Data Warehousing

A data warehouse is a collection of data marts representing historical data from different operations in the company. This data is stored in a structure optimized for querying and data analysis as a data warehouse. Table design, dimensions and organization should be consistent throughout a data warehouse so that reports or queries across the data warehouse are consistent. A data warehouse can also be viewed as a database for historical data from different functions within a company. The term Data Warehouse was coined by Bill Inmon in 1990, which he defined in the following way: "A warehouse is a subject-oriented, integrated, time-variant and non-volatile collection of data in support of management's decision making process". He defined the terms in the sentence as follows:

 Subject Oriented: Data that gives information about a particular subject instead of about a company's ongoing operations.

 Integrated: Data that is gathered into the data warehouse from a variety of sources and merged into a coherent whole.

 Time-variant: All data in the data warehouse is identified with a particular time period.

 Non-volatile: Data is stable in a data warehouse. More data is added but data is never removed. This enables management to gain a consistent picture of the business. It is a single, complete and consistent store of data obtained from a variety of different sources made available to end users in what they can understand and use in a business context. It can be Used for decision Support, Used to manage and control business, Used by managers and end-users to understand the business and make judgments.

Business Analysis Framework

The business analyst get the information from the data warehouses to measure the performance and make critical adjustments in order to win over other business holders in the market.

To design an effective and efficient data warehouse, we need to understand and analyze the business needs and construct a **business analysis framework**. Each person has different views regarding the design of a data warehouse. These views are as follows −

- **The top-down view** − This view allows the selection of relevant information needed for a data warehouse.
- **The data source view** − This view presents the information being captured, stored, and managed by the operational system.
- **The data warehouse view** − This view includes the fact tables and dimension tables. It represents the information stored inside the data warehouse.
- **The business query view** − It is the view of the data from the viewpoint of the end-user.

Components of Data Warehouse

1.2.1 Source Data Component

Source data coming into the data warehouses may be grouped into four broad categories:

Production Data: This data originates from the various operating systems used across the enterprise. Depending on the specific requirements of the data warehouse, selected segments of this data are extracted from different operational environments.

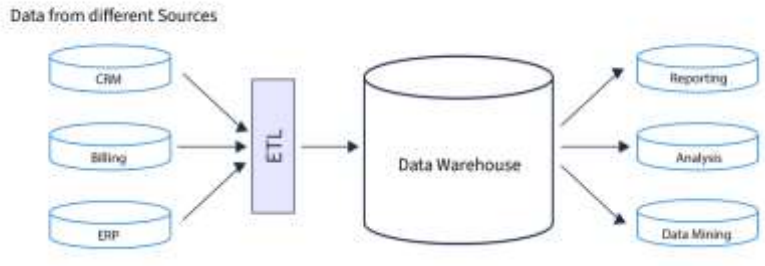
Internal Data: In each organization, the client keeps their "**private**" spreadsheets, reports, customer profiles, and sometimes even department databases. This is the internal data, part of which could be useful in a data warehouse.

Archived Data: Operational systems are mainly intended to run the current business. In every operational system, we periodically take the old data and store it in achieved files.

External Data: Most executives depend on information from external sources for a large percentage of the information they use. They use statistics associating to their industry produced by the external department.

1.2.2 Data Staging Component

After we have been extracted data from various operational systems and external sources, we have to prepare the files for storing in the data warehouse. The extracted data coming from several different sources need to be changed, converted, and made ready in a format that is relevant to be saved for querying and analysis. The three primary functions that take place in the staging area.



1) Data Extraction: This method has to deal with numerous data sources. We have to employ the appropriate techniques for each data source.

2) Data Transformation: As we know, data for a data warehouse comes from many different sources. If data extraction for a data warehouse posture big challenges, data transformation present even significant challenges. We perform several individual tasks as part of data transformation.

First, we clean the data extracted from each source. Cleaning may be the correction of misspellings or may deal with providing default values for missing data elements, or elimination of duplicates when we bring in the same data from various source systems.

Standardization of data components forms a large part of data transformation. Data transformation contains many forms of combining pieces of data from different sources. We combine data from single source record or related data parts from many source records.

