DESIGNING A VARIETY OF DATA WAREHOUSE SCHEMAS SUITABLE FOR META-SEARCH ENGINES

Book · May 2016

1 author:

Atheer Alrammahi
University of Al-Qadisiyah
3 PUBLICATIONS **0** CITATIONS

Over the last twenty years, the benefit to analyze the data has increased considerably, because the competitive advantages that information can provide the decision-making process. Today, one of the keys to survival in the business world is to be able to analyze, plan and respond to changing economic circumstances as soon as possible. Several organizations have billions of bytes of data, but they suffer from multiple problems that make it difficult to leverage data: the data are spread across various computer systems, data from different sources are inconsistent, the data be found too late, etc. to resolve these problems, new concepts and tools have evolved into a new information technology known Data Warehousing. Data Warehouse Projects (DW) are costly: they often need several years to properly implement and require millions of dollars of hardware, software and consulting services. Data Warehouse Projects (DW) are expensive: they often require several years to properly implement and need millions of dollars of hardware, software and advisory services.

My name is Atheer Hadi Alrammahi. I have completed the master degree in Computer Science and Communication of in Faculty of Sciences and Fine Arts in Lebanon June 2013. Workplace : Director of the E-Learning center in university of AL-Qadisiya

978-3-659-88939-4

Computer Science and Communication

Alrammahi

Atheer Hadi Alrammahi

Designing a variety of data warehouse schemas suitable for meta-search

Computer Science and Communication

LAP LAMBERT Academic Publishing

Atheer Hadi Alrammahi

Designing a variety of data warehouse schemas suitable for meta-search

Atheer Hadi Alrammahi

Designing a variety of data warehouse schemas suitable for meta-search

Computer Science and Communication

LAP LAMBERT Academic Publishing

Impressum / Imprint
Bibliografische Information der Deutschen Nationalbibliothek: Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Bibliographic information published by the Deutsche Nationalbibliothek: The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.d-nb.de.

Coverbild / Cover image: www.ingimage.com

Verlag / Publisher:
LAP LAMBERT Academic Publishing
ist ein Imprint der / is a trademark of
OmniScriptum GmbH & Co. KG
Bahnhofstraße 28, 66111 Saarbrücken, Deutschland / Germany
Email: info@omniscriptum.com

Herstellung: siehe letzte Seite /
Printed at: see last page
ISBN: 978-3-659-88939-4

Abstract

Communication and information sharing has been synonymous with databases as long as there have been systems to accommodate them. Now more than ever, users expect the exchange of information for immediate, effective and secure way. However, due to the large number of databases within the company, obtaining data effectively requires coordinated efforts between existing systems. There is a real need today to have a single location for storing and sharing data that users can easily use to make business decisions improved, rather than trying to go through the multiple databases that exist today and can do so by using enterprise data warehouse.

Modern databases have included use of DSS (Systems Decision Support) to increase their business decision function and enable detailed analysis of offline data by business leaders high. Data warehousing and meta search engine are two of the areas fastest growing technologies in the past information.

The data warehouse is an environment that can be easily adjusted to maximize the effectiveness of the implementation of decision support functions. But the advent of commercial uses of the Internet on a large scale has opened up new possibilities for data entry and inclusion in the warehouse.

The thesis includes a description of the techniques of data storage, design, expectations and challenges for data cleansing and transforming existing data, as well as other challenges to the extraction of transactional databases. The thesis also includes a technical item discuss the requirements and technologies used to create and update the warehouse database data. The thesis deals with how data databases and other data repositories could integrate. In these thesis, I will discuss the design of a variety data warehouses (star, snowflake, and fact constellation/galaxy) that are more suitable for meta-search engines & data web housing environments and architectures. In addition, I will discuss the requirements analysis, logical design and physical design issues in the search engine metadata. I gathered a wide range of interesting OLAP queries for Meta search engines and categorize. On the basis of these OLAP queries, I illustrate our design of the data warehouse architecture bus structures dimension tables, a basic outline of a star, and an aggregation star schema. I present to you different physical design considerations for implementing the dimensional models. I think my collection of OLAP queries and dimensional models would be helpful in the development of data warehouses from the real world in search of metadata. Also, a technical comparison will

be done once the design of the various data warehouses are designed to help decision makers to select the most appropriate scheme to carry out their daily activities. There are many curricula in designing a data warehouse both in conceptual and logical design phases. The famous conceptual design approaches are dimensional fact model, multidimensional E/R model, starER model and object-oriented multidimensional model. And in the logical design phase, star schema, fact constellation schema, galaxy schema and snowflake schema.

Keywords: Data Warehousing, Data Web Housing, Business Intelligence, Meta-Search Engine, Performance Tuning, Optimization, Star Schema, Snowflake Schema, Fact Constellation/Galaxy Schema.

Dedication

I dedicate this thesis to carry your name with pride and to whom I miss you since childhood and to the wavering of my heart for remembrance dedicate this thesis to my father Mr. Hadi Issa (may Allah have mercy on him).

To my wisdom, scientific, and literary and dream, and to my way of the rectum and into the path of guidance to the fountain of patience, optimism and hope and to everyone in the presence of God and His Prophet after my mother dear.

To my wife .. Which bore the me alienation frost .. And nomadic trouble ..

To my brother and companion of my way, and this life without you nothing with you I am, and without you I like anything ..At the end of my career I want to thank the noble attitudes to Mr. Hadeer Hadi.

To my children, daughter (Fatima Alzahraa) and both sons (Hussein and Ali) .. My dream big .. To see them build their home to Iraq .. Under the sky filled with goodness and love and peace ..

To those who have contributed even a word in the Lighting candles ..

I attribute the level of my Master to my advisor Dr. Jaber Jaber. Without him, this thesis would not have been completed. We could simply not wish for a better or friendly advisor.

Acknowledgement

Praise be to God the creator of heaven and earth, and peace and blessings be upon our Prophet Muhammad and upon his family and after.

At the end of my thesis I would like to thank all those people who made this thesis possible and an unforgettable experience for me.

First of all, I would like to express my deepest sense of Gratitude to my advisor Dr. Jaber Jaber, who offered his continuous advice and encouragement throughout the course of this thesis. I thank him for the systematic guidance and great effort he put into training me in the scientific field.

I also extend my gratitude thanks to my precious family my mother and my brother, and my wife and my children who have credited with completing my thesis. Also do not forget to thank the Iraqi Ministry of Higher Education and the University of Qadisiyah to help me to complete my service to our beloved Iraq.I also extend my thanks and gratitude to my friends Dear Mr. Bilal Ezzeddine Nakhal and Mr. Haroun Aziz and Mr. Maythem Hazem Aziz and Mr. Amir Maan Furman and Mr. Bassam Nuri Shakir to help me in my work and would stand beside me in all my work thank so much for them and I hope to God that reconciles both helped me in my life.

Atheer

List of Contents

List of Figures

List of Tables

List of Abbreviations

DW Data Warehouse.

OLAP Online Analytical Processing.

DSS Systems Decision Support.

ER Entity Relationship

OLTP Online Transaction Processing

DB Database

XML Extensible Markup Language

ETL Extract, Transform, Load

ODBC Open Database Connectivity

DBMS Database Management Systems

RDBMS Relational Database Management Systems

SMP Symmetric Multiprocessing

MPPS Massively Parallel Processor

MDDBs Multidimensional Database

EIS Executive Information System

HIPO Hierarchical Input Process Output

DFD Data Flow Diagram.

ERD Entity Relationship Diagram

DIS Data Item Set

I/O Input/Output

DF Dimensional Fact

ME/R Multidimensional E/R

HOLAP Hybrid OLAP

MOLAP Multidimensional OLAP

ROLAP Relational OLAP

SQL Structured Query Language

UID Unique Identifier Distinguishes.

DDL Data Description Language

DML Data Manipulation Language

URL Uniform Resource Locator

TOD Time of day

HTTP The Hyper Text Markup Language

IP Internet Protocol

KDD Knowledge Discovery in Database

Chapter 1: Introduction

1.1 Problem Statement

Over the last twenty years, the benefit to analyze the data has increased considerably, because the competitive advantages that information can provide the decision-making process. Today, one of the keys to survival in the business world is to be able to analyze, plan and respond to changing economic circumstances as soon as possible.

Several organizations have billions of bytes of data, but they suffer from multiple problems that make it difficult to leverage data: the data are spread across various computer systems, data from different sources are inconsistent, the data be found too late, etc. to resolve these problems, new concepts and tools have evolved into a new information technology known Data Warehousing.

Data Warehouse Projects (DW) are costly: they often need several years to properly implement and require millions of dollars of hardware, software and consulting services.

Data Warehouse Projects (DW) are expensive: they often require several years to properly implement and need millions of dollars of hardware, software and advisory services.

Sales DW and associated products continue year after year, more and more. The market for DW tool has reached 7.9 billion in 2003 and grew by 11 percent this year, more than three times the growth rate of the previous year. [3] In the meantime, according to [6], the online analytical processing (OLAP) market increased from $ 1 billion in 1996 to $ 4.3 billion in 2004 and demonstrated valued at 15.7 percent growth 2004.

Well as much progress has been made in the field of DW, there now is no standard method or data model for the design of DW. Moreover, several reports indicate that about 40-50% of projects fail DW [5, 27]. Consequently, a new method based on DW standards can help develop DW.

Meta-search engine is the answer to provide comfortable, efficient and effective access to information from multiple sources. It is designed to address the key issues that affect the process of research as a database, the selection of materials and the results of a merger, who need additional knowledge about the components of search engines such as representatives of the base detailed data, similarity functions underlying weighting schemes term, indexing methods, and so on. No effective methods exist to find such information without the

cooperation of the engines underlying research after including the right solutions based on different degrees of knowledge of each local search engine that will be applied accordingly. Best solution must evolve thousands of databases, with many of them containing millions of documents, and accesses a day. Construction of a meta-search engine is launching the movement to solve these problems. therefore must create data warehouse for meta search engine. [1].

1.2 Why Not take advantage of the entity relationship model?

Data models and traditional techniques, like the famous entity-relationship (ER) and various extensions of ER [2], are not suitable for DW design, due to the complexity of the corresponding models. Various authors have highlighted this problem. For example, Ralph Kimball States in [3]:

Data models and entity relationship is a disaster for the querying because they are unable be understood by users and they cannot be successfully navigated by the DBMS software. Entity relationship models cannot be used as the basis for enterprise data warehouses.

But the data models later adapted for DW, as the famous star schema Ralph Kimball [3], or they are capable to examine the main features of DW. In addition, each method has its own set of symbols and terminology, which causes a lot of perplexity and letdown.

1.3 What is the aim of this thesis?

The aim of this thesis is to design a variety of data warehouses (star, snowflake, and fact constellation/galaxy) that are more suitable for meta-search engines & data web housing environments and architectures and A technical comparison will be done once the design of different data warehouses are made to help decision makers choose the most appropriate schema to run their daily business.

The objective of this thesis is to define a method that enables the designer to deal the various phases and stages of design of a DW. Our approach consists of the following:

Design a Data Warehouse for Meta-search engines and data webhouse (A meta search engine is a research tool [1] that sends user queries to many other search engines and / or databases and aggregates the results in a list or shows them depending on their source. Meta search engines allow users to enter search options once and access multiple search engines

simultaneously. Meta search engines work on the principle that the Web is too large for any search engine to index it all and that more comprehensive search results can be obtained by combining the results of several search engines, It can also save the user from having to use multiple search engines separately).

Data Webhouse defined by Ralph Kimball [3] Two separate focus:

- Bringing the web to the warehouse
  - Clickstream data as a source of information
- Bringing existing data warehouses to web
  - Fully distributed environment

Also an important requirement of our work is to define a method with a set of models that can be utilized by the DW designer to connect the design to the end user.

1.4 Meta search engine

Whereas the number of search engines on the World Wide Web is increased, meta-search engine is the solution to supply the convenience, efficient and actual access to information from many sources. It is designed to address the key issues that affect the process of research as a database, the selection of materials and the results of a merger, who require additional knowledge about the components of search engines such as officials of the base Detailed data, similarity functions underlying weighting schemes term, indexing methods, and so on. Not have effective methods are available order to find such information without the cooperation of the engines underlying research post including the correct solutions based on varying degrees of knowledge of each local search engine that will be implemented accordingly. Better solution must evolve thousands of databases, with many of them containing millions of documents, and accesses a day. Construction of a meta-search engine is launching the movement to solve these problems [1].

Meta search is to utilize multiple other search systems (called component search systems) to perform simultaneous search. A meta search engine is a search system that enables meta search. To perform a basic meta search, a user query is sent to multiple existing search engines by the meta search engine; when the search results returned from the search engines are received by the meta search engine, they are merged into a single ranked list and the

merged list is presented to the user. Key issues include how to pass user queries to other search engines, how to identify correct search results from the result pages returned from search engines, and how to merge the results from different search sources. More sophisticated meta search engines also perform search engine selection (also referred to as database selection), i.e., identify the search engines that are most appropriate for a query and send the query to only these search engines. To identify appropriate search engines to use for a query requires to estimate the usefulness of each search engine with respect to the query based on some usefulness measure [1].

1.5 Data Webhouse:

The data webhouse is a web instantiation of the data warehouse. The webhouse plays a central and a crucial role in the operations of the web enabled business. The Data Webhouse will,

- Houses and publishes click stream data and other behavioral data from the web that drive an understanding of customer behavior.
- Act as a foundation for web enabled decision making. The data webhouse must allow its users to make decisions about the web, as well as make decisions using the web.
- Act as a medium that publishes data to the customers, business partners, and employees appropriately, but at the same time protects the enterprise's data against unattended use.[1, 3].

1.6 Structure of the Thesis

This thesis is divided into 4 chapters. The content of this thesis is organized so that readers need not read all the chapters to obtain the information they need. Therefore, some concepts are repeated through several chapters. In addition, a wide pointing to other sections related cross-references to enable the reader to develop the content. The following list briefly describes each chapter :

- Chapter 2 (**Introduction to Data Warehouses**) gives a short introduction to data warehousing and associated technologies and Data Warehouse Architecture.

- Chapter 3 (**DESIGNING A DATA WAREHOUSE**) discusses our data warehouse design methodology and Conceptual Design Models and a comparison of data warehouse design models.

- Chapter 4 (**IMPLEMENTING A DATA WAREHOUSE**) discusses implementation of the process of building data warehouses for meta-search engines.

This thesis ends with a list of the references used during the research, an index of important terms.

Chapter 2: INTRODUCTION TO DATA WAREHOUSE

2.1 Background

Data warehousing is a process involved in the collection, organization and analysis of data usually several heterogeneous sources in order to increase the business functions of the end user [5]. It is designed to provide a working model for the easy flow of data from operational systems to systems of decision support. The structure of the data warehouse includes three main levels of granular information data, archival and data and summary metadata to support data [6].

Data warehousing:

1. Focuses on the ad hoc queries posed by end-user business users rather than by experts in information systems.
2. Focuses mainly on offline data, historical data rather than operational type volatile online.
3. Must effectively manage large volumes of data than those treated by standard operating systems.
4. Needs present the data in a form that coincides with the expectations and understanding of business users of the system rather than the architects of information systems.
5. Needs to strengthen data elements. Various operating systems refer to the same data in different ways.
6. Largely based on metadata. The role of metadata is particularly important because the data must be kept in context over time.

The major problems in the design of data warehouses are:

1. performance compared to flexibility
2. cost
3. maintenance.

Business intelligence is a measurement currently associated with data warehousing. Actually, most tool suppliers position their products as business intelligence software instead of data warehousing software. There are other opportunities where the two terms are used interchangeably. So, exactly what is intelligence company?

Business intelligence usually relates to information that is available for the company to make decisions about. A data warehousing system (or data mart) is the backend, or infrastructure element for implementation intelligence business. Business intelligence further comprises the perspicacity acquired to the analysis of data mining, and data unstructured (hence the content management systems need). For our purposes, I will discuss business intelligence through the use of a data warehouse infrastructure.

This section contains the following:

- Business intelligence tools: tools widely used for business intelligence.

- Business Intelligence uses: Various forms of business intelligence.

2.2 Definition of Data Warehouse

A data warehouse (DW) refers to a database that is different from the organization's Online Transaction Processing (OLTP) database and that is used for the analysis of consolidated historical data.

According to Barry Devlin, IBM Consultant, "a DW is simply a single, complete and consistent store of data obtained from a variety of sources and made available to end users in a way they can understand and use it in a business context"[2, 7].

According to W.H. Inmon, "a DW is a **subject-oriented**, **integrated**, **time-variant**, and **nonvolatile** collection of data in support of management's decision making process" [2, 27, 8, 9].

The description of the four main characteristics of DW is provided below.

Subject-oriented: In general, an enterprise contains very detailed information to meet all the requirements for related subsets of the organization (sales department, human resources dept, Marketing, etc..) and optimized for processing the transaction. Commonly, this kind of data is not suitable for decision makers to use. Decision-makers need to subject-oriented data. DW must contain only key business information. The data warehouse should be organized according to the subject and single subject oriented data should be transferred to a warehouse.

If the decision maker needs to find all the information about a specific product, he / she would need to use all the systems such as hire purchase, system sales order and sales system catalog, which is not the best and practical way. Instead, all the key information must be consolidated in a warehouse and organized into domains as shown in Figure 1.

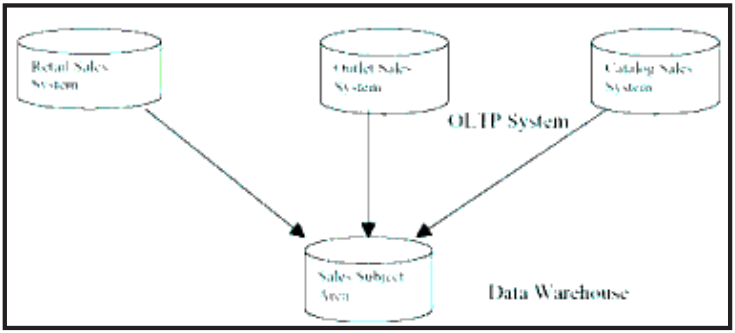


Figure 1 Standardization of OLTP information

Integrated DW is an architecture built by integrating data from multiple heterogeneous sources (such as relational database (DB), flat files, Excel spreadsheets, XML data, data from existing systems) to support structured and / or ad hoc queries, analysis and decision making reports. DW also provides mechanisms for the cleaning and standardization of data.

Time-variant: DW provides information from a historical prospective study. Each key structure in the DW contains, implicitly or explicitly, a time element. A DW usually stores data that is 5-10 years, which will be used for comparisons, trends and forecasting.

Nonvolatile: data in the warehouse is not updated or changed (see Figure 2), so it does not require control mechanisms treatment, recovery and competitive transaction.

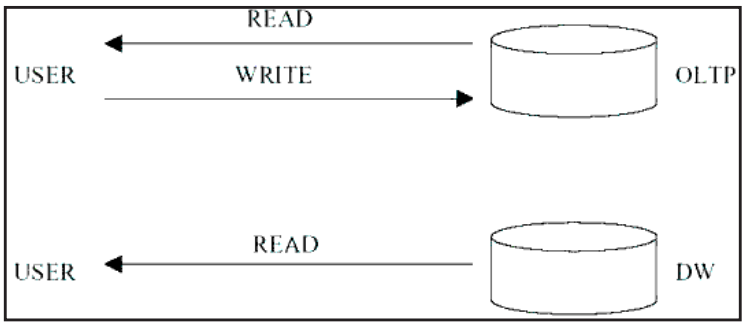


Figure 2 Simple comparison of OLTP and DW systems

Ralph Kimball gave a concise definition of a data warehouse:

A data warehouse is a copy of transaction data specifically structured for query and analysis. It is a working of a data warehouse view. Kimball does not cover how the data warehouse is designed as Inmon did, rather he concentrated on the features of a data warehouse.

2.3 The Goals of a Data Warehouse

The basic goals of the data warehouse [10]:

1 - "Provides information in an accessible organization." The content of the data warehouse are properly labeled and obvious. It is very easy data access, because they are a click and there is not necessary to wait for it. These properties are well called in the order above, comprehensible waterways and speedy performance.

2 - "Give information coherent organization." As information is of paramount importance for data warehouses, because they can get data from multiple parts of an organization. They must correspond properly. If both measures of organization have the same name then they should say the same thing.

In contrast, both measures do not mean the same thing, they are tagged differently.

3 -. "To be a source of adaptation and resilience of information" allows him to add new data and new questions, without any change in the existing data and technology owed to are intended for continuous change.

4 - The data warehouse not only control access to data efficiently, but also gives its owners a high visibility uses "To be a secure bastion that protects the information assets of the owner." and misuse of these data even after it has left the warehouse data.

5 -. "To be the foundation of decision-making" The data warehouse delivers the correct information for decision makers. Decisions are emerged from data warehouse.

2.4 Data Warehouse concepts:

Dimensional Data Model: Dimensional data model is usually used in data warehousing systems. This section explains the modeling technique, and two kinds of common patterns, star schema, snowflake schema and fact constellation/galaxy

Dimensional data model is usually used in data warehousing systems. This is different from the third normal form, commonly used for transactional systems (OLTP). As you can imagine, the same data would be stored differently in a dimensional model from a model third normal form.

to understand of dimensional data modeling we will define some of the terms currently used in this type of modeling:

Dimension: A category of information. For example, the time dimension.

Attribute: A unique level within a dimension. For example, the month is an attribute of the time dimension.

Hierarchy: The specification of levels showing the relationship between one attribute in a dimension. For example, a possible hierarchy in the dimension of time is Year → Quarter → Month → Day.

A dimensional model comprises fact tables and lookup tables. Fact tables connect to one or more lookup tables, but fact tables do not have a direct contact between them. Dimensions and hierarchies are represented by lookup tables. Attributes are nonkey columns in the lookup tables. While designing data models for data warehouse / data marts, schema types are most widely used star schema or snowflake schema.

As a star or snowflake is used depends largely on your personal preferences and the needs of enterprises. Personally, I'm partial to snowflakes, when there is a business case in analyzing information in that particular level [7].

Data Mart: A data structure that is optimized for access. It is designed to help analyze end-user data. It commonly argued a single empirical analysis used by a distinct set of workers.

Staging Area: Any database that is designed primarily to receive data in a storage environment.

Slowly Changing Dimension: This is a common problem opposite data warehousing practitioners. This section explains the problem and describes three ways to handle this problem with examples.

The "Slowly Changing Dimension" problem is a specific common in data warehousing. In a word, this applies to cases where a file attribute varies in time. We provide an example below:

Sara is a client with MOF Inc. She first lived in Illinois. Thus, the original entry in the lookup table customer has the following record: [9]

Customer Key	Name	State
2001	Sara	Illinois

In later stages, she moved to Los Angeles, California, January 2006. How should MOF Inc. now change its customer table to reflect this change? This is the "Slowly Changing Dimension" problem.

There are usually three ways to solve this type of problem, and they are categorized as follows:

Type 1: The new record replaces the original. No trace of the previous record exists.

Type 2: A new record is added into the customer dimension table. Therefore, the client is treated primarily as two people.

Type 3: The original record is edited according to the change.

In **Type 1** Slowly Changing Dimension, the new information merely substitutes the original information. In other words, no history is recorded.

In our example, we first remind following table [9]

Customer Key	Name	State
2001	Sara	Illinois

Customer Key	Name	State
2001	Sara	California

advantages:

- This is the best way to handle the problem of slowly changing dimension, because it is not necessary to keep track of the old information.

disadvantages:

- All history is lost. Using this method, it is not possible to go back in history. For example, in this case, the company would not be able to know that Sara has lived in Illinois before.

usage:

Approximately 50% of the time.

When to use Type 1:

Type 1 slowly changing dimension should be used when it is not necessary for the data warehouse to keep track of in historical changes.

In **Type 2** Slowly Changing Dimension, a new record is added to the table to represent the new information. Consequently the original and the new record will attend. The new record obtains its own primary key.

In our example, we first remind following table: [10]

Customer Key	Name	State
2001	Sara	Illinois

Customer Key	Name	State
2001	Sara	Illinois
2001	Sara	California

advantages:

- This allows us to precisely maintain all historical data.

disadvantages:

- This will entail size of the table to grow rapidly. In the case where the number of rows of the table is very high to start, storage and performance can become a concern.
- This inevitably complicates the ETL (Extract, Transform and Load) process.

usage:

Approximately 50% of the time.

When to use Type 2:

Type 2 slowly changing dimension should be used when it is necessary for the data warehouse to track historical changes.

In **Type 3** Slowly Changing Dimension, there will be two columns to show the attribute of special interest, indicative of the original value, and an indication of the current value. There will also be a column that indicates the current value becomes active [7].

In our example, we first remind following table

Customer Key	Name	State
2001	Sara	Illinois

To greet Type 3 Slowly Changing Dimension, we now meet the following columns: Key Client

- Name
- State of origin
- current Status
- Entry into force

After Sara moved from Illinois to California, the original information is maintained, and we have the following table (assuming the actual date of the change is January 15, 2006):

Customer Key	Name	Original State	Current State	Effective Date
2001	Sara	Illinois	California	15-JAN-2006

advantages:

- This does not mean expand the size of the table, because new information is updated.
- This allows us to spare some of history.

disadvantages:

- Type 3 will not be able to retain all history where an attribute is further modified than once. For example, if Sara moves later in Texas on December 15, 2006, the California information will be lost.

usage:

Type 3 is seldom used in practice.

When to use Type 3:

Type III slowly changing dimension should be used only when required for the data warehouse to track historical changes, and when these changes will not occur in finitely many of times.

Operational Data Store: A collection of data that responds to operational needs of the various business units. This is not a component of a data warehouse architecture, but a solution to the business needs [9].

Multidimensional Analysis: The ability to manipulate information in a range of relevant categories or "dimensions" for easier analysis and understanding of the underlying data. It may also appear called "drilling down", "drilling through" and "slicing and dicing"[35].

Conceptual Data Model: What is a conceptual data model, its characteristics, and an example of this kind of data model.

A conceptual data model identifies the relationships at the highest level between the different entities. Characteristics of conceptual data model include:

- Consists the significant entities and relations between them.
- No attribute is specified.
- No primary key is specified.

The following figure is an example of a conceptual data model [10].

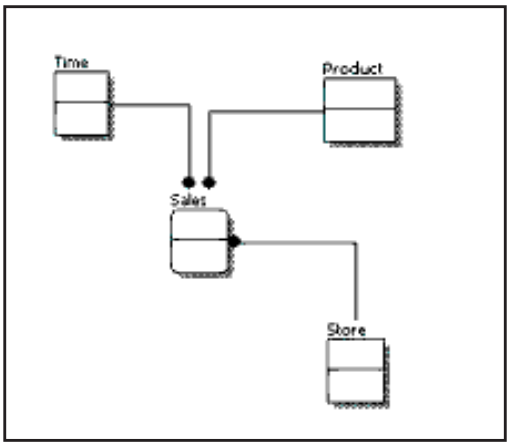


Figure 3 Conceptual Data Model

From the figure above, we can see that the only information displayed via the conceptual data model are entities that describe the data and the relationships between these entities. No other information is available across the conceptual data model.

Logical Data Model: What is a logical data model, its characteristics, and an example of this type of data model.

logical data model describes the data in many details as possible, irrespective how they will be put in located in the physical database. Characteristics of a logical data model include:[10]

* Consists of all entities and relationships between them.
* All attributes of each entity are specified.
* The primary key for each entity is specified.
* Foreign keys (keys identifying the relationship between various entities) are specified.

* Normalization takes place at this level.

The steps in the design of logical data model are:

1- Specify primary keys for all entities.
2- Find the relationship between the various entities.
3- Find all the attributes of each entity.
4- Resolve many-to-many.
5- Standardization.

The following figure is an example of a logical data model.

