CHAPTER - 2

DESIGN OF DIMENSIONAL MODEL FOR

CLINICAL DATA STORAGE AND ANALYSIS
2.1 Introduction

A major problem being faced by most of the organizations and industries around the world is efficient storage of large amount of data and its maintenance. Most of the financial services, telecom giants and other service providers hence forth have tried to take help from information technology for getting storage solutions. Beside storage, they are also interested in using the available data for future predictions for growth of the business. Ralph Kimball suggested [1] about operational systems and data warehouse that can be associated with a organization corresponding to their data storage needs. If operational system is meant for turning the wheel of the organization then a data warehouse, on the other hand, watch the wheels of the organization getting turned [2]. It is widely recognized that the data warehouse has profoundly different needs, clients, structures, and rhythms than the operational systems of record. A Datawarehouse (DW) is a specialized form of relational database that stores information oriented to satisfy decision-making requests [3]. A very frequent problem in enterprises is the impossibility for accessing to corporate, complete and integrated information of the enterprise that can satisfy decision-making requests. In general, a DW is constructed with the goal of storing and providing all the relevant information that is generated along the different databases of an enterprise.

A similar kind of scenario is being faced now in the field of clinical science, where large amount of data is generated on daily basis. Every day in a different country in a different state in a different city in a different hospital lands a new patient, a new case, a new heap of data but an old problem still persists, i.e. of data storage for nearly every hospital or research institute. It’s time now for change and advancement. The extraordinary explosion of medical knowledge, technologies, and ground-breaking drugs may vastly improve healthcare delivery for the welfare of its consumers, but the key is to implement these technologies, to extract as much as we can. Since clinical informatics is a multidisciplinary field, it combines data representation, cognitive science, policies, telemedicine and data discovery. The ability to quickly and efficiently retrieve information makes the creation of an organized database indispensable, and thus clinical informatics makes the representation and interpretation of complex medical terms quite simple for a specialized form of clinical database. Cognitive science comes into play to help those in the medical community, understand process and perceive artificial intelligence and computing.

Once diagnosis process of an individual is being completed, what hospitals usually consider to be the junk data might be as important and meaningful as a medicine given to a patient. It’s all about fetching information from this raw data which can form a base for knowledge discovery. The information may be of help to a patient corresponding to temporal analysis of clinical data as and
when studied. Also on a larger scale this information can help in prevention, proactive treatments and early detection of certain life threatening diseases at population level.

Clinical informatics, deals majorly with the clinical data concerned with a patient or a group of patients, which may include a patient’s health records, history with the disease, and treatment description etc. Technology allows clinical research and patient care to become more integrated and interactive. These data can then be used to answer questions relevant to specific communities and can be extrapolated to a national level, a classical example of which is the Slim Prim Biomedical Database [4]. Furthermore, information can be assimilated for community education to help improve healthcare.

Datawarehouse is usually developed using a specific blueprint design, said to be the dimensional model. A dimensional model is a “specific discipline for modeling data that is an alternative to entity relationship modeling” [2]. Like an entity relationship model, a dimensional model reflects a data structure and is specifically designed to model data in a way that emphasizes user understandability, enhances query performance, and tracks change [2,5,6]. To achieve these design characteristics, a dimensional model is typically being kept in a dc-normalized state. There are two kinds of tables in a dimensional model - dimension and fact. Dimensional tables consist of descriptive attributes which can help in describing business entity whereas a fact table consists of measure corresponding to each of the feature. Dimension tables contain primary keys which associate the dimension attributes to the fact table, and textual descriptions. Fact tables contain foreign keys and measurements. An effective data warehouse can be built and maintained only when, it has an effective design and well defined grain of its dimensional model.