On the other hand, data transformation also contains purging source data that is not useful and separating outsource records into new combinations. Sorting and merging of data take place on a large scale in the data staging area. When the data transformation function ends, we have a collection of integrated data that is cleaned, standardized, and summarized.

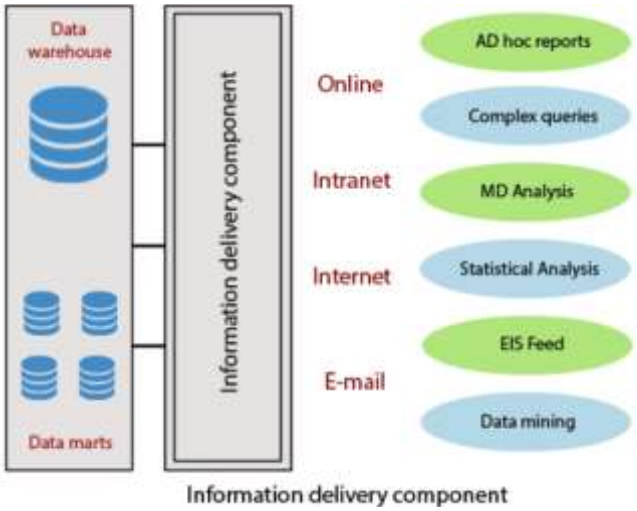
3) Data Loading: Two distinct categories of tasks form data loading functions. When we complete the structure and construction of the data warehouse and go live for the first time, we do the initial loading of the information into the data warehouse storage. The initial load moves high volumes of data using up a substantial amount of time.

1.2.3 Data Storage Components

Data storage for the data warehousing is a split repository. The data repositories for the operational systems generally include only the current data. Also, these data repositories include the data structured in highly normalized for fast and efficient processing.

1.2.4 Information Delivery Component

The information delivery element is used to enable the process of subscribing for data warehouse files and having it transferred to one or more destinations according to some customer-specified scheduling algorithm.



1.2.5 Metadata Component

Metadata in a data warehouse is equal to the data dictionary or the data catalog in a database management system. In the data dictionary, we keep the data about the logical data structures, the data about the records and addresses, the information about the indexes, and so on.

1.2.6 Data Marts

It includes a subset of corporate-wide data that is of value to a specific group of users. The scope is confined to particular selected subjects. Data in a data warehouse should be a fairly current, but not mainly up to the minute, although development in the data warehouse industry has made standard and incremental data dumps more achievable. Data marts are lower than data warehouses and usually contain organization. The current trends in data warehousing are to developed a data warehouse with several smaller related data marts for particular kinds of queries and reports.

1.2.7 Management and Control Component

The management and control elements coordinate the services and functions within the data warehouse. These components control the data transformation and the data transfer into the data warehouse storage. On the other hand, it moderates the data delivery to the clients. Its work with the database management systems and authorizes data to be correctly saved in the repositories. It monitors the movement of information into the staging method and from there into the data warehouses storage itself.

Building a Data Warehouse

Building a data warehouse involves several key steps that ensure the collection, transformation, storage, and accessibility of data for business analysis and decision-making. The process typically includes:

1. Requirement Analysis

- Understand the business objectives and identify the types of data needed.

- Define the users, reports, and queries that the data warehouse must support.

2. Data Source Identification

- Identify operational systems (e.g., ERP, CRM, flat files, external data).
- Analyze data formats, quality, and availability.

3. Data Extraction

- Extract relevant data from multiple heterogeneous sources.
- Use tools like ETL (Extract, Transform, Load) to automate this process.

4. Data Cleaning and Transformation

- Clean the data by removing duplicates, correcting errors, and filling missing values.
- Transform data into a uniform format suitable for analysis (e.g., date formats, units).

5. Data Loading

- Load the cleaned and transformed data into the data warehouse.
- This can be done in batch mode or in real-time (streaming).

6. Data Modeling

- Design the schema using star schema, snowflake schema, or galaxy schema.
- Create fact and dimension tables based on analysis needs.

7. Metadata Management

- Store information about data sources, structures, and transformations.
- Helps users understand and access data efficiently.