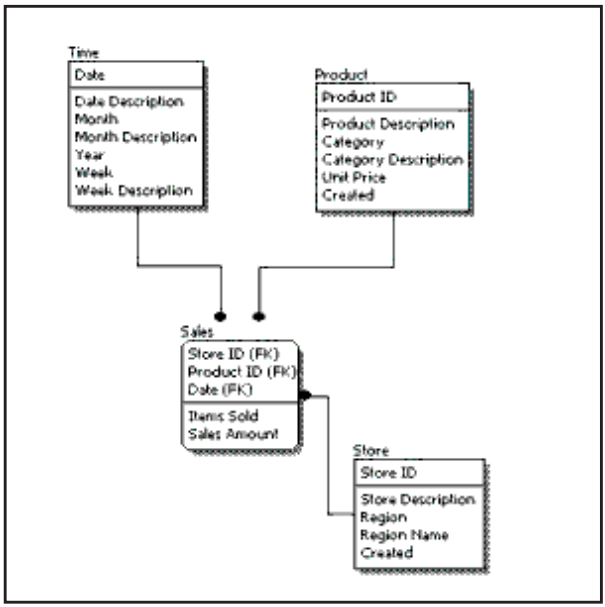


Figure 4 logical Data Model

Comparing the logical data model above with the conceptual schema data model, we see the principal differences between the two:

- In a logical data model, primary keys are present, whereas in a conceptual data model, the primary key is not present.
- In a logical data model, all attributes are specified in an entity. No attributes are specified in a conceptual data model.

- The relationships between entities are specified using primary keys and foreign keys in a logical data model. In a conceptual data model, relationships are simple, not specified, we just know that the two entities are related, but we do not indicate which attributes are used for this relationship [9].

Physical Data Model: What is a physical data model, its characteristics, and an example of this type of data model.

physical data model shows how the template will be built in the database. A model of physical database shows all the table structures, including the column name, data type column, the column constraints, the primary key and foreign key relationships between tables. Characteristics of a physical data model include:

- Specification all tables and columns.
- Foreign keys are used to identify relationships between tables.
- DE standardization may happen as required of the user.
- Physical considerations can cause physical data model to be completely different from the logical data model.
- Physical data model will be different for different RDBMS. For example, the data type for a column can be different between MySQL and SQL Server [40].

The steps in the design of physical data model are:

1- Convert entities into tables.
2- Convert relationships into foreign keys.
3- Convert attributes into columns.
4- Modifying physical data model based on physical constraints / requirements.

The following figure is an example of a physical data model.

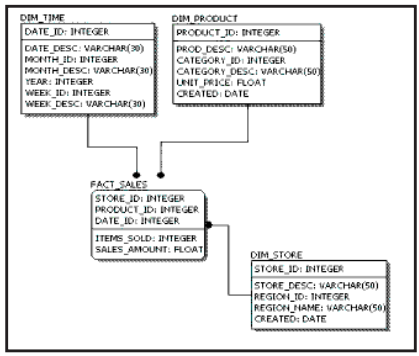


Figure 5 Physical Data Model

Comparing the logical data model above with the diagram of the logical data model, we are seeing the principal differences between both:[9]

1- Entity names have become table names.
2- Attributes are maintaining the column names.
3- The data type for each column is specified. Data types can be different depending on the database used.

Data Integrity: refers to the validity of the data, significance the data is consistently and correctly. In the area data warehousing, we often hear the term "Garbage In, Garbage Out." If there is no data integrity in the data warehouse, reports and analysis result will not be necessary.

In a data warehouse or a data mart, there are three areas where data integrity should be applied:

- **Database level**

We cannot guarantee the integrity of data in the database. Common ways to respect the integrity of the following:

- Referential integrity

The relationship between the primary key table and foreign key in another table should always be maintained. For example, a primary key cannot be deleted if there is still a foreign key that references the primary key.

- Key constraint / Unique primary

Primary keys and UNIQUE constraints are used to ensure that each row in a table can be uniquely identified.

- Not NULL NULL vs-measurement

For columns defined as NOT NULL, they might not have a null value.

- Valid values

Only permitted values is allowed in the database. For example, if a column can have only positive integers, a value of "-1" cannot be allowed.

- **ETL process (Extract, Transform and Load)**

For each step of the ETL process, data integrity monitoring must be established to ensure that data is the same as the data destination. The most common controls include the number of records or record amounts.[11]

- **Access level**

We must ensure that data is not altered by any unauthorized means or during the ETL or data warehouse. To do this, there must be guarantees cons unauthorized access to data (including physical access to servers) access and logging of all historical data access. Data integrity cannot insure if there is no Allowed access to data.

What is OLAP: OLAP (Online Analytical Processing) is a data processing which allows a user to easily and selectively extract and view data from different perspectives. For example, a user can request that the data be analyzed to display a spreadsheet showing all products beach ball sold a company in Florida in July, compare revenue figures with those of the same products in September then see a comparison of other product sales in Florida in the same period. To facilitate this type of analysis, OLAP data is stored in a multidimensional database. While a relational database can be considered as two-dimensional, a multidimensional database considers each data attribute (such as product, geographic sales region, and time period) as a separate "dimension ". OLAP software can locate the intersection of dimensions (all products sold in the eastern region above a certain price for a certain period of time) and display them. Attributes such as time periods can be decomposed into sub-attributes.

OLAP can be used for data mining or the discovery of previously undiscerned relationships between data elements. An OLAP database does not need to be as large as a data warehouse, as all transactional data is necessary for the analysis of trends. Using Open Database Connectivity (ODBC), data can be imported from existing relational data bases to create a multidimensional database for OLAP [10,4].

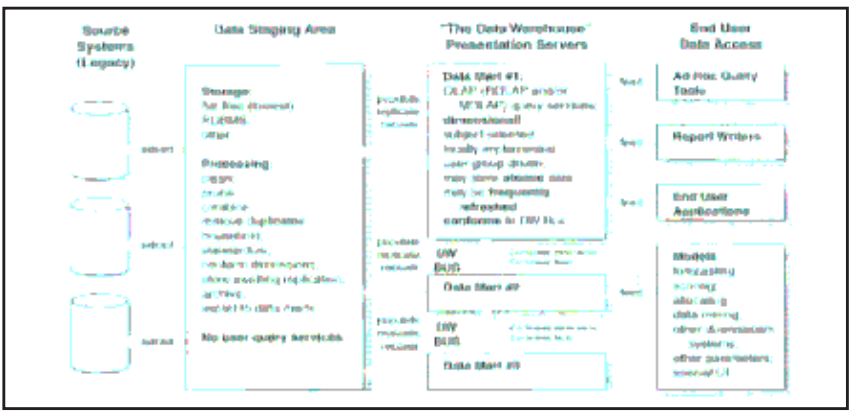


Figure 6 The Basic Elements of the Data Warehouse [4]

2.5 Data Warehouse Architecture

The structure which carries all the components of a data warehouse is well known in the architecture. It contains a number factors. Mainly, it includes the integrated data which is the centerpiece. The architecture consists all for provide information from the data warehouse and is also composed of the rules, procedures and functions which enable data warehouse to work and serve the needs of the business. Finally, the architecture is composed of technology that allows the data warehouse. The architecture supplies the framework for development and implementation of the data warehouse [11].

DW components offer:

• The overall technical architecture.

• Source, acquisition, cleansing and transformation tools.

• Database data warehouse.

• Metadata

• Access Tools

• query and reporting tools (OLAP, data mining)

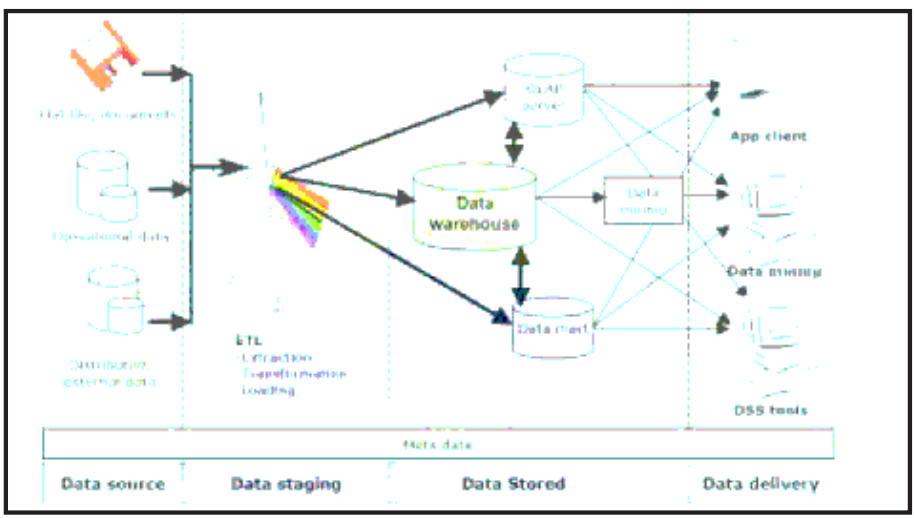


Figure 7 DW architectural and components

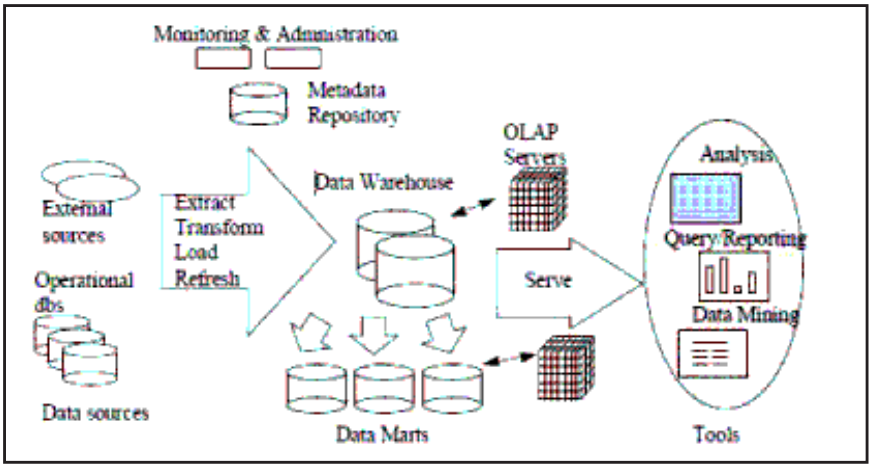


Figure 8 Data Warehousing Architecture

2.5.1 Overall architecture

The architecture is not merely a suite of tools which are necessary to carry out the features and supply services. When referring to the function of extracting data inside one of the architectural components, it is easy to talk about the function itself and the tasks associated with this function. The tools are the means to implement the architecture. Therefore, the architecture is chosen first and tools second. Selection tools should be allocated to the data warehouse architecture. Whenever the architecture is established, the architectural plan must include all related components. The plan also includes in detail all functions, processes, procedures and data warehouses of each architectural element. Architectural plans serve as a model for the design and development. It will also list the masters of the selection tools [12, 27].

2.5.2 Source system

Source data [2] in the data warehouse may be grouped into four categories;

1. Production data

This category of data is different from existing operations' of the company based on the information needs in data warehouse systems. while processing the data, many variations in data formats and data reside on different hardware platforms can be found. Therefore, the DW must support all different types of database[12].

2. The internal data

In many organizations, users can maintain their spreadsheets "private" documents, client profiles, and even database ministry. These are internal data, parts of which might be useful in the data warehouse. If an organization conducts business with customers on a one-to-one basis and the contribution of each client's bottom line is important, then the profiles with detailed demographic data of the sample are important customers in a warehouse data [10].

3. Archived data

Operating systems are mainly designed to execute the current business. In any operating system, the old data is stored in archive files. The circumstances of the organization determine the frequency and portions of operational databases are archived for storage. Some data are archived after one year. Sometimes the data are left in the database of the operating system, as long as five years [11].

Lot of different archiving methods exist. There are steps archiving methods. In the first step, recent data are stored in a database separate archiving data may still be online. In the second stage, the oldest data are stored in a flat file on disk storage. In the next step, the oldest data is archived to tape cartridges or microfilm and even be stored off site.

4. External data

Most leaders rely on data from external sources for a large percentage of the information they use. They utilize statistics for their area produced by external organizations. They can use data on market shares of competitors and standard from financial indicators for their business to check their return values.

The objectives of these external data sources cannot be met by the available data within the organization itself. The knowledge gained from production data and archived data are somewhat limited. They give an image based on what we do or have done in the past. To identify industry trends and compare performance against other organizations, we need data from external sources [8].

2.5.3 Staging area

This is where all the extracted data is gathered and prepared for loaded into the data warehouse. The staging area [11] is like a joining or a construction zone factory. In this field, we look each extracted file, review the business rules, perform various functions for transforming data, sort and merge data, resolve any discrepancies, and clean data. When the data is ultimately prepared either for enterprise data enterprise-wide or a conformed data

marts, data resides temporarily in the staging area need to be attended charged to the referential warehouse data.

In major data warehouses, data staging area is kept in sequential or flat files. However, these flat files include fully incorporated and cleaned data formats suitable ready for loading. Generally these files are in the format may be charged by the utility tools RDBMS data warehouse. Data staging are stored for long periods. Although the extraction data loading can be readily obtained in the relational database form with suitable index, the establishment and maintenance of these relational databases implies overhead for index creation and migration from the system data source.

The staging area can contain data in any grit that fills the tables containing trade measures. Are also frequently for the aggregated data to be stored in the staging area for loading.

Other types stored in the staging area data relate to the basic dimensions of business data such as product, time, sales region, customers and promotional programs.

- **ETL (data extraction / transformation / loading)**

Sights that pertain to ETL [9, 10, 11, 12] in a data warehouse are far more time and intensive human because all activities should be actually achieved in improving the quality and integrity of the data. Consequently, all activities must be performed with the utmost care

Data Extraction

To clearly identify all of the internal data source. Include all computing platforms and source files from which the data is to be pulled. If we inclusion of external data sources, determine the consistency of our data structures with those external sources. Also report the methods of data extraction.

Data transformation

Many kinds of transformation function is required before the data can be mapped and developed for loading into the data warehouse repository. such features include input selection, the separation of input structures, normalization and destandardize source structures, aggregation, conversion and solving values of the mission and name and address conversions. Look each data item provided to be stored in the data warehouse cons the components of data and determine the mappings and transformations. And to determine the-mappings and transformations.

Data loading

Set Baseline load. Determine how many times each major group of data must be maintained in the data warehouse. How many updates will be updated every night? Does the mandate of

the environment over a cycle update in a day? How the changes will be captured in the source systems? Define how updates daily, weekly and monthly updates will be initiated and completed.

2.5.4 Data warehouse database

The central database data warehouse [1, 4, 10, 11] is a cornerstone of data warehousing environment. This database is nearly always brought into play the management system relational database (RDBMS) technology. However, implementation of the warehouse based on the traditional RDBMS technology is often constrained by the fact that traditional RDBMS implementations are optimized for the treatment of transactional data. Certain data warehouse attributes such as size quite large database, the processing of ad hoc queries, and the necessity of for the creation of hose user, including aggregate multilink joined, and drill-downs have become drivers various technological approaches to the warehouse data base data. These approaches consist.

• The basic design of parallel relational database that need a parallel calculation platform, such as symmetric multiprocessing (SMP), massively parallel processor (MPPS), and / or groups of unilateral or multiprocessors.

• An innovative approach to speed up a traditional RDBMS using the new index structure around scans of relational tables.

• Multidimensional Database (MDDBs) which are based on proprietary technology database or implemented utilizing previously familiar RDBMS.

• Basics of multidimensional data are designed to overcome the restrictions located on the warehouse by the nature of the relational database model. this approach is closely coupled with the tools of online analytical processing which act as clients to store multidimensional data. architecturally these tools part of a group of common data warehouse categorized as data query, reporting components, analysis and extraction tools.

2.5.5 Database model

2.5.5.1 Star schema

The star schema is the single data warehouse schema. This is known a star schema because the diagram of a star schema looks like a star, with points beaming from a center. The center of the star is composed of one or more fact tables and the points of the star are the dimension tables [11].

A star schema is marked by one or more very large fact tables that include the primary information in the data warehouse and a certain number of tables of much smaller dimensions

(or lookup tables), each of which includes information the entered for a specific attribute in the fact table.

A star query is a join between a fact table and a certain number of lookup tables. Either lookup table is joined to the fact table using a primary key to foreign key join, but the lookup tables are not connected to each other. A typical fact table includes keys and measurement A measurement is typically a numeric column or character, and can be taken from a column in a table or a derivative of two columns in a table or two columns from multiple tables.

A star join is a primary key to foreign join the dimension tables to a fact table key. The fact table normally has a chained index on the key columns to ease such joins.

The main advantages of star schemas are that they:

- Provide a direct and predictive cartography between the business entities during analysis by end users and the schema design.
- Delivers highly optimized for the performance of typical data warehouse queries.

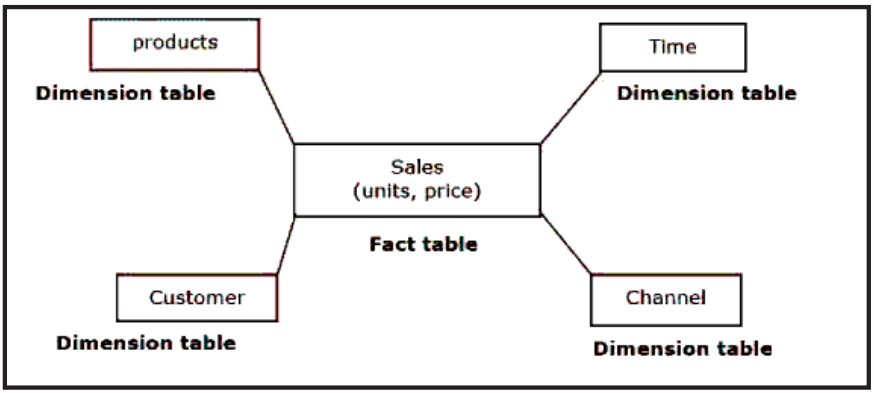


Figure 9 Star schema[11]

2.5.5.2 Snowflake schema

The snowflake schema is a warehouse model more complicated than star schema data, and is a form of star schema. This is known a snow flake schema since the diagram looks like a snowflake [11].

Snowflake schemas standardize dimensions to eliminate redundancy. In other words, the dimension data has been divided into multiple tables rather than one large table. For example, a product dimension table in a star schema might be standardized in a table of

product, Product_Category table and table Product_Manufacturer in a snowflake schema. Although it saves space, it increases the number of dimension tables and calls for a more foreign key joins. The outcome is more complex queries and lowered query performance.

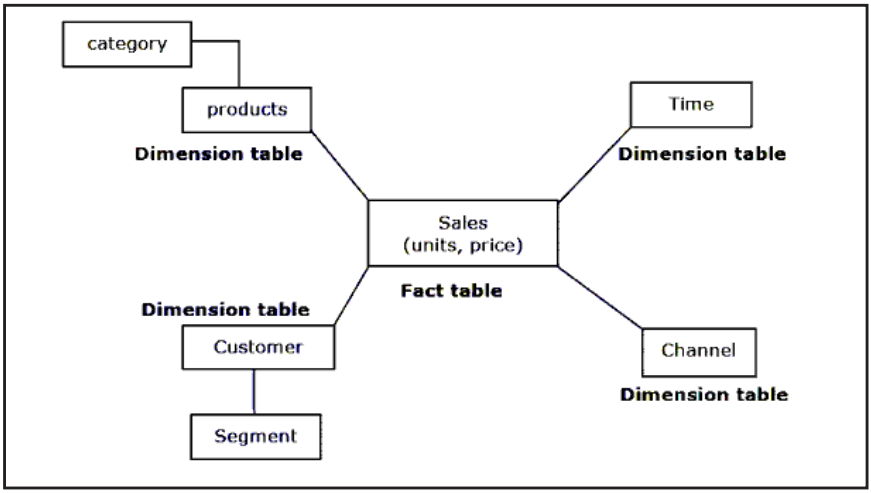


Figure 10 Snowflake schema[11]

2.5.5.3 Fact Constellation Schema

A fact constellation schema is composed of a set of star schemas with hierarchically related tables facts. The links between the different fact tables provide the ability to "drill down" between levels of detail [10, 11]. The figure below, Figure 11 shows an example of constellation diagram of fact.

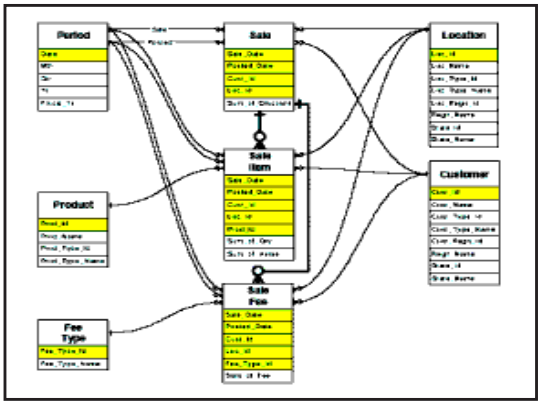


Figure 11 Fact Constellation Schema[11]

2.5.5.4 Galaxy Schema

Galaxy schema is a schema where several tables of dimension tables of information sharing. Unlike a constellation diagram of fact, fact tables in a galaxy does not need to be directly linked [11]. The figure below, Figure 12 shows an example schema of the galaxy.

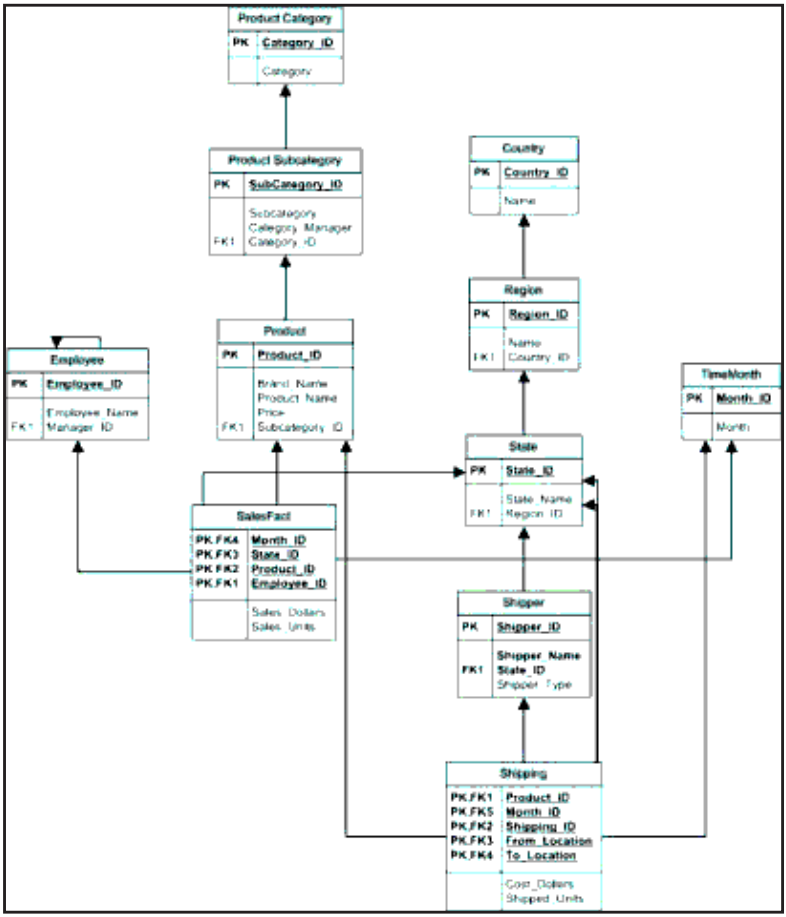


Figure 12 Galaxy Schema

2.5.5.5 Fact and Dimension table

- Dimension tables

Dimension tables [10, 11] represent the business dimensions along which the parameters are tested. It includes attributes that describe that record in the fact table.

Some of these attributes provide descriptive information, others are used to specify how the data in the fact table should be summarized to provide relevant information to the analyst. Dimension tables contain hierarchies of attributes that aid summary.

Dimensional modeling product dimension tables in which each table contains facts attributes that are separate from those in other dimensions. Dimension tables are used to specify how a measure in the fact table must be summarized.

- **Fact tables**

Every data warehouse or data mart comprises one or more fact tables [4, 10, 11]. Plant to a star schema or snowflake, a fact table captures data to measure business operations of the organization. Fact tables usually contain a large number of rows may be several hundred of millions of records when they include one or more years of history to a large organization. A key feature of a fact table is that it contains numerical data (facts) that can be summarized to supply information on the history of functioning of the organization. Each fact table also comprises a multi-part index containing foreign keys as primary keys related dimension tables, which contain attributes that records. Fact tables must not hold narrative information or other data fields of digital measurement and index fields that report the facts to the corresponding entries in the dimension tables

2.5.5.6 Surrogate Keys

It is important that the primary keys of dimension tables remain stable. it is highly recommended that surrogate keys [1, 27] be created and used for primary keys for every dimension tables. Surrogate keys are key components that are maintained within the warehouse rather than keys collected from source systems of data. There are many reasons for the use of surrogate keys:

- Data tables in different source systems may use different keys for the same entity. Older systems that supply historical data might have used another system of numbering system being transaction processing online. A surrogate key uniquely identifies each entity in the dimension table whatever its source button.

A separate field may be used to contain the key used in the source system. Systems developed independently in the divisions of the company cannot utilize the same keys, or

they can use the keys that are in conflict with data in the systems of other divisions. This may not cause issues in each division irrespective reported summary data, but it cannot be tolerated in the data warehouse where the data is consolidated.

- Key may change or be reused in systems sources. This condition is generally less likely than others, but some systems have been known to reuse keys belonging with outdated data. But, the key can be even used in the historical data warehouse and the same key cannot be used to identify different entities.

- Changes in organizational structures are movable keys in the hierarchy. this may be a common situation. For example, if a salesperson is moved from one area to another, the company may prefer to monitor two things: sales data for the seller with the region of origin of the person for the data prior to the date transfer, and sales data for the seller in the new region of the person after the date of transfer. to represent this organization data, the record of the real estate agent must be in two locations on the dimension table of the sales force, which cannot be if the corporation identification number of the employee business is used as the primary key of the dimension table. A surrogate key allows the same seller to participate in various places in the dimension hierarchy.

In this case, the seller will be represented twice in the dimension table two keys with various substitution. These surrogate keys are used to join the files seller for sets of facts relevant to different parts of the hierarchy occupied by the seller. The identification number of the employee must be done in a separate table in the order that information on the employee column may be revised or summarized regardless of the number of times the employee's file appears in the dimension table.

The implementation and management of surrogate keys is the responsibility the data warehouse. OLTP systems are rarely affected by these situations and the goal of these keys is to accurately monitor history in the data warehouse. Surrogate keys are held in the staging area of the data in the process of transforming data.

2.5.6 Metadata

Metadata [29] (The catalog information) describes the data in the data warehouse (eg, data elements and business-oriented description) and the source of the data and the transformations or branches that may have been carried to create the data item.

2.5.7 Delivery Information

Deliver information [2, 11] consisting makes it easy for users to access information directly from the company-wide DW, dependent data marts, or all data marts consistent. The majority

of access to information in a DW is through online queries and interactive analysis sessions. However, DW will also produce scheduled and ad hoc reports.

In many DW, data flow also helps the decision specialized downstream applications as executive information system (EIS) and data mining. other most common flow rate information is the foundation of exclusively data for multidimensional OLAP. In the component of the information delivered, we can supply application services form the user's desktop, and application server, or from the database itself. This will be a crucial for the architecture design decisions.

Duties and Services in the area of information delivery:[11, 12]

• Ensure security to control access to information.

• Access for users to monitor and improve the service for future improvements.

• Enable users to browse the contents of the data warehouse.

• Facilitate access by concealing the internal complexity of the data warehouse users.

• Auto reformat queries for optimal performance.

• Allow queries to be familiar of global arrays for quicker results.

• Direct requests and control runaway queries.