To address the clinical data integration issues and to have a data warehouse based storage structure which can effectively handle the clinical data temporally, a clinical dimensional model is being proposed, that will address the concerned dimensionality issue. Ralph Kimball [2] addressed about the typical health care cycle, but has discussed the entities in detail concerned with typical billing cycle. However, with respect to challenges highlighted and research being carried out in various fields of genomics, proteomics, etc. along with clinical sciences, it can be associated with personalized medication and therefore would need storing the data at the granular level of a person. The various domains which can be said to associated with effective recording of an individual health data is being depicted in Figure 3, which depicts in near future along with clinical and drug related data, in addition, the genomics and proteomics data are also going to play a major role with respect to an effective treatment process. Each of the said domains can lead to development of a specific data mart associated with the data warehouse. The current study is focusing on one of the
said domains of clinical data by providing a dimensional model design which can be used for development of a clinical data mart.

![Diagram of domains associated to health care](image)

**Figure 3** - Domains associated to health care.
(different domains that can be associated with health care in future)

### 2.2 Materials and Methods

#### 2.2.1 Data processing

The data obtained from different hospitals could not be straight away used for analysis, "*data in the real-world is dirty*" i.e. have errors, unusual values, and inconsistencies. Data quality can be assessed in terms of accuracy, completeness and consistency. The data that we obtained was:

- **Incomplete**: Some of the records were lacking attribute values. Eg: few of the patient's record did not have 'basophil' count while few missed 'serum bilirubin', that leads to incompleteness in the data.

- **Noisy**: Means that the data contains errors or outliers Eg: A record had value of 10 in platelet count which was an error.

- **Inconsistent**: Containing discrepancies in codes or names or format Eg: The date in few records were in mm/dd/yyyy format while in others as dd/mm/yyyy format.

There are many reasons that lead to such kind of data. Incomplete, noisy, and inconsistent data are commonplace properties of large, real-world databases and data warehouses. Attributes of interest may not always be available, such as kidney function test may not be performed for a patient every time visiting the hospital. Other data may not be included simply because it was not considered important at the time of entry. Relevant data may not be recorded due to
misunderstanding, or because of equipment malfunctions. Data that were inconsistent with other recorded data may have been deleted. Furthermore, the recording of the history or modifications to the data may have been overlooked. Missing data, particularly for tuples with missing values for some attributes, may need to be inferred. Data can be noisy, having incorrect attribute values, owing to the following:

- The data collection instruments used may be faulty.
- There may have been human or computer errors occurring at data entry.
- Errors in data transmission can also occur.
- There may be technology limitations, such as limited buffer size for coordinating synchronized data transfer and consumption.
- Incorrect data may also result from inconsistencies in naming conventions or data codes used.

2.2.1.1 Level of Redundancy

It is another important factor in data processing. It is useful to know how much of the data is repeated from the various sources. Redundant data can slow down or confuse the knowledge discovery process. Data reduction and cleaning methods, carefully employed, can aid in removing duplicated data prior to its usage.

2.2.1.2 Major Tasks in Data Preprocessing

- Data cleaning: Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies.
- Data integration: Integration of multiple databases, data cubes, or files.
- Data transformation: Normalization and aggregation.
- Data reduction: Obtains reduced representation in volume but produces the same or similar analytical results.
- Data discretization: Part of data reduction but with particular importance, especially for numerical data.

Accuracy of data is an important criteria to be considered during development of a clinical warehouse especially when there are no Electronic Health/Medical Records (EHR/EMR) implemented [7]. Data incorrectness usually exists because of design or operational deficiency and can be identified where the mapping between the information system state and the real world state break down [7]. Henceforth, with utmost care the dimensional model (data model) of the clinical warehouse was designed based on the descriptive and measurable features of the clinical data. Further, it consists of date and time dimension that ensures temporal storage of data for a patient.
Also, to check the operational deficiencies, the quality assurance of data was ensured by implementing appropriate data processing codes for range and data validation checks [8], re-entering samples of data to assess for accuracy, checks for data completeness and attention for data consistency [9].

The dimensional model had being designed using Erwin data modeller 8.2 [10]; Kettle 3.1 [11] was used to encode ETL mappings for processing of data from source files into the data warehouse; and MySQL 5.0.19 [12] relational database management system was used for the physical creation of warehouse.

