different approaches to a proper definition of Design Pattern (DESIGN). Different parties such as IT vendors, press groups and business consultants have their own approach to this subject. Below a few examples are described. Together they should illustrate the main concept of Design Pattern.

Gartner Group describes DESIGN as a process of transformation from data to information, and after a voyage of discovery transforming this information to knowledge.

Vriens & Philips, found out DESIGN as a process of acquiring and processing of information in order to support an organization’s strategy.

De Tijd, 2006, defines DESIGN as all applications supporting analyzing and reporting of corporate data in order to improve decision making which leads to better steering of the company. The Decision makers need to be provided by reliable information, filtered from all raw data the company has acquired in the past. The main purpose is to transform these raw data into valuable, actionable information. Common transactional software automates daily based processes such as the creation of invoices and registers
them into the system. Unlike this, DESIGN sets a step backwards to provide a holistic view on these transactions. Figures from the past are not reported in a very detailed way, instead they are aggregated, analysed and linked to each other with the purpose to forecast future activities.

David M. Kroenke, 2006, mentions business intelligence systems fall into these broad categories, namely reporting, including OLAP, and data mining. Aronson, Liang and Turban, 2005, also divide DESIGN tools into reporting, OLAP and data mining. In this chapter I will divide DESIGN tools also into these 3 categories. Apparently most of the definitions agree Design Pattern should support defining the fundamental direction of a company by analyzing and reporting data.

2.3 DESIGN PATTERN ARCHITECTURE

1. OPERATIONAL APPLICATIONS VS. DESIGN PATTERN APPLICATIONS

Figure 2.1 illustrates the 2 main components of DESIGN applications, and their relation with operational applications. Notice that Kroenke, 2006, defines reporting and data mining as the 2 main DESIGN components. In my opinion OLAP is situated anywhere between reporting and data mining. Therefore, in this dissertation reporting, OLAP and data mining are described separately.
Figure 2.1 Showing Relationships of Operational and Design Pattern Applications (Source: Kroenke, 2006)

The operational business applications such as order entry, manufacturing and purchasing read from and write data to the operational database via the operational database management system (DBMS). Entering orders for example into a corporate system is mainly situated on a company’s operational level and do not mainly require high level decisions to make.

According to this scheme, management, situated on tactical and strategic level, are supposed to relatively use Design Pattern applications to improve decision making. Notice that this distinction touches the core of this dissertation, as the main goal is to find out what corporate level(s) is DESIGN really contributing to.
The DESIGN applications on the other hand might only read data directly via the operational DBMS from the operational database, as long as simple reporting and/or small databases are applied. Data from extractions of this operational database as well as purchased data from external data vendors are read through the DESIGN DBMS. DESIGN applications then can both make reports as well as carry out some advanced analyses based upon these data. In further paragraphs a more detailed subdivision of these components will be explained (Kroenke, 2006).

2. **NEED FOR A DATA WAREHOUSE**

For complex Design applications running on large databases difficulties might occur while reading directly from the operational database. Besides slowing down the DBMS and its applications, errors might occur when values are missing or in a wrong format.

Therefore, a separate database, an extraction from the operational database, needs to be set up and prepared for Design use. This process of data warehousing is done in 3 main steps, also known as extraction, transforming and loading (ETL).

Extraction programs retrieve data from a variety of heterogeneous operational databases based upon a certain model. The metadata describes this model and the definition of the source data elements. For example, a model describing Design the regional sales performance is defined by metadata containing sales
data in integer format created by salespeople of a particular region. Notice that using indexes improves the speed of this extraction process. Transformation of data is sometimes needed to ensure consistency of all data in the data warehouse.

