Data warehouse

10.1 Data Warehouse Overview

In computing, a **data warehouse** (**DW**, **DWH**), or an **enterprise data warehouse** (**EDW**), is a system used for reporting and data analysis. Integrating data from one or more disparate sources creates a central repository of data, a data warehouse (DW). Data warehouses store current and historical data and are used for creating trending reports for senior management reporting such as annual and quarterly comparisons.

The data stored in the warehouse is uploaded from the operational systems (such as marketing, sales, etc., shown in the figure to the right). The data may pass through an operational data store for additional operations before it is used in the DW for reporting.

10.2 Types of systems

Data mart

A data mart is a simple form of a data warehouse that is focused on a single subject (or functional area), such as sales, finance or marketing. Data marts are often built and controlled by a single department within an organization. Given their single-subject focus, data marts usually draw data from only a few sources. The sources could be internal operational systems, a central data warehouse, or external data.[1]

Online analytical processing (OLAP)

Is characterized by a relatively low volume of transactions. Queries are often very complex and involve aggregations. For OLAP systems, response time is an effectiveness measure. OLAP applications are widely used by Data Mining techniques. OLAP databases store aggregated, historical data in multi-dimensional schemas (usually star schemas). OLAP systems typically have data latency of a few hours, as opposed to data marts, where latency is expected to be closer to one day.

Online Transaction Processing (OLTP)

Is characterized by a large number of short on-line transactions (INSERT, UPDATE, DELETE). OLTP systems emphasize very fast query processing and maintaining data integrity in multi-access environments. For OLTP systems, effectiveness is measured by the number of transactions per second.

OLTP databases contain detailed and current data. The schema used to store transactional databases is the entity model (usually 3NF).[2]

Predictive analysis

Predictive analysis is about finding and quantifying hidden patterns in the data using complex mathematical models that can be used to predict future outcomes. Predictive analysis is different from OLAP in that OLAP focuses on historical data analysis and is reactive in nature, while predictive analysis focuses on the future. These systems are also used for CRM (Customer Relationship Management).[3]

10.3 Software tools

The typical extract-transform-load (ETL)-based data warehouse uses staging, data integration, and access layers to house its key functions. The staging layer or staging database stores raw data extracted from each of the disparate source data systems. The integration layer integrates the disparate data sets by transforming the data from the staging layer often storing this transformed data in an operational data store (ODS) database. The integrated data are then moved to yet another database, often called the data warehouse database, where the data is arranged into hierarchical groups often called dimensions and into facts and aggregate facts. The combination of facts and dimensions is sometimes called a star schema. The access layer helps users retrieve data.[4]

This definition of the data warehouse focuses on data storage. The main source of the data is cleaned, transformed, cataloged and made available for use by managers and other business professionals for data mining, online analytical processing, market research and decision support.[5] However, the means to retrieve and analyze data, to extract, transform and load data, and to manage the data dictionary are also considered essential components of a data warehousing system. Many references to data warehousing use this broader context. Thus, an expanded definition for data warehousing includes business intelligence tools, tools to extract, transform and load data into the repository, and tools to manage and retrieve metadata.

Benefits

A data warehouse maintains a copy of information from the source transaction systems. This architectural complexity provides the opportunity to :

- Congregate data from multiple sources into a single database so a single query engine can be used to present data.

- Mitigate the problem of database isolation level lock contention in transaction processing systems caused by attempts to run large, long running, analysis queries in transaction processing databases.
- Maintain data history, even if the source transaction systems do not.
- Integrate data from multiple source systems, enabling a central view across the enterprise. This benefit is always valuable, but particularly so when the organization has grown by merger.
- Improve data quality, by providing consistent codes and descriptions, flagging or even fixing bad data.
- Present the organization's information consistently.
- Provide a single common data model for all data of interest regardless of the data's source.
- Restructure the data so that it makes sense to the business users.
- Restructure the data so that it delivers excellent query performance, even for complex analytic queries, without impacting the operational systems.
- Add value to operational business applications, notably customer relationship management (CRM) systems.
- Making decision–support queries easier to write.

Generic data warehouse environment

The environment for data warehouses and marts includes the following:

- Source systems that provide data to the warehouse or mart;
- Data integration technology and processes that are needed to prepare the data for use;
- Different architectures for storing data in an organization's data warehouse or data marts;
- Different tools and applications for the variety of users;
- Metadata, data quality, and governance processes must be in place to ensure that the warehouse or mart meets its purposes.

In regards to source systems listed above, Rainer states, “A common source for the data in data warehouses is the company’s operational databases, which can be relational databases”.