2.2.2 Proposed Design for the Clinical Dimensional Model

The advanced form of clinical data warehouse would be complex and time consuming to review a series of patient records. However, its going to be a efficient data repository existing to deliver quality patient care. Data integration tasks of medical data store are challenging scenarios when designing clinical data warehouse architecture.

A few decades ago, physicians knew pretty much everything that is to be known about medicines; most doctors could recollect the names of their patients. However, today, no doctor can keep up with the explosion of medical and health information. While health care organizations have recognized the use of computers, but in comparison to other industries its application in healthcare have not been encouraging. This is because, among other factors, it takes too long to get information in many cases; there is no easy accessibility to data, and no uniform standard among various vendors. But once the data warehouse is ready, it's worth spending the time and money in it.

With its current advents, the clinical domain associated with the health cycle needs major attention. The major problem being faced is of varied dimensionality, ranging from images to numerical form of data which needs to be answered. Based on the same we propose for an appropriate clinical dimensional model for the structure of a clinical data mart that will store data at granularity of an individual corresponding to time. The given model has been designed using Erwin data modeller (version 8.2) [10].

2.2.3 Creation of Clinical Warehouse

MySql 5.0.19 RDBMS package was used to physically create the data warehouse. The data extraction, cleaning and processing process was done using Kettle 3.1 [Extraction, Transformation, Loading (ETL) technology].
CHAPTER - 2

The aim of building this warehouse is to lead to a platform for applying data mining technique to find correlation among various attributes, applying association mining studies, etc. which would help us in deciphering new translational paradigms which could be used by doctors, physicians, other health professionals and even by a common man who has got knowledge about how to use computer and internet.

2.2.3.1 Staging Schema

The **staging database** is a separate data cache (storage area) that helps users in continuous access to application data. Its access continues even when data is being imported from the various external sources and prepared for loading. This minimizes the downtime that user experiences during data loading or data refreshing. Here the data is dumped as it is, without any changes being made to it i.e. the data here is in its original form.

The data obtained from varied sources were formatted into a common input form. All the files were converted into 'csv' and images into 'jpeg' form. The files were categorized into date, time, patient, disease, diagnostic test and image categories respectively. A staging database named as **Clinical Staging Data** was created for storage of data from source files. Each category file types were processed to a particular table in the staging database as defined in Table I. For uniquely identifying data from a particular hospital an additional field was added as "hospital name and location", that was picked from the name of file being processed. SQL queries for creation of tables for staging database can be accessed from Appendix I - A1. ETL mapping were designed using Kettle 3.1 platform to process the files from different source files into respective tables of staging database.

<table>
<thead>
<tr>
<th>TABLE NAME (STAGING SCHEMA)</th>
<th>TYPE OF DATA STORED</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Date</strong></td>
<td>This table stores the date records. Here each date has been given a unique id and other details like week of the year, in which quarter of the year is the dates falling are given.</td>
</tr>
<tr>
<td><strong>Time</strong></td>
<td>This table stores the time records. Here each time value (in terms of hour, minute &amp; second) has been given a unique id.</td>
</tr>
<tr>
<td><strong>Patient</strong></td>
<td>It stores the data collected from hospitals with details of patients records.</td>
</tr>
<tr>
<td><strong>Disease</strong></td>
<td>It stores the data corresponding to all the disease for which a patient may be diagnosed</td>
</tr>
</tbody>
</table>
Diagnostic_test

It stores the data corresponding to all diagnostic tests which may be subjected to a patient.

Image

It stores numerical information of all the images along with the path details of their physical location in the system.

Table I - Details of Clinical_Staging_Data (Staging Schema).

2.2.3.2 Functional Schema (Clinical Warehouse)

A functional schema was created by the name Clinical_Warehouse in which tables were created based on the proposed dimensional model (Figure 4). SQL queries for creation of tables for staging database can be accessed from Appendix I - A2. The data from the Clinica_Staging_Data database, was subjected to cleaning and conversion process so that its appropriate form can be stored in the functional schema which can be used further for analysis and to find correlation. Appropriate mappings designs were developed to process and store the data based on the dimensional model designed. For example values of a particular diagnostic test coming from different sources should be in a common format and stored in Dim_Diagnostics_Test table. The tables like patient dimension have only selected attributes so we accordingly use appropriate function to load the required attributes in the table.