• Provide self-service report generation for the user, consisting of a variety of flexible options to create, schedule and run reports.

• Sets store results of queries and reports for future use.

• Provide multiple levels of grain of the data.

• Deliver event triggers to control the loading of data.

• Arrange for users to perform complex analysis through online analytical processing (OLAP).

• Permit the flow of data to support systems dedicated downstream decisions as EIS and data mining.

2.5.8 Access tools

The main objective of data warehousing is to supply information to business users for strategic decision making. These users interact with the data warehouse using front-end tools. A lot of these tools require an information specialist, although multiple users and develop expertise in tools. Access tools into five major groups:

- Request data and reporting tools
- Tools for application development
- Information Management System (EIS) tools

- OLAP tools (online analytical processing)
- Tools for Data Mining

2.5.9 Data mart

The data mart is a model that represents the same data structure with the data warehouse. They are prepared for the specific needs of the entire organization or a part of it. The data mart contains less data that gives users the benefits. First, it allows to work with faster queries. Another advantage is mobility through it requires less hard drive space so that the user can perform the data warehouse with the laptop. During the process of designing data marts, it is possible to follow two different methods to collect data. One option is to collect granular data warehouse enterprise data and process them according to the needs around which the data mart was prepared. The second option is to collect data in the form directly to the data mart. The data, which is designed to meet the demands of data mart and are stored in the central repository of all enterprise data. In Figure , the options can be considered [10].

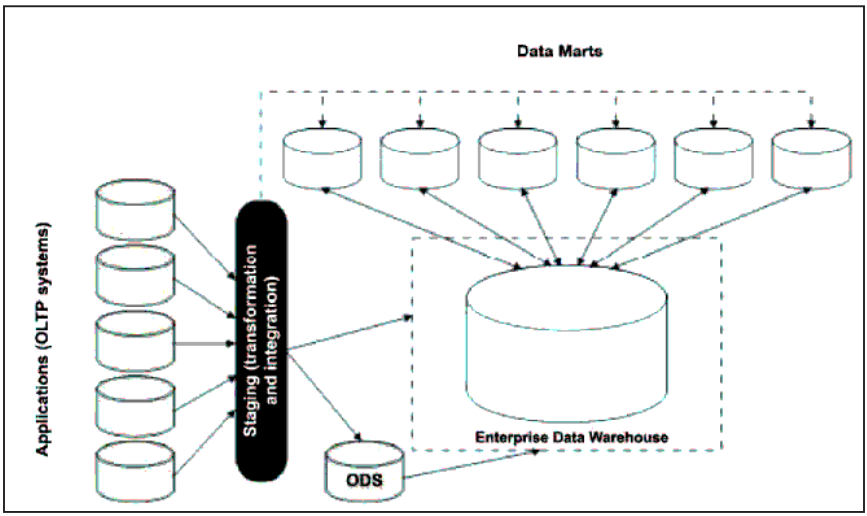


Figure 13 Figure Enterprise Information Architecture [11]

While data marts appear to have advantages over DW, there are questions that must be addressed about data marts.

Size: While data marts are considered smaller than the data warehouse, the size and complexity of some data stores can match a small company DW. As the size of a data warehouse increases, it is likely to have a decline in performance.

Load performance: The time and load performance of the end-user response data are essential tasks of data marts. To increase the response time, data marts typically contain many summary tables and aggregations that have a negative effect on the performance of loading.
User access to data from multiple data Marts: a solution to this issue is the construction virtual data marts that are views multiple physical data marts.
Administration: With the increasing number data marts, management needs derived from coordinate data mart such as version handling the coherence, integrity, security, and performance optimization.

2.6 Data Warehouses, OLTP, OLAP, and Data Mining

A relational database is designed for a specific use. Because the goal of a data warehouse differs from that of an OLTP, the design characteristics of a relational database that supports a data warehouse different design features an OLTP database.

Data warehouse database	OLTP database
Conceived for the analysis of business measures by categories and attributes	Conceived for business operations in real time
Optimized for bulk loads and large, complex, unpredictable queries that access many rows per table	Optimized for a shared set of transactions, generally added or recover one row at a time per table
Loaded with valid, coherent, do not require real-time validation	Optimized for validation of entry data during transactions, utilizes data validation tables
Support for multiple concurrent users compared to OLTP	Support thousands of concurrent users

2.6.1 A Data Warehouse Supports OLTP

A data warehouse supports an OLTP system by providing a place for the OLTP database data unloading as it builds up, and supplying services that would complicate and degrade OLTP operations whether they were made in the database OLTP database. Without a data warehouse to store historical information, the data is stored on a static media, such as magnetic tape or accumulate in the OLTP database [10].

If the data is simply stored for conservation, it is not available or held for analysts and policy makers. If data is allowed to accumulate in the OLTP so it can be used for analysis, OLTP database continues to grow in size and needs more analysis to index and query service report. These requests access process and large portions of historical data continues to grow and add a significant burden to the database. The major indices required to support these applications as taxing OLTP transactions with additional index maintenance. These queries can also be complicated to develop because the scheme OLTP database generally complex.

A data warehouse relieves historical data OLTP, allowing to run OLTP transaction efficiency peak. The volume of records and the high ratios requests are handled by the data warehouse and do not charge the OLTP, which does not need additional evidence to support. Because the data is moved to the data warehouse, it is also reorganized and consolidated so that analytical queries are easier and more effective [33].

2.6.2 OLAP is a Data Warehouse Tool

Online analytical processing (OLAP) is a technology designed to deliver superior performance for ad hoc queries of Business Intelligence. OLAP is designed to work effectively with data organized in accordance with the dimensional model commonly used in data warehouses.

A data warehouse provides a multidimensional view of data in an intuitive model designed to meet the types of questions asked by analysts and policy makers. OLAP organizes data warehouse data into multidimensional cubes based on this dimensional model, then pretreat these cubes to provide maximum performance for queries that summarize the data in different ways. For example, a query that asks for the total sales revenue and quantity sold for a range of products in a given period of time specific geographical area can usually be answered in a few seconds or less regardless of how many hundreds of millions of lines of data are stored in the database of data warehouse data.

OLAP is not designed to store large amounts of text or binary data, it is not designed to support operations update at high volume. The inherent stability and consistency of historical data in an OLAP data warehouse can provide the outstanding performance by quickly summarizing information for analytical queries [31, 37].

2.6.3 Data Mining is a Data Warehouse Tool

Data mining is a technology that applies sophistication and complexity algorithms to analyze the data and display valuable information for analysis by policy makers. While OLAP organizes data into a model adapted to the exploration by analysts, data mining analysis makes on the data and delivers the results to decision makers. Thus, OLAP supports model-driven analysis and extraction of data supports the analysis focused on the data. Data mining is traditionally used only on raw data in the database data warehouse data or, more commonly, the text data fetch from the database data warehouse of data files. In addition, the results of data mining can be incorporated into OLAP cubes to further improve the model-driven analysis by providing additional dimensionally in the OLAP model. For example, data mining can be used to analyze sales data against customer attributes and create a new cube dimension to assist the analyst in the discovery of the information contained in the data cube [10, 27].

Chapter 3: DESIGNING A DATA WAREHOUSE

3.1 Requirements for Data Warehouse Management Systems Database

In the implementation of a DW solution, many technical aspects need to be considered. When one of OLTP systems (DBMS) data management includes only performance transaction processing (essentially, a transaction must be completed within the minimum time and without obstacles, and with the support of thousands of transactions per second) The relational DBMS (RDBMS) suitable for storage of data has the following needs [9];

- **Performance Load**: Data warehouses require incremental loading data periodically whereby the performance of the charging process must be as gigabytes of data per hour.
- **Load processing**: Data Conversion, filtering, indexing and reformatting may be required for loading data into the data warehouse. This process should be performed as a single unit of work.
- **Management of data quality**: The warehouse must ensure coherence and referential integrity despite different data sources and large data size. The measure of success of a data warehouse is the capacity to meet business needs.
- **Query Performance**: Complex queries must fulfill within reasonable periods.
- **Terabyte scalability**: The data warehouse RDBMS should have no boundaries database size and will deliver recovery mechanisms.
- **Mass scalability of the user**: The warehouse RDBMS must be capable of support hundreds of simultaneous users.
- **Administration Warehouse**: Simple to use and flexible administration tools there for warehouse management data.
- **Advanced query capabilities:** The warehouse RDBMS must supply advanced to enable end users perform advanced computations and analysis analytical operations.

3.2 Designing a Data Warehouse

Designing a warehouse means to fulfill all the requirements outlined in Section 3.1 and, of course, is a complex process. There are two main components to build a DW, the design of

the interface between operating systems and the design of the DW [6]. DW design is different from a design-driven classical systems demands.

3.2.1 Beginning with Operational Data

Establishment of the DW does not only imply the extraction of operational data and enter the warehouse (Figure 14).

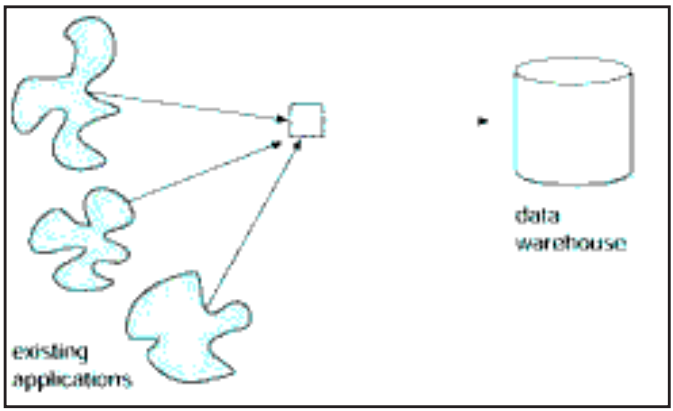


Figure 14 Data Extraction

Revoke the data in the DW without integrating is a big mistake (Figure 15).

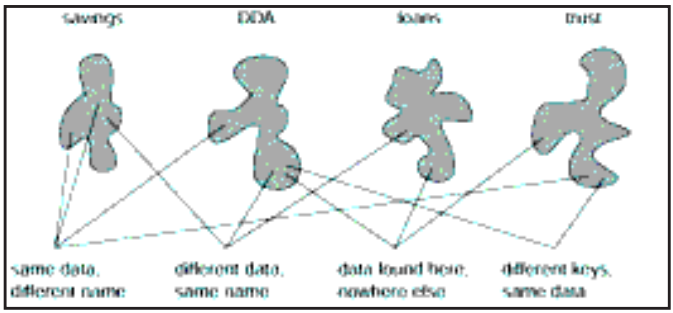


Figure 15 Data Integration

Existing applications have been designed with their own requirements and integration with fellow applications that do not care much. These results in data redundancy, ie the same data

may occur in other applications with the same sense, with different name or different measurement (Figure 16).

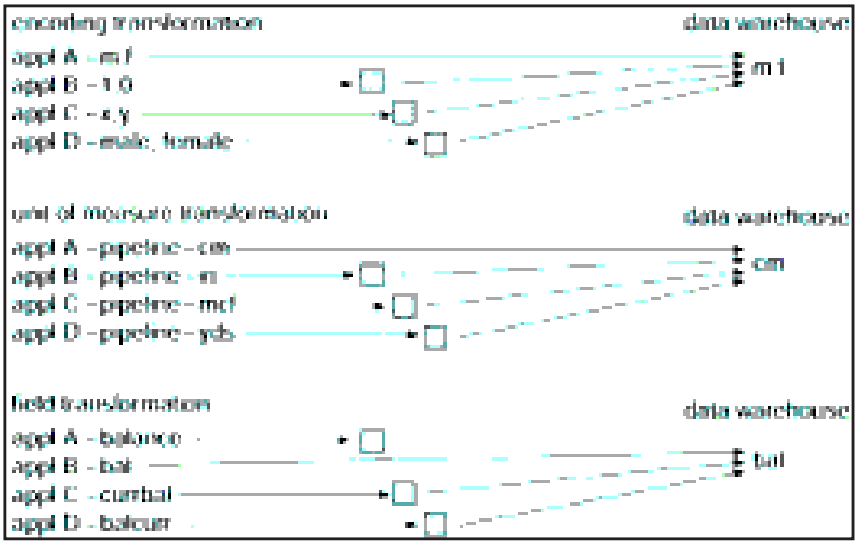


Figure 16 Same data, different use

Other problem is the performance of data access of existing systems. The environment of existing systems has gigabytes and maybe terabytes of data, and try to sweep all this every time a load DW needs to be done is the resource and time consuming and unrealistic.

Three kinds of data are loaded into the DW operating system:

- Archival data
- The data currently included in the operating environment
- Amendments to the environment DW updates that took place in the operating system since the last update.

Five joint methods are used to reduce the amount of operational scanned to update the DW data:[27]

- The scan data was stamped in the operational environment.
- Analyze a "delta" file. A delta file contains only the changes in demand due to the operations that were performed by the operating environment.
- Analyze a log file or an audit file created by the transaction processing system. A log file contains the same data in a delta file.
- Modify the application code.

- Rubbing a 'before' and 'after' image of all file operations.

Is another challenge that the operational data undergoes a lag time basis during his stay in the DW. The accuracy of data mining is valid when it is available, after that it can be updated. However, when the data is loaded into the warehouse, it can not be updated anymore, so a time element to be attached. Other problem when passing data is the need to manage the volume of data that can be found in and enters the warehouse. Volume of data in the DW will grow rapidly.

3.2.2 Data/Process Models

The process model is applicable only to the operational environment. The data model is applied to both the operational environment and environmental DW.

A process model consists:[12]

- Functional decomposition
- Context-level zero diagram
- DFD (Data Flow Diagram)
- Structure chart
- State transition diagram
- Hierarchical input process output(HIPO) chart
- Pseudo code

A process model is priceless, for example, during the construction of the data mart. The process model is based on the requirements, it is not suitable for DW. The data model is applicable to both the environment of existing systems and the DW environment. A comprehensive data model business was built without taking into account the distinction between existing operational systems and DW. The enterprise data model focuses only on primitive data. Performance factors are added to the enterprise data model as the model is transported to the environment of existing systems. Even if some changes are made to the data model for the business operating environment, other changes are made to model business data for use in a DW environment. First, the data are used only in the operating environment is deleted. Then, the key structures of the data model of business are reinforced with a time element. Derived data is added to the data model of business where the derived data is used publicly and calculated once, not repeatedly. Finally, the data relationships in the operational environment are transformed into "artifacts" in DW. A final design activity in the transformation of enterprise data model for the data model data warehouse is to perform a

"stability" analysis. Stability analysis involves grouping of attributes and data on the basis of their tendency to change.

3.2.3 The DW Data Model

Contain three levels in the process of data modeling: high-level modeling (called ERD, the level of the relationship of the entity), the modeling of intermediate level (called the set of data item, or DIS) and modeling of low level (called the physical model).

3.2.3.1. High-Level Modeling

The high level of modeling characteristics entities and their relationships. The entity name is surrounded by an oval. Relationships between entities are represented by arrows. The direction and the number of arrowheads indicate the cardinality of the relationship, and direct relationships are shown ....

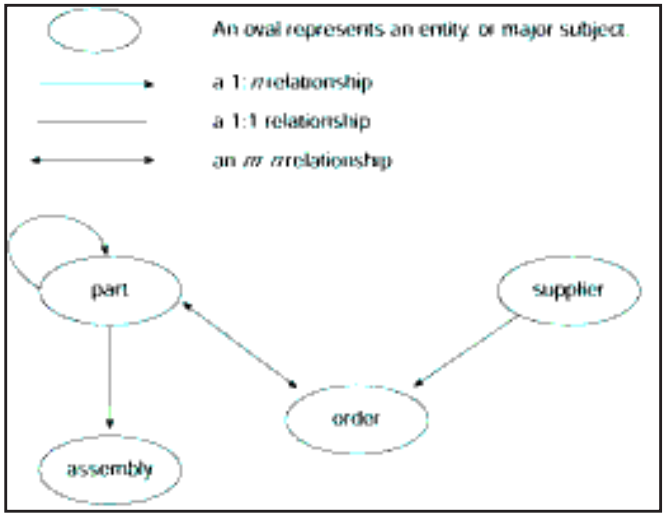


Figure 17 A easy ERD for a manufacturing environment

Entities that are presented in the ERD level (see Figure 17) are at the highest level of abstraction. The ERD company as shown in Figure 18 consists of several individual reflect the different views of people across society ERDs. Data at high level separate models were created for the different communities in society. Together they form the company ERD [40].

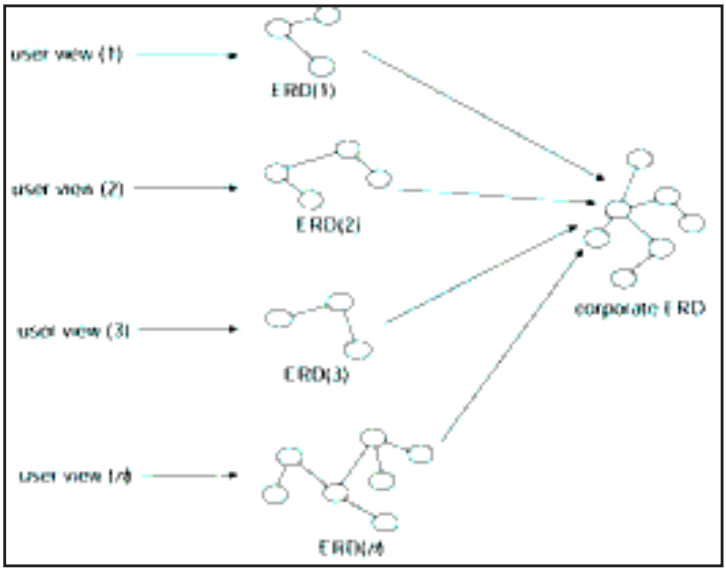


Figure 18 ERD company established by departmental ERDs

3.2.3.2 Mid-Level Modeling

Post the data model level is created, the next level is established, the model of intermediate level or DIS. For each major natural or legal field, identified in the model High level data, a model of intermediate level is created. Each area is then developed its own model of intermediate level (see Figure 19) [40].

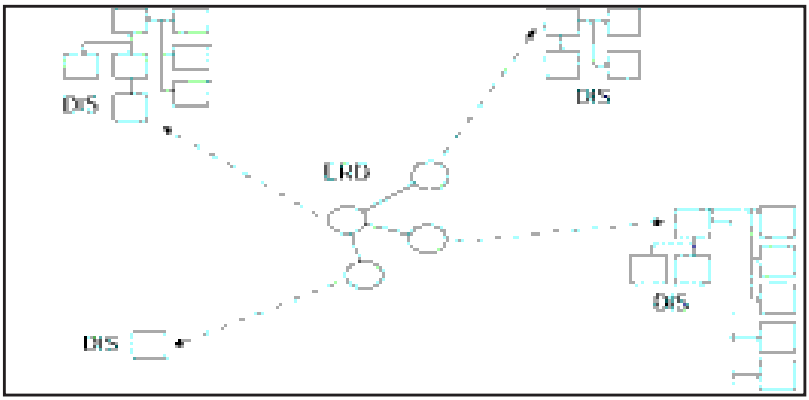


Figure 19 Relationship between ERD and DIS

Four basic concepts are at intermediate level model (also shown in Figure 20):

- A elementary group of data.
- A secondary group of data.
- A connector, suggesting the relationship between the major data fields.
- "type" of data.

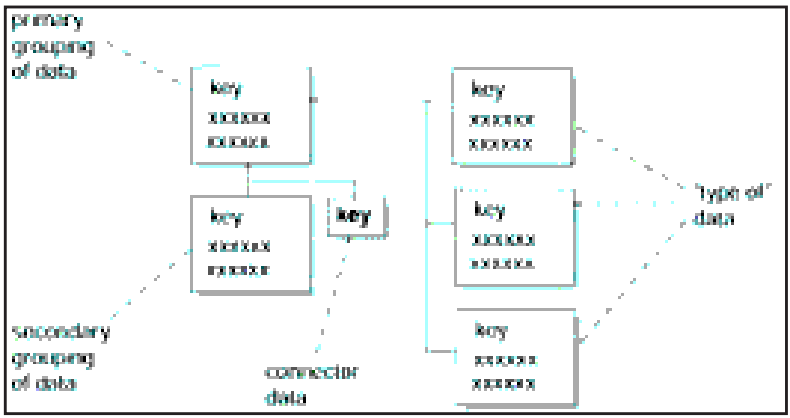


Figure 20 Model Midlevel members

The elementary group is one and only one time for each major area. It contains attributes that exist only once for each primary domain. As with all groupings data, the main group contains attributes and keys for each key area.

The secondary group contains data attributes, which can be several times each primary domain. This grouping is indicated by a line drawn down from the elementary group data. There may be as many as there are secondary groups distinct groups of data that can occur multiple times.

The third construct is the connector. The connector for data from one group to another. Relationship identified level results in a ERD recognition DIS level. The convention used to indicate a connector is a variant of a foreign key.

The fourth building in the data model is "type" of data. "Type" data is indicated by a line to the right of a group of data. The aggregation of data to the left is great. The aggregation of data to the right is the sub-type of data.

These four data modeling concepts are utilized to identify the attributes of data in a data model and the relationship between attributes. When a relationship is identified ERD level, it is manifested by a pair relations connection at DIS [12].

A model example is drawn in Figure 21 below.

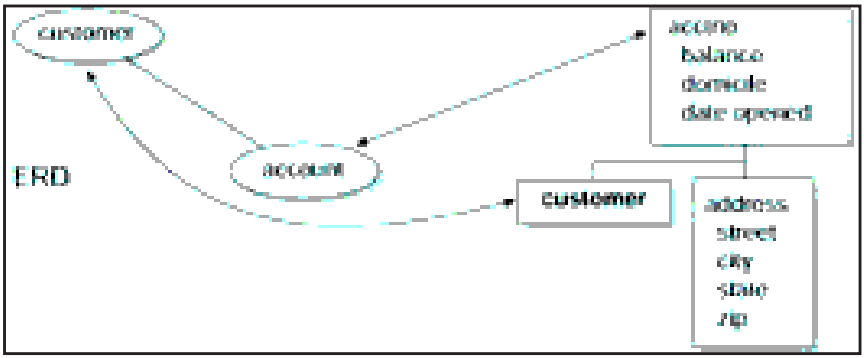


Figure 21 A Midlevel model sample

As ERD company which is created from various ERDs reflecting the user community, DIS company be created from several DIS. Figure 22 shows an example DIS company formed by several departments DIS [12].

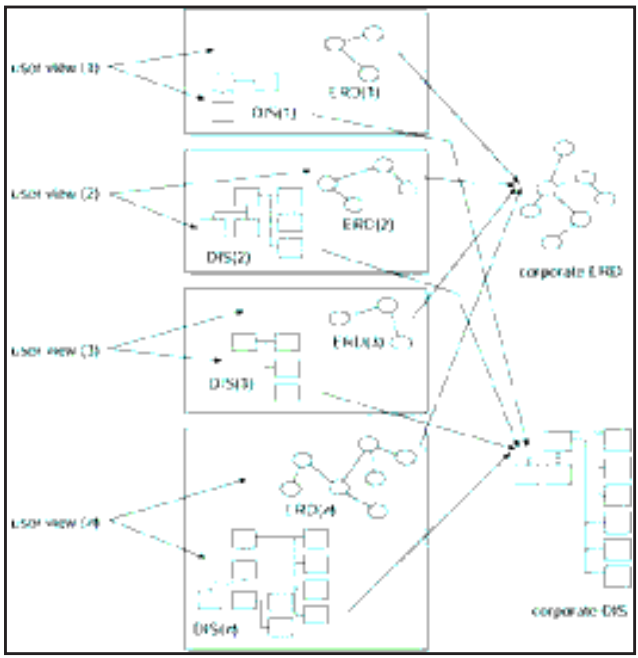


Figure 22 Corporate DIS formed by departmental DISs.

Figure 23 shows DIS of a particular department.

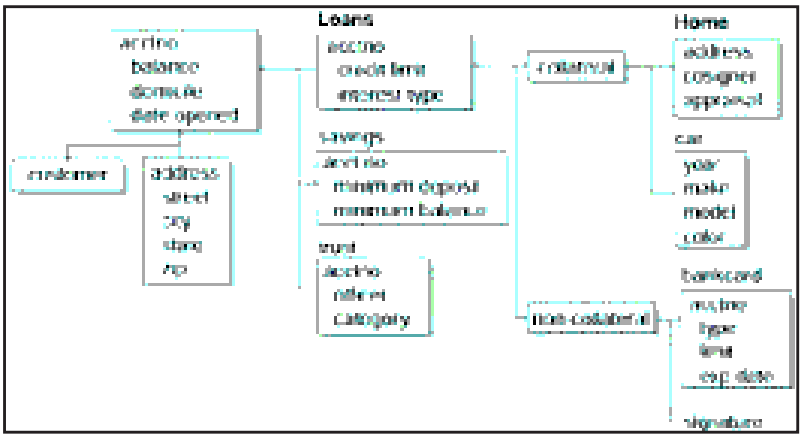


Figure 23 An example of a departmental DIS

3.2.3.3. Low-Level Modeling

The physical data model is built from the data model midlevel only by expanding the data model to include intermediate key and the physical characteristics of the model.

At this stage, the physical data model looks like a series of paintings, sometimes called relational tables. With DW, the first step in this way is to decide on the granularity and data partitioning.

After the granularity and distribution are taken into account a variety of other physical design activities are integrated into the design. At the heart of the physical design considerations is the use of physical input / output (I / O). Physical I/O is the activity that brings data into the computer from storage or sends data to storage from the computer.

The work of designer DW organize data physically to return the maximum number of records from the execution of a physical I / O. Figure 24 illustrate the major considerations in low-level modeling [12].

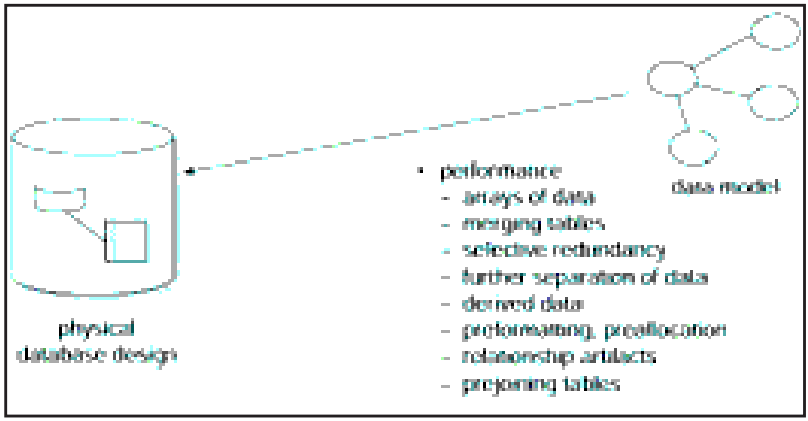


Figure 24 Considerations in low-level modeling

There is another mitigating factor on the physical placement of data in the data warehouse: The data in the warehouse are not normally updated. This frees the designer to use physical design techniques that otherwise would not be acceptable if it is regularly updated.

3.2.4 Methodology data base design for DW

In the next sections of this thesis, I will discuss design methods both conceptual and logical data warehousing. Adopting the terminology of [12, 13, 14, 15, 16] three various design phases are distinguished; manages design concepts that are close to the way users perceive data, discusses the logical design related to a certain type of DBMS concepts , physical design depends on the specific DBMS and describes how data is actually stored [35, 40].

Before starting the discussion, the basic concepts of dimensional modeling must be listed are: facts, dimensions and measures [12, 17, 18].

- A fact is a collection of associated data elements, consisting of measures and context data. It is typically business items or business transactions.
- A dimension is a set of data describing a dimension of the business. Dimensions determine the context for the background facts, which are the parameters that we want to perform OLAP.
- A measure is a numerical attribute of a fact, which represents the performance or behavior the business regarding to the dimensions

Prior this discussion, I also prefer to sum up the proposed methodology Kimball [4], which is accepted as a guru on data warehousing and whose education has prompted many scholars to the study of data warehousing.

