ISSN No:-2456-2165

Data Analytics for Food Distribution Industry

Ananthan Subburaj
Technology Architect
New Jersey, United States of America

Abstract:- The Wholesale distribution and retail industries faces huge challenges in quickly identifying how the supply chain is performing, how to manage supply in real time, how to further improve the supply chain. The warehouse operations and transportation data gathering and analysis is the key to give insight into historic and current status of the business process and enable key decision making to improve the efficiency of the business. The objectives of this paper are to introduce effective and cost effective data analytics feature using the key Oracle database features to analyze the current and historic supply chain data to extract actionable information.

Keywords:- *Data Analytics; Supply chain analysis; Data warehouse Performance; Dustribution Industry.*

I. INTRODUCTION

Real time data capture and optimum data Analytics helps make confident decisions, increase operational efficiency and reduce costs. The food supply chain industries like wholesale distributors or retail chains generate huge volumes of data related to inventory, transportation and sales. This data helps the supply chain industry to evaluate the performance of the entire supply chain as well as zoom in to the details at item level to understand the inventory, sales and margin history. The insight into the current and historical performance of items helps to plan and predict demand and rank products based on profit margins. There are so many different database solutions, in memory data analytics products and cloud based database solutions available in the market to extract information out of the data collected from the business operations, but they all need additional technical resources in the respective product area and involves additional cost. The purpose of this paper is to address the key challenges in adopting the appropriate data analytics process to power growth and reduce cost.

II. KEY CHALLENGES TO ADOPT DATA ANALYTICS

There are few common challenges in implementing the data analytics to improve the operational efficiency and reduce cost.

- Real time data availability at centralized data warehouse system
- Data standardization across different systems and entities in the organization
- Tools and technology to process large volume of data
- Performance latency to extract usable information from the data
- As there are so many technologies emerging in the market for the data analytics, lack of skilled resource to adopt the technology
- Most important point is the cost. How expensive is the problem that the system is intended to solve and how expensive is the technology and infrastructure required to implement data analytics solution

III. POWER GROWTH WITH DATA ANALYTICS

Wholesale distribution companies can extract actionable information with proper data analysis to monitor the warehouse operations, sales and inventory to predict and respond to changes in demand and supply. They can significantly benefit by adopting technics with the existing oracle database systems to advanced analytics:

- Analyze operational data to understand what retailers have purchased in the past to predict future demand
- Visualize sales trends based on retail chain, warehouse, product level details
- Quickly identify the gap in estimate vs actual sales and demand
- Predict optimal stock levels for perishables items to prevent spoilage and out-of-stocks to meet supply
- Predict if the warehouse will achieve target shipments for the day based on the real time operational progress data
- Identify space utilization issues by looking at the inventory and shipment data
- Track sales trends with target growth projections and identify gap

IV. DATA COLLECTION

The wholesale distribution and larger retail chain have multiple warehouses in different locations to service goods difference area of retail stores and customers, each warehouse may have local database and applications to manage the individual warehouse operations. The data from all the warehouses need to collect at a central data warehouse infrastructure. The data collection process can be done via any of the ETL tools or the oracle Golden gate technology to achieve real time data capture. There are three main categories of data captured from all the warehouse management systems Master data, Transaction Data and Business data, once the data is loaded into central data warehouse database, the data needs to be standardized from various applications. The data standardization and grouping

ISSN No:-2456-2165

has to be done in the data warehouse before preparing the data for the analysis.