<table>
<thead>
<tr>
<th>TABLE NAME (FUNCTIONAL SCHEMA)</th>
<th>TYPE OF DATA STORED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date dimension (DIM_DATE)</td>
<td>This tables stores the date information which have been processed from the ‘date’ staging table.</td>
</tr>
<tr>
<td>Time dimension (DIM_TIME)</td>
<td>This tables stores the date information which have been processed from the ‘time’ staging table.</td>
</tr>
<tr>
<td>Disease dimension (DIM_DISEASE)</td>
<td>This table contains description for all the diseases associated to humans. Each of the disease have been assigned a unique id. Data processed from ‘disease’ staging table.</td>
</tr>
<tr>
<td>Diagnostic test dimension (DIM_DIAGNOSTIC_TEST)</td>
<td>This table contains details for all the diagnostic tests that a patient gets done e.g. platelet count, TCL, KFT, LFT etc. All the</td>
</tr>
</tbody>
</table>
tests have been given a unique id. Data processed from 'diagnostic test' staging table.

**Patient dimension**

(DIM_Patient)

Patient's personal details such as name, patient_id, age, sex etc. are stored in this table. For each patient an unique id is being autogenerated (given as patient id) whenever information is entered for the first time. Data processed from 'patient' staging table.

**Patient fact table**

(FACT_PATIENT)

This table stores the patient id and corresponding measurements corresponding to a particular diagnostic test with a disease conducted on a particular date & time. It helps to store all the historical information corresponding to any number of tests being conducted for a patient.

**Image dimension**

(DIM_IMAGE)

Stores descriptive information of various types of medical images.

**Patient Image Table**

(FACT_PATIENT_IMAGE_DETAIL)

All the numerical parameters associated with a patient's image are stored in this table.

---

**Table II - Details of Clinical_Datawarehouse (Functional Schema).**

---

2.2.2.3 **ETL mappings (using Kettle 3.1)**

The process of cleaning and transforming data is known as ETL, or Extraction, Transformation, and Loading. Proper care of the data is an important part of maintaining a successful data warehouse. Kettle is an open source ETL (Extraction, Transformation and Loading) tool. The product name is spelled as K.E.T.T.L.E, which is a recursive acronym for "Kettle Extraction, Transport, Transformation and Loading Environment". It's a platform-independent ETL tool by Matt Casters [11]. Being an ETL tool, Kettle is an environment that's designed to:

- Collect data from a variety of sources (extraction)
- Move and modify data (transport and transform) while cleansing, denormalizing, aggregating and enriching it in the process.
- Frequently store data (loading) in the final target destination, which is usually a large, dimensionally modeled database called a data warehouse.
**Figure 4** – Flow structure of data in the warehouse managed by ETL codes.

Figure 4 depicts how flow of data from various source files into the clinical data mart is managed using ETL codes in Kettle. Table III enlists description of ETL mappings for processing of data from source files into respective tables of staging schema (**Cancer_Staging_Data**). Mappings can be accessed from **Appendix I - B1**.

<table>
<thead>
<tr>
<th>ETL Mapping Name</th>
<th>Source</th>
<th>Target</th>
<th>Operators Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Staging_date</td>
<td>Date.csv</td>
<td>Date</td>
<td>CSV file input, Table Output</td>
</tr>
<tr>
<td>Staging_time</td>
<td>Time.csv</td>
<td>Time</td>
<td>CSV file input, Table Output</td>
</tr>
<tr>
<td>Staging_patient</td>
<td>CSV files based on data of patients obtained from different hospitals</td>
<td>Patient</td>
<td>CSV file input, Get File Name, Target</td>
</tr>
<tr>
<td>Staging_disease</td>
<td>Disease.csv</td>
<td>Disease</td>
<td>CSV file input, Table Output</td>
</tr>
<tr>
<td>Staging_test</td>
<td>Test.csv</td>
<td>Diagnostic_test</td>
<td>CSV file input, Table Output</td>
</tr>
<tr>
<td>Staging_image</td>
<td>Image files along with numerical measures in csv files</td>
<td>Image</td>
<td>CSV file input, Table Output</td>
</tr>
</tbody>
</table>