There are nine steps to represent the Kimball methodology are as follows [12, 19, 20]:

- Selection Process: The process (function) refers to the subject of specific data mart. The first data mart building should be one that is liable to be delivered on time and budget with the answer to the most important business
- Grain selection: This means resolution precisely what represents a fact table record. Only when the grains are selected to table the fact that we determine the dimensions of the fact table. Resolution grain of the fact table also determines the grain from each of the dimension tables.
- Identify and adapt to the dimensions: Dimensions define the context of questions about the facts in the fact table. A well-constructed set of dimensions makes the data mart comprehensible and easy to use. A set of dimensions poorly presented or incomplete will reduce the usefulness of a data warehouse to a company. When a dimension is used in more than one data mart, the rating is designated as being compliant.
- Select the facts: The grain of the fact table decides which facts can be used in the data mart. All facts should be expressed implied by the grain level. The facts must be numeric and additive. Extra facts can be added to a fact table at any time, provided they are compatible with the grain of the table.
- Equipment for pre-computations in the fact table: Once the facts have been selected every should be reviewed to determine whether there are possibilities to use the pre-calculations.
- To complete the dimension tables: We return to the dimension tables and add a description text for both dimensions. Literal descriptions should be as predictive and comprehensible to users. The utility of a data warehouse is determined by the extent and nature of the attributes of the dimension tables.
- The choice of the term of the database: The term measures the gap in time is the fact table. It is necessary to look at the same period a year or two earlier. Super large fact tables raise at least two very high design issues DW. First of all it is often hard to source more old data. Older data, plus there will be more problems in the reading and interpretation of old files or old cassettes. Second, it is imperative that the older

versions of the important dimensions will be used, not the newer versions. This is called the problem of 'slowly changing dimension'.

- Monitoring of slowly changing dimensions: There are three basic kinds of slowly changing dimensions:
- Type 1: where an attribute modified dimension crashed,
- Type 2: where a changed dimension attribute leads a new dimension record must be created,
- Type 3: altered dimension attribute leads another attribute to be established so that both old and new attribute values are accessed simultaneously in the same file size.
- Deciding priorities and query modes queries: We believe the problems of physical design. The most critical problems affecting the perceived end user of the physical design data warehouse are around physical sorting of the fact table on the disk and the presence of pre-recorded summaries or aggregations. Some more physical design problems relating to the administration, backup, performance indexing and security. We have a design data mart that supports the requirements of a given business process and also allows for easy integration with other data marts linked to finally form the DW company wide.

3.2.5 Conceptual Design Models

The main objective of conceptual modeling is growing a complete range abstract, formal design, based on the needs of users [12, 21].

During this stage of a DW, there is the need to:

- Depicting facts and their properties: Facts properties are generally digital and can be summarized (associate).
- Plug the dimension of the facts: The time is always associated with it
- Represent objects and capture their properties with organizations of these: the object properties (properties summary) can be numeric. In addition, there are three specific types of associations; specialization / generalization (showing objects subclasses of other objects), aggregation (showing objects as parts of an object layer), membership (showing that an object is a member of another class than the same object characteristics and behavior). Strict adherence (or not) (all members belong to one class higher object), full membership (or not) (all members belong to a class of higher object and the object class is made only by members).

- Note the associations between objects and facts: The facts are connected to objects.
- Characterize between dimensions and classify them into hierarchies dimensions governed by membership type associations forming hierarchies that specify different levels of granularity.

3.2.5.1 The Dimensional Fact Model

This model is constructed from ER diagrams [12, 22, 23, 24, 25]. The Dimensional Fact (DF) model is a collection of tree structured patterns whose components are facts, attributes, dimensions and hierarchies sheets. The existence Sheets attributes additively, the optional attributes dimension and dimension attributes cannot be shown on fact schemes. Consistent fact schemes may be overlapped in order to join and compare data.

A fact scheme is structured as a tree whose root is a fact. Fact is represented by an area that indicates the fact name.

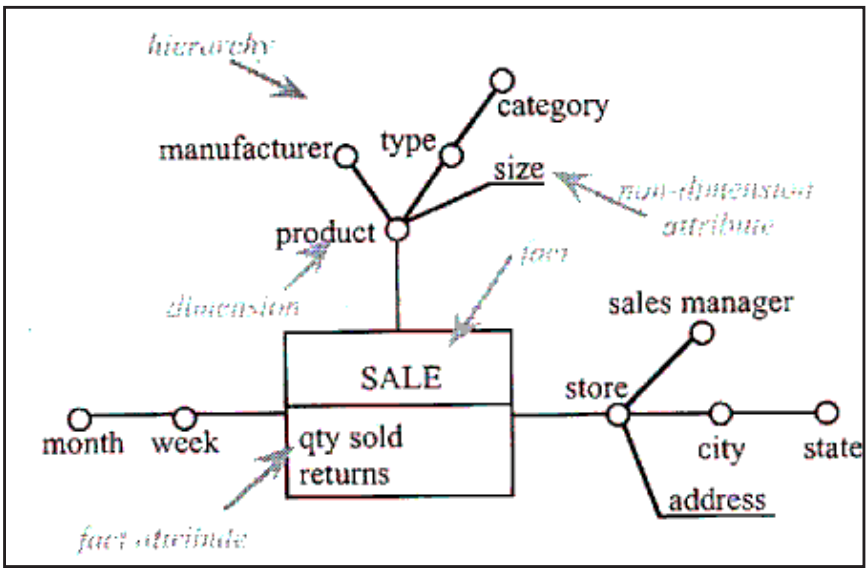


Figure 25 A dimensional fact schema sample

Sub trees rooted in dimensions hierarchies. The circles represent the attributes and edges show the relationship between pairs of attributes. Unlisted attributes (address attribute as shown in Figure 25) are shown by lines rather than circles. Not an attribute dimension contains additional information about an attribute hierarchy is connected to it by one-to-one

and cannot be used for aggregation. Arcs represented by lines express relations between pairs of optional attributes.

A fact expresses a relation many-to-many between dimensions. Each sequence of dimension values that defines an instance of a value for each attribute is. Many of the attributes are additive along all dimensions. This means that the operator amount can be utilized for global and all hierarchies attribute values. An attribute that is referred semi-additive if it is not additive in one or more dimensions, non-additive, except additive along the longest dimension.

DF model composed of five steps:

- Resolution facts (a fact can be represented on the E / R unit with a scheme is F or a n-ary relation between entities E1 to En).
- For every fact;

- Construct the attribute tree

  Every vertex corresponds to an attribute of the schema; root corresponds to the identifier F, for each vertex v, the corresponding attribute operatively shall determine the attributes corresponding to all offspring of v When F is recognized by the combination of two or further attributes, identifier (F) denotes their concatenation must be add some other notes:. It is useful to emphasize that there scheme of relationships between optional attributes in a hierarchy. Associations optional or optional attributes of the E / R scheme should permeate by a dash, a one-to-one relationship can be considered a special kind of many-to-one, therefore it can be inserted into the attribute trees generalization hierarchies of E / R equal to one-to-one relationships between the super-entity and each sub-entity scheme, x-to-many relationships cannot be inserted into the attribute tree. In fact, the representation of these relations at the logical level, for example by a star schema, could not without infringing first normal form, an n-ary relation is the same as n binary relationships. The majority of n-ary relations have maximum plurality greater than 1 in all their branches they identify n one-to-many binary relations that cannot be inserted into the attribute tree.

- Pruning and grafting the attribute tree

  Not every attributes represented in the attribute tree are useful for DW. Thus, the attribute tree can be pruned and spliced to remove levels superfluous detail. pruning is carried out by dropping any sub-tree of the tree. fell the attributes will not be included

in the scheme is, therefore it will be impossible to use aggregated data grafting is used when its offspring must be preserved.

- Setting dimensions

  Dimensions shall be selected from the tree tops attribute amongst children of the root E / R schemes can be categorized as a instantaneous and temporal. Instantaneous scheme describes the state of the field of application; Previous versions of variables on time data is constantly replaced by new versions a timing diagram describes the evolution of the field of application over a range of time; older versions of data are explicitly represented and stored during the design a DW from a timing diagram, time is explicitly touted as a E / R attribute and so it is an obvious candidate to set a dimension. time is unclear shown, but should be added as a dimension of fact schema.

- Attributes that define

  Fact attributes are usually either the number of cases of F, the sum / average / maximum / minimum expressions involving numeric attributes of the tree attribute. A fact may not have attributes, if the only information to be registered is the appearance of the fact.

- Setting hierarchies

  Alongside each hierarchy, the attributes must be organized in a tree such as x-to-one relationship between each node and its descendants. It is still possible to cut and graft the tree to remove unnecessary details. It is also possible to add new levels of aggregation by defining ranges for numeric attributes. In this phase the attributes that should not be utilized for aggregation, but only for informative purposes can be identified as unlisted attributes.

3.2.5.2 Multidimensional E/R Model

Argument is made that the ER approach is not suited for the multidimensional conceptual modeling since the semantics of the major features of the model are unable be effectively represented.

Multidimensional Model E / R (ME / R) contains certain key considerations [11. 12, 26]:

- All trades ER model.
- Minimum extension of ER model, it should be easy to learn and operate for an experiment ER Modeler. A few additional items.

- Representation of multidimensional aspects, in spite of the minimality, specialization must be powerful enough to voice the multidimensional aspects of basis, namely the qualification data and the quantification and hierarchical data structure of qualification.

This model allows the concepts of generalization. There are a few specializations:

- A set of special features: dimension level.
- Two sets of special relations connecting dimension levels:
- a special n-ary relationship set: the "fact" ensemble of relations.
- a special binary relationship set: the 'roll-up for all relations.

The "roll-up" for all relationships, it provides a dimension level A to level B dimension representing concepts of a higher level (city roll-up to another) abstraction.

The "fact" set of relationships is a specialization of a set of n-ary relations in general. It connects n various entities dimension level.

The fact relationship is modeled natural partition to qualify and quantify the data. The attributes of the fact model of the relationship that measures the effect while levels dimension model quantifying data. Design utilizes a special scoring chart that a notation of the sample is illustrated in Figure 26.

Figure 26 The scoring chart of ME / R components

Individual characteristics of ME / R model can be summarized as follows;

- A core element in the multidimensional model is the concept of dimensions that span the multidimensional space. The ME / R model does not include an explicit counterpart to this idea. This is not necessary because a dimension is composed of a set of dimension levels. The data that dimension levels belong to a particular dimension is implicitly included in the graph structure roll-up.
- The structure of hierarchical classification of dimensions is given by the level of entity sets of dimensions and relations roll-up. The relationship rolls-up sets identify a directed acyclic graph of the dimension levels. This enables the simple modeling of

more than one hierarchy, alternative paths and levels of shared hierarchy for various dimensions. Therefore, no redundant modeling shared levels is necessary. Congestion level attributes are modeled as attributes of entity sets dimension level. This enables a different attribute structure for each rating level.

- In modeling the multidimensional cube in a determined relationship, it is possible to include an arbitrary number of elements in the schema representing a "model multi cube". Superior scheme also includes information on the level of granularity at which dimensions are shared.
- Regarding measures and their structure, the ME / R model provides structured records as measures for several attributes that ensemble of relations. The semantic information that certain measures are derived can not be counted in the model. As the E / R model of ME / R model captures the static structure of the application domain. Calculating measures and operational information is cannot be included in the static model. An orthogonal functional model must capture these dependencies.
- Schema consists rolls-up relationship between entities. Consequently the levels of various dimensions can roll up a common parent level. This information can be used to avoid redundancies.
- This model is used "is a" relationship.
- ME / R and ER models and notations can be utilized jointly.

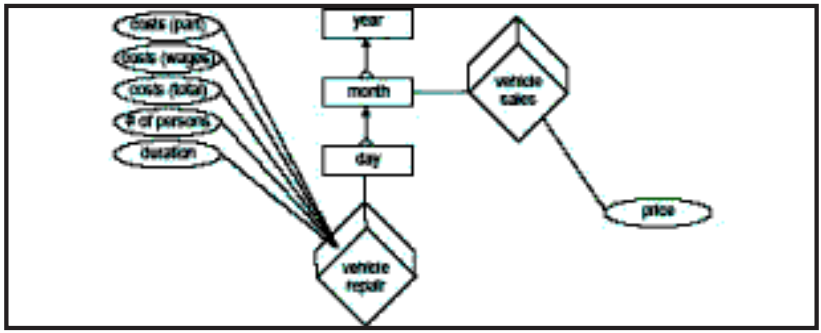


Figure 27 Cubic several sharing dimensions at various levels

As noted above, the ratings of ME / R and ER model can be used together as shown in Figure 28.

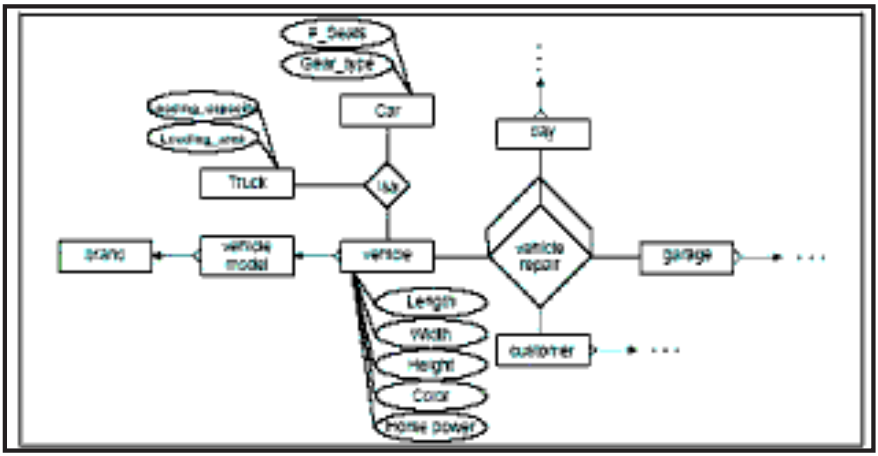


Figure 28 Combining ME/R notations with E/R

3.2.6 Logical Design Models

DW logical design involves setting up structures that allow efficient access to information. The designer built multidimensional structures considering conceptual diagram showing the information needs, the source databases and non-functional requirements. This phase also includes specifications for data mining tools, data loading process, and access to the storage methods. At the end of the logical design phase, a working prototype should be established to the end user.

Dimensional models represent the structure data "cube", which makes longer compatible with the management of OLAP logical representation of data. Goals the dimensional model are [12, 27]:

- To generate structures of databases that are easy for end users to understand and write queries cons.
- Order to maximize the query efficiency.

  It achieves these objectives by minimizing the number of tables and relationships them. Standardized databases have certain characteristics that are suitable for OLTP systems, but not for DW [17]:
- Its structure is not easy for end users to understand and use. In OLTP systems is not an issue because, generally end users interact with the database through a software layer.

- Data redundancy is minimized. This maximizes the efficiency of updates, but tend to hit retrievals. Data redundancy is not an issue in DW since the data is not updated online.

Modeling Dimensionality utilizes the ER model with some important restrictions. Three-dimensional model consists of a table with a composite primary key, called the fact table, and a plurality of smaller tables called dimension tables. Every dimension table has a simple primary key (non-composite) which is exactly one of the components of the composite key in the fact table. This feature structure is called star schema or star join.

Other important characteristic every natural keys are replaced with surrogate keys. It means that each join between fact and dimension tables is based on using surrogate keys, not natural keys. Every surrogate key should have a general structure based on easy integers. The use of surrogate keys enables data in the DW to have some independence from the data utilized and produced by the OLTP systems.

As discussed stated before the logic design is an abstract concept, and if it does not address the physical implementation yet. We only deal to set different types of information in the logical design. The entities of the entity-relationship modeling show a piece of data storage schemes (logical design), but in relational databases, an entity usually maps table. The attribute is a component of an entity that helps define the unique nature of the entity. In relational databases, an attribute corresponds to a column, to ensure that the data is consistent. So we use unique identifiers, it is something that we add to tables so that we can make a distinction between the same point when it appears in different places in the physical design, it called primary key. An entity-relationship helpful for the design of the data warehouse as dimensional modeling, as it is associated with highly standardized models such as applications (OLAP). Within dimensional modeling, we can use the fact table to identify the information owned by a central information part of its own associated dimension tables. [34]. Our logical design should translate: (1) a set of entities and attributes corresponding to fact tables and dimension tables. (2) a model of operational data from your source for information-oriented subject in our destination data warehouse schema.

3.2.7 Physical Design

While logical design phase, you defined a model for your data warehouse constituted of entities, attributes and relationships. The entities are connected to each by relations. Attributes are used to describe entities. The unique identifier distinguishes (UID) between an

instance of an entity to another. During this phase, we translate the expected structures real database schemas, and we deal with the:

- Entities tables.
- Relations with foreign key constraints.
- The column attributes.
- The primary key constraints unique identifiers.
- The unique identifiers to the constraints of unique keys.

To reflect the logical design into physical design, we need to build some or all of the following structures (table spaces, tables and partitioned tables, views, integrity constraints, dimensions, Indexes and partitioned indexes and materialized views).

3.2.7.1 Table spaces

A table space is composed of one or more data files, that are physical structures in the operating system you are running. A data file is associated with one table space. From the standpoint of design, table spaces are containers for physical design structures. Table spaces need to be separated by differences. For example, tables should be separated from their indexes and small tables should be separated from large tables. The table space should also represent logical business units if possible. Since a table space is the crudest granularity for backup and recovery or the transportable table spaces mechanism, the logical business design assigns availability and maintenance.

3.2.7.2 Partitioned tables

Using partitioned instead of those non-partitioned tables addresses the key problem of supporting very large volumes of data, allowing you to break them down into smaller, more manageable pieces. The main design criterion for partitioning is manageability, but you will also see performance benefits in most cases because of the size of a partition or an intelligent parallel processing. For example, you can choose a partitioning strategy based on the date of the sale transaction and a monthly granularity. If you have the data value of four years, you can delete the data of one month as it becomes more than four years with one quick (DDL) and load new data while only affecting 1/48th the complete table. Business issues for the last quarter will only affect three months, which is equivalent to three partitions, or 3/48ths of the total volume.

Partitioning big tables enhances performance because each partitioned piece is more manageable. In general, your score based on transaction dates in a data warehouse. For example, each month, the value of one month's data can be assigned its own partition.

Sectoral data compression can save disk space by compressing heap organized tables. A typical type of organized heap table, you should consider compressing data segment is partitioned tables. To reduce disk use and memory use (specifically, the cache), you can store tables and partitioned tables in a compressed format inside the database. This often leads to a better place at the level of read-only operations. Compression data segment can also speed up the execution of queries. There is, however, a cost borne by the processor. Compression segment data should be used with redundant data, such as tables with many foreign keys. You should avoid compressing tables with much update or other activity (DML). Although compressed tables or partitions are changed, there is an overload to update these tables, and high update activity may work against compression by making a little space to be wasted.

3.2.7.3 Views

A view is a customized presentation of data in one or more tables or other views. A view makes the output of a query and treats it as a table. Views do not demand space in the database.

3.2.7.4 Integrity Constraints:

Integrity constraints are used to enforce business rules associated with your database and to avoid having invalid information in the tables. Integrity constraints in data warehousing differ from constraints in environments (OLTP). Environments (OLTP), they primarily impede the insertion of invalid data in a record, which is not a big problem in data warehousing environments because accuracy has already been guaranteed. In data warehousing environments, constraints are used to rewrite the query. NOT NULL constraints are particularly common in data warehouses. In some specific circumstances, constraints need space in the database. These constraints are in the form of the behind unique index.

Indexes and Partitioned Indexes are facultative structures are involved in tables or clusters. Besides the classic B-tree indexes, bitmap indexes are very common in data warehousing environments. Bitmap indexes are optimized index structures for set operations. In addition, they are necessary for some optimized access methods such as data transformations stars. Indexes are just like tables in that you can partition them, although the partitioning strategy does not depend on the structure of the table. Partitioning indexes renders more easily managed the warehouse during refresh and enhance query performance [36].

3.2.7.5 Dimensions

A dimension is a schema object that defines hierarchical relationships between columns or column sets. Is a hierarchical relationship starting a functional dependence of a

level of a hierarchy to the next. A dimension of one container is logical relationships and does not need to space in the database. A typical dimension is city, state (or province), region and country. Choose the Dimension object from within the Schema icon to display all dimensions. Choose a specific dimension to graphically display its hierarchy, levels and all the attributes that have been defined [36].

3.2.7.6 Indexes and Partitioned Indexes:

Indexes are elective structures are involved in tables or clusters. Besides the classic B-tree indexes, bitmap indexes are very common in data warehousing environments. Bitmap indexes are optimized index structures for tasks on the bundles. Moreover, they are necessary for selected optimized access methods such as data transformations stars. Indexes are just like tables in that you can partition them, although the partitioning strategy does not depend on the structure of the table. Partitioning indexes renders more manageable the data warehouse during refresh and enhance query performance [10, 36].

3.2.7.7 Materialized Views:

Are the results of queries that have been registered in advance for long-term calculations are not needed when you actually run your statements (SQL)? From a design point of view physical, materialized views are like tables or partitioned tables and act as indexes in that they are used transparently and enhance performance [27, 36].

3.2.7.8 Hierarchies:

Hierarchies are logical structures that use ordered levels as a way of organizing data. A hierarchy can be used to determine data aggregation. For example, in a time dimension, a hierarchy might aggregate data from the month level quarter level to the year. Dimension hierarchies equally group levels from general to granular. Query tools utilize hierarchies to let you get into your data to see different levels of granularity. This is one of the main advantages of a data warehouse. During the design hierarchies, you must take into account the relationship between the structures, hierarchies are also essential to allow more sophisticated rewrites.

3.2.7.9 Unique Identifiers:

Unique identifiers are rated for a different record in a dimension table. Artificial unique identifiers are frequently used to prevent the potential problem of unique identifiers changing. Unique identifiers are represented by the # character. For example, #User_id.

3.2.7.10 Relationships:

Relationships are the connections between the data warehouse objects and how to deal these objects together. Figure (29) Example of storage data objects and their relationships click stream fact table and dimension table, keyword, category, date, time and user. As we know, five types of data relationships must be considered when designing databases [38]:

* One-to-one
* One-to-many relationship
* Many-to-many relationship
* Relationship recursive many-to-many.

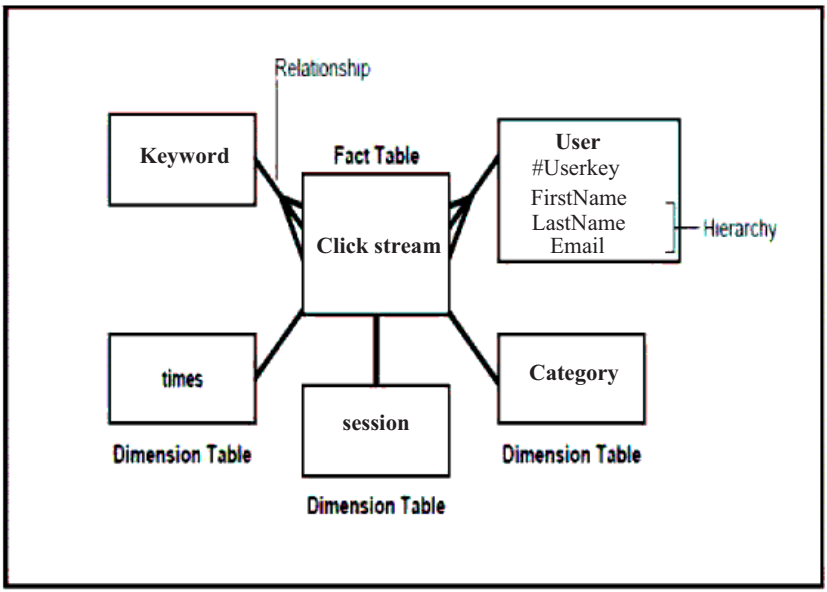


Figure 29 Typical Data Warehousing Objects

3.2.8 On-Line Analytical Processing (OLAP):

Is the interface for the final data warehouse and utilize data in the forms of multi-dimensional (eg, data mining Data Mining and Knowledge Discovery in Databases, Knowledge Discovery in Database (KDD). It is a field of interlaced with other areas, such as databases and methods of artificial intelligence and machine learning and neural networks as well as statistical and field systems knowledge and information retrieval .. Etc.

(OLAP) technology to enhance database query optimization and reporting, instead of analytical operations. The data source (OLAP) databases, online transaction processing

(OLTP), which is generally stocked in data warehouses. (OLAP) data are derived from data gathered in earlier structures allow sophisticated analysis. (OLAP) data is also organized in the hierarchy and stored in cubes instead of tables. It is a complicated technology that utilizes multidimensional structures to supply fast access to data for analysis. This facilitates the organization in a PivotTable report or PivotChart report summaries higher levels. There are five basic operations OLAP [36, 37]:

1 - Roll Up, which is used to reach the lower levels of detail for a given data cube. This command requires the current data cube (object) and makes a GROUP BY on one dimension.

2 - Drill Down, which is used to reach higher levels of detail. This control is the opposite of roll.

3 - Facility, which provides a section through a given data cube. This order enables users to concentrate on a particular portion of the data inside the cube, for example, the user may want to take a look at the data on the user came to the site to find specific information using a meta-search engine.

4 - DICE, which provides a single cell of the cube (the smallest group), for example, it can provide data on a number of special categories that asks the user domain.

5- PIVOT, which rotate the cube to change the perspective, for example, the question of time "perspective can be changed to " instead of time". These commands in terms of indicated and execution are generally made using a point-and-click interface, and consequently we do not describe their syntax. Instead, we provide examples for each of the above commands (OLAP).

Naturally, OLAP engines have business users with multidimensional data from data warehouses or data marts, not concerns about how and where the data is stored. However, the physical architecture and implementation of OLAP engines shall consider issues of data storage. Implementations of a warehouse server engine for OLAP are [12, 27].

In the OLAP world, there are mainly two different types: Multidimensional OLAP (MOLAP) and Relational OLAP (ROLAP). Hybrid OLAP (HOLAP) refers to technologies that combine MOLAP and ROLAP.

MOLAP

This is the more traditional manner of OLAP analysis. In MOLAP, data is stored in a multidimensional cube. The storage is not in the relational database, but in proprietary formats.

advantages:

Strong performance: MOLAP cubes are built for rapidly retrieval of data, and is optimal for slicing and dicing operations.
Can perform complex calculations: All calculations have been pre-generated when the cube is created. Therefore, complex computations are not only feasible but they return quickly. disadvantages:
Restricted in the amount of data it can manage: Because all calculations are performed when the cube is built, it is not possible to include a large amount of data in the cube itself. This does not to say that the data in the cube cannot be derived from a large amount of data. In fact it is possible. But in this case, only summary information will be included in the cube itself.
Requires extra investment: Cube technology are often proprietary and do not already exist in the organization. Therefore, to adopt MOLAP technology, chances are additional investments in human and financial resources are needed.

ROLAP

This methodology is based on handling of data stored in relational database data to give the look of the slicing and dicing capability of conventional OLAP. Essentially, every action slicing and dicing is the same as adding a "WHERE" clause in the SQL statement. advantages:
Can handle vast amounts of data: Limiting ROLAP technology data size is limiting the size of the data base underlying relational database. This means, ROLAP itself imposes no limitation on the amount of data.
Features can leverage incidental to relational database: Often, relational database already comes with a multitude of features. ROLAP technology, because they are sit on top of the relational database, can therefore take advantage of these features.
disadvantages:
Performance can be sluggish: Because each ROLAP report is basically a SQL (or multiple SQL queries) query in the relational database, the request time can be long if the size of the underlying data is important.
Limited by SQL functionalities: Because ROLAP technology is based primarily on generating SQL statements to query the relational database and SQL statements does not fit all needs (eg, it is difficult to carry complex calculations using SQL), ROLAP technologies are thus traditionally limited by what SQL can do. ROLAP vendors have reduced this risk by structure in the complex as well as the capability to allow users to define their own functions out-of-box-office tool.