8. Data Indexing and Partitioning

- Index and partition large tables for faster query performance.

9. Testing and Validation

- Ensure the data is accurate, consistent, and reliable.
- Validate reports and queries against expected results.

10. Deployment and Maintenance

- Deploy the warehouse for end-user access through reporting tools or OLAP systems.
- Maintain and update the system regularly to incorporate new data or business needs.

Data Warehouse Architecture

Data Warehouse Architecture is complex as it's an information system that contains historical and commutative data from multiple sources. There are 3 approaches for constructing Data Warehouse layers: Single Tier, Two tier and Three tier. This 3 tier architecture of Data Warehouse is explained as below.

Single-tier architecture

The objective of a single layer is to minimize the amount of data stored. This goal is to remove data redundancy. This architecture is not frequently used in practice.

Two-tier architecture

Two-layer architecture is one of the Data Warehouse layers which separates physically available sources and data warehouse. This architecture is not expandable and also not supporting a large number of end-users. It also has connectivity problems because of network limitations.

Three-Tier Data Warehouse Architecture

This is the most widely used Architecture of Data Warehouse. It consists of the Top, Middle and Bottom Tier.

Bottom Tier:

The bottom tier is a warehouse database server that is almost always a relational database system. Back-end tools and utilities are used to feed data into the bottom tier from operational databases or other external sources (such as customer profile information provided by external consultants). These tools and utilities perform data extraction, cleaning, and transformation (e.g., to merge similar data from different sources into a unified format), as well as load and refresh functions to update the data warehouse. The data are extracted using application program interfaces known as gateways. A gateway is supported by the underlying

DBMS and allows client programs to generate SQL code to be executed at a server. Examples of gateways include ODBC (Open Database Connection) and OLEDB (Open Linking and Embedding for Databases) by Microsoft and JDBC (Java Database Connection). This tier also contains a metadata repository, which stores information about the data warehouse and its contents.

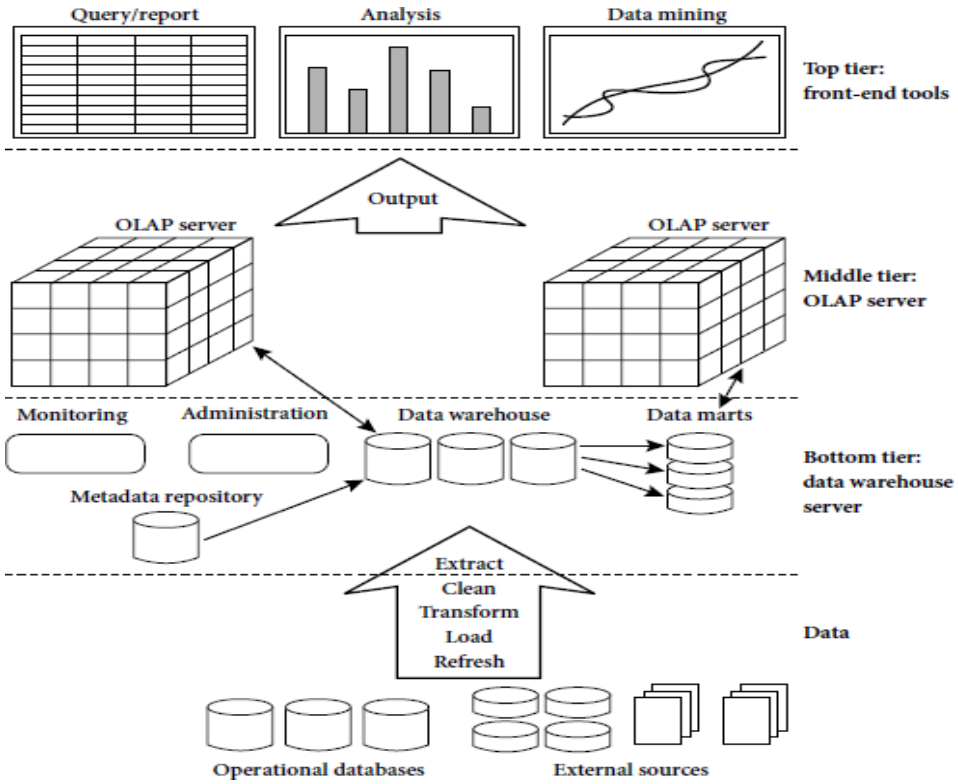
Middle Tier

The middle tier is an OLAP server that is typically implemented using either a relational OLAP (ROLAP) model or a multidimensional OLAP.