The Data need to be transformed into the right format or missing values need to be filled in. Certain aspects of operational data, such as low-level transaction information, are also removed as they slow down query times. Finally the DBMS loads these processed data into a data warehouse. This process of extraction, transforming and loading (ETL) is crucial in DESIGN processes as they are the link to the source data. Once ETL has finished, users can start producing information or intelligence. Obviously, people preparing data warehouses are experts in data management. They consider the preparation of data warehouses as their final product. In contrast, from business point of view, data warehouses are only the beginning of a business analyst’s job. People of marketing or financial departments might rather work with data marts. These are small subdivisions of data warehouses containing data on specific business components. The marketing analyst might for example analyse a data mart containing sales data of a particular market segment. Figure 9 illustrates how data warehouse DBMS links operational databases and DESIGN tools graphically. It Notice that metadata, data concerning the data’s author, format, time of creation, etc., is also stored through the data warehouse DBMS.
2.4 ANALYSIS AND REPORTING FOR IMPROVE DECISIONS MAKING

1. REPORTING

Reporting technology in DESIGN contains much more functionalities than just distribution of information. Reporting is applied in business processes to generate reports for applications such as logistics and financial management. Based upon user skills, DESIGN distinguishes 3 main types of reporting tools, namely production reporting tools, desktop report writers and managed query tools.

Production reporting tools are used to generate operational reports or extremely voluminous batch tasks such as counting and printing salaries.
Generating reports requires support from the IT department. As these reports cover large amount of data, queries are processed in batch mode. In contrast, desktop report writers enable users to design queries and reports quite simply and easily on their desktop, without interference of the IT department. Via a graphical interface report writers have access to multiple databases, make selections out of them and present and distribute the results via a large variety of report formats. Figure 2.3 shows examples of different types of reports. The Desktop report writers enable users to design quite simple reports, based upon a rather small pool of data. When complex source data need to be accessed, managed query tools are to be applied.

Managed query tools enable users to access complex source data on a fairly simple way. This connection requires an interface between the data sources and the user, which defines the relation between the physical data in the databases and the user language. This interface contains a graphical SQL environment generating the SQL code according to a graphical command. Standard Query Language (SQL) is a standardized database language for data access (read, insert, update, delete) and manipulation in relational database management systems (Aronson, Liang and Turban, 2005).

the SQL enables users also to carry out some simple calculations on data such as generating overviews of past (trends), current, and likely future business activities (forecasting) (Kroenke, 2006).
As a result of this user friendly interface, users can totally focus on defining questions without worrying about aspects such as the location of the data, consistency and the like. For example, an overview of total units sold per year, customer type and geographical area can be shown. Important to stress is this interface enables nontechnical users to create their own customized reports. In my opinion holistic overviews are very useful at high level management; this reporting tool might contribute a lot to a company’s strategic management. Sometimes managed query tools provide OLAP functionalities as well. OLAP enables users to further dig into the general overviews of managed query tools.

The Report delivery and distribution is crucial in the decision making process. Reports need to be delivered to the right and authorized users, in the right format and on the right time. The report output might be delivered on

Figure 2.3 Showing Components of a Reporting System

[Diagram of reporting system components]
paper, via a browser, over the telephone, or in any other format. Figure 2.3 shows both input and output components of a reporting system. A digital dashboard is an electronic customized display of the report (Kroenke, 2006). A financial analyst might for example prefer the company’s financial stock price as well as European and American stock prices on his dashboard. Alerts are reports that are automatically triggered when an event has occurred, e.g. An e-mail is sent if the company’s stock price has reached a predefined limit.

The RFM analysis is an example of such a report. This report ranks all customers based upon how recently (R) they bought something, the frequency (F) they buy, and how much money (M) they spent. This RFM technique enables users for example to identify clients tenting to go to competition (Kroenke, 2006). Although this is not a complex technique, it already unveils, in my opinion, extremely valuable intelligence for a company.

### 2.5 DATA WAREHOUSE DATA MODEL

Inmon (1993) argues that there are three levels in data modeling process: High-level modeling (called the ERD, entity relationship level) which features entities, attributes and relationships, Mid-level modeling (called the data item set) which is data set by department, and Low-level modeling (called the physical model) optimize for performance.
After the high-level data model is created, the next level is established the midlevel model. For each major subject area, or entity, identified in the high level data model, a midlevel model is created. Each area is subsequently developed into its own midlevel model. The physical data model is created from the midlevel data model just by extending the midlevel data model to include keys and physical characteristics of the model. At this point, the physical data model looks like a series of tables, sometimes called relational tables.