HOLAP

HOLAP technologies attempt to combine the benefits of MOLAP and ROLAP. For more summary Type, HOLAP builds on cube technology for quicker performance. Where detailed information is needed HOLAP can "break" in the cube in the underlying relational data.

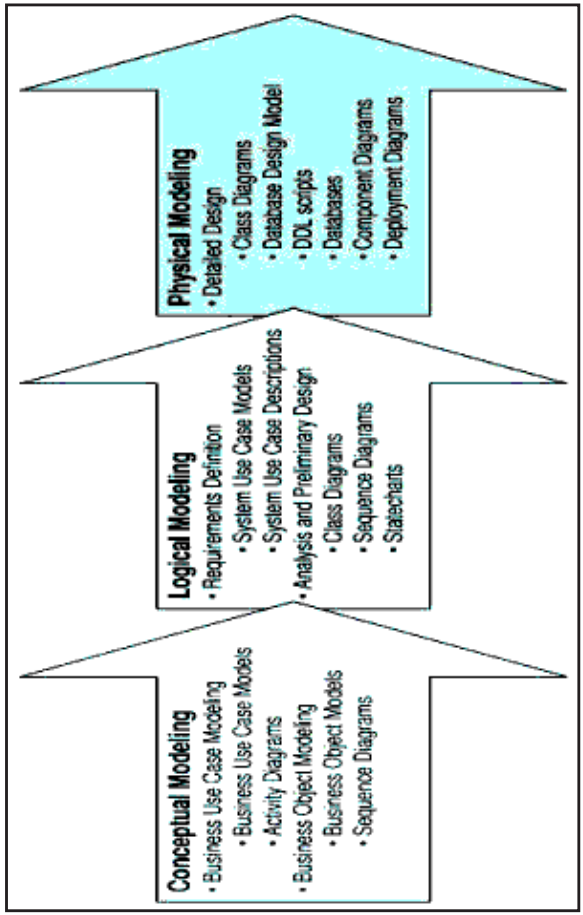


Figure 30 Stages of modeling and related Meta search engine constructs

3.2.9 Compare between Logical and Physical Design

The logical design is that you draw with a pen and paper or design with Oracle Warehouse Builder or Designer before constructing your warehouse. The physical design is the establishment of the database with SQL statements. In the physical design process, you

convert the data collected during the logical design phase into a description of the structure of the physical database. Physical design decisions are primarily motivated by query performance and maintenance aspects of database [37, 35].

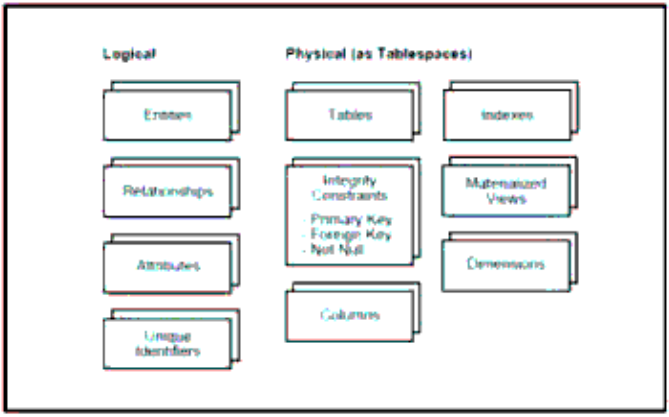


Figure 31 logical design compared with the physical design

3.2.9.1 Dimensional Model Design

This section explains a method to develop a dimensional model from an Entity Relationship model [28].

This data model is utilized by OLTP systems. It does not include redundancy, but a high performance updates, displays all the data and the relationships between them. Simple queries require several table joints and complex sub-queries. It is adapted for technical specialist.

- Rank entities: To generate a dimensional model from the ER model, first classify entities into three categories.

- Entities transaction: Such entities are the most significant entities a DW. They have the highest priority. They build fact tables in a star schema. These entities recording details on special events (orders, payments, etc.) that decision makers wish to understand and analyze. There are some features;

  ♣ It also outlines a event that happens at a point in time.

  ♣ It includes measures or quantities which can be summarized (amount of sales volumes)

- Component entities: They are directly linked to a transaction entity with a one-to-many relationship. They have the lowest priority. They set out details or components

of each transaction. They correspond to the "who", "what", "when", "where", "how" and "why" of the event (customer, product, time, etc..) Time is an important component of any transaction . They build dimension tables in the star schema.

- Entities classification: Such entities are linked with entities comprising a chain of one-to-many relationship. They are operatively dependent on a consisting entity. Such entities represent hierarchies included in the data model, which can be folded to form the entity component dimension tables in the star schema.
- Identify hierarchies: Most dimension tables in the star schema comprise shipped hierarchies. A hierarchy is referred maximum amount if it can not be extended up or down, including another entity. One entity is called minimal if it has no relationship one-to-many. An entity is referred maximum amount if it has no relationship many-to-one.
- produce dimensional models: There are two traders to produce three dimensional models from ER.
- Collapse Hierarchy: Higher-level bodies can be divided into lower level entities in the hierarchy. Collapse of the hierarchy is a kind of destandardization. We cannot go on until we reached the down the hierarchy and end up with a single table.
- Aggregation: This operator may be applied to an entity transaction to establish a new containing entity summary data provided.

A schema is a collection of database objects, including tables, views, indexes, and synonyms. Different ways to organize these objects in models of patterns designed for data warehouses.

There are three types of this scheme:

1- Star schema.
2- Snowflake schema models.
3- Fact constellation/galaxy

In this work we have tried to provide a simple summary about each type of these schemes, and the use of each species for the design of data warehouses for (meta-search engines and data web housing environments), and we will focus on one of these species is (star schema).

And use a variety of database management systems (DBMS) for the design of data warehouses, not all of this (DBMS) can provide all these schemes. For this reason prefers to use Oracle (DBMS) in the practical side of it is designed to support all the schemes of the

data warehouse and the vast majority of Oracle data storage features also apply to the star schemas, snowflake schemas and Fact constellation/galaxy.

3.2.9.2 Star Schema

This is the basic structure of a dimensional model. It has a fact table and a set of tables placed around smaller dimensions of the fact table. The fact data will change over time. Tables of the most relevant facts are numeric and additive for the data warehouse rare access to a single recording applications. They have access to hundreds, thousands, millions of records at a time and aggregates. The fact table is related to all the dimension tables by one to many relationships. It includes measures that can be grouped in different ways [27, 28, 29].

Dimension tables contain descriptive textual information. Dimension attributes are utilized as constraints in the data warehouse queries. Dimension tables provide the foundation for aggregate measures in the fact table. They typically consist of embedded hierarchies. Each star schema is made as follows;

- A fact table is formed for each entity of the transaction. The key table is the combination of keys its related components.
- A dimension table is formed for each component entity by the collapse of entities related hierarchical classification in it.
- When the hierarchical relationships between entities transaction, the child entity inherits all the dimensions (and key attributes) of the parent entity. This allows " drill down" in the level of transaction.
- Numeric attributes in the transaction entities should be aggregated by the material attributes (dimensions). Attributes and aggregation functions used depend on the application.

Star schemas can be used to accelerate the performance of applications by the denormalization reference information in a single dimension table. Denormalization is appropriate when there are a number of entities associated with the dimension table that is often asked, avoiding the overhead of having to attach additional tables to access these attributes. Denormalization is not appropriate when additional data is not accessible very often, because the overhead of scanning the table larger dimension can not be offset by increases in query execution.

The benefit of using this scheme, it reduces the number of tables in the database and the number of relationships between them and also the number of joins necessary in the user queries. Figure 32 shows a star schema in the sample.

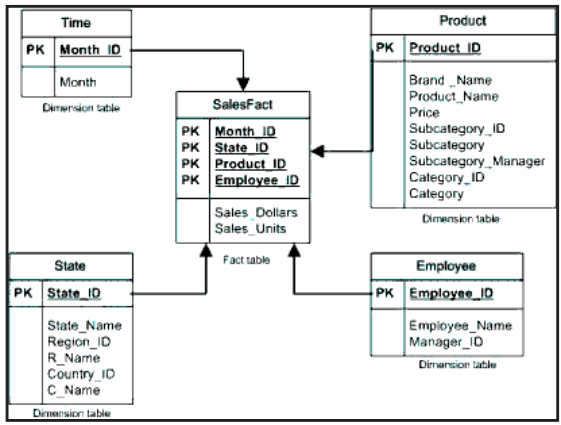


Figure 32 Star Schema

3.2.9.3 Fact Constellation Schema

A constellation diagram is consisting of a set of star schemas with fact tables related hierarchically. The links between the various fact tables offer the opportunity to "drill down" in the level of detail [27, 28]. The figure below, Figure 33 shows an example of constellation diagram of fact.

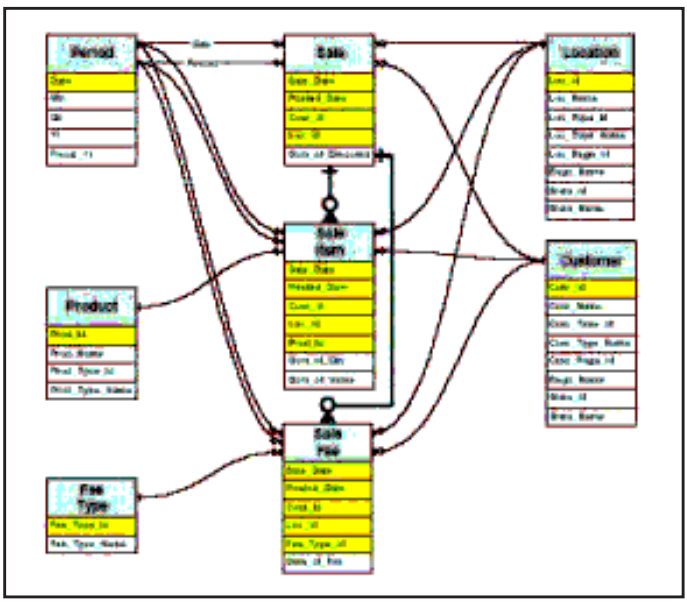


Figure 33 Fact Constellation Schema

3.2.9.4 Galaxy Schema

Galaxy is a schematic diagram where several tables of dimension tables of information sharing. Unlike a constellation diagram of fact, fact tables in a galaxy does not need to be directly linked [12, 28]. The figure below, Figure 34 shows an example schema of the galaxy.

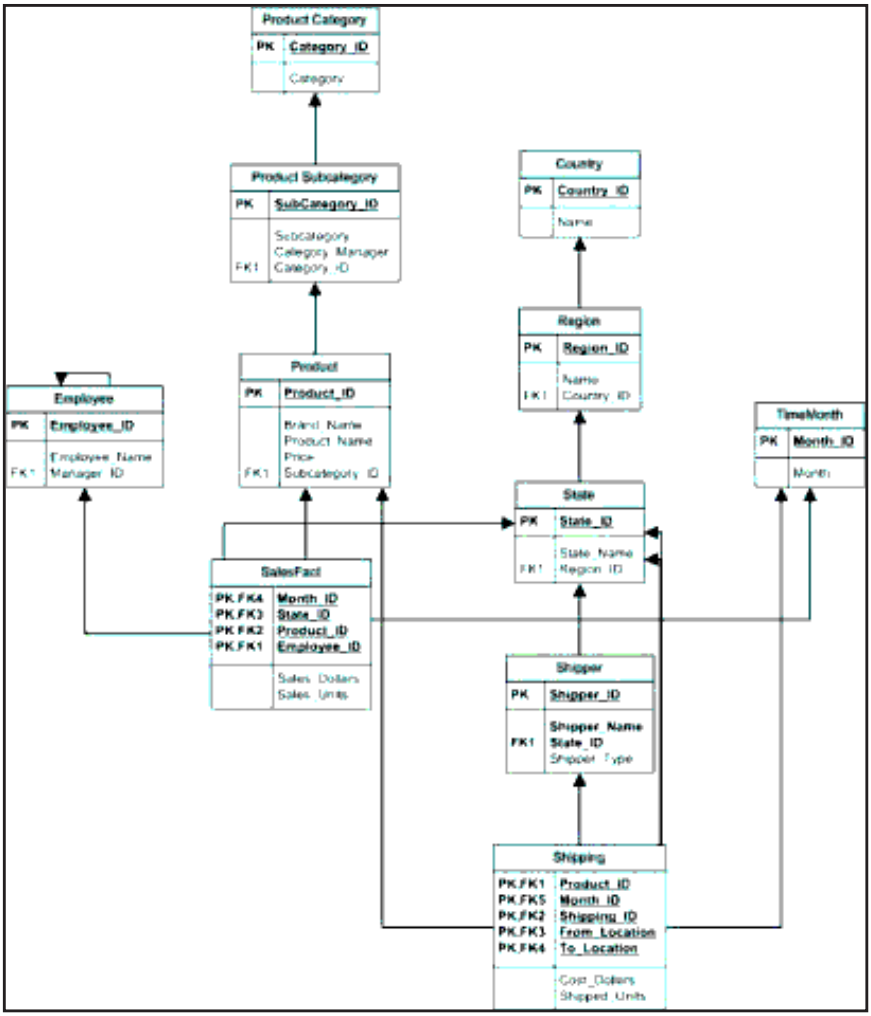


Figure 34 Galaxy Schema

3.2.9.5 Snowflake Schema

In a star schema, hierarchies in the data model originally collapsed to form or denormalized dimension tables. Every dimension table may contain several independent hierarchies. A snowflake schema is a variant of the star schema with all the hierarchies expressly indicated and dimension tables do not include non-standardized data [12, 27, 28].

The relationship many-to-one between sets of attributes of a dimension can separate new dimension tables, forming a hierarchy. The structure displays the decomposed flake hierarchical structure of dimensions very well.

A snowflake schema can be generated by the following procedure:

- A fact table is formed for each entity of the transaction. The key table is the combination of keys related component entities.
- Every component entity becomes a dimension table.
- When the hierarchical relationships between entities transaction, the child entity inherits all relations with its original entities (and key attributes) of the parent entity.
- Numeric attributes in the transaction entities should be aggregated by key attributes. Attributes and functions used depend on the application.

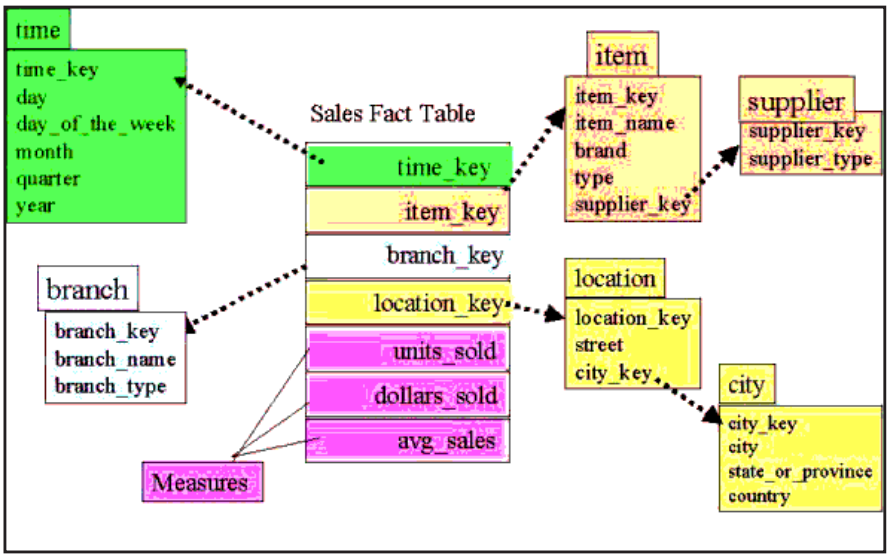


Figure 35 Snowflake Schema

3.2.9.6 Cube

Cubes are logical storage structures of OLAP databases. A cube specifies a set of related dimensions, each cube cell maintain a value, the value of each cell is a point of intersection of the dimensions.

OLAP (Online Analytical Processing) cubes are a new feature that capitalize on the existing data warehouse infrastructure to deliver self-service Business Intelligence capabilities for the end user.

An OLAP cube is a data structure which overcomes the limitations of relational databases by providing rapid data analysis. The cubes can view and summarize large quantities of data while providing users with a searchable access to more granular data so it can be rolled, sliced, diced as needed to handle the widest variety of issues germane to a user's domain of interest.

The database that the company uses to hold all their transactions and records are called online transaction processing (OLTP) databases. Such databases are generally entered in both cases and contain a wealth of information that can be utilized by policy makers to take informed decisions about their business. The databases are used to store data, however, were not designed for the analysis and time and expense required to retrieve the responses of these databases is prohibitive. Databases, online analytical processing (OLAP) databases specialized data designed to help retrieve this information from business intelligence data.

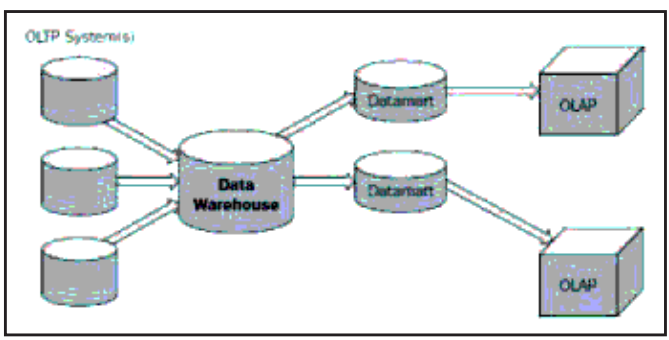


Figure 36 Topology of DW

The new functionality of OLAP is that data cube may be present in an aggregated form. For the user, the cube seem to know the answers in advance as a set of values have been pre-calculated. Free having to query the OLAP data source, the cube can return responses to a wide variety of issues almost instantly.

3.2.10 Meta Data

An important component of the environment DW metadata. Metadata, or data about data, provides the most effective utilization of DW. The metadata enables the user / DSS analyst end to browse through the possibilities. This means, when a user approaches a data warehouse where there is no meta-data, the user is not clear where to start the analysis.

Meta data serves as an index of the contents of the data warehouse. It is above the warehouse and keeps a record of what is the case in the warehouse. Generally, items meta tracks of data stores are as follows [9]:

- Data structure knowledge programmer and analyst at DSS.
- Source data.
- Data Transformation.
- Data Model.
- DW
- History of extracts

Metadata has multiple functions within the DW regarding the processes associated with data transformation and loading, management and query generation DW.

Metadata associated with the transformation and loading data must describe the source data and the changes that were made to the data. Metadata associated with data management describes the data as it is stored in the DW. Each object in the database must be described, including the data for each table, index, and view and constraints. Metadata is also needed by the request handler to generate appropriate queries.

3.3 Comparison of Logical Design Models

Among the models of logic design, star schema, snowflake schema and the schema of the constellation of information are the most used in trade models. In this section, I want to compare these three models in terms of quality factors of efficiency, ease of use, reuse and flexibility. I think efficiency is the most important factor in DW modeling. A DW is usually a very large database. Due to numerous requests have access to large amounts of data and multiple join operations, efficiency becomes an important factor [12, 27]. A star schema is usually the most effective form for two reasons. First, a design with denormalized tables need less joints. Second, most optimizers recognize star schemas and can generate efficient operations "star join". A constellation diagram is actually a set of star schemas with

hierarchically. Is a constellation diagram may need more join the fact tables operations. Similarly, a snowflake schema will require several joins on dimension tables. In some cases where non-standard dimension tables in the star schema is very large, a snowflake schema approach may be the most efficient design. In terms of ease of use, a number of advantages can be considered an approach to design star schema. The star schema is the simplest structure among the three schemes. Because a star schema has the smallest number of tables, users must perform fewer operations of joins that makes it easier to formulate analytical queries. It is easier to learn a star schema in relation to two other schemes. diagram of the constellation of facts and snow flake schema are more complex than star schema which is a disadvantage in terms of ease of use. Considering reuse, snowflake pattern that is reusable and made stars of the constellation diagrams. The dimension tables in a snowflake schema does not contain non-standard data. This makes it easier to share dimension tables between patterns snowflake in a DW. In a star schema and design approaches constellation diagram of fact, the dimension tables are denormalized, which makes it less convenient to share dimension tables between schemas. In terms of flexibility, a star schema is more flexible to adapt to changes in user requirements. The star schema can adapt to the changing needs of users easier because all dimensions are equivalent in terms of access to the fact table supply.

Table 1 Comparison of Logical Design Models

	Star Schema	Snowflake Schema	Fact Constellation
Efficiency	High	Low	Moderate
Usability	High	Low	Moderate
Reusability	Low	High	Moderate
Flexibility	High	Low	Moderate

3.4 Data Warehousing Schemas:

There are three types of schemas are available in data warehouse. Out of which the star schema is mostly used in the data warehouse designs for meta search engine. The second mostly used data warehouse schema is snow flake schema. We will see about these schemas in detail:

3.4.1 Star Schemas:

The star schema is possibly the single data warehouse schema. This is known a star schema as the entity-relationship diagram of this schema looks like a star, with points radiating from a central table. The center of the star composed of a large fact table and the points of the star are the dimension tables.

A star schema is marked by one or more very large fact tables that contain the primary information in the data warehouse, and a number of tables much smaller dimensions (or lookup tables), each includes information about a particular attribute for entries in the fact table.

A star query is a join between a fact table and a number of dimension tables. Each dimension table is joined to the fact table using a primary key to foreign key join, but the dimension tables are not interconnected. The cost-based optimizer recognizes star queries and generates efficient execution plans for them. [34] A typical fact table contains keys and measures. For example, in the diagram of the sample, the fact table, click streams, contain the measurement session, user, page, date, time, keyword and category, and session_id keys TIME_ID, user_id, keyword_id and category_ id. The dimension tables are session, user, time, keyword, document, and category. The dimension table session, for example, contains information about the dimension of session is more than just a label that includes all the page events that constitute a single user session.

The dimension of the session is where we mark the session and trace its activity [39], to describe the characteristics of the session.

A star join is a primary key to foreign key join dimension tables to a fact table [34].

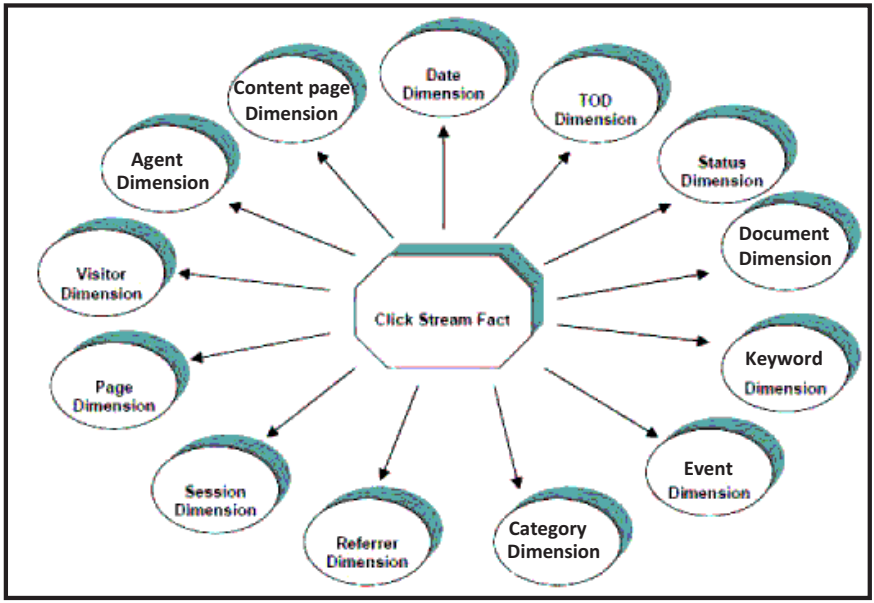


Figure 37 Star Schema of the Meta search engine -Click Stream Fact and its Associated Dimensions

This section illustrates the capabilities of detailed analysis of the model by listing the simple fact that the user will be able to analyze and corresponding dimensions that gives the user the ability to drill and drill down and slicing and in deciding on the basic fact. Before the design of meta search engine clicks of specific flows, it is necessary to collect as many dimensions as you may think may be relevant in an environment flow click. The unique dimensions of warehouses click stream data page, the visitor session and referrals. The size of the page describes the background to the event page for a Web page. It contains key attributes as the page source of the page, the page function. The size of the visitor gives details of visitor. The main attributes are userId, CookieId, operating system and browser. The dimension of session provides one or more levels of diagnostic session the visitor as a whole. For example, the local context of the session could be seeking product information, but the overall context of the session would be to order a product. The reference dimension describes how the client arrived at the current page.

3.4.2 Snowflake Schemas:

Snowflake schema is a warehouse model more complicated than star schema data, and is similar to a star pattern with more braches terminal. This is known a snow flake pattern as the diagram resembles a snowflake.

Snowflake schemas normalize dimensions to eliminate redundancy. In other words, the dimension data has been grouped into multiple tables instead of one large table. For example, a keyword dimension table in a star schema might be normalized into a keyword table, a table of categories, and a table of category in a snowflake schema. While this saves space, it increases the number of dimension tables and requires more foreign key joins. The result is more complex queries and reduced query performance. [34]

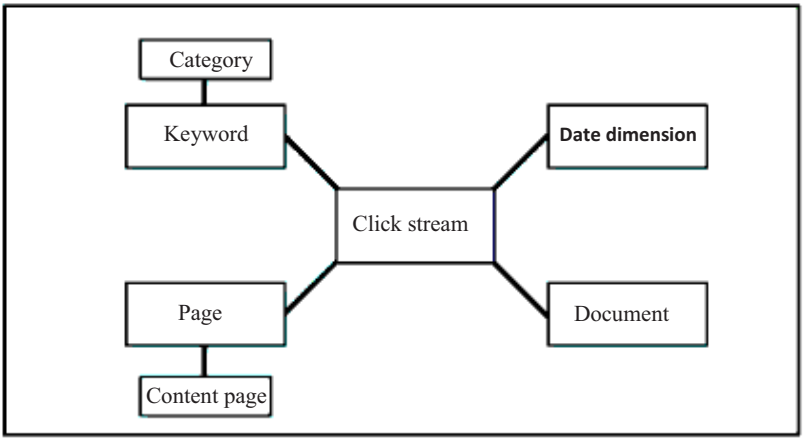


Figure 38 Snowflake Schema of example the Meta search engine -Click Stream Fact and its Associated Dimensions

3.4.3 Fact constellation/galaxy:

Fact constellation diagram is also known as galaxy schema. This is nothing but a schema that contains multiple fact tables dimensions actions. It is a collection of star schemas sharing their size. So it is called as a diagram of the galaxy. For each star schema or snowflake schema it is possible to construct a fact constellation schema. This schema is more complex than star or snowflake architecture, which is because it contains multiple fact tables. This allows dimension tables to be shared amongst many fact tables.That solution is very flexible, however it may be hard to manage and support. The main disadvantage of the fact constellation schema is a more complicated design because many variants of aggregation

must be considered. In a fact constellation schema, different fact tables are explicitly assigned to the dimensions, which are for given facts relevant. This may be useful in cases when some facts are associated with a given dimension level and other facts with a deeper dimension level. Use of that model should be reasonable when for example, there is a Click stream fact table and a fact table with order item which is calculated based on month,visitor id and keyword id. In that case using two different fact tables on a different level of grouping is realized through a fact constellation model [40].