- OLAP model is an extended relational DBMS that maps operations on multidimensional data to standard relational operations.
- A multidimensional OLAP (MOLAP) model, that is, a special-purpose server that directly implements multidimensional data and operations.

Top Tier

The top tier is a front-end client layer, which contains query and reporting tools, analysis tools, and/or data mining tools (e.g., trend analysis, prediction, and so on).



Designing the Data Warehouse Schema

After understanding the requirements for the data warehouse, the next step is to design the schema. The schema is the structure of the data warehouse, including the tables, columns, and relationships between them.

There are several approaches to designing a data warehouse schema,

Star schema: A star schema consists of a central fact table surrounded by dimension tables. The fact table contains the measures or facts, and the dimension tables contain the attributes or context for the measures. The schema is called a star because the dimension tables are connected to the central fact table through foreign key relationships, forming a star shape.

Snowflake schema: A snowflake schema is an extension of the star schema, where the dimension tables are normalized into multiple tables. This results in a more complex schema, but it can improve query performance by reducing the amount of data stored in the dimension tables.

Hybrid schema: A hybrid schema is a combination of the star and snowflake schemas, where some dimension tables are normalized and others are not. This can be useful when some dimensions are highly granular and require normalization, while others are less granular and can be denormalized.

ETL Process in Data Warehouse

ETL (Extract, Transform, Load) is a key process in data warehousing that prepares data for analysis. It involves:

- Extracting data from multiple sources
- Transforming it into a consistent format
- Loading it into a central data warehouse or data lake

ETL helps businesses unify and clean data, making it reliable and ready for analysis. It improves data quality, security, and accessibility, enabling better insights and faster decision-making in a world of diverse data sources.

ETL Process

The ETL process, which stands for Extract, Transform, and Load, is a critical methodology used to prepare data for storage, analysis, and reporting in a data warehouse. It involves three distinct stages that help to streamline raw data from multiple sources into a clean, structured, and usable form. Here's a detailed breakdown of each phase:

Extraction

The Extract phase is the first step in the ETL process, where raw data is collected from various data sources. These sources can be diverse, ranging from structured sources like databases (SQL, NoSQL), to semi-structured data like JSON, XML, or unstructured data such as emails or flat files. The main goal of extraction is to gather data without altering its format, enabling it to be further processed in the next stage.

Types of data sources can include:

- **Structured:** SQL databases, ERPs, CRMs
- **Semi-structured:** JSON, XML
- **Unstructured:** Emails, web pages, flat files

2. Transformation

The Transform phase is where the magic happens. Data extracted in the previous phase is often raw and inconsistent. During transformation, the data is cleaned, aggregated, and formatted according to business rules. This is a crucial step because it ensures that the data meets the quality standards required for accurate analysis.

Common transformations include:

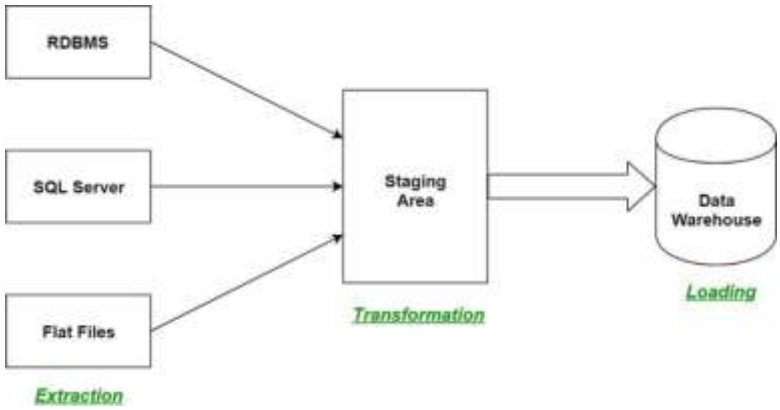
- **Data Filtering:** Removing irrelevant or incorrect data.
- **Data Sorting:** Organizing data into a required order for easier analysis.
- **Data Aggregating:** Summarizing data to provide meaningful insights (e.g., averaging sales data).