Stuart Mullins (2007) in his blog titled “Data Warehouse Data Model Design” explains what can be used to differentiate the DW from an ordinary archive database which can easily become a dumping ground. Data is conformed (Data elements are conformed so that the definitions of "customer" or "revenue" mean the same thing no matter where the originated), Data is historical (view of the business at a particular point in time), Data is shared (Can be queried or otherwise accessed has little value), Data is comprehensive (Can be captured and consolidated from multiple systems).

2.5.1 **DW MODELING TECHNIQUES**

Ballard (1998) gave an exploration of the evolution of the concept of data warehousing, as it relates to data modeling for the data warehouse, they defined database warehouse modeling is the process of building a model for the data in order to store in the DW. There are two data modeling techniques
that are relevant in a data warehousing environment are Entity Relationship (ER) modeling and dimensional modeling.

The ER modeling produces a data model of the specific area of interest, using two basic concepts: entities and the relationships between those entities. Detailed ER models also contain attributes, which can be properties of either the entities or the relationships.

The Dimensional modeling uses three basic concepts: measures, facts, and dimensions. Dimensional modeling is powerful in representing the requirements of the business user in the context of database tables. Measures are numeric values that are can be added and calculated.

Ballard (1998)

2.5.2 DW DATABASE DESIGN MODELING

There are three levels of data modeling. They are conceptual, logical, and physical. For the purpose of this thesis, we would discuss only the first two. Conceptual Design manages concepts that are close to the way users perceive data; logical Design deals with concepts related to a certain kind of DBMS; physical Design depends on the specific DBMS and describes how data is actually stored.

The main goal of conceptual Design modeling is developing a formal, complete, abstract Design based on the user requirements. DW logical
Design involves the definition of structures that enable an efficient access to information. The Designer builds multidimensional structures considering the conceptual schema representing the information requirements, the source databases, and non-functional (mainly performance) requirements. This phase also includes specifications for data extraction tools, data loading processes, and warehouse access methods. At the end of logical Design phase, a working prototype should be created for the end-user. Basaran (2005)

### 2.5.3 DEVELOPING DATA WAREHOUSE

Demarest (2008) was explicit when it says that planning the developing and deployment of a standard data warehouse should be taken as an IT project, hence what made IT project fail applies also applies when developing data warehouse; thus the need for Project Planning and following the system development life cycle. There is the need for careful planning, requirements specification, Design, prototyping and implementation. The cyclical model entails five stages which are described below

![Figure 2.4 Showing DW Development Lifecycle (DWLC) Model](image)

**Figure 2.4** Showing DW Development Lifecycle (DWLC) Model
Where the Design stage takes information from both available data inventories and analyst requirements and analytical needs, of robust data models and turns it into data marts and intelligent information. The Prototype deployment stage, where group of opinion-makers and

Certain end-user clientele, are brought in contact with a working model of the data warehouse or data mart Design, suitable for actual use. The purpose of prototyping shifts, as the Design team moves back and forth between Design and prototype. Deploy stage is the stage of formalization of user-approved prototype for actual production use. The Operation is the day to-day maintenance of the data warehouse or mart, the data delivery services and client tools that provide analysts with their access to warehouse and the management of ongoing extraction, transformation and loading processes that keep the warehouse current with respect to the authoritative transactional source systems. Enhancement stage is where external business conditions change discontinuously, or organizations themselves undergo discontinuous changes enhancement moves seamlessly back into fundamental Design, if the initial Design and implementation didn’t meet requirements.

2.6 DESIGN PATTERN CONCEPTS

Initially, Design was coined as a collective term for data analysis tools. Meanwhile, the understanding broadened towards Design as an encompassment of all components of an integrated decision support infrastructure.
In Design systems, data from OLTP is combined with analytical front ends to “present complex and competitive information to planners and decision makers”. A central component of Design systems is the data warehouse (DW), which integrates OLTP data for analytical tasks. From the managerial approach, Design is seen as a process in which data from within and out the organization are consolidated and integrated in order to generate information that would facilitate quick and effective decision-making.