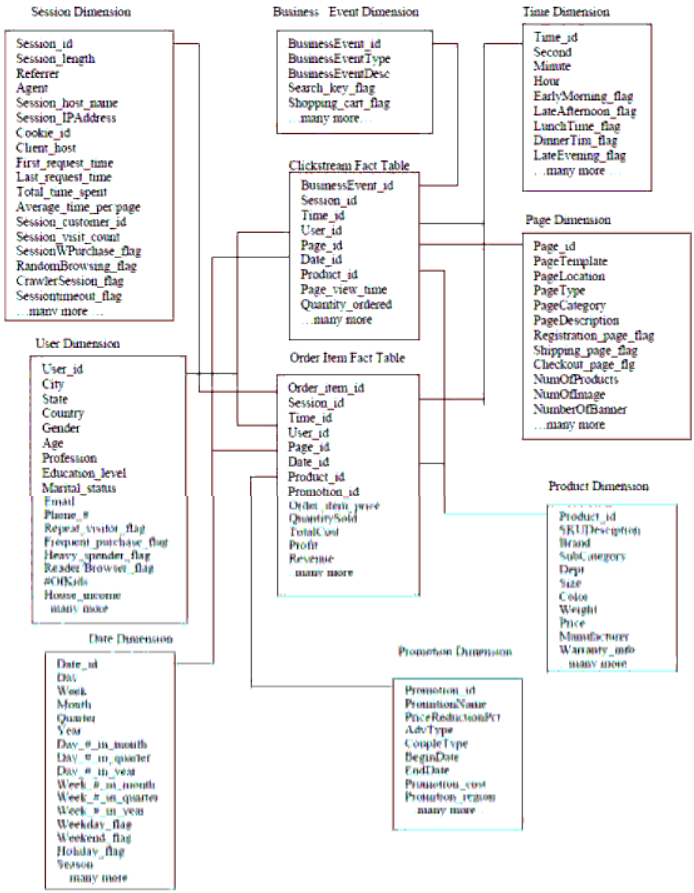


Figure 39 star schema and Fact constellation/galaxy Schema of the search engine -Click Stream Fact and its Associated Dimensions

3.5 Data Webhouse Construction

A data warehouse provides the source data for online analytical processing and data mining. Designing an appropriate schema data warehouse and OLTP complete data warehouse system is very long and complex. A well-designed data warehouse feed business with the right information at the right time to make the right decisions in Meta search engine system [10, 41, 42]. we discussed data capture methods for the website, which collect clickstream, keyword, visitor, session, referrer and date information, etc. This data is transaction data online and are stored in the database transaction processing (OLTP). Patterns of databases of the OLTP are based on modeling ER, normalized to reduce the redundancy in the data base and adapted to maintain atomicity, consistency and integrity so as to maintain the speed and efficiency of use in business operations on a daily basis as insert, update and delete a transaction. For an OLTP application, it normally only need to access a small set of records in databases but require very quick responses. For mining purposes to use the Web, it must have a schema database (called a data warehouse), designed to support decision making and data analysis (On-Line Analytical Processing). Conventional relational data bases are designed for online transaction processing (OLTP) and do not meet the requirements of effective online analytical processing.

As a result, data warehouses are designed differently from traditional relational data bases. Data warehouses using OLTP read-only data historical analysis. The data in a data warehouse system are usually organized in multidimensional modeling with star schema (fact tables and tables of similar dimensions). The requirement of clickstream data in the data warehouse makes the design of a more complicated pattern. Web challenges the current view of the data warehouse with several new requirements. [10] The data warehouse is required to make available to the customer clickstream analysis, if a new WEBHOUSE term was coined by Ralph Kimball [10,43]. A WEBHOUSE plays a vital role in the Internet revolution as a platform for analysis of all behavioral data from the clickstream, as well as many websites that rely on the data warehouse to customize and control the web experience of the end user in real time [10].

WEBHOUSE we use to refer to the data warehouse system for the extraction of Web usage. The WEBHOUSE is the data source of data mining and business intelligence reports in Warehouse / OLAP data and it contains the basic business content of what an online store sells Web services and capabilities.

A WEBHOUSE should allow you to analyze all visits to a website, all products sold in the shop many points of view site. Many systems have developed to exploit the web log

records that can find patterns of association and sequential patterns on the Internet, but in order to understand customers as repeat visitors against unique visitors, customers single purchase over several clients purchase, it is necessary to include additional information such as order information from the online store, product information on the product, the navigation sequence of user clickstream and customer information from the user table. Below we discuss the needs analysis and the technique of dimensional modeling to design WEBHOUSE.

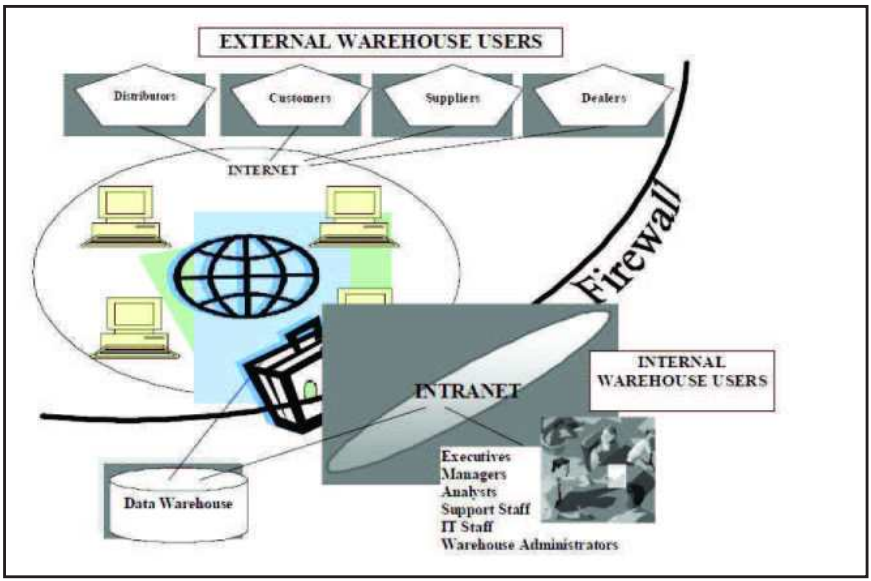


Figure 40 The Webhouse and Data Warehouse

It is necessary to construct an overview of the flow of clicks dive arriving on websites. We want to build a WEBHOUSE provide useful information and answer important questions for the search engine Meta. The design of a WEBHOUSE begins with a needs analysis. We spent a lot of time talking with our customers, businessmen, engineers / developers and end users analysts collect their needs and what types of activities they hope to get answers from the WEBHOUSE. Their questions cover a wide range and areas:

- Web site activity (hourly, daily, weekly, monthly, quarterly etc)
- document
- visitor
- Referrers (by domain, by sale amount, by visit numbers etc)

• Navigational behavior pattern (top entry page, top exit page, killer age, hot page etc)

• Click conversion-ratio • Keyword (category)

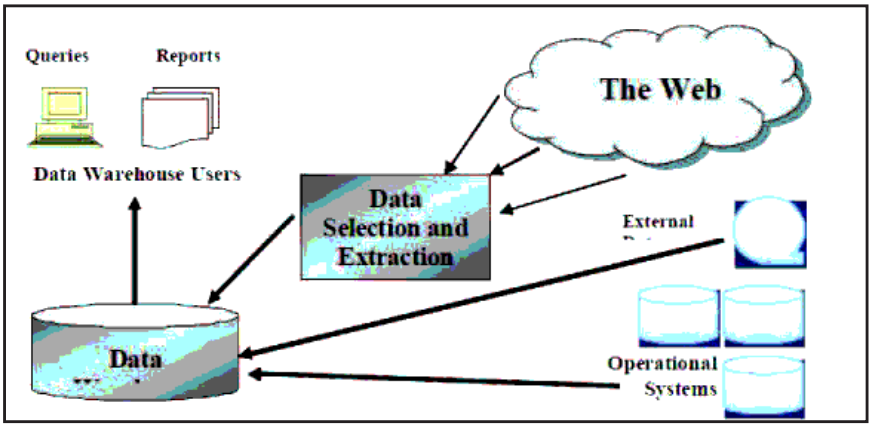


Figure 41 Web data for the data warehouse

3.6 Extraction, Transformation, and Loading (ETL):

ETL: the data extraction process from source systems and bringing it into the data warehouse. LTE, which stands for extraction, transformation and loading [44]. A common sequence of processes (ETL) are the following: 1 - Data Extraction 2- Data Verification 3 - Data Cleansing 4 - Integration 5 - Aggregation 6 - Loading

3.6.1 Data Extraction

This is the process that captures data from the source system (s) and moves to the intermediate database. The extraction can be a complete refresh where all the data specified by the source system is copied or an incremental update which only copies the specified data that have changed since the last time the process (ETL) has been executed. Different methods can be used to extract such as copying data objects, using (SQL) to capture all or part of the data (FTP) to download data files, replication, etc. [44].

3.6.2 Data Verification

This step compares the data compares the data from the previous step with the rules as specified in the specifications of the company of (DW) data. He audit files are stored in the source system can also be compared to those for sure that we extract the correct data, ie the number of rows and number of bytes extracted. The rules for data quality control that we can

be the responsibility of the owner of the source system or the responsibility of the administrator (DW) and responsibilities must be declared in the specifications of the company. If the data are not consistent with the rules of data quality, then there are a number of options:

- Discard the data in order not to enter the (DW).
- Keep the data in a holding area so that it may be fixed prior to insertion into the data warehouse.
- Adding the data warehouse with errors.
- Which can lead to serious problems down the line.

When data are released or located in a waiting area at the source system administrator must be informed and give details / access to data released so they can try to correct the problems that are causing the problems. [44].

3.6.3 Data Cleansing

Data cleansing and data cleansing is the process by which valid data are made more precise. General aspects of data transformations include cleaning [44]:

- Search and replace - such as to synchronize the name of the building where there were instances of the same building referred to in various acronyms.
- Change cases - for example in the column 'title' in table 'a customer, converting all instances of "MRS", "Mrs." and "MR" to "Madame".
- Fusion of data from different data sources.
- Handling NULL - conversion to a default value.
- Conversion of data types - to syncronise data from different systems, namely the client system can be an integer, but another system can store around as a character data type.
- Fractionation source files into several intermediate files that can be used to drive multiple loads the target table.
- Inclusion of meta-data to help describe the data in the data warehouse such as identifiers of the source system, end date, etc.

3.6.4 Data Integration

Data integration is the incorporation and fusion of data from different source systems in a unified data interface that can be used for the front end analysis. Integration of data

normally leads to the creation of new entities and structures of columns and involves operations that join together data from different sources. These new entities and columns should be an easier way to understand the structure that disparate source systems [44].

3.6.5 Aggregation

Once we have tables that are ready to be loaded into the DW, we can make summary calculations (aggregation) and store the summary data to allow rapid development of applications. When creating our dimensional data model, it is essential that good roads as part of the aggregation design dimension tables [44].

3.6.6 Loading

It is at this point that "uploads" the transformed data to the data warehouse directly. The loading rate can be influenced by factors such as table size and the proportion of updates / inserts. Most database management systems (DBMS) are supplied with a utility such as bulk loading (SQL) program Bulk Copy server. It can often be quicker to delete entire tables and insert all rows using these utilities bulk loading rather than trying to update all the rows that have changed since the last update [44].

Chapter 4: IMPLEMENTING A DATA WAREHOUSE

The meta search engine - database schema is designed to make the structure more understandable for users of the underlying data and simplify the application process. The modeling approach recommended data warehouse data is to follow an approach called dimensional modeling star schema, snowflake schema and schema Constellation done. . The star schema has a central fact table with dimension tables at the points of the star. Composite of a table that needs a foreign key field for the primary key of each dimension table primary key. The dimension tables are denormalized very hierarchical [30]. A fact table is the primary table in the metadata search engine that contains business facts and dimension tables are tables of company at the table of facts that are essential aspects of the business and contain "attributes critical dimensions of the business. The central fact table provides users with the ability to perform analysis on business facts and dimension tables provide users with the ability to make these analyzes made in different critical dimensions of the business of the company [31, 37].

The snow flake schema is a warehouse more complicated than star schema data model, and is a form of star schema. This is called a snowflake schema from the schema resembles a snowflake. Snowflake schemas normalize dimensions to eliminate redundancy.

This chapter provides you with an in-depth understanding of data warehousing its application to business intelligence. You will learn the concepts and skills necessary to build a successful data warehouse to enable your business intelligence program on the first implementation. In addition, this chapter show how is the design of data warehouses on the meta-search engines.

A Data Warehouse (DW) is a database that stores information geared to meet the demands of decision-making. A very common problem in business is the lack of access to corporate information, comprehensive and integrated company that can meet the demands of decision-making [32, 35]. In the past, many education issues were hard to answer. But with the capacity provided by Data Warehouse data can be aggregated and disaggregated, compare disparate data and generate custom reports, you are able to answer more detailed and useful focus and clarify issues through the process of data analysis. [33]

The search engines epitaxial which was created by a student AUL Bilal Nakhal, which adopted the studies it and because of its great importance in the field of information retrieval order, and ease of use for the new user so it needs this search engine to set up a data warehouse to assist in the analysis and decision-making. Besides an relational database, a

data warehouse environment consists an extraction, transportation, transformation and loading solution (ETL), an online analytical processing (OLAP) engine, analysis tools client, and other applications that manage the process of collecting data and deliver business users.

When you have decided to build the data warehouse for the meta search engine, you need to determine the needs and agree on the scope of your application, and established a conceptual design. Finally, you need to translate your requirements into a deliverable system. You can do this translation by creating logical and physical data warehouse design. You then define [34, 1]:

- The particular data content.
- The relationships within and between groups of data.
- The system environment support your data warehouse.
- Data transformations necessary.
- The frequency with which data is updated..

Focus your design to the needs of end users. End users usually want to perform analysis and look at aggregated data, not individual transactions. But, end users may not know what they need until they saw. In addition, a well thought out design enables the growth and changes that the user needs change and evolve. Starting with the logical design, you concentrate on the information needs and recording the implementation details for later.

The logical design is more conceptual and abstract the physical design. Within logical design, you watch the logical relationships between objects. In the physical design, you watch the most efficient way to store and retrieve objects and their manipulation from a standpoint of transportation and backup / restore.

4.1 Facts and Dimensions in the Meta search engine

Table 2 below shows the objects is done and sizes available in the Meta search engine for the purpose of analysis.

Table name	Fact/ Dimension	Levels
Click Stream Fact	Fact	-
Date	Dimension	Year, Quarter, Month, Week, Day
Universal Date	Dimension	Year, Quarter, Month, Week, Day

Universal TOD	Dimension	Period of the day, Hour, Minute, Second
TOD	Dimension	Period of the day, Hour, Minute, Second
Visitor	Dimension	IP Address (or) Visitor Id (or) Cookie Id
Page	Dimension	Object Type, File Type, Page Type, URL (a) Domain, Site, Directory, URL
Session	Dimension	Session Type
Referrer	Dimension	Referrer Type, URL (a) Domain, Site, URL
Status	Dimension	Type of the Status, Status description
Visit	Dimension	-
Content page	Dimension	-
Document	Dimension	-
Keyword	Dimension	Type of keyword, keyword field
Category	Dimension	Category type, category name, category description
User	Dimension	-

4.1.1 Fact table

The big tables in the data warehouse schema that store facts and foreign keys in the dimension tables. Fact tables represent data, usually numeric and additive, that can be analyzed and discussed. A fact table generally two types of columns: those that contain numeric facts (often called measurements), and those that are foreign keys to dimension

tables. A fact table provides facts or facts that have been aggregated at the retail level. Fact tables that contain aggregated facts are often referred summary tables.

When we want to establish a new fact table, we must define a fact table for each star schema. The primary key of the information table is normally a composite key that consists of all of its foreign keys [36].

Clickstream Fact table

Layout, development and strengthening are the clearest tasks implicated in building a Web service. But the analysis of how end users use a service may be more important to the success of this service. Understanding how users achieve your website and how they perform after they are there, you can enhance the experience of prospective users of the site. It also allows you to improve your return on investment by increasing sales and increasing the "stickiness" of user visits. We look to clickstreams, the flow of requests (clicks) users generate as they move from page to page inside a website to provide such information. Clickstreams include a huge amount of quantitative data that can be used to enhance the user experience with a site. When correctly operated, course analysis can provide responses to very specific matters such as:[12, 36]

- What are the more popular pages on a site?
- How do people arrive at the site? Where do they go when they leave? How long do they stay?
- What are the pages of this website appear to be "session killers", which stops the remote user session and leaves Make a certain percentage of people leave my site carts loaded with a web page?
- Which browsers and operating systems do not utilize visitors?
- What are the most popular ways that users take center stage?
- How successful are the various banner campaigns? How many people go to my website if they reached through a banner in particular?

Answers says us where to direct the development and promotion efforts in the future. If we find that certain services we've put a lot of effort into being neglected by users, and we want to feature them near the top of the site to improve its popularity. If we find that users leave the site after vote yes or no and completely at a certain point in the checkout process meta-search engine and we may want to hire a consultant usability to simplify the user interface so that users can complete their transactions. The standardized approach now to analyze user behavior and utilization of a website through log analysis of web server logs file. Web servers connect each HTTP request to a text file in the file system, a log file analyzer

can analyze log files and displaying data in a variety of useful ways. But the Web server log files may not respond to some of the questions that can be answered by analyzing the course. Standard log file analyzers cannot supply the level of analysis rendered possible by storing clickstream data for several reasons. Firstly a standard log file does not include all the information that the web server knows a visitor, such as the identity of the user. In addition, building a data warehouse for data path enables data path must be correlated from various sources.

Table 3 Presents various business on Measures will be all which the user able to do analysis using the click stream fact table and the associated dimension tables

Field Name	Description
Datekey	Foreign key for the Date dimension
TimeDayKey	Foreign key for the TOD dimension
Visitorkey	Foreign key for the Visitor dimension
Pagekey	Foreign key for the Page dimension
Sessionkey	Foreign key for the Session dimension
Keywordkey	Foreign key for the keyword dimension
Categorykey	Foreign key for the category dimension
Documentkey	Foreign key for the document dimension
Referrerkey	Foreign key for the Referrer dimension
Statuskey	Foreign key for the Status dimension
Visitkey	Foreign key for the Visit dimension
ContentPagekey	Foreign key for the Content Page dimension
TimeView	Time spent in seconds by the visitor on a particular object as page, file, …..
BytesTransfer	Bytes transferred to the client machine.

4.1.2 Dimension table:

Dimension tables, also known as research or reference tables, include the fairly static data in the warehouse. Dimension tables store the information that you normally use to contain queries. Dimension tables are generally textual and descriptive and you can use them

as the row headers of the result set. A dimension is a structure, usually consisting of one or more hierarchies, that categorizes data. Dimensional attributes help to describe the dimensional value.

The dimension table provides users with the capability to analyze the trade measures in various dimensions by enabling users to explore and drill down and slice and dice with the attributes of the dimensions. Drilling down adds detail to an existing application lines and is nothing more than asking to give more details. Drilling up by subtracting row and is nothing other than view the data in a more comprehensive / consolidated form. Slicing is to constrain the data that is viewed on an attribute found in one dimension and a cut is to oblige the data displayed by the attributes in several dimensions.

4.1.2.1 Date Dimension

Which day of the week was the request? Day in the month? Day in the year? Which week in the year?. Figure 42 shows the date dimension with all its attributes. Hierarchical dimension attributes are represented using connections arrowhead and general characteristics are represented using line connections. The date is the level of grain unless the user will be able to break up, and the year is the highest level that the user will be able to drill down. Drill down path can be marked with following the arrowheads.

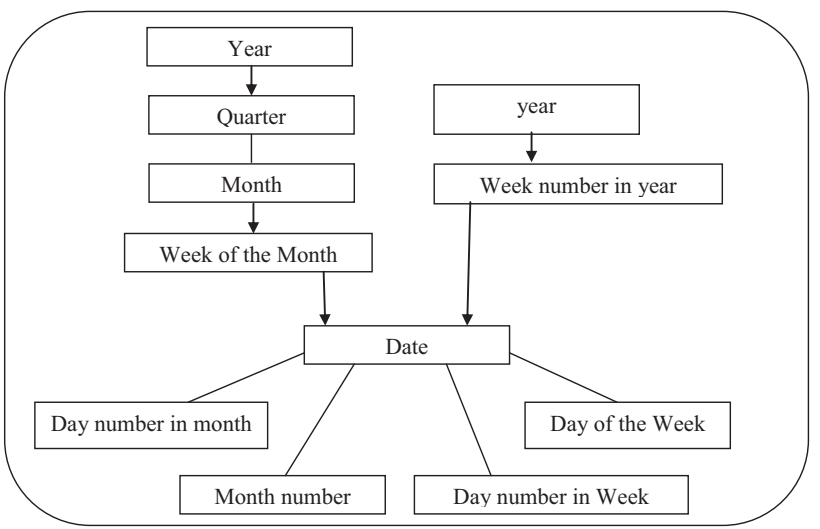


Figure 42 Data Dimension with the hierarchy

Table 4 Describes the structure of the Universal date dimension table:

Field Name	Description	Values/Examples
Datekey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
Date	Date	07/02/2013
DayWeek	Day of the week	Thursday
DayWeekNumber	Day of week number	1-7, Thursday being 5
WeekMonth	Week number in the month	1-5
WeekYear	Week number in the year	1-52
DayMonth	Day number in the month	1-31
MonthNumber	Month of the year in number	1-12
Month	Month of the year	February
Quarter	Quarter of the year	1-4
Year	Year of the date	2013

4.1.2.2 Visitor dimension:

What is the segment of the user given the visitor belongs for example, the visitor is in the `` saw" the privacy policy area? Note that the representation of the size of the user in the model illustration labels data user ID is with individual identities, which do not occur in the real data model clickstream. The dimension of visitor follows anonymous visitors using the specified visitor identification method. Figure 43 shows the dimensions of the visitors with all its attributes. Hierarchical dimension attributes are represented using connections arrowhead and general characteristics are shown using line connections. User ID / Cookie ID / Domain name is the level of grain unless the user will be able to drill down, and the country is the highest level that the user will be able to drill down. The lowest granularity will be determined on the client site. Drill down path can be marked with following the arrowheads.

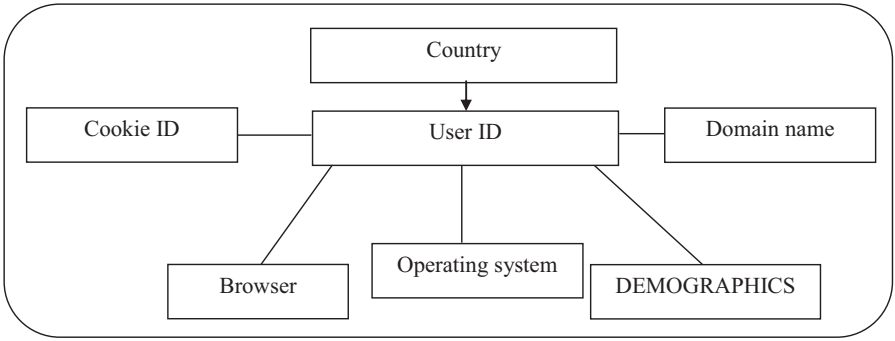


Figure 43 Visitor Dimension with the hierarchy

- Demography is gathering many areas. It is also possible to form a hierarchy in the demographic information. This hierarchy contains the following levels (Visitor Type,Visitor).

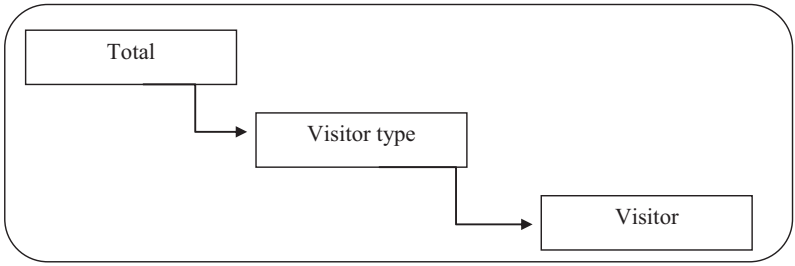


Figure 44 Show the Visitor hierarchy of the Visitor dimension. This hierarchy contains the following levels, in descending order: Visitor Type, Visitor.

Table 5 Describes the structure of the Visitor dimension table:

Field Name	Description	Values/Examples
Visitorkey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
UserId	Identification of the Visitor (name or login user id).	-
CookieId	Value of the Cookie	A string
IPAddress	IP address of the requesting	192.168.22.81
Country	The country of the visitor. Predicted from the domain of the visitor	Iraq, Lebanon,…
OperatingSystem	The name of the operating system with version	Windows NT 3.0, Unix 6…

Browser	The name of the browser with version	Internet Explorer9.0, Google Chrome 3.0
+FirstName	The first name of the Visitor	Atheer, Ahmed,…
+LastName	The Last name of the Visitor	Hadi, Hussein
+DateOfBirth	The Date of birth of the Visitor	07/05/1981
+Agegroup	The age group of the Visitor	18-26, 30-44,…..
+Gender	The Gender of the Visitor	Male, Female
+Occupation	The occupation of the Visitor. Limited to a set of categories.	Computer, communication
+IncomeGroup	The income group of the Visitor. The groups are defined by the business.	-
+ZipCode	The Zip code of the place of the visitor	
+State	The state of the visitor.	
+VisitorCountry	The country of the Visitor specified by the Visitor.	Iraq, Lebanon,…

+ Optional fields. Collected from the web site visitor through registration forms.

4.1.2.3 Time of Day Dimension (TOD):

Time dimension of the day offers a second granularity during which an incident occurred. The time of day of the dimension hierarchy contains: The Time of Day Hierarchy, Figure 45 shows the time of day dimension with all its attributes. Hierarchical dimension attributes are represented using connections arrowhead and general characteristics are represented using line connections. Second, the level of grain unless the user will be able to break up, and the time of day is the highest level that the user will be able to drill down. Drill down path can be marked with following the arrowheads.

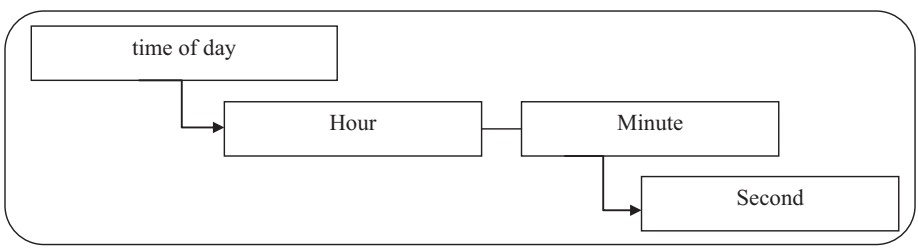


Figure 45 Time of day Dimension with the hierarchy

Table 6 Describes the structure of the time of day dimension table:

Field Name	Description	Values/Examples
TimeDayKey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
Second	Second of minute	1-60
Minute	Minute of Hour	1-60
Hour	Hour of a day	9-10, 12-13
TimeDay	Time of the day	11:19:10, 22:12:18
PeriodDay	Collection of hour in a day	Evening, morning

This hierarchy contains the following levels, listed from top to bottom (Hour, Minute, Second).

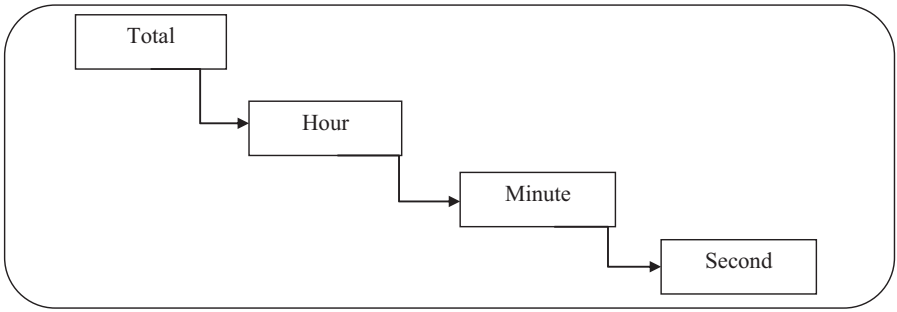


Figure 46 Show the Time of Day hierarchy of the Time of Day dimension. This hierarchy contains the following levels, in descending order: Hour, Minute, Second.

