Study of Design and Development of Data Warehousing framework

Auchinto Chatterjee[1], Dr. Sharvani GS[2]

[1]Under-Graduate, Dept. of CSE, RV College of Engineering, Bengaluru, Karnataka, India
[2]Associate Professor, Dept. of CSE, RV College of Engineering, Bengaluru, Karnataka, India

---

Abstract - *Over the years data generation and utilization have drastically increased for organizations of various sizes. The data is utilized for analysis and reporting to enhance the decision prospect and the business strategy. Data warehousing is one of the biggest schemes that has grown to suffice the requirement of storage of data and making it available and accessible to the end consumer such as analysts. This paper highlights the various phases of development and integration of data warehouse. Furthermore, this paper targets to discuss the suitable approaches at each phase of the design process.*

Key Words: Data Warehouse, Data Pipelining, Data Integration, Data Staging.

1. INTRODUCTION

Traditionally, the industries had been anchored onto using value-based databases. The value based databases are storage with defined schema requirements that is meant to store the only the latest information in the record. For instance, the phone number of an employee, only the latest active version is necessary. Thus value-based databases were incapable of carrying historical information and cannot be further used for business strategy. With the ascending need of bulk storage of data and accessibility for analysis and reporting, the concept of information distribution centers has evolved. The information distribution hubs accumulate data from various value-based sources without any exchanges. Thus, they separate analysis and operation load. The data can be form of association from multiple sources available to multiple audience.

The development of these kind of information hubs, and their integration with the existing systems have an elaborate value across each and every share holder. Such a typical system may have members as the management team, the end consumers, developers and other key stake holders. With the distinction of interests of these stake holders the system can have various perspectives during the design process. One can separate these views into four categories, Top - Down View, Data Source View, Data Warehouse View and Business Query View.

- **Top-Down View**: Demarcation of information valuable for present and posterior business needs.
- **Data Source View**: Raw data being accumulated and maintained at the source, This data can be visualized at various phases from individual tables to integrated tables.
- **Data Warehouse View**: It stores the multidimensional form of the data, where the native data is supported with meta-data that has been calculated prior to storage, giving information about native data.
- **Business Query View**: Value prospect from the end-consumer's perspective.

There are dedicated teams for variety of data utilization such as business analysts, business process analysts, risk analysts and system analysts. These teams analyze the data under multiple attributes such as the business prospects that can be delivered to the end users, the resources invested for the development, deployment and maintenance of such systems and expecting returns against the same, and finally the challenges involved, the expectancy of faults in the model, reiterating to strengthen and performance deductions. These kind of reports are accompanied by overcoming of inefficiencies like lost opportunities, rework, and undermining the potential benefits.

Data warehousing is primarily attributed to certain applications, Processing of information, Analysis and Data Mining. Processing of information deals with basic statistical query and analysis of the data available in the warehouse. Analytical processing deals with Online Analytical Processing (OLAP) queries over summarized or detailed historical data, thus providing multidimensional analysis. And data mining focuses on discovering hidden patterns by classification and prediction algorithms, visualized graphically. The following section discusses the various phases and approaches that can be incorporated in the designing of data warehouses.

2. LITERATURE REVIEW

2.1 Components of Data Warehousing framework

The entire system can be broadly divided into three segments as, User Interface Layer or the VIEW, Architecture Layer (MODEL) and Pipeline Layer (CONTROLLER). Thus, structured upon Model - View - Controller architectural design [3].

- **User Interface Layer**: Portal available to the end user.
- **Structure Layer**: It carries the structure and schema of the warehouse.
- **Pipeline Layer**: It contains the middle-ware that governs the Structure and Interface layer. Here the pipe-lining process of extract, transform and load is executed.

2.2 Phases of Development of Data Warehousing framework

The development of the framework is assorted into five phases: Analysis, Requirements, Conceptual design, Logical Design and Physical Design [1]. As highlighted in the following table (Table - 1).

Table -1: Five phases of Development Process of the Data warehouse framework.