The transformation stage can also involve more complex operations such as currency conversions, text normalization, or applying domain-specific rules to ensure the data aligns with organizational needs.

3. Loading

Once data has been cleaned and transformed, it is ready for the final step: Loading. This phase involves transferring the transformed data into a data warehouse, data lake, or another target system for storage. Depending on the use case, there are two types of loading methods:

- **Full Load:** All data is loaded into the target system, often used during the initial population of the warehouse.
- **Incremental Load:** Only new or updated data is loaded, making this method more efficient for ongoing data updates.



Metadata

Metadata is data about data which defines the data warehouse. It is used for building, maintaining and managing the data warehouse.In the Data Warehouse Architecture, meta-data plays an important role as it specifies the source, usage, values, and features of data warehouse data. It also defines how data can be changed and processed. It is closely connected to the data warehouse.

EXAMPLE

- 4030 KJ732 299.90

- This is a meaningless data until we consult the Meta that tell us it was

  - Model number: 4030

- Sales Agent ID: KJ732
- Total sales amount of $299.9

Meta Data are essential ingredients in the transformation of data into knowledge. Metadata can be classified into following categories:

- **Technical Meta Data**: This kind of Metadata contains information about warehouse which is used by Data warehouse designers and administrators.
- **Business Meta Data:** This kind of Metadata contains detail that gives end-users a way easy to understand information stored in the data warehouse.

Query Tools

One of the primary objects of data warehousing is to provide information to businesses to make strategic decisions. Query tools allow users to interact with the data warehouse system. These tools fall into four different categories:

- Query and reporting tools
- Application Development tools
- Data mining tools
- OLAP tools

Query and reporting tools

Query and reporting tools can be further divided into

- Reporting tools
- Managed query tools

Reporting tools:

- Reporting tools can be further divided into production reporting tools and desktop report writer.

  - ✓ Report writers: This kind of reporting tool is tools designed for end-users for their analysis.

- ✓ Production reporting: This kind of tools allows organizations to generate regular operational reports.

- It also supports high volume batch jobs like printing and calculating.
- Some popular reporting tools are Brio, Business Objects, Oracle, PowerSoft, SAS Institute.

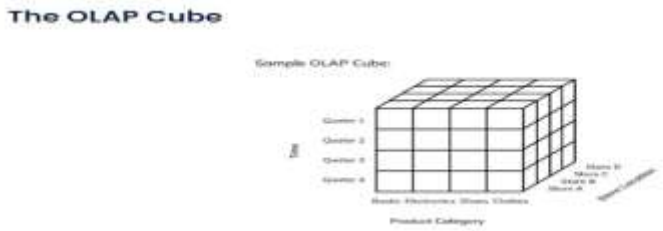
Managed query tools:

This kind of access tools helps end users to resolve snags in database and SQL and database structure by inserting meta-layer between users and database

OLAP (Online analytical Processing):

- OLAP is an approach to answering multi-dimensional analytical (MDA) queries swiftly.
- OLAP is part of the broader category of business intelligence, which also encompasses relational database, report writing and data mining.
- OLAP tools enable users to analyze multidimensional data interactively from multiple perspectives.

OLAP consists of three basic analytical operations:



Many of the OLAP applications include sales reporting, marketing, business process management (BPM), forecasting, budgeting , creating finance reports and others. Each OLAP cube is presented through measures and dimensions. Measures refers to the numeric value categorized by dimensions. In below diagrams, dimensions are time, item type and courtiers/cities and the values inside them (605, 825, 14, 400) are measures.
The OLAP approach is used to analyze multidimensional data from multiple sources and perspectives. The three basic operations in OLAP are:

- **Roll-up (Consolidation)**
- **Drill-down**
- **Slicing and dicing**

Roll-up or consolidation refers to data aggregation and computation in one or more dimensions. It is actually performed on an OLAP cube. For instance, the cube with cities is rolled up to countries to depict the data with respect to time (in quarters) and item (type).