4.1.2.4 Session Dimension:

The session dimension provides one or more levels of diagnosis for the user's session. For example, the overall session context might be retrieving information from the Web. The success status would diagnose whether the mission was completed. The user status attribute is a convenient place to label the user for periods of time, with labels that are not immediately clear either from the page of the immediate session. This dimension is extremely important because it provides a way to group sessions for insightful analysis.[1]

During a given session she started? When will it end? That visitor? How many clicks were part of the session? What was the total value of visits of the session?. The dimension of the session, be used to characterize sessions according to the rules defined by the user. The Session dimension contains one hierarchy: The Session Type Hierarchy, Figure 47 shows the

dimensions of session with all its attributes. The general characteristics are represented using line connections, the connection with a circle at one end indicates that the specified item is a set of fields. Session Type gives meaning to visit.

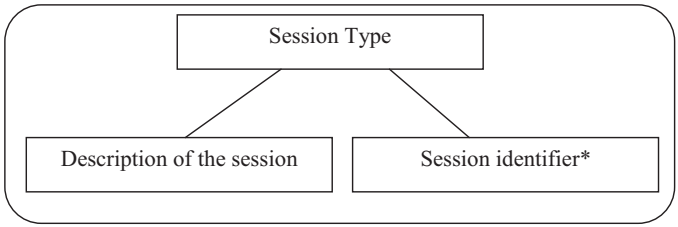


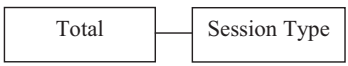
Figure 47 Session Dimension with the hierarchy

* Session identifier: are collection of fields, which describes the conditions for characterizing a session type.

Table 7 Describes the structure of the session dimension table:

Field Name	Description	Values/Examples
SessionKey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
SessionType	The type of the user session. Session is defined on the basis of management rules	Quick hit and gone,..
SessionDescription	The description of a specific session	-
Sessionidentifier	The parameters that characterize the particular session. It can be split into multiple fields, based on the business rules provided by the customer.	Ex: If Time Spent is in the range of 1-10 min and the pages visited in general info, then it is a 'Looking for Info' Session

This hierarchy contains one level (Session Type)



4.1.2.5 Status Dimension

The dimension of the state of server-class and describes the status codes returned by the server in response to a request. Figure 48 shows the status dimension with all its attributes. Hierarchical dimension attributes are represented using connections arrowhead and general characteristics are represented using line connections. Id status is the level of grain unless the user will be able to break up, and the type of status is the highest level that the user will be able to drill down. Description of the report provides a description of the status ID. Drill down path can be marked with following the arrowheads.

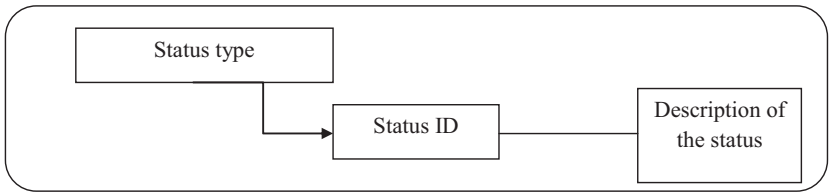


Figure 48 Status Dimension with the hierarchy

Table 8 Describes the structure of the status dimension table:

Field Name	Description	Values/Examples
StatusKey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
StatusId	The status code	023, 102, 440,…
SatusDescription	The description of Status	Found error, felicitous
StatusType	Type of status	Error in file

The Server Status dimension contains one hierarchy: The Server Status Hierarchy, This hierarchy contains one level (Server Status).

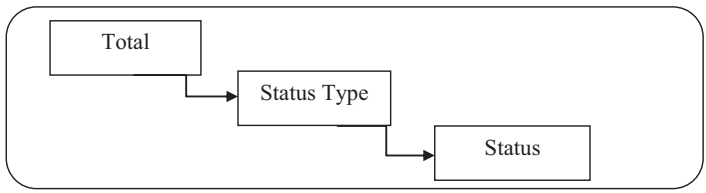


Figure 49 This figure displays the Status hierarchy of the Status dimension. This hierarchy contains the following levels, in descending order: Server Status, Status.

4.1.2.6 Referrer Dimension

The Referrer dimension allows a multilevel analysis. The information obtained from the domain name of the sponsor can be used in the country high-level reports / type field. If the URL of origin is recognized as belonging to a search engine, the URL can be parsed for keywords used in the search. (What kind of Referrer was there (eg, search engine)? What was the referring URL? What search text, if any, was used? If it was within the site, which is the reference page id?). Figure 50 shows the dimensions of session with all its attributes. Hierarchical dimension attributes are represented using connections arrowhead and general characteristics are represented using line connections. URL is the level of grain unless the user will be able to break up, and the area and the type of source are higher than the user will be able to drill down. Drill down path can be marked with following the arrowheads.

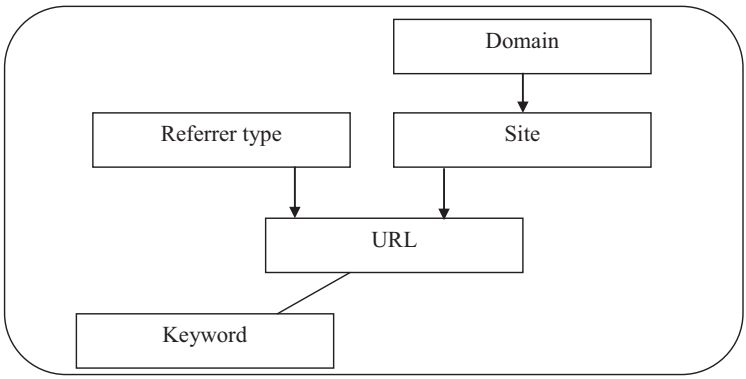


Figure 50 Referrer Dimension with the hierarchy

Table 9 Describes the structure of the Referrer dimension table:

Field Name	Description	Values/Examples
ReferrerKey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
URLReferring	The URL of the referring page	E:\..\Fact.html
SiteReferring	The Site of the referring page	-
DomainReferring	The domain of the referring page	-
Keyword	The keyword given by the user as search criteria to reach the page.	Meta search engine, data warehouse
Referrer Type	The type of the referrer	Search engine

The Referrer dimension include the following hierarchies: (The Referrer Campaign Hierarchy, The Referrer Category Hierarchy , The Referrer Geography Hierarchy , The Referrer Organization Hierarchy). This hierarchy contains the following levels, listed from top to bottom (Campaign, Referring URL)

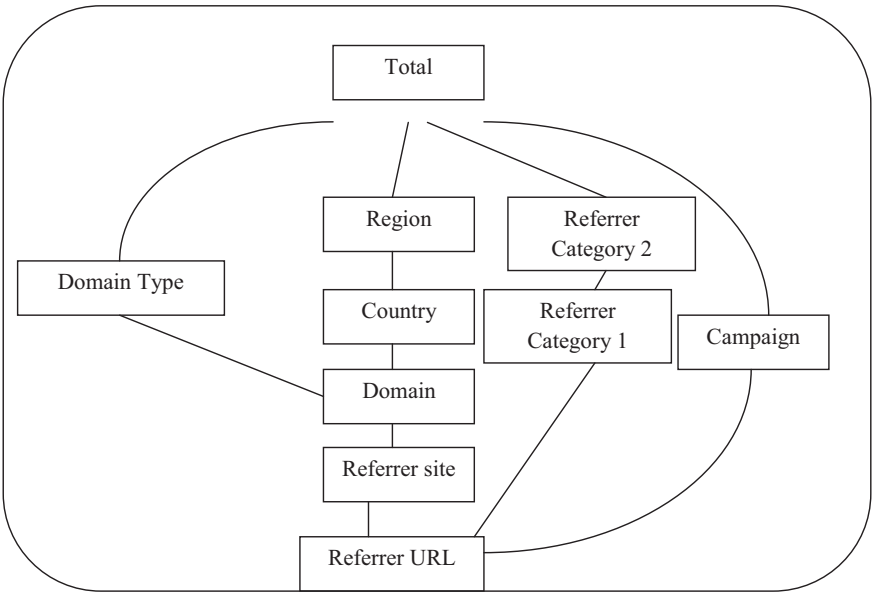


Figure 51 Show the Referrer Organization hierarchy of the Referrer dimension. This hierarchy includes the following levels, in descending order: Domain Type, Domain, Referring Site, Referring URL

4.1.2.7 Page Dimension

The page dimension describes the page context for a Web page event. The grain of this dimension is the individual page type. Our definition of "page" must be flexible enough to handle the evolution of Web pages from the current mostly static page delivery to highly dynamic page delivery in which the exact page the user sees is unique at that instant in time. We will assume even in the case of the dynamic page that there is a well-defined function that characterizes the page, and we will use that to describe the page. We will not create a page record for every instance of a dynamic page because that would yield a dimension with an astronomical number of records. These records also would not differ in interesting ways. What we want is a record in this dimension for each interesting distinguishable type of page. Static pages probably get their own record, but dynamic pages would be grouped by similar function and type. When the definition of a static page changes, because it is altered by the

Webmaster, the record in the page dimension can either be overwritten or can be duplicated. This decision is a master of policy for the data Webhouse, and it depends on whether the old and new descriptions of the page differ materially, and whether the old definition should be kept for historical analysis purposes. The dimension of the page comprises a single record for every logical page hosted on a Web site. URI rods conjunction with a set of contained identify parameters query string identify impressionable page. Figure 52 shows the dimensions of a page with all its attributes. Hierarchical dimension attributes are represented using connections arrowhead and general characteristics are represented using line connections. URL is the level of grain unless the user will be able to break up, and the area and the type of object are higher than the user will be able to drill down. Drill down path can be identified by following the arrowheads. What was the name of the requested page? What kind of page is asked? What was the URL? What sort of content?

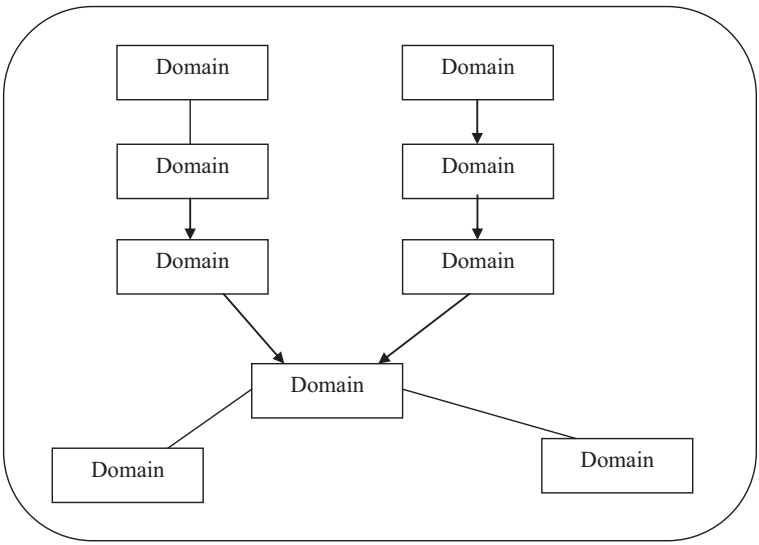


Figure 52 Page Dimension with the hierarchy

Table 10 Describes the structure of the page dimension table:

Field Name	Description	Values/Examples
PageKey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
URL	The complete path of the page on the server	E:\..\Fact.html

PageName	Name of the web page	Meta search page
PageType	Classification of pages in the website	New page, load page
FileType	The file type of the object	Html, gif
ObjectType	the object type	Application, page Content
FileName	Name of the file are accessible by the user	Fact.html
Directory	The server directory of the file accessed	E: \New folder\doc…
Site	The site where the particular page is possible	
Domain	The domain name of the site where the page resides	

The Page dimension contains the following hierarchies: (The Page Category Hierarchy, The Page Resource Hierarchy), This hierarchy contains the following levels listed from top to bottom (Resource Type, Resource, Page).

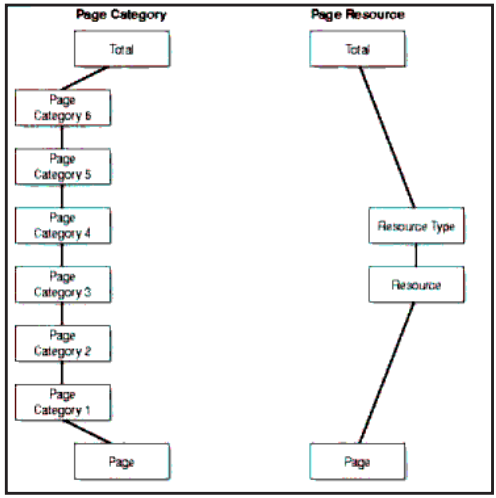
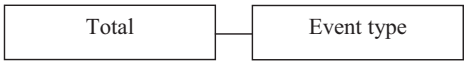


Figure 53 Show the Page Category and the Page Resource hierarchies of the Page dimension. The Page Category hierarchy includes the following levels, in descending order: Page Category 6, Page Category 5, Page Category 4, Page Category 3, Page Category 2, Page Category 1, Page. The Page Resource hierarchy includes the following levels: Resource Type, Resource, Page.

4.1.2.8 Event Dimension

The event dimension describes what happened on a particular page at a particular point in time. The main interesting events are Open Page, Refresh Page, Click Link, and Enter Data. As dynamic pages based on XML become more common, the Event dimension will get much more interesting because the semantics of the page will be much more obvious to the Web server. Each field in an XML document can be labeled with a user-defined tag [1]. The Event Type dimension point out whether a given impression is a entry page or a page of output, both or neither. This dimension can also be expanded to other types of events. The type of event has a dimension hierarchy: (The Event Type Hierarchy: This hierarchy includes one level, Event Type.



4.1.2.9 The Agent Dimension

The dimension of the agent contains attributes that are drawn from the "User Agent" matched string in several Web server log files. User agents are typically web browsers, but other agents like Web spiders can also access sites. (Which browser is the user using? Which version? What operating system?). The Agent dimension includes the following hierarchies: (The Agent Client Software Hierarchy: This hierarchy contains the following levels, listed from top to bottom {Client Type , Client , Client Version , Agent } and The Agent Operating System Hierarchy: This hierarchy contains the following levels, listed from top to bottom { Platform, Operating System , Agent }).

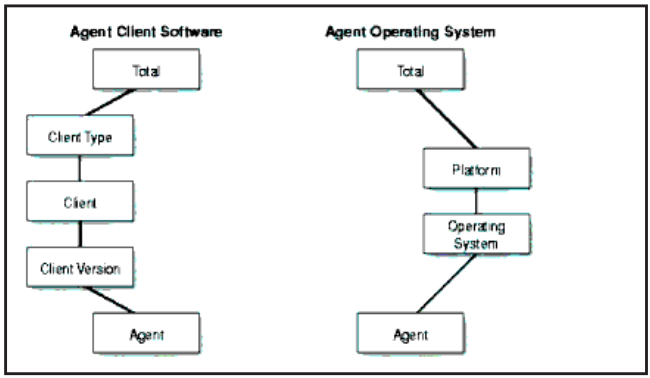


Figure 54 This figure displays the Agent Client Software and the Agent Operating System hierarchies of the Agent dimension.

4.1.2.10 The Visit Dimension

This dimension has no hierarchy. This dimension is used to identify the beginning and end of the visit, it is a spectacle in table 11:

Table 11 Describes the structure of the visit dimension table:

Field Name	Description	Values/Examples
VisitKey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
Description	The value or description. For the beginning of the visit, it is "Start" to an end of the survey page "End"	Start, End.

4.1.2.11 The Content Page Dimension

This dimension has no hierarchy. This dimension is used to identify a page as a content page or not. indicated in table 12

Table 12 Describes the structure of the Content page dimension table:

Field Name	Description	Values/Examples
ContentPagekey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
Description	"Yes" to specify a content page. "No" to specify other files	Yes or No

4.1.2.12 The Document Dimension

The dimension of document describes some information about web pages that can be retrieved from the Web that appears on the screen after a specific keyword. This dimension consists of many attributes that define each document and assign its position in the list of results that should appear on the screen. The dimension of document provides structured bibliographic information on the document, indicated in table 13.

Table 13 Describes the structure of the document dimension table:

Field Name	Description	Values/Examples
Documentkey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
Description	The dimension of document provides structured bibliographic information on the document	-

4.1.2.13 The Keyword Dimension

The dimension of keyword defines one of the two main services that our engine is working, where the dimension will contain words and information about these words. It will be permanently attached to the clickstream and takes significant values in many contexts. Dimension keyword well understood by the design community data warehouse. It will contain the necessary attributes to describe each keyword. Some of these attributes can create a hierarchy of goods that allows groups of keywords to be rolled into ever more massive. Other attributes have nothing to do with the hierarchy of goods, but are useful simple descriptors. In other words, each group of keywords will belong to a certain category. We can have an attribute in this dimension that looks like "Category". In addition, the "PROGRAMMING" category can be for example belonging to a higher category is "Java". In this case, we can see another attribute in this dimension as that is as "Category". This dimension will contain both "subcategory" and "Category" attributes.

The dimension as a search captures phrases, keywords or Boolean expressions that are part of a referral from an external search engine or part of a local search, the Search dimension includes one hierarchy: The Search Hierarchy (Search Category 3, Search Category 2, Search Category 1 , Search). The category dimension describes the various categories of subjects used to retrieve the document (eg Legel, medical). Each dimension is located in a category of related work. this category may be obtained from a thesaurus or hierarchy of concepts such as WordNet or manually derived from series of related work.

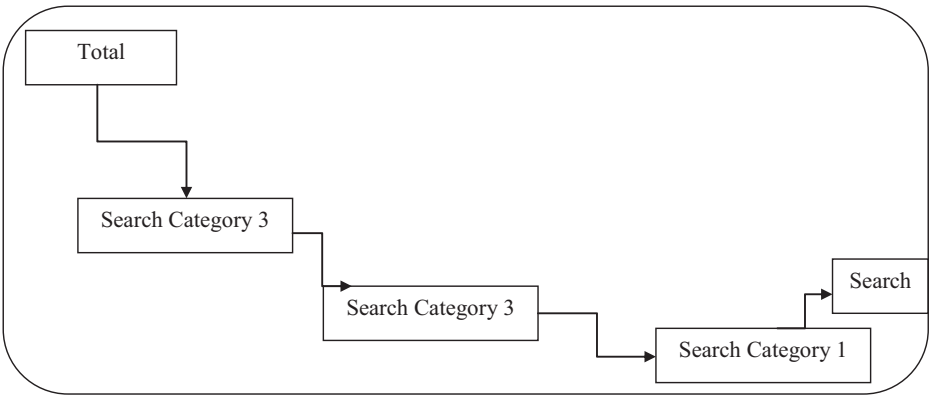


Figure 55 displays the Search hierarchy of the Search dimension. This hierarchy contains the following levels, in descending order: Search Category 3, Search Category 2, Search Category 1, Search.

Table 14 Describes the structure of the Keyword dimension table:

Field Name	Description	Values/Examples
Keywordkey	Primary key (Surrogate key) for the dimension	1, 2, 3, 4, ……
Description	the description of the "keyword" dimension where that dimension contains "Sub-Category" and "Category" attributes	-

4.2 Practical application

In this section, we will discuss the practical application of the above in the theoretical side of design data warehouse and explain each step of the design of tables, dimensions and cube cherished figures in clearer vision and to make this thesis as a reference for those who want to take the design of the data warehouse as a starting point or base on which to build future essays. We will now proceed to develop a practice of the data warehouse and the user in the creation of our own in order to preserve privacy in the creation of everything that needs the data warehouse. "In this thesis, we used Oracle 9i Warehouse Builder to design the data warehouse".

Oracle9i Warehouse Builder is a data warehousing tool you can use to:

- Design: Warehouse Builder user interface contains graphical editors that enable you to design a complete logical model of your warehouse.
- Extract, transform, and load: Warehouse Builder includes a graphical interface that enables you to plan how to extract data from a variety of sources, transform it, and configure it for loading into the data warehouse.
- Deploy and Update: Warehouse Builder code generator enables you to deploy and populate a data warehouse or data mart without manual coding.
- Manage: Warehouse Builder integrates with Oracle Workflow and Oracle Enterprise manager, enabling you to schedule data warehouse updates.
- Integrate: Warehouse Builder integrates with Oracle database and query tools for a complete business intelligence product. [44]

Warehouse Builder enables you to design the following relational objects in the database :

There are many types of business intelligence systems: relational, multi-dimensional, OLAP, and any number of combinations of these. All serve specific needs and have unique characteristics. The typical problem is that these systems are built with separate tools on

separate environments and cannot be combined. The ideal situation would be to combine an OLAP environment with pre-calculated data and with and ad-hoc query environment, thus giving people optimal flexibility. Choosing a tool set that provides this integration reduces the development costs and increases the users' satisfaction and productivity . Warehouse Builder is ideally positioned to solve the design issue resulting from various best-of-breed solutions. It sits on top of the Oracle database and enables you to design all relevant objects in that database .

- Tables – including constraints and indexes
- Views – including the view query
- Materialized Views – including the refresh options
- Fact tables – including bitmap indexes and keys to the dimensions
- Dimensions – including the relational dimension and storage object

As of Oracle9i ،the OLAP constructs are also available in the Oracle database. Warehouse Builder is uniquely positioned to make these constructs available from a design perspective. Within the same user interface, it enables you to design for the OLAP catalog, reusing some of the relational design objects :

- Dimensions – representing dimensions in OLAP
- Facts – representing cubes in OLAP
- With this, Warehouse Builder is one of the very few tools to have full design capabilities for both environments, even reusing the objects from one environment in the creation of the other. It is then obvious that Warehouse Builder saves time in designing the end user environment. An additional benefit is the user experience, which is now limited to one interface for both systems.

Using the graphical editors and wizards available user interface in Warehouse Builder, you can design a data warehouse logical model. When you create a logical model of your data warehouse using Warehouse Builder, you define the metadata that describes the complete data warehouse, from data sources through processing of data warehouse schema finished. Metadata defined in Warehouse Builder is stored in the repository and used to generate the installation scripts to create your data warehouse. All design work in Warehouse Builder is done in a thesis. A thesis contains all objects and metadata definitions for building a data warehouse. These objects and definitions are distributed among the modules of a maximum of three different types:[44]

- Target Data Warehouse
- Source file
- Source database

Warehouse Builder consists of the following parts:

- Repository: A set of relational objects in an Oracle8*i*/9*i* database that store metadata definitions used in the data warehouse. The repository also stores transformation libraries.
- Client: A graphical user interface that enables you to interact with the repository using wizards and editors. Use the client to logically design your data warehouse and generate the code scripts used for deploying and loading a data warehouse.
- Runtime Environment: A set of Oracle objects that you use to set up your target schema. It provides auditing capabilities and a graphical user interface for viewing audit messages.
- Browser: Integration with Oracle Portal provides reports on your metadata, including lineage and impact analysis.
- Application Integrators: Warehouse Builder integrates with Business Intelligence tools, such as Oracle Discoverer, for reporting, analysis, and data mining capability. You can also integrate application data and metadata from sources such as SAP.

4.2.1 Name of use:

The name of the user to create. Enter the new user name. The user name can contain only characters in your character set database and can be more than 30 bytes long. Profile The profile assigned to the user. Use the drop-down list to select the profile you want to assign to the user. The default profile is assigned if you do not make a selection. The Oracle authentication method used to authenticate the user.

- Password: Requires a password for the connection. Enter the password in the entry field adjacent text. Enter the password in the textbox Confirm text for verification.
- External: Indicates that the operating system checks the user.
- Global: Indicates that the user is identified worldwide among multiple databases.

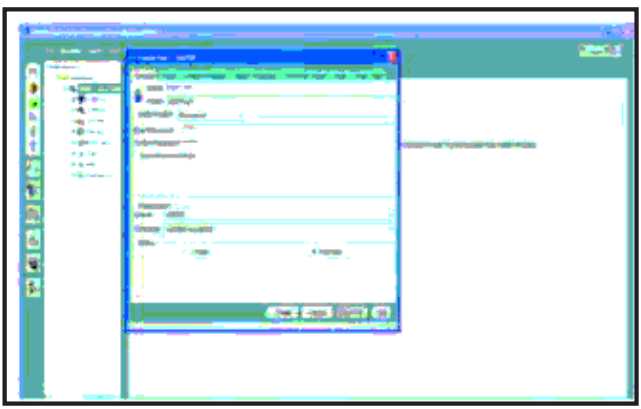


Figure 56 to create user in oracle

4.2.2 Table creation:

The wizard introduction page table allows you to set the table name, schema, and table space belongs to the new table. What do you want the name of the new table to be? What model do you want the table to be part of? Select a schema from the list. The list contains all available models in the database. Tablespace that you create the table? Select a tablespace from the dropdown list. The list contains all the tables spaces in the selected schema.

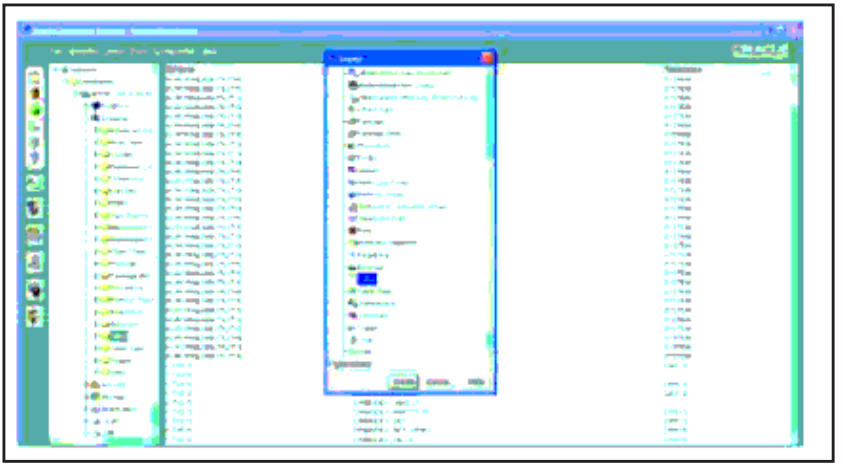


Figure 57 Table creation

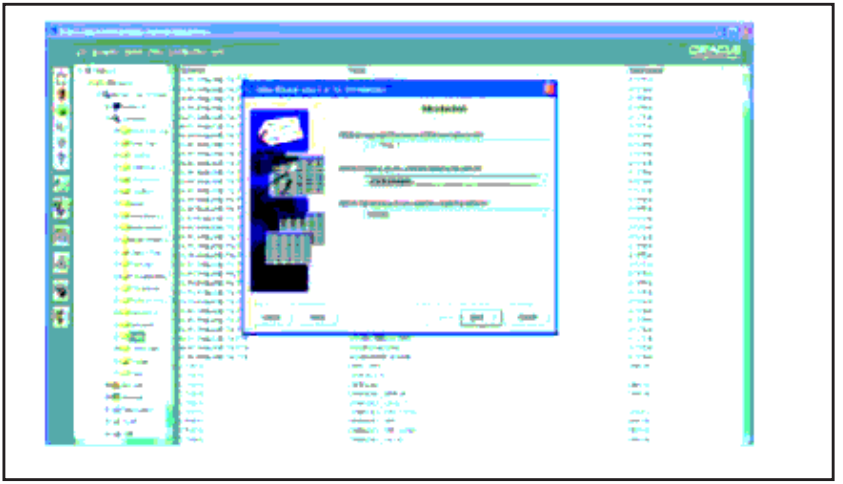


Figure 58 Table creation

4.2.2.1 Table Columns Definition

Defining the columns of the table wizard page allows you to define the columns of the table. Name column enter a valid Oracle identifier. For more information about naming conventions. Data type of the column select the specific data type for the column in the list. For more information on the use of data types, Size Enter the number of bytes allowed for values specified in the column. Indicate scale the number of digits to the right of the decimal point. Does this column has a default value? If so, please enter. Enter an expression that serves as default value for this column in all rows for which the INSERT statement omits a value for the column.