Phases	Function	Output
Analysis	Information of systems over whom warehouse is developed	Database schemes capturing the system behaviors
Requirements	Gathering user requirements	Specifications for the warehouse
Conceptual Design	Visualizing Schema, system specifications	Conceptual schema like ERD/UML
Logical Design	Generating a detailed logical model with respect to involved modules	Logical Schema
Physical Design	Implementation with respect to logical schema	Physical Schema

2.3 Approaching the development and integration

The development of the system can be approached using one of the following methodologies, Top-Down as described Inmon, Bottom-Up as elaborated by Kimball and Data-Vault approach as elicited by Lintsedt [2,8].

2.3.1 Top-Down Approach

1. Data is accumulated from different sources into a third average shape.
2. This is suitable for scenarios where end-client details are not depicted but the source is extremely determined.
3. This is not beneficial for organizations where data is susceptible to be replaced in its entirety in every 5-10 years along with its context.
4. The data is staged into segregated data marts and analysis and reporting is done for each mart individually.
5. Data marts aid in strict segregation of data as per context, thus facilitating clients to gather context-specific data.
6. Inmon has divided the complete pipeline into 4 levels: Operational (Source), Atomic (Pipeline), Departmental (Data marts) and Individual (Analysis, Reporting and Mining).

2.3.2 Bottom-Up Approach

1. Data is accumulated from various data sources and later stored in to connected Data stores
2. The Data stores are connected either in Star or Snowflake Schema. The connections are based on the facts and dimensions of the multidimensional nature of the captured data. With the fact tables (FT) making the central node of division and Dimension tables (DT) making the terminal nodes.
3. This is suitable for scenarios where source specifications are not deterministic, while the client-side requirements are thoroughly laid out.
4. This enables high request execution.
5. But the challenge remains that the client requests keep changing rapidly.

2.3.3 Data-Vault Approach

1. Lintsedt has suggested a hybrid model that keeps the best of both, customary database and information distribution centers.
2. Every record in the table should be supported by the time-stamp and identification of the source out of the multiple sources, thus enabling users to trace the record back to source with temporal context.
3. Supports and stores long term historical data unlike value-based databases.
4. Offers resilience to change in customer needs.
5. Offers sustenance to different sources.

Figure-1 demonstrates Inmon's and Kimball's approach in parallel with all the phases of a data warehouse framework [2,8]. The four phases of Data warehousing pipeline: Operational, Atomic, Departmental and Data Access. Operational level marks the Source, Atomic signifies the

Extract-Transform-Load Pipeline, Departmental signifies the Data marts and Data Access marks Analysis, Reporting and Data Mining. As Inmon suggested, in the Departmental Level, the system should have isolated Data Marts and thus each independently responding to data access phase. While Kimball suggested to have Fact tables (FT) and Dimension Tables(DT) in Star/Snowflake Schema.

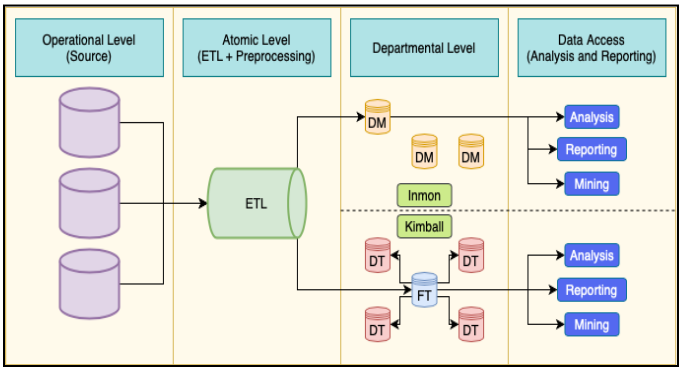
Operational Level (Source)
Atomic Level (ETL + Preprocessing)
Departmental Level
Data Access (Analysis and Reporting)
ETL
DM
DM
DM
Inmon
Kimball
DT
DT
FT
DT
DT
Analysis
Reporting
Mining
Analysis
Reporting
Mining

Fig -1: Phases of Data warehousing pipeline: Operational, Atomic, Departmental and Data Access.