The role of Design here is to create an informational environment and process by which operational data gathered from transactional systems and external sources can be analyzed and to reveal the “strategic” business dimensions. From this perspective emerge concepts such as “intelligent company”: One that uses Design to make faster and smarter decisions than its competitors (Liautaud, 2000). “Intelligence” means reducing a huge volume of data into knowledge through a process of filtering, analyzing and reporting information.

The technological approach presents Design as a set of tools that supports the storage and analysis of information. The focus is not on the process itself, but on the technologies that allow the recording, recovering, manipulation and analysis of information. For instance, Scoggins (1999) classifies data mining (DM) as a Design technique; Hackathorn (1999) includes all resources (DW, DM, hypertext analysis and web information) in the creation of a Design system; and finally, linking Design and the Internet, Giovinazzo (2002) posit
the integration of DW and customer relationship management (CRM) applications.

Whether managerial or technological, there is one shared idea among all these studies are given below

- The core of Design is information gathering, analysis and use
- The goal is to support the decision making process, helping the company’s strategy. Taking into account the scarce literature, we looked for other areas that could help us reach a more comprehensive understanding of Design.

We find contributions in three distinct topics: information planning, balanced score card and competitive intelligence. Here are some benefits that Design Pattern offers and how they can help the entertainment industry to make and distribute creative substance and stay aloft of the game:

- **Product Profitability Design:** How much profit does a particular item contribute? How does item’s profit break down across business units, media and distribution channels? What are the specific costs and expenses associated with producing the item? What percent of revenue or profit do they represent?

- **Customer and market analysis:** What are the key demographic characteristics of customers by product? Which other products do they
tend to buy? Does the data indicate that an underserved market segment has greater revenue potential?

- **Channel analysis:** Which channels reach what types of consumers? How profitable is each channel? How will channels be affected by changing technologies, as well as the emergence of new channels?

- **Forecasting and planning:** What are the market potential of a new product, and how much investment should be made? How will a new release perform and what will its profit contribution be? What level of supply will adequately meet demand?

The result – employees can now access detailed sales data from around the world, which was previously not possible, and they are also able to run sophisticated self-service reports that provide granularity and a near real-time view into sales performance, ultimately helping these users make informed decisions that drive the results of the business. In addition to sales data, Media companies can measure marketing and promotion effectiveness and monitor corporate performance and results. Design not only converts raw data into intelligent information, but also allows business users to access the right information at the right time and able to transform it into smart decisions.

Media companies with its business processes based on such intelligent information can disrupt its competitor’s moves, strategize a sustainable competitive edge, tap into new customer bases, retain existing customer bases, increase operational efficiencies and be better prepared for the future.
2.7 DATA WAREHOUSING OLTP

Data warehouse are also known as Online Analytical processing (OLAP) system because they serve managers and knowledge workers in the field of data analysis.

The Online transaction processing (OLTP) systems or operational systems are those information systems that support the daily processing that an organizational does. The OLTP system’s main purpose is to capture information about economic activities of an organization. On might argue that the purpose of OLTP system is to get data into computers, whereas the purpose of data warehouse is to get data or information out of computers Han and Kamber (2001) argue that an OLTP system is customer-oriented as opposed to a data warehouse that is market-oriented. It is a Design difficult to combine Design data warehousing (OLAP) and OLTP capabilities Design in one system.

The dimensional data Design model used in data warehouse is much more effective for querying that the relational model used in OLTP systems. Furthermore, the data warehouses may use more than one database as a data source. The dimensional Design is not suitable for OLTP systems mainly due to redundancy and loss of referential integrity of the data. Organizations choose to have two separate information systems, one OLTP and One OLAP system.
Poe (1998) stress the fact that analysis using OLAP systems are primarily done through comparisons, or by analyzing patterns and trends. For example, sales trends are analyzed along with marketing strategies to determine the relative success of specific marketing strategies with regard to sales patterns; such analysis may not be possible with OLTP.