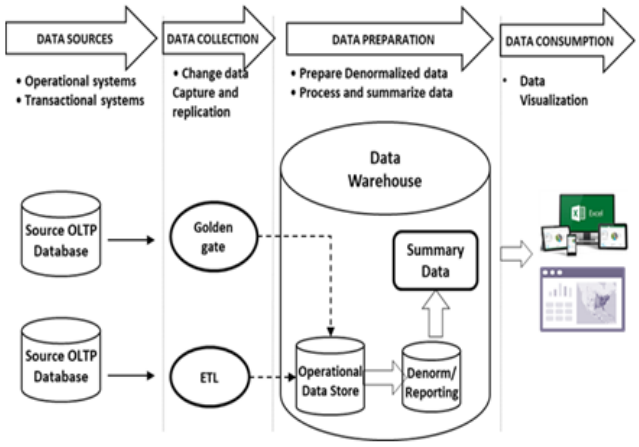

Fig. 1. Data Analytics Topology

V. DATA ANALYTICS

Data collected from warehouse operational system databases are loaded into data warehouse database on periodic schedule. For real time data analytics the source warehouse database system can be implemented with oracle Golden Gage for real time change data capture and replicate to the central data warehouse database. The data warehouse database generally of size from hundreds of gigabytes to a few terabytes. The analytics data tables can be classified in to master data tables and transaction tables. The Master data or dimension tables represent the business entities of an organization, represented as hierarchical information such as sales date, shipping date, warehouse name, store name, locations, and product category. The dimension tables usually small in data volume and change slowly over time based on the business changes. The dimension tables are used to aggregate and analyze data at appropriate business hierarchy level. The transaction tables are fact tables which stores all the business transaction data. The fact table stores majority of data and generally very large in size that are updated periodically or in real time with data streaming from one or more operational databases.

The data analytics involves aggregating and analyzing all the transaction data across the multiple dimensions with different Hierarchies. The hierarchies describe the business model and organization structure for common grouping of transaction data to understand the business. As the fact table contains huge volume of data ranging from gigabytes to terabytes aggregating and processing requires huge system capacity and performance optimization. The performance optimization can be done by precalculating summaries using the oracle schema object Materialized views. The materialized views are special types of database objects to improve query performance by precalculating performance intensive aggregations and joins prior to query execution and storing the results in a table.

VI. MATERIALIZED VIEWS FOR DATA ANALYTICS

The Materialized views are database objects referred as summary table which stores the summarized data from the base tables. The Materialized views are also used to precompute joins on multiple dimension and fact tables with or without aggregation functions. The precomputed data is stored as table to eliminate runtime overhead associated with the performance intensive aggregation and join operations for a large data analytics queries.

The end user queries or the data visualization tools can access the huge transaction data with the base tables or regular views defined on the base tables, the query rewrite feature in the Oracle server automatically rewrites the SQL query to use the summary tables to provide query results. The Materialized views in the oracle database are transparent to the end user and the database application. Although the materialized views are generally accessed through query rewrite mechanism, an end user or database application can access the materialized views. Before using the materialized views for direct query access consider the impact of stale data, when the data is changed in the base table depending on the type of materialized view the data may become stale. For data sensitive applications it advised to avoid reading data directly from materialized views and also set up the appropriate setting for the database parameter query_rewrite_integrity. The query rewrite integrity database parameter controls the rewrite process based on the status of the materialized view.

The materialized view data freshness is estimated based on the base tables directly used by the materialized view. When a materialized view is created based on another materialized view, the freshness of the child materialized view is estimated based on the changes in the parent materialized view and does not depend on the tables referenced by the parent materialized view.

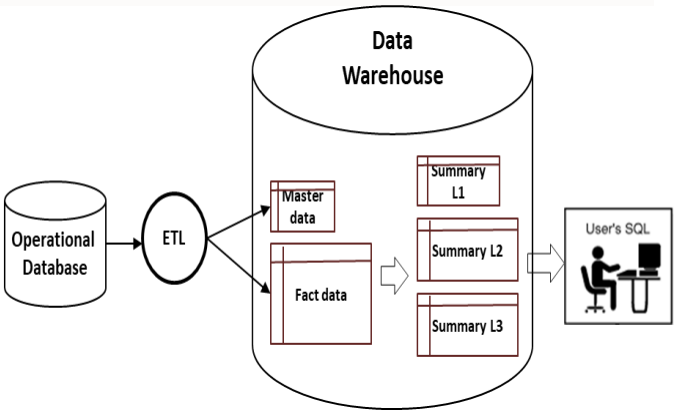

Fig. 2. Data warehouse Design

ISSN No:-2456-2165

VII. TABLE PARTITION FOR DATA ANALYTICS

As the fact table in data warehouse database store huge volume of data, partitioning the fact tables is useful for performance optimization as well as to improve the manageability. The table partition is extremely useful to simplify the administration tasks, effectively use appropriate storage options like compression level or using different type of storage for hot and cold data. Also using local indexes helps with data drill down, improves the index rebuild process, and helps with the materialized view fast refreshing with partition change tracking (PCT) refresh.

VIII. REAL-TIME MATERIALIZED VIEWS

The materialized views becomes stale when the data is changed in the table referenced by the materialized view and the oracle optimizer does not rewrite the query to read from materialized view. The real-time materialized views are created with on query compute option to provide fresh data to user queries even when the materialized view is in stale status using on query computation technique.