**Table III** - ETL mapping description for processing data from source files into **Cancer_Staging_Data**.

Table IV enlists description of ETL mappings for processing of data from tables of **Clinical_Staging_Data** staging schema into respective tables of **Clinical_Warehouse**. Mappings can be accessed from **Appendix I - B2**.
<table>
<thead>
<tr>
<th><strong>ETL Mapping Name</strong></th>
<th><strong>Source</strong></th>
<th><strong>Target</strong></th>
<th><strong>Operators Used</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>DW_dim_date</td>
<td>Date</td>
<td>DIM_DATE</td>
<td>Table Input, Add Sequence, Table Output</td>
</tr>
<tr>
<td>DW_dim_time</td>
<td>Time</td>
<td>DIM_TIME</td>
<td>Table Input, Add Sequence, Table Output</td>
</tr>
<tr>
<td>DW_dim_disease</td>
<td>Disease</td>
<td>DIM_DISEASE</td>
<td>Table Input, Add Sequence, Data Validator, Table Output</td>
</tr>
<tr>
<td>DW_dim_test</td>
<td>Diagnostic_test</td>
<td>DIM_DIAGNOSTIC_TEST</td>
<td>Table Input, Add Sequence, Data Validator, Table Output</td>
</tr>
<tr>
<td>DW_dim_patient</td>
<td>Patient</td>
<td>DIM_PATIENT</td>
<td>Table Input, Add Sequence, Select Values, Table Output</td>
</tr>
<tr>
<td>DW_fact_patient</td>
<td>Patient</td>
<td>FACT_PATIENT</td>
<td>Table Input, Select Values, DB Lookup, Data Validator, Table Output</td>
</tr>
<tr>
<td>DW_dim_image</td>
<td>Image</td>
<td>DIM_IMAGE</td>
<td>Table Input, Add Sequence, Select Values, Table Output</td>
</tr>
<tr>
<td>DW_fact_image</td>
<td>Image</td>
<td>FACT_PATIENT_IMAGE_ DETAIL</td>
<td>Table Input, Select Values, DB Lookup, Data Validator, Table Output</td>
</tr>
</tbody>
</table>

**Table IV** - ETL mapping description for processing data from Cancer_Staging_Data (Staging Schema) into Clinical_Warehouse (Functional Schema)
2.3 Results

Figure 5 - Logical representation of clinical dimensional model
(logical data model for the clinical data mart)

The logical design form of the dimensional model (figure 5) is in a star schema representation [13]. Proposed design consists of two fact tables - Fact_Patient and Fact_Patient_Image_Detail, which stores the textual measures and numerical measures obtained from the images respectively. In the given dimensional model, the Fact_Patient table (which would keep track of the numerical measures for diagnostic factors) is referencing to Dim_Date, Dim_Time, Dim_Patient, Dim_Disease, and Dim_Diagnostic_Test and the Fact_Patient_Image_Detail is referencing to Dim_Date, Dim_Time, Dim_Patient and Dim_Image
dimension respectively. Patient_Id serve as the primary key of the Dim_Patient dimension table, which is the unique id that is provided to each patient and this is the id which is majorly linking all other information related to that patient. The dimension further includes other descriptive information associated to a patient like name, age, gender, etc. Keeping in consideration the data in patient dimension may change, Start_date, End_date and Flag attributes have been added and so it can act as slowly changing dimension (SCD) [14-16]. During its physical implementation for a data mart an SCD-Type II implementation can be made for Dim_Patient dimension [14]. While designing the clinical dimensional model the temporal based storage prospect was taken into consideration, henceforth Date and Time dimension are included. Date_Id is the primary key for Dim_Date dimension, which assigns unique id to each of the date value. The dimension also include various date based attributes like month, week, calendar year, quarter, etc., which can help to make an analysis considering different period. Time_Id is the primary key for Dim_Time which assign unique id corresponding to each second of a minute and hour. Separate inclusion of Time dimension ensures irrespective of number of times a test is conducted for a patient on any given date, each measure would be recorded uniquely in the Fact_Patient table. Disease_Id and Test_Id are the primary keys of Dim_Disease and Dim_Diagnostic_Test dimensions respectively. They include various attributes which would describe diseases and various diagnostic tests, respectively. Patient_Id, Disease_Id, Test_Id, Date_of_Measurement_Id and Time_of_Measurement_Id act as composite primary key for Fact_Patient table. It stores with respective to unique key each of the measured values. The Image Dimension (Dim_Image) is linked to Fact_Patient_Ime_Details; here Patient_Id and Image_Id (in combination) with Date_Id and Time_Id act as the composite key. The Fact_Patient_Ime_Details include attributes which would store measures corresponding to numerical conversion of images like area, skewness, mean gray valu, etc.