2.4 Design Approaches

As mentioned in the Section 2.2 (Phases of Development), after the operational systems are analyzed and the user requirements have been gathered, it is required to design the visualizable conceptual model of the system. Over the years several methodologies have been employed as per convenience[1], some of them have been listed below:

- Entity Relationship Diagram (ERD) modeling the scheme of the available source endpoints
- Unified Modeling Language (UML) such as Class Diagram or Structure chart signifying the various structural elements. As Juan Trujilio describes four contexts of UML designs, Multidimensional, Mapping, ETL, and Deployment.
- Dimensional Fact Modeling, described by Stefano Rizzi, assembling facts and dimensions.
- Object Oriented UML, dividing the pipeline into two levels, Requirements Level and Design Level. The former deals with elicitation of user requirements and analysis of source, while the latter describes the major classes gathered for the data in the form of class diagrams, either in Star schema or Snowflake schema.

2.5 Data Pipelining Approaches

In this section the pipeline process is described which is governed by the principle capture, transform and flow (CTF). The pipeline is modeled structure that connects and transfers the data from the source to warehouse, making it accessible to end users [5,6,7].

2.5.1 Extract - Transform - Load (ETL)

The data is extracted from the source, then transformed into reduced form to suffice the analytical requirements of business intelligence (BI) and finally loaded into the warehouses where BI tools can further analyze or query the manipulated data and utilize for reporting [5,7]. In ETL approach, the raw data can not be query from the warehouse end because the data is not accessible in the Extract and transform phase to the end consumer. This approach requires special subject matter experts who can design queries based on available data that can provide details regarding the business aspects.

2.5.2 Extract - Load - Transform (ELT)

The data is extracted from the source, loaded onto an intermediate remote temporary bulk storage known as data lakes, and finally it's made available in the warehouse also where it can be further utilized through BI tools [6]. The data lake stage is also known as staging. In this approach the end-consumer can query the raw data (or cleaned raw data) from the data lakes directly and need not be constricted to just the warehouse where processed data is available for analytics. In the recent years this approach has gained popularity because of the onset of remote bulk storage over cloud, thus decreasing the cost of storage. Traditionally, cost of storage was exaggerated, thus it was required to process data and store in reduced forms only. Therefore, first bulk storage is prioritized to facilitate raw data querying.

2.6 Data Integration Approaches

Data integration means the replenishment of data or the updation of the data in target site (warehouse) when there is an update at source site. Some of the approaches are at the operational stage and some are based on the warehouse side [3].

- **Push**: As when changes occur at source, same is pushed into the warehouse subsequently.
- **Pull Data**: Warehouse pulls the changes from the source, when required.
- **Event based**: Data is synced when a particular event occurs at the source, such as COMMIT or ROLLBACK. Or if the type of column is changed to a particular data type.
- **Polling**: Warehouse keeps checking for changes at the source continuously and syncs new updates if found any at the source. This exhausts the resources at the warehouse since always busy in checking the source even when there are no potential updates.
- **Near Real time**: Syncing the data from the source at scheduled time, thus periodic in nature and making latest data available to the client-side.

2.7 Data Staging Approaches

There are certain methodologies to incorporate the continuous changes to the data in a real time environment [3].

- **Data trickle feed**: The real-time data can be stored either along with the historical data in the same fact tables or they can be stored separately, with real-time partition. If same fact table then no special modeling is required. But in case of partition, the end user is shown a single logical table with both data. It is difficult to scale up in such scenarios.
- **Trickle & flip flop**: A staging table is maintained which carries the copy of current data. Periodically, this data is transferred to existing historical tables in the warehouse. The loading of the old data into warehouse from staging table and current data into staging is known as flip time. This methodology is subject to flip time and the data load being changed.
- **External Real time Data Cache (RTDC)**: A dedicated database is used as cache, where the data base is structurally is similar to the warehouse, thus no extra modelling is required. The function of the cache is to make those real-time data segments available to the user along with historical data from warehouse. If the dedicated cache is external then lookup tables are used to to keep track of data segments available in the cache.

2.8 OLAP queries in real time

Online Analytical Processing (OLAP) queries are executed over historical data along with the real time data for multidimensional analysis of the data [3]. The data is said to be multidimensional because it is analyzed based on two components facts and dimensions [1]. Facts contain measure of business process and dimensions are specific views of the facts. Similar to how the schema of tables define the various fields or attributes that are taken in consideration and tuples or records carry collection of specific instances of each field. To manage the queries in real-time certain procedures can be employed.

