*Impact Factor (SJIF): 3.632*

**International Journal of Advance Research in Engineering, Science & Technology**

*e-ISSN: 2393-9877, p-ISSN: 2394-2444*

***(Special Issue for ITECE 2016)***

# Detail Study of Dynamic Data Warehousing

**Prof.Chirag Vala[1], Prof.Krunal Vasani[2]**

*[1]Computer Science & Engineering, SLTIET*
*[2]Computer Science & Engineering, SLTIET*

***Abstract*** — ***Dynamic warehousing is not a product, tool or simple one-off solution. It is an approach that enables you to hand over more dynamic business seeing into by getting mixed together, making great change, getting together and getting at details seeing into from structured and unstructured information. The result is a framework for delivering right-time, contextual information for both strategic planning and operational purposes. Enabling dynamic warehousing requires a set of services that extends old and wise knowledge for computers storing and going to person in authority to support the increasing number of businessprocesses and applications having need of given to getting details powers, and to house the demands for more force full business power.***

**Keywords: DataWarehosing, Capture Transform and Flow, Business Model**

## I. INTRODUCTION

The Dynamic Data Warehouse (DDW) represents more than the evolution of the EDW approach. To advance from traditional data warehousing to dynamic data warehousing requires a low-latency environment, not just a large atomic data warehouse. Most of the capabilities required to get from vanilla data warehousing to near real-time, on-demand data warehousing are implemented by the proprietary data warehouse vendors as workarounds to a legacy database and operating system.

The proprietary data warehouse vendors make it sound like loading inconsistent, diverse data into its atomic data store automatically rationalizes it and renders it consistent. Not so. What is required is a dynamic ecosystem that makes data warehousing as simple as front, middle and back end. The time horizon extends upstream and downstream to encompass the atomic data warehouse in an architecture designed to reduce latency at key points in the information supply chain. Key chokepoints that create delay are in-bound processing, data rationalization, closing the loop between transactional and business intelligence systems and information delivery.

First section of this paper describes the Overview of the Dynamic Data Warehouse. Second section of this paper explains about structure of DDW. Third section of this paper explains about analysis of different techniques. And in last section we conclude with the importance of DDW.

## II. Platform required to implement DDW

The platform should be able to:

- Process transactions and analytical requests.
- Handle varying service level agreements (SLA).
- Scale easily as the number of applications grows.
- Analyze structured as well as unstructured data.
- Provide real-time analytics that can be embedded in business processes.
  Support advanced analytics such as data mining within the data warehouse.

Dynamic warehousing requires an extended infrastructure to:

- Implement changes to the business model without impacting usage.
- Monitor and analyze data sources for structure and content to ensure the best data is being accessed for each application.
- Provide tools that enable business users and IT staff to collaborate on data requirements and definitions.

- Deliver impact analysis and data lineage reports to coordinate changes and provide visibility to critical data flows.
- Deliver data that has been cleansed and harmonized to the warehouse regardless of volume or latency requirements.
- Synchronize master data for key business entities across operational systems.

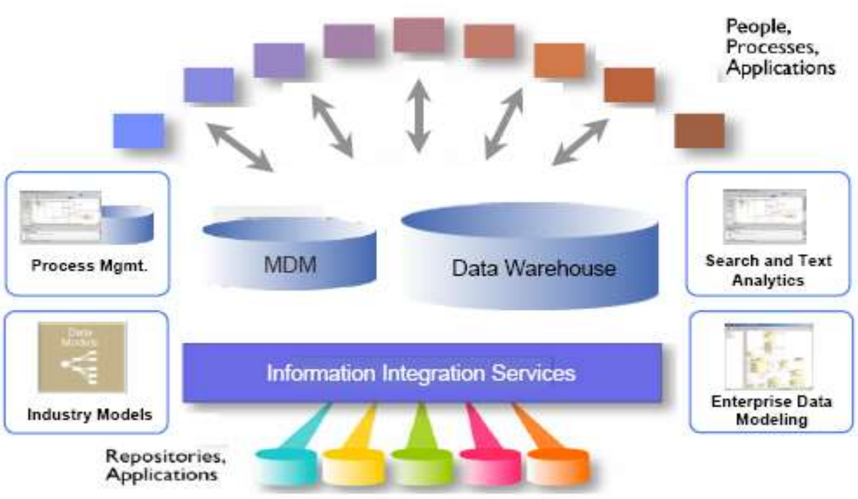

Figure - 1 : Architecture of Dynamic Data Warehouse

## III. COMPONENTS OF REAL TIME DATA WAREHOUSING

An intelligent warehousing solution and framework can commonly be divided into three fundamental tiers with data flows between them.

The three layers are:

### 1. Presentation Layer

The presentation layer manages the flow of information from the warehouse to the analyst, providing an interface that makes it easier for the analyst to view and work with the data.

### 2. Architecture Layer (Structure, Content/Meaning)

The architecture layer describes the structure of the data in the warehouse. An important component of the architecture layer is flexibility. The level of flexibility is measured in terms of how easy it is for the analyst to break out of the standard representation of information offered by the warehouse in order to do custom analysis. Custom analysis is where semantic thickness becomes important.