Regarding data integration, Rainer states, “It is necessary to extract data from source systems, transform them, and load them into a data mart or warehouse”. Rainer discusses storing data in an organization’s data warehouse or data marts.”.

Metadata are data about data. “IT personnel need information about data sources; database, table, and column names; refresh schedules; and data usage measures“.

Today, the most successful companies are those that can respond quickly and flexibly to market changes and opportunities. A key to this response is the effective and efficient use of data and information by analysts and managers. A “data warehouse” is a repository of historical data that are organized by subject to support decision makers in the organization. Once data are stored in a data mart or warehouse, they can be accessed.

10.4 History

The concept of data warehousing dates back to the late 1980s when IBM researchers Barry Devlin and Paul Murphy developed the "business data warehouse". In essence, the data warehousing concept was intended to provide an architectural model for the flow of data from operational systems to decision support environments. The concept attempted to address the various problems associated with this flow, mainly the high costs associated with it. In the absence of a data warehousing architecture, an enormous amount of redundancy was required to support multiple decision support environments. In larger corporations it was typical for multiple decision support environments to operate independently. Though each environment served different users, they often required much of the same stored data. The process of gathering, cleaning and integrating data from various sources, usually from long-term existing operational systems (usually referred to as legacy systems), was typically in part replicated for each environment. Moreover, the operational systems were frequently reexamined as new decision support requirements emerged. Often new requirements necessitated gathering, cleaning and integrating new data from "data marts" that were tailored for ready access by users.

Information storage

Facts

A fact is a value or measurement, which represents a fact about the managed entity or system. Facts as reported by the reporting entity are said to be at raw level.

E.g. if a BTS received 1,000 requests for traffic channel allocation, it allocates for 820 and rejects the remaining then it would report 3 **facts** or measurements to a management system:

- tch_req_total = 1000
- tch_req_success = 820
- tch_req_fail = 180

Facts at raw level are further aggregated to higher levels in various dimensions to extract more service or business-relevant information out of it. These are called aggregates or summaries or aggregated facts.

Dimensional vs. normalized approach for storage of data

There are three or more leading approaches to storing data in a data warehouse — the most important approaches are the dimensional approach and the normalized approach.

The dimensional approach refers to Ralph Kimball's approach in which it is stated that the data warehouse should be modeled using a Dimensional Model/star schema. The normalized approach, also called the 3NF model (Third Normal Form) refers to Bill Inmon's approach in which it is stated that the data warehouse should be modeled using an E-R model/normalized model.

In a dimensional approach, transaction data are partitioned into "facts", which are generally numeric transaction data, and "dimensions", which are the reference information that gives context to the facts. For example, a sales transaction can be broken up into facts such as the number of products ordered and the price paid for the products, and into dimensions such as order date, customer name, product number, order ship-to and bill-to locations, and salesperson responsible for receiving the order.

A key advantage of a dimensional approach is that the data warehouse is easier for the user to understand and to use. Also, the retrieval of data from the data warehouse tends to operate very quickly. Dimensional structures are easy to understand for business users, because the structure is divided into measurements/facts and context/dimensions. Facts are related to the organization's business processes and operational system whereas the dimensions surrounding them contain context about the measurement (Kimball, Ralph 2008).

The main disadvantages of the dimensional approach are the following:

1. In order to maintain the integrity of facts and dimensions, loading the data warehouse with data from different operational systems is complicated.

2. It is difficult to modify the data warehouse structure if the organization adopting the dimensional approach changes the way in which it does business.

In the normalized approach, the data in the data warehouse are stored following, to a degree, database normalization rules. Tables are grouped together by *subject areas* that reflect general data categories (e.g., data on customers, products, finance, etc.). The normalized structure divides data into entities, which creates several tables in a relational database. When applied in large enterprises the result is dozens of tables that are linked together by a web of joins. Furthermore, each of the created entities is converted into separate physical tables when the database is implemented (Kimball, Ralph 2008). The main advantage of this approach is that it is straightforward to add information into the database. Some disadvantages of this approach are that, because of the number of tables involved, it can be difficult for users to join data from different sources into meaningful information and to access the information without a precise understanding of the sources of data and of the data structure of the data warehouse.

Both normalized and dimensional models can be represented in entity-relationship diagrams as both contain joined relational tables. The difference between the two models is the degree of normalization (also known as Normal Forms). These approaches are not mutually exclusive, and there are other approaches. Dimensional approaches can involve normalizing data to a degree (Kimball, Ralph 2008).