Adds a new column definition of the table. Remove removes the definition of the selected column of the table. Up Arrow Move the definition of the selected before the previous column. For this button is active, the selection on the list should not be on the first point. Down Arrow Move the definition of the selected after the next column. For this button is active, the selection on the list should not be the last item.

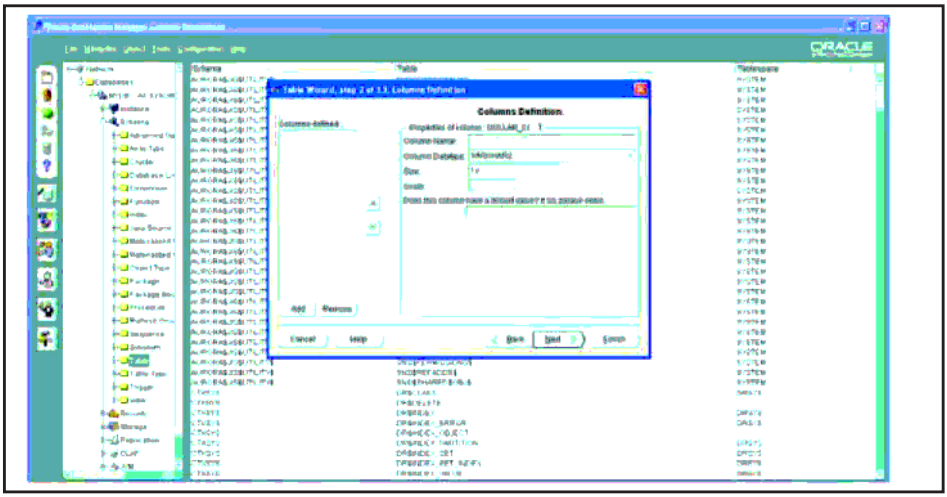


Figure 59 Columns Definition

4.2.2.2 Primary Key Definition

The page definition primary key of the table allows you to define primary key constraints on table columns. Do you want to create a primary key for this table? No, I do not want to create a primary key.

Selected by default, specifies that no primary key must be defined for the table. Yes, I want to create a primary key. Indicates that a primary key must be created. When this option is selected, all previously defined on the Definition page of the columns of the table columns in the multicolumn list. Click the control field for each column to display the position of the column in the primary key. To remove a column from the primary key, click on the entry of the New Order of the column out of the sequence of columns. Re-order columns.

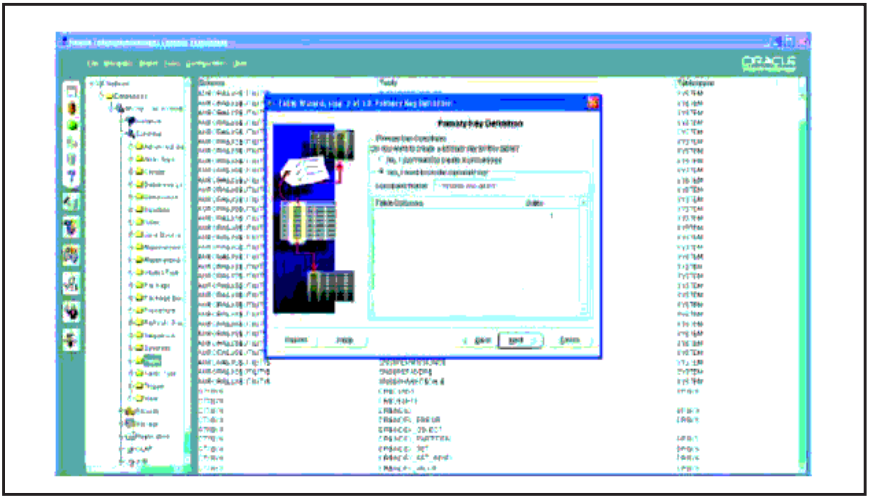


Figure 60 Primary Key Definition

4.2.2.3 Null and Unique Constraints:

Page null and unique constraints wizard table allows you to define null and unique constraints on the columns of the table. The column name and data type are displayed for each column. Click on a column in the list of defined columns to view or change its null and unique constraints. Then the column value is null? Yes is selected by default. Is the value of the column to be unique? No is selected by default. Left Arrow Go to the column defined above. At lease one column must be defined for this button to be active. The Status column displays the number of columns and the number of columns in the table. Go right arrow next to the column defined.

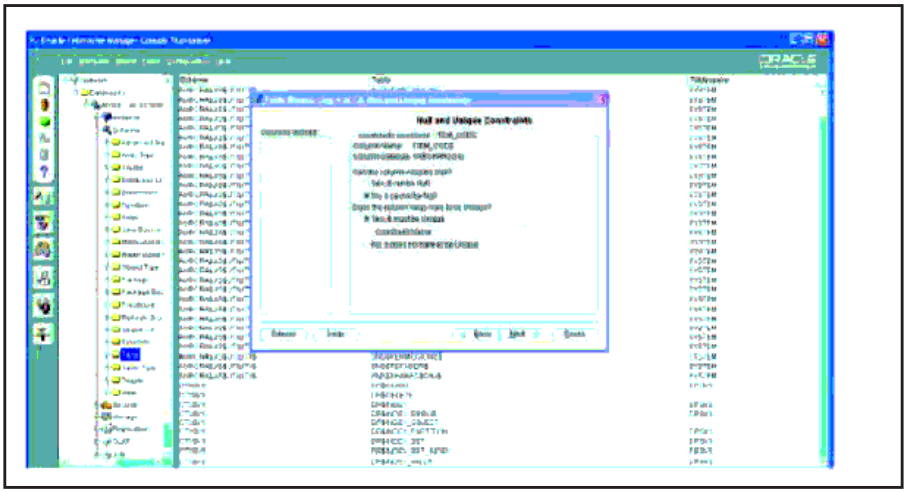


Figure 61 Null and Unique Constraints

4.2.2.4 Foreign Constraint:

Page foreign constraint table allows you to define foreign key constraints. The column name and data type are displayed for each column. Is a foreign key column? No, the column is not a foreign key this setting is selected by default. Yes, the column is a foreign key referenced schema: drop-down list displaying all available diagram in the current database. By default, the schema of the user appears in the text entry field. Referencing table: drop-down list shows all available tables in the schema reference. This field is only active if a schema is selected.

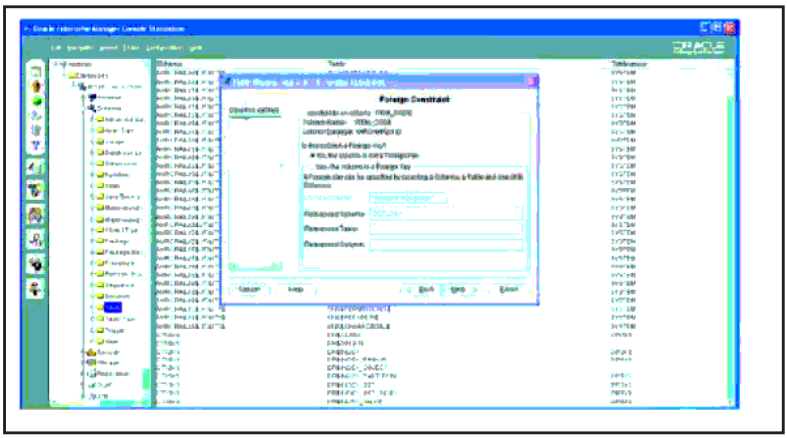


Figure 62 Foreign Constraint

4.2.2.5 Check Constraints:

Constraints validation page table allows you to specify the control requirements for table columns. The column name and data type are displayed for each column. Is this column a check condition? No, the column does not have a check condition. This setting is selected by default. Yes, the column has a check condition. Once selected, the "What is the condition of verification of this column?" text entry field becomes active. Examples are given for check constraints.

4.2.2.6 Summary Page Name:

Name of the table being defined. Diagram wherein the table is defined. See the SQL generated SQL that you set in the previous pages. If you wish to change your table and index definitions, go to the appropriate page of the table.

4.2.3 Dimension:

Provide a name for the dimension object, then select the schema in which it is created. The name of the dimension will be capitalized by default. To include uppercase and lowercase letters and special characters in the name, enclose the name in quotes.

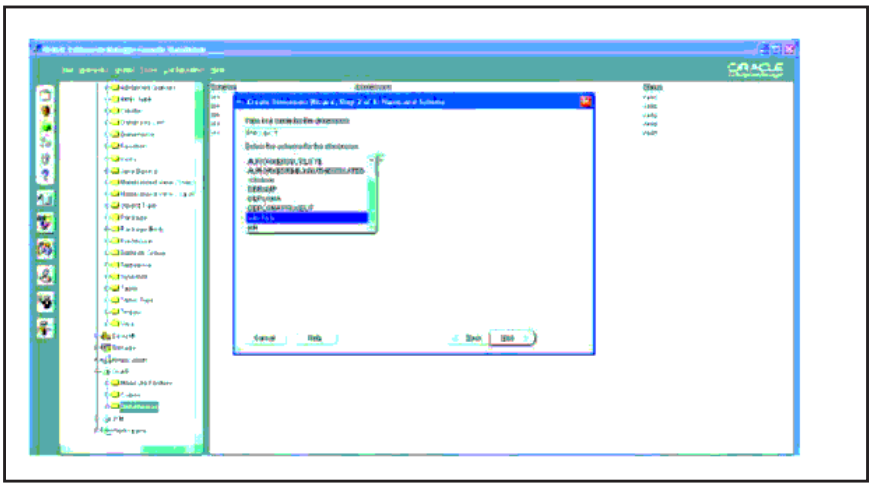


Figure 63 Specify name of Dimension

4.2.3.1 Define the levels in the dimension:

☒Level identifies the dimension members stored in one or more columns in a dimension table. You can specify the hierarchical relationships between levels in a later step. To set a level: Click New and enter a name for the level in the Properties dialog box. By default, the name will be in uppercase. To include uppercase and lowercase letters and special characters in the name, enclose the name in double name quotes. The level appear in the Levels box. If the dimension is a time dimension, you can choose to specify a level combo guy. The drop-down list of the types include Year, Quarter, Month, and types of levels of day as long as you have not yet created these levels. If the dimension is not a dimension of time, the type of level will be normal. In the Properties dialog box, enter the name and schema of the dimension table. Select one or more columns of the dimension table as columns for source level. Use the arrow button to move the column names from the Available Columns list to the Selected Columns list.

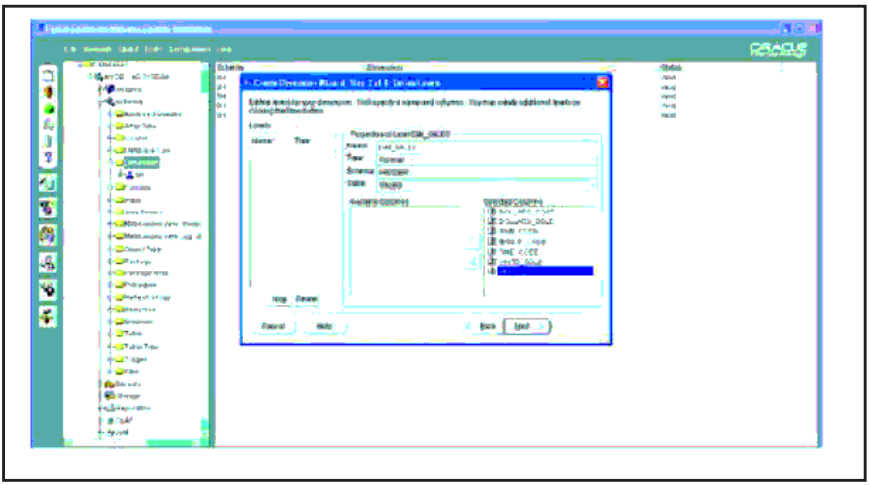


Figure 64 define the levels in the dimension

4.2.3.2 Define Attributes (with OLAP) :

Define sets of attributes in the dimension. Level attributes represent information about the dimension members at a given level. For example, in a document dimension, it could be a brand name associated with each word and each category. The set of attributes, which could be called BRAND, contain two-level attributes: a column with keywordkey level and another involving a column with the level of the category is associated. A level may have more than

one attribute level, provided that each of them is defined within a set of separate attribute. To define a set of level attributes: Click New to create a new set of attributes, or choose one of the attribute names (if available) in the Attribute box. For a new attribute, type the name in the Properties dialog box. If the attribute is already listed in the attribute box, select it to display its name in the Name box. By default, the name will be in uppercase. To include uppercase and lowercase letters and special characters in the name, enclose the name in quotes. Choose the type of attribute. It can be a normal attribute or a short or long description.

The default value is normal. If you selected one of the default attributes for the dimensions of time, the attribute type is End date or time span and no list of choices available. Choose one or more levels of the box available levels and use the arrow button to move them to the Selected Levels area. For long and short (and END_DATE and Time_Span for time dimensions) descriptions, you must choose the available levels. For each level you selected, click the box of the corresponding source column. A drop-down list of columns appears.

4.2.3.3 Define Hierarchies

Define hierarchies of the dimension. Hierarchies to establish a parent-child relationship between levels. The dimension can have a single hierarchy, multiple hierarchies, or no hierarchies. If the dimension has multiple hierarchies, the same levels can be used in more than one hierarchy. If the dimension is used for OLAP and has more than one hierarchy, all hierarchies must share the same leaf level. To define a hierarchy: Click New and enter the name of the hierarchy in the Properties dialog box. By default, the name will be in uppercase. To include uppercase and lowercase letters and special characters in the name, enclose the name in quotes.

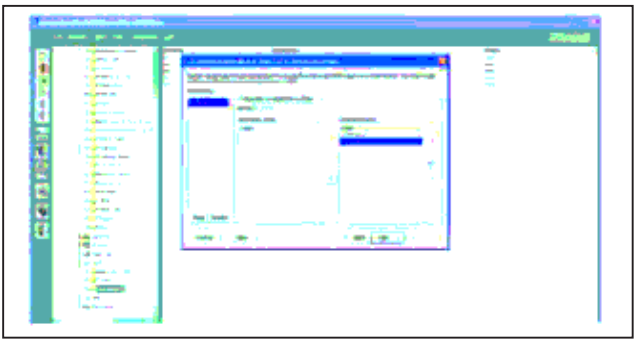


Figure 65 Define Hierarchies

4.2.3.4 Specify Joins Specify:

The foreign key relationships between the levels of a hierarchy. Specify the join page is displayed only when your dimension hierarchies with levels from tables distinct dimensions (as in a snowflake schema). To specify keys to join a hierarchy: Click the name of the hierarchy in the hierarchy area. Pairs of parent-child level in the hierarchy that require joints appear in the box pair level. When you select a pair of level, the source column or a column assembly for the parent level appear in parent columns. Click the Columns area of the corresponding child. A drop-down list of columns appears. Select the key column in the child table that contains a column value from parent table.

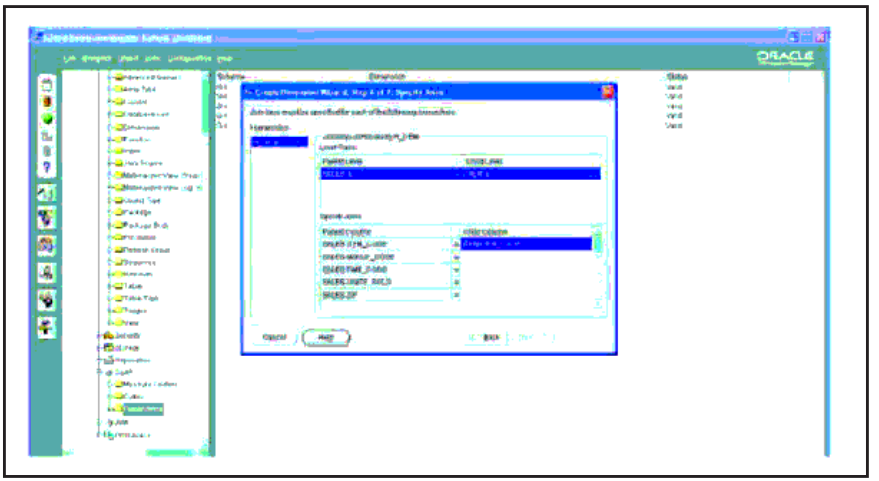


Figure 66 Specify Joins Specify

4.2.4 Cube

What is a Cube?

A cube is a logical organization of multidimensional data. A cube is derived from a fact table, which must already exist in your data warehouse. The cubes are purely informational objects and do not provide data storage. The actual data resides in the fact tables at the source. Cubes exist only for OLAP and are not part of the data dictionary. The cube data is classified by size. All data must be exactly the same size. The cubes are the mechanism for the creation of measures, metadata objects recognized by business intelligence applications using Oracle OLAP Services.

A measure can be a single fact, or it can be a fact derived from Existing data using a mathematical operation or a data transformation. To create a cube, use the Create Cube Wizard or the OLAP cube property sheet Within Management. To edit a cube, use icts property sheet.

A cube is defined Within a Specific scheme. Its component parts (fact table and dimension objects) May exist in the cube's gold scheme in different scheme. You can associate with a cube materialized views. By Storing summary data from the cube in materialized views, You May Improve the performance of queries generated by OLAP Services.

The principal steps of Creation Cube are indexed as:

1 - Supply general information:

Enter a name for the cube, then select the schema in which it is created. Deliver a display name and description for use in the OLAP management.

2 - Choose a fact table:

Choose a fact table as the basis for the cube. The fact table contains the data for the measures in the cube. To select a fact table, use the arrow button to move the name of the table or view name from the list of available tables in the window of the selected table.

3 - Add Dimensions

Choose the dimensions that will be associated with the cube. To select a dimension, use the arrow to move the name of the dimension from the Available Dimensions list to the list of selected dimensions. To create a new dimension, click Create to launch the Create dimension.

4 - Display Dimension Properties

Display the properties of each dimension associated with the cube. Dimension properties consist of an optional alias name, a default hierarchy, and a join between the fact table and dimension table. The dimensions that you specified in the previous step are listed in the dimension (Alias) box.

For each dimension, specify the properties of dimension by providing information in the mapping box size. To define the properties of a dimension: In the Alias box dimension, you can provide a logical name for the dimension. Alias dimension are logical names for the dimensions. Different logical dimensions can be defined according to the same real dimension, but using different hierarchies and different mappings to the fact table. If you do not do providing an alias name of the dimension of the object is used. Note that the mapping of the information table is independent of the hierarchy. In the box hierarchy calculation, specify the default hierarchy for aggregating data on the dimension or dimension alias. Join

in the Level box, specify the level where the size or alias dimension joined to the fact table. By default, the leaf level of the dimension is selected as the level of join. It be readily available to OLAP Services, the dimension requires a unique leaf level (all hierarchies in the dimension must share the same level of the sheet). However, if the dimension is more than one leaf level, the first level of the paper in alphabetical order is the default join level. All levels of leaves are marked with an icon in the list of levels. Once you have specified the level of join, the key column for this level appears in the key column in the table Dim Select foreign key column in the fact table that corresponds to this key column by selecting a column in the drop-down list under Fkey Pass fact Table.

5- Specify Measures

Clarify measures to be included in the cube. "The cube must have at least one measure" By default, all columns of type NUMBER data in the fact table are listed as measures in the area of action. Click New to create a new measurement. Click Delete to delete a measurement. To specify the properties of a measurement: In the Name of the measure, give a name to the measure. The name of the measure will be capitalized by default. To include uppercase and lowercase letters and special characters in the name, enclose the name in quotation marks in the Name box of the display, provide a display name to be used by the OLAP Management. In the Column area source, select a column in the list. The data type of the column is displayed in the box data type. Do not select columns with the following data types: BLOB, CLOB, NCLOB, RAW or LONG RAW. In the Description box, you can optionally provide a description of the cube.

6- Summary

Display brief information on the cube. Measures and dimensions of the cube are listed in the Components box. The source column for each measure and the source dimension tables for each dimension are listed in the Sources area. Click on graphic display for displaying a graphical representation of the cube. Click Show SQL to display the SQL code that will be generated to create the cube. If you want to generate materialized views to optimize queries that run on the cube, select the Summary Advisor Wizard Launch box. Use Summary Advisor wizard to optimize the cube. Click Finish to create the cube.

References

[1] NAKHAL, BILAL (2012). "META-SEARCH ENGINES & DATA WEBHOUSING: CHALLENGING Technologies & Performance Tuning Techniques". *MS thesis, University of ARTS, SCIENCES AND TECHNOLOGY UNIVERSITY IN LEBANON. Beirut, Lebanon.*

[2] Romm M., Introduction to Data Warehousing, San Diego SQL User Group.

[3] Kimball Ralph, Merz Richard (2000), "*The Data Webhouse: Building the Web-enabled Data Warehouse*", New York, John Wiley & Sons.

[4] Kimball R., "The Data Warehouse Toolkit", John Wiley, 1996

[5] Hui-Huang Hsu (2008). Data Warehousing and Mining: Concepts, Methodologies, Tools, and Applications (pp. 93-102).

[6] W. H. Inmon, "Building the Data Warehouse, 3th Edition", John Wiley, 2002

[7] Franconi E., Introduction to Data Warehousing, Lecture Notes, http://www.inf.unibz.it/~franconi/teaching/2002/cs636/2 ,2002

[8] Goyal N., Introduction to Data Warehousing, BITS, Pilani Lecture Notes

[9] Thomas Connolly & Carolyn Begg., "Database Systems, 3th Edition", Addison-Wesley, 2002.

[10] Kimball R., Letting the Users Sleep, Part 1, DBMS, 1996, http://www.dbmsmag.com/9612d05.html

[11] SRITHONG, P. (2005). "DATA WAREHOUSING WITH AN OLAP SYSTEM: A CASE STUDY FOR THESIS OPERATIONS OF THE SCHOOL OF NURSING, RAMATHIBODI HOSPITAL, MAHIDOL UNIVERSITY" MS thesis, MAHIDOL UNIVERSITY.

[12] BERIL, P. (2005). A COMPARISON OF DATA WAREHOUSE DESIGN MODELS. MS thesis, Atilim University.

[13] Elmasri R., Navathe S., "Fundamentals of Database Systems", 3rd Edition,Addison-Wesley, 2000.

[14] Batini C., Ceri S., Navathe S., "Conceptual Database Design-An Entity Relationship Approach", Addison-Wesley, 1992.

[15] Abello A., Samos J., Saltor F., A Data Warehouse Multidimensional Data Models Classification, Technical Report, 2000.

[16] Abello A., Samos J., Saltor F., A Framework for the Classification and Description of Multidimensional Data Models, Database and Expert Systems Applications, 12th International Conference, 2001.

[17] Gatierrez A. and Marotta A., An Overview of Data Warehouse Design Approaches and Techniques, Uruguay, 2000.

[18] Ballard C., Herreman D., Schau D., Bell R., Kim E., and Valencic A., "Data Modeling Techniques for Data Warehousing", IBM Redbook, IBM International Technical Support Organization, 1998.

[19] Thomas Connolly & Carolyn Begg., "Database Systems, 3th Edition", Addison- Wesley, 2002.

[20] Kimball R., Letting the Users Sleep, Part 2, DBMS, 1997, http://www.dbmsmag.com/9701d05.html

[21] Phipps C., Davis K., Automating Data Warehouse Conceptual Schema Design and Evaluation, DMDW'02, 2002

[22] Rizzi S., Open Problems in Data Warehousing., http://sunsite.informatik.rwthaachen.de/Publications/CEUR-WS/Vol-77/ DMDW 2003, Berlin, Germany.

[23] Golfarelli M., Maio D., Rizzi S., Conceptual Design of Data Warehouses from E/R Schemas, Proceeding of the 31st Hawaii International Conference on System Sciences (HICSS-31), Vol. VII,1998.

[24] Golfarelli M., Maio D., Rizzi S., The Dimensional Fact Model: A Conceptual Model For Data Warehouses, International Journal of Cooperative Information Systems (IJCIS), Vol. 7, 1998

[25] Golfarelli M., Rizzi S., Designing the Data Warehouse: Key Steps and Crucial Issues, Journal of Computer Science and Information Management, 1999

[26] Sapia C., Blaschka M., Höfling G., Dinter B., Extending the E/R Model for the Multidimensional Paradigm, Proceeding 1st International Workshop on Data Warehousing and Data Mining (DWDM98), 1998

[27] J. Han and M. Kamber, "Data Mining: Concepts and Techniques", Chapter2: Data Warehouse and OLAP Technology for Data Mining, Barnes & Nobles, 2000

[28] Moody D. L. and Kortink M. A. R., From Enterprise Models to Dimensional Models: Methodology for Data Warehouse and Data Mart Design, http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS//Vol-28/ DMDW 2000 , Stockholm, Sweden.

[29] Teklitz F., The Simplification of Data Warehouse Design, Sybase, 2000.

[30] Manjunath T.N, Ravindra S Hegadi, Ravikumar G K."Analysis of Data Quality Aspects in DataWarehouse Systems", International Journal of Computer Science and Information Technologies, Vol. 2 (1), 2010, 477-485.

[31] Valery A. Petrushin and Latifur Khan, "Multimedia Data Mining and Knowledge Discovery", 2007 - London: Springer-Verlag, pp. 3- 17.

[32] Alejandro Gutiérrez, Adriana Marotta, An Overview of Data Warehouse Design Approaches and Techniques,Instituto de Computación, Facultad de Ingenieria, Universidad de la República, Montevideo, Uruguay, October 2000.

[33] Data Warehouse 101 Handbook, Introduction to the Data Warehouse, Professional Development for the Massachusetts Education Data Warehouse, Version 2.0 ,September 2008 ,www.doe.mass.edu.

[34] Paul Lane, Viv Schupmann and Ingrid Stuart (Change Data Capture), Oracle Database Data Warehousing Guide, 10g Release 2 (10.2) B14223-02, Copyright © 2001, 2005, Oracle.

[35] Ravikumar G K et al., (2011). A STUDY ON DESIGN AND ANALYSIS OF WEB MART MINING AND ITS RELEVANCE TODAY. International Journal of Engineering Science and Technology, ISSN : 0975-5462, Vol. 3 No. 4.

[36] Hayder, K. (2001). Design and Implementation of Framework for a Data Warehouse. Dissertation High Diploma. College of Science University of Baghdad.

[37] Ralph Kimball, The Data Warehouse ETL Toolkit, Wiley India Pvt Ltd., 2006.

[38] Donald K. Burleson, Oracle Tuning , The Definitive Reference, February 2010 for Second Edition.

[39] S. Kotsiantis, D. Kanellopoulos, P. Pintelas, Multimedia Mining, SEAS Transactions on Systems, Issue 10, Volume 3, December 2004, pp. 3263-3268

[40] K. Duncan, J. Thomann, D. Wells, W. McKnight, D. Marco, M. Haisten, D. Wells Proceedings of the Fourth Annual Implementation Conference, TDWI - February 1999.

[41] Song I., Whang K., "Database Design for Real-World E-Commerce Systems", IEEE Data Engineering Bulletin, March 2000, Vol. 23, No. 1, 23-28

[42] Song I., LeVan-Shultz K., "Data Warehouse Design for E-Commerce Environment",WWWCM99

[43] Ralph Kimball, "Clicking with your Customer, Intelligence Enterprise", Intelligent Enterprise, Jan 05, 1999, Vol 2, No. 1.

[44] Oracle9i,Data Warehousing Guide,Release 2 (9.2),March 2002,Part No.

A96520-01.