### 3. Middleware Layer (Interfaces and Replenishment)

The middleware layer is the glue that holds the data warehouse together. It integrates the data warehouse with production and operational systems. Data needed for warehouse applications often must be copied to and from computers of different types in different locations. Warehousing often implies transformational data integration. Production data needs to be secured and is frequently not in the format needed for warehousing. Real-time integration and replenishment tools that help businesses deal with the data management issues of implementing a data warehouse can add real value.

A real-time integration and replenishment solution or a **Capture, Transform and Flow (CTF)** tool can contribute to the simplicity and efficiency of a real-time data warehouse.

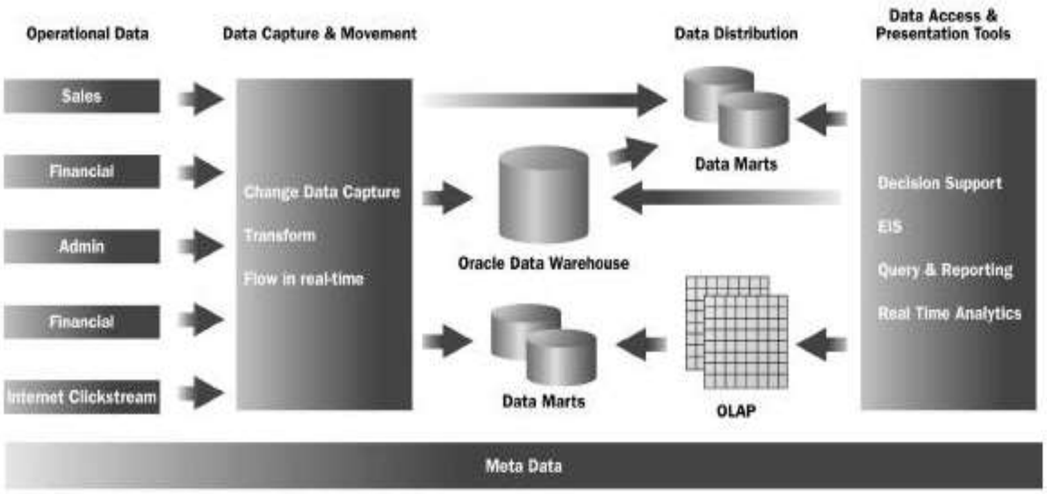

**Figure – 2 :** Typical data warehouse implementation utilizing

## IV. ISSUES IN DYNAMIC DATA WAREHOUSING

The issue of query contention(conflict) and scalability is the most difficult issue facing organizations deploying real-time data warehouse solutions. Data warehouses were separated from transactional systems in the first place because the type of complex, analytical queries run against warehouses don't "play well" with lots of simultaneous inserts, updates or deletes.

Real-time data warehousing and OLAP are possible using today's technology, but obstacles exist. For the determined team armed with the right knowledge and experience, it is possible to make real-time reporting, analysis and alerting systems work. The challenge is making the right tradeoffs along the way so the systems meet the needs of the user base while ensuring that they don't collapse under their own weight or cause existing production warehouses to malfunction.

The benefits of data warehousing in real-time are becoming clearer every day. With the right tools, designs, advice, approaches and, in some cases, tricks, real-time data warehousing is possible and will only become easier in the future. In any case, the time to begin planning and prototyping is now.

A data warehouse can only be considered real-time, or near real-time, when all or part of the data is updated, loaded or refreshed on an intra-day basis, without interrupting user access to the system.

However, most ETL tools, whether based on off-the-shelf or custom-coded products, operate in a batch mode. They assume that they have free reign to drop tables, re-load tables and conduct other major database operations without disturbing simultaneous end-user queries.

To work around this challenge, data that changes throughout the day can be loaded into a parallel set of tables, either through a batch process that runs every several minutes or through a continuous trickle feed. In a Teradata-based system, a batch load is best conducted using either the FastLoad or MultiLoad utility, and a trickle feed is best performed using TPump. Once the load interval, say five minutes, is up, the freshly loaded tables are simply swapped into production and the tables with the now-stale data are swapped out of production. This can be accomplished through the dynamic updating of views or by simple table renaming.

The downside of this type of n-minute cycle-loading process is that the data in the warehouse is not truly real-time. This type of system is more accurately described as near real-time. For many data warehousing applications, near real-time is good enough. Also, using a near real-time approach helps relieve some of the other challenges of real-time warehousing.

For applications where true real-time data is required, the best approach is to trickle-feed the changing data from the source system directly into the data warehouse.

With TPump, Teradata handles this type of real-time updating much better than many other database platforms, and this architecture can be used even in systems with large data volumes and high numbers of concurrent users.

## V. CONCLUSION

Dynamic warehousing is part of the next generation of technology that enables organizations to gain more business insight and deliver relevant information on demand. An effective solution can help financial institutions access and analyze existing data and ultimately boost business in a highly competitive market. Whereas traditional data warehouses can make it difficult to keep up with today's fast-paced banking environments, dynamic warehousing delivers immediate, integrated information. With this information at their fingertips, employees can take action and make timely decision.

## REFERENCES

[1] Mastering Master Data Management http://www.dmreview.com/article_sub.cfm?articleID=1060131.
[2] Desmond A. Martin. "Active Data Warehouse: Where Agile Retailers Win by Capitalizing on Time." Teradata. August 2006. p. 11.
[3] ftp://ftp.software.ibm.com/software/data/bi/dynamic-whitepaper.pdf.