In *Information-Driven Business*, Robert Hillard proposes an approach to comparing the two approaches based on the information needs of the business problem. The technique shows that normalized models hold far more information than their dimensional equivalents (even when the same fields are used in both models) but this extra information comes at the cost of usability. The technique measures information quantity in terms of information entropy and usability in terms of the Small Worlds data transformation measure.

Top-down versus bottom-up design methodologies

Bottom-up design

Ralph Kimball created an approach to data warehouse design known as *bottom-up*. In the *bottom-up* approach, data marts are first created to provide reporting and analytical capabilities for specific business processes.

These data marts can eventually be integrated to create a comprehensive data warehouse. The data warehouse bus architecture is primarily an implementation of "the bus", a collection of conformed dimensions and conformed facts, which are dimensions that are shared (in a specific way) between facts in two or more data marts.

Top-down design

Bill Inmon has defined a data warehouse as a centralized repository for the entire enterprise.[17] The *top-down* approach is designed using a normalized enterprise data model. "Atomic" data, that is, data at the lowest level of detail, are stored in the data warehouse. Dimensional data marts containing data needed for specific business processes or specific departments are created from the data warehouse. In the Inmon vision, the data warehouse is at the center of the "Corporate Information Factory" (CIF), which provides a logical framework for delivering business intelligence (BI) and business management capabilities. Gartner released a research note confirming Inmon's definition in 2005[18] with additional clarity. They also added one attribute.

Hybrid design

Data warehouse (DW) solutions often resemble the hub and spokes architecture. Legacy systems feeding the DW/BI solution often include customer relationship management (CRM) and enterprise resource planning solutions (ERP), generating large amounts of data. To consolidate these various data models, and facilitate the extract transform load (ETL) process, DW solutions often make use of an operational data store (ODS). The information from the ODS is then parsed into the actual DW. To reduce data redundancy, larger systems will often store the data in a normalized way. Data marts for specific reports can then be built on top of the DW solution.

The DW database in a hybrid solution is kept on third normal form to eliminate data redundancy. A normal relational database, however, is not efficient for business intelligence reports where dimensional modelling is prevalent. Small data marts can shop for data from the consolidated warehouse and use the filtered, specific data for the fact tables and dimensions required. The DW effectively provides a single source of information from which the data marts can read, creating a highly flexible solution from a BI point of view. The hybrid architecture allows a DW to be replaced with a master data management solution where operational, not static information could reside.

The Data Vault Modeling components follow hub and spokes architecture. This modeling style is a hybrid design, consisting of the best practices from both 3rd normal form and star schema. The Data Vault model is not a true 3rd normal form, and breaks some of the rules that 3NF dictates be followed. It is however, a top-down architecture with a bottom up design. The Data Vault model is geared to be strictly a data warehouse. It is not geared to be end-user accessible, which when built, still requires the use of a data mart or star schema based release area for business purposes.

10.5 Data warehouses versus operational systems

Operational systems are optimized for preservation of data integrity and speed of recording of business transactions through use of database normalization and an entity-relationship model. Operational system designers generally follow the Codd rules of database normalization in order to ensure data integrity. Codd defined five increasingly stringent rules of normalization. Fully normalized database designs (that is, those satisfying all five Codd rules) often result in information from a business transaction being stored in dozens to hundreds of tables. Relational databases are efficient at managing the relationships between these tables. The databases have very fast insert/update performance because only a small amount of data in those tables is affected each time a transaction is processed. Finally, in order to improve performance, older data are usually periodically purged from operational systems.

Data warehouses are optimized for analytic access patterns. Analytic access patterns generally involve selecting specific fields and rarely if ever 'select *' as is more common in operational databases. Because of these differences in access patterns, operational databases (loosely, OLTP) benefit from the use of a row-oriented DBMS whereas analytics databases (loosely, OLAP) benefit from the use of a column-oriented DBMS. Unlike operational systems which maintain a snapshot of the business, data warehouses generally maintain an infinite history which is implemented through ETL processes that periodically migrate data from the operational systems over to the data warehouse.

Evolution in organization use

These terms refer to the level of sophistication of a data warehouse:

Offline operational data warehouse

Data warehouses in this stage of evolution are updated on a regular time cycle (usually daily, weekly or monthly) from the operational systems and the data is stored in an integrated reporting-oriented data

Offline data warehouse

Data warehouses at this stage are updated from data in the operational systems on a regular basis and the data warehouse data are stored in a data structure designed to facilitate reporting.

On time data warehouse

Online Integrated Data Warehousing represent the real time Data warehouses stage data in the warehouse is updated for every transaction performed on the source data

Integrated data warehouse

These data warehouses assemble data from different areas of business, so users can look up the information they need across other systems