2.4 Discussion

Interpreting data across multiple systems has been always challenging, and various integration techniques, with varying levels of complexity, have been proposed in the past to solve the problem of data integration and storage [17-20]. Nagarajan et al. [20] identified the potential utilization of solutions using relational database management systems (RDBMSs) for assembling and integrating the data for data-warehousing-based solutions. A relational database model is composed of classes of data, with each class characterized by a set of attributes. This conventional design is ideal for data sets composed of classes with a limited and fixed number of attributes. When each instance has values for all attributes (or columns) within a class (or table), then the database is not filled with numerous null entries and memory is used efficiently. However, research
has revealed that this design is not effective for data sets with large numbers of attributes that vary taking into consideration the time dimensionality [21]. Some of the researches propose use of knowledge-based terminology for identifying data dimensions in clinical informatics [22] and on the conceptual development of IDs using ontology-based systems for the design and integration of clinical data [23]. The inherent variation between databases due to the demands on each system means that there is no consensus on ontology and metadata descriptions. It might therefore be necessary to define a new ontology for each database. Although this approach gives the database designer freedom at the outset, inexperienced designers can spend excess time in researching previous knowledge, seeking an optimum design. Where possible, designers should use pre-existing ontologies. These can be modified as necessary to improve accessibility. The Bio-mediator system provide a theoretical and practical foundation for data integration across diverse biomedical domains via a “knowledge-base-driven centralized federated database” model [21]. However, the efficiency of query processing time and the need to filter out unnecessary query results still are concerns. The data architecture required for clinical data warehousing has been researched in applications such as clinical study data management systems (CDMSs) and clinical patient record systems (CPRSs). They both use an entity-attribute-value (EAV) system (i.e., row modeling) as opposed to conventional database design [22]. The EAV system has the advantage of remaining stable as the number of parameters increases when knowledge expands, a common situation in the basic sciences and in clinical trials [23]. The characteristics of clinical data as it originates during the process of clinical documentation, including issues of data availability and complex representation models, can make data mining applications challenging. Data preprocessing and transformation are required before one can apply data mining to clinical data.

The lacunae's reported can be addressed to an extent by the proposed clinical dimensional model. Further the data storage structure formed would acts as a data collector, data integrator and data provider in the data mining process that could be used by doctors, physicians and other health professionals. The application of classical data warehousing process should be thus able to answer the queries being raised and also be able to mitigate issues like appropriate storage structure of clinical data, able to handle varied sources of data, reduce the dimensionality constraint, and handling of multiple data variables. The data mart for clinical data should be able to render the data in appropriate structures, provide metadata that adequately records syntax/semantics of data and reference pertinent medical knowledge.
2.5 Conclusion

Clinical Informatics is one of the most versed fields and new IT solutions are being designed for its effective management. However, there still a gap in the effective storage solution along with techniques for correlation of the data. We dream of an era in which all the genetic (genomic and proteomic) information of an individual along with drugs data will be correlated with his/her clinical and information aspect.
References