On the contrary, Drill-down operation helps users navigate through the data details. In the above example, drilling down enables users to analyze data in the three months of the first quarter separately. The data is divided with respect to cities, months (time) and item (type).

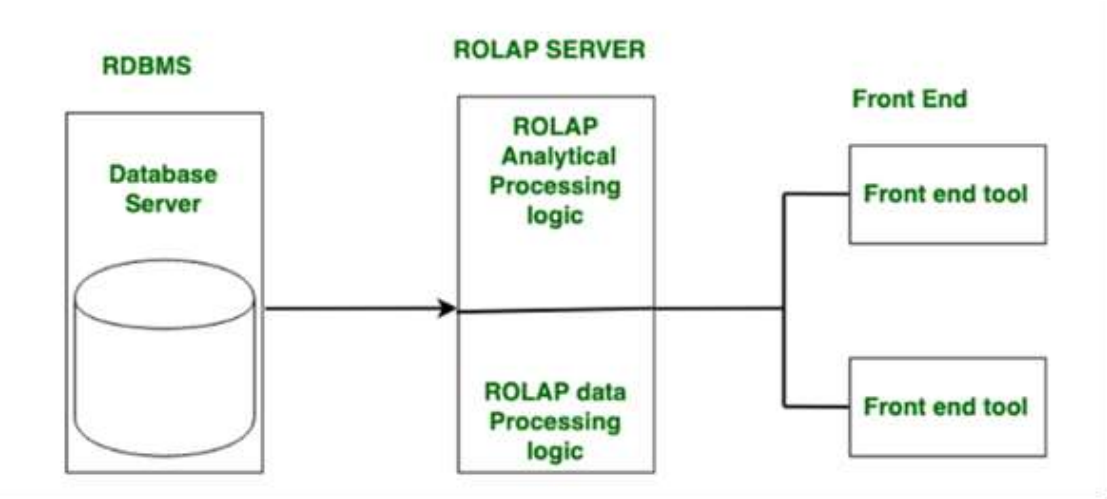
Slicing is an OLAP feature that allows taking out a portion of the OLAP cube to view specific data. For instance, in the above diagram, the cube is sliced to a two dimensional view showing Item(types) with respect to Quadrant (time). The location dimension is skipped here. In dicing, users can analyze

data from different viewpoints. In the above diagram, the users create a sub cube and chose to view data for two Item types and two locations in two quadrants.

Types of OLAP:

1. Relational OLAP (ROLAP):

- ROLAP works directly with relational databases. The base data and the dimension tables are stored as relational tables and new tables are created to hold the aggregated information. It depends on a specialized schema design.
- This methodology relies on manipulating the data stored in the relational database to give the appearance of traditional OLAP's slicing and dicing functionality. In essence, each action of slicing and dicing is equivalent to adding a "WHERE" clause in the SQL statement. ROLAP tools do not use pre-calculated data cubes but instead pose the query to the standard relational database and its tables in order to bring back the data required to answer the question.
- ROLAP tools feature the ability to ask any question because the methodology does not limit to the contents of a cube. ROLAP also has the ability to drill down to the lowest level of detail in the database.



Benefits:

- It is compatible with data warehouses and OLTP systems.
- The data size limitation of ROLAP technology is determined by the underlying RDBMS. As a result, ROLAP does not limit the amount of data that can be stored.