When a query is executed the optimizer checks for the status of the materialized view, if the materialized view is fresh then the query is rewritten to provide data from the materialized view as is. If the real time materialized view status is stale then the optimizer computes the fresh data using the materialized view log and provides the result set. The query rewrite for the real time materialized view combines the data from the materialized view and the delta change data from the materialized view log to provide the correct query result. The process of on query compute does not update the materialized view with this fresh computed data, it is only used to provide data for the current query.

IX. HIERACHICAL MATERILIZED VIEW DESIGN PRINCIPLES

The Hierarchical materialized view design helps to analyze huge fact tables with very fast high level summary access and provides ability to drilldown with optimum performance. The level 1 summary materialized view should be created to have only few hundred thousand records out of the base fact table with billions of rows, this can be yearly, monthly and weekly where house level summary data. This helps to executive level dash board to get within few seconds out of billions of records on fact table. The seconds level of materialized view can have addition medium level drill down to product category or retail chain level of details, the summary materialized view can have up to few million records to provide detailed dashboard reports relatively faster. Then third level or materialized view with additional dimensions added to the aggregation can have lot more millions of records. The depth of the hierarchy and can vary based on the volume of data in the fact table and the number of summary records produced at the given aggregation dimension level taking in to account of the visualization performance.

The final drill down at the granular level can be handled by partitioning the table with data range and warehouse or location as sub partition. This gives the ability the requirement to analyze the granular level details for a given store and item level details when the materialized views becomes unusable.

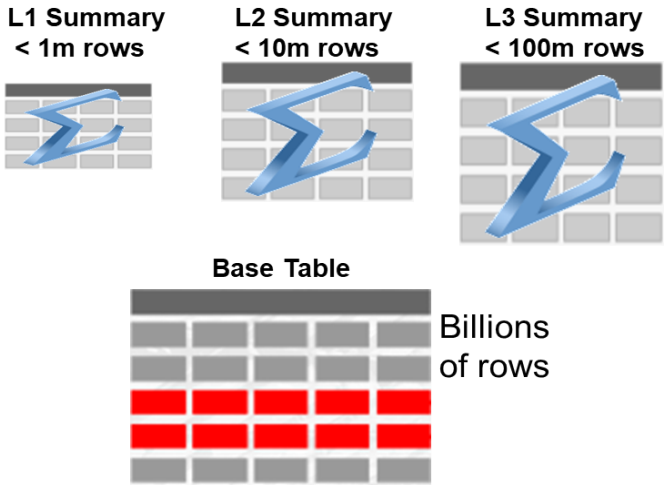

Fig. 3. Heirarchial Mview Design

The data analytics design with right combination of hierarchical materialized views, real time materialized view refresh and table partition to meet the analytics requirement provides best performance with oracle data warehouse environment.

X. CONCLUSIONS

The transaction data generated in wholesale distribution industries have great useful information on the procurement, inventory, transportation and sales. Analysis of this massive operational data can help to understand supply chain efficiency, warehouse space utilization, customer behavior and so on. This helps wholesale distribution industry improve operational efficiency and reduce cost. The growing need to analyze customer behavior and demand forecasting is driven by globalization and increasing market competitions as well as the surge in supply chain digitization practices. In this study, I introduced the new approach to the data analytics with advanced oracle database features in the proper combination. The large fact table partition with right date based dimension provides the ability to drill down to the details for the specific date, the level one summary with very high level summary materialized view provides the executive dashboard level of key overview of the business performance. The subsequent level of analytics and dimension requirement will drive the design for multi-level summary materialized views. The key considerations to implement any analytics system is the cost and the skill set required to implement and maintain the system. This paper provided insight into the techniques using the exiting oracle warehouse system and the existing technical skill set to achieve effective data analysis.

REFERENCES

[1]. https://docs.oracle.com/en/database/oracle/oracle-database/19/dwhsg/index.html
[2]. https://docs.oracle.com/en/database/oracle/oracle-database/19/vldbg/index.html
[3]. A Comparative Study Analysis of Materialized View for Selection Cost T. Nalini, A. Kumaravel and K. Rangarajan
[4]. Materialized Views In Oracle by Randall G. Bello, Karl Dias, Alan Downing, James Feenan, Jim Finnerty, William
[5]. D. Norcott, Harry Sun, Andrew Witkowski, Mohamed Ziauddin
[6]. https://www.oracle.com/applications/supply-chain-management/what-is-supply-chain-management-system.html