- **Near Real time**: Using Trickle-Flop method with longer flip time. Thus updating the staging table after longer duration.
- **Restrict complex time queries**: Users are not allowed to directly make complex queries on the real time data. A separate partition can be maintained with a snapshot of less frequently changing real time data that can be used for complex analysis.
- **External RTDC**: Maintaining external database as cache that caters to all OLAP queries with respect to real-time data, which is being stored in the cache.

2.9 Multidimensional Data mining

This section discusses integration of data mining strategies with OLAP analysis over multidimensional data. Data Mining deals with exploratory approaches to deduce hidden patterns using classification and prediction algorithms to generate deeper analytical insights in context to business intelligence [4].

- **High quality data in warehouse**: For efficient data mining so as to lead tangible results, a huge investment should be made towards cleansing and preprocessing of the data. This preprocessing can serve OLAP and Data mining both. The data mining can also be used as an integral step in the preprocessing and integration phases.
- **Info processing architecture**: Dedicated infrastructures are required that can manage multiple heterogeneous sources and integrate tools like that of OLAP. SO it is better to use existing infrastructures in form of OLAP tools instead of developing from point zero.
- **OLAP based exploration of data**: Since data mining is exploratory in nature, analysts may need to move across the data, or focus on a particular segment, or analyse different segments with varied degree to finally generate tangible results. Thus the infrastructure should also support traversing across the data, and slicing or segmentation of the data.
- **Online selection of data mining functions**: The users may not know the various features that can be extracted from the data, thus OLAP infrastructures can provide options to choose automated data mining functionalities to extract and visualize deductions.

3. CONCLUSIONS

In this paper, the various phases of the design and development of a data warehousing framework is discussed. The analysis extends from components of such a framework that defines the structure of the system. The various approaches towards the development are discussed, top-down for cases with defined source specifications and bottom-up for scenarios with client side requirements and data-vault carrying best of both.

The various design methodologies and visualization approaches have been discussed that would aid in the development of the conceptual model of the system. Furthermore, the pipeline strategies are discussed that integrate data from the source and stage them for end usage. The paper highlights optimized staging strategies to capture the continuous changes in real-time environment and further use them for analysis under Online Analytical Processing (OLAP). The future scope of this paper discusses integration of the OLAP infrastructures along with data mining to utilize the prospects of Data Warehousing.

REFERENCES

[1] Rajni Jindal, Shweta Taneja, 'Comparative Study of Data Warehouse Design Approaches: A Survey', International Journal of Database Management Systems (IJDMS) Vol.4, No.1, February 2012.

[2] Panacea Makele, Srinath Doss, 'A Survey on Data Warehouse Approaches for Higher Education Institution', International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE) Volume 1, Issue 11, May 2018.

[3] Vaishali Wangikar, 'Study of Different Approaches for Real Time Data Warehouse Environment', Conference: VIT Pune: National Conference on Modeling, Optimization and Control NCMOC - 2015.

[4] Dishek Mankad, Preyash Dholakia, 'The Study on Data Warehouse Design and Usage', International Journal of Scientific and Research Publications, Volume 3, Issue 3, March 2013.

[5] Gustavo V. Machado, Ítalo Cunha, Adriano C. M. Pereira, Leonardo B. Oliveira, 'DOD-ETL: distributed on-demand ETL for near real-time business intelligence', Journal of Internet Services and Applications volume 10, Article number: 21, 2019.

[6] Florian Waa, Tobias Freudenreich, Robert Wrembel, Maik Thiele, Christian Koncilia, Pedro Furtado, 'On-Demand ELT Architecture for Right-Time BI: Extending the Vision', International Journal of Data Warehousing and Mining 9(2):21-38 · April 2013.

[7] Panos Vassiliadis, 'A Survey of Extract-Transform-Load Technology.,' International Journal of Data Warehousing and Mining 5:1-27, July 2009.

[8] Yessad. L, Lahiod. A, 'Comparative Study of Data Warehouses Modeling Approaches: Inmon, Kimball and Data Vault', Interanational Conference on System Realiability and Science, 2016.

BIOGRAPHIES

Auchinto Chatterjee, Under-Graduate B.E. (2016-2020) in Computer Science and Engineering, RV College of Engineering, Bengaluru, Karnataka, India.

Dr. Sharvani GS, Associate Professor, M.Tech - Computer Networks and Engineering, Dept. Of Computer Science and Engineering, RV College of Engineering, Bengaluru, Karnataka, India.