Kimball supported same idea but Inmon (1993) was a Design different on the approach to the development of data warehouse system. He argue that although OLTP are developed from requirements as a starting point, data warehousing starts at implementing the data warehouse and ends with a clear understanding of the requirements. The data warehouse development lifecycle is data-driven and OLTP are requirements driven. Kimbal (1996) differ from this approach by following a requirements-driven development lifecycle.

2.8 DATA WAREHOUSE AND DESIGN PATTERN HIGH LEVEL ARCHITECTURE

Eckerson (2003) from the Data warehouse institute did study on the success factor in implementing Design, systems in organizations and the role of data warehouse in this process. Eckerson (2003) views the DESIGN process holistically as a “data refinery” Data from different OLTP systems are integrated, which leads to a new product called information. The data warehouse staging process is responsible for the transformation. Users equipped with program such as specialized reporting tools, OLAP tools and
data mining tools transform the information into knowledge. Kimball (1996) includes this as part of the data warehouse.

Kimball, the aim of the data warehouse is to give end-users (mostly managers) easy access to data in the organization. In order to do this, it is necessary to capture everyday operational data from the operational systems of the organization. These are the OLTP system. The data from the source systems go through a process called data staging to the presentation servers (Kimball et al 1996). The data at the staging process involves four processes namely Extract, Transformation, Loading and finally presentation. It is on the presentation stage that the data marts, which represent business areas in the organization is build on.

The data in the data mart or data warehouse is stored as star schemas consisting of FACT and DIMENSION tables. This is different from the entity relational diagram (ERD) used in traditional systems.

There is a difference between the data warehouse and Design Pattern architecture as advocated by the two known scholars in the industry, (Inmon, 1993) advocates the use of data-driven method. This means that the Decision Support System process begins with data and ends with requirements. In contrast to Inmon’s approach, advocate the use of requirements-driven methods. The data warehouse starts with the project planning to determine the readiness of the organization for a data warehouse and to set the staff requirement for the data warehouse team. A clear understanding of business
requirements is the most important success factor and Kimball, (1996) state that this process of requirements

Collection differs substantially from data-driven requirements analysis. The business requirements establish the foundation for the three parallel tracks focused on technology, data and end user applications.

2.9 DATA WAREHOUSE DESIGN CONCEPTS

The Design of the database depends on the approaches of the father of data warehouse developers. The two-Design processes are referred to as Top-down process, as described by Inmon and Bottom-up as described by Ralph Kimball. These are explained in detail below.

2.9.1 TOP-DOWN-MODEL

These was Introduced by Inmon, The process begins with an Extraction, Transformation, and Loading (ETL) process working from legacy and/or external data sources. Extraction transformation, process data from these sources and output it to a centralized Data Staging Area. Following this, data and metadata are loaded into the Enterprise Data Warehouse and the centralized metadata repository. Once these are constituted, Data Marts are created from between the data warehouse and the data marts are automatic as long as the discipline of constituting data marts as subsets of the data warehouse is maintained.
2.9.2 BOTTOM UP DATA WAREHOUSE MODEL

The central idea in Bottom-up model is to construct the data warehouse incrementally over time from independently developed data marts. The process begins with ETL for one or more data marts. No common data staging area is required. There is generally a separate area for each data mart. There may not even be standardization on the ETL tool. The Model was introduced by Ralph Kimball.

For the purpose of this project, Bottom-up model approach would be adopted, which is the Kimball’s development lifecycle, this states with one data mart (e.g. Sales) later on further data mart are added e.g. Marketing and Collection. Data flows from sources into data marts, then into the data warehouse. It is also implemented in stages (faster) Due to the time constraint and project limitation, it is easier to complete a process for a subset of a company based on the data mart and link it up as the business grows. The stages proposed for the process include Investigation, Analysis of the current environment, identify requirements, and identify architecture, data warehouse Design, implementation and ongoing data administration.