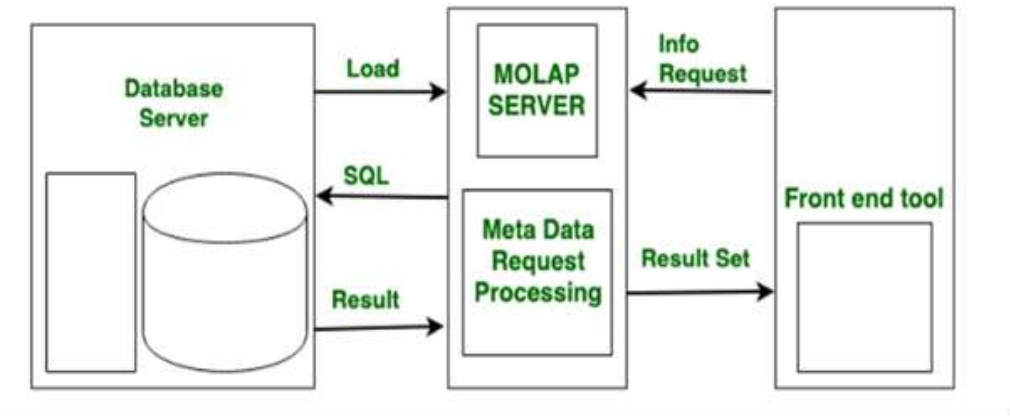
Limitations:

- SQL functionality is constrained.
- It's difficult to keep aggregate tables up to date.

2. Multidimensional OLAP (MOLAP):

- MOLAP is the 'classic' form of OLAP and is sometimes referred to as just OLAP.

- MOLAP stores this data in an optimized multi-dimensional array storage, rather than in a relational database. Therefore it requires the pre-computation and storage of information in the cube - the operation known as processing.
- MOLAP tools generally utilize a pre-calculated data set referred to as a data cube. The data cube contains all the possible answers to a given range of questions.
- MOLAP tools have a very fast response time and the ability to quickly write back data into the data set.



Benefits:

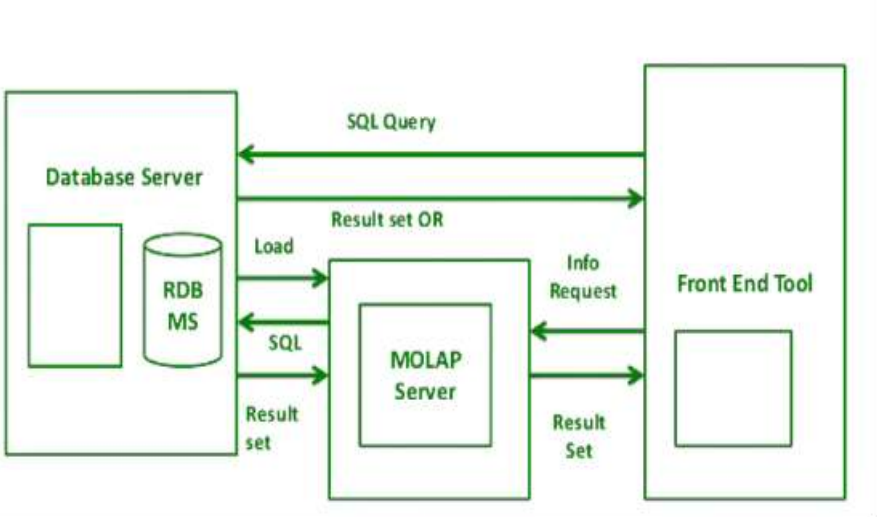
- Suitable for slicing and dicing operations.
- Outperforms ROLAP when data is dense.
- Capable of performing complex calculations.

Limitations:

- It is difficult to change the dimensions without re-aggregating.
- Since all calculations are performed when the cube is built, a large amount of data cannot be stored in the cube itself.

3. Hybrid OLAP (HOLAP):

- There is no clear agreement across the industry as to what constitutes Hybrid OLAP, except that a database will divide data between relational and specialized storage.
- For example, for some vendors, a HOLAP database will use relational tables to hold the larger quantities of detailed data, and use specialized storage for at least some aspects of the smaller quantities of more-aggregate or less-detailed data.
- HOLAP addresses the shortcomings of MOLAP and ROLAP by combining the capabilities of both approaches.
- HOLAP tools can utilize both pre-calculated cubes and relational data sources.



Benefits:

- HOLAP combines the benefits of MOLAP and ROLAP.
- Provide quick access at all aggregation levels.

Limitations

- Because it supports both MOLAP and ROLAP servers, HOLAP architecture is extremely complex.
- There is a greater likelihood of overlap, particularly in their functionalities.