LECTURE SESSION EIGHT

DATA WAREHOUSING

Lecture objectives

8.1 Introduction

Hello and welcome to today's class. Last week we learned about distributed database systems, today we shall learn about data warehousing particularly the motivation towards data warehousing, benefits of data warehousing, architecture of data warehousing, types of data in a data warehouse and information flow, the problems of data warehousing, data warehouse design, data marts, data analysis tools and finally the various types of OLAP servers. I hope you will enjoy the class.

8.2 Learning Outcomes

At the end of this lecture, you should be able to:

1. Explain the concept of data warehousing.
2. Describe the architecture of data warehousing.

3. Discuss different types of data in a data warehouse and information flow.
4. Discuss the problems associated with data warehousing.
5. Apply the various data analysis tools in a data warehousing environment.

8.3 Understanding a Data Warehouse

The term "Data Warehouse" was first coined by Bill Inmon in 1990. According to Inmon, a data warehouse is a subject oriented, integrated, time-variant, and non-volatile collection of data. This data helps analysts to take informed decisions in an organization.

Definition

A data warehouse is a subject oriented, integrated, time-variant, and non-volatile collection of data.

An operational database undergoes frequent changes on a daily basis on account of the transactions that take place. Suppose a business executive wants to analyze previous feedback on any data such as a product, a supplier, or any consumer data, then the executive will have no data available to analyze because the previous data has been updated due to transactions.

A data warehouses provides us generalized and consolidated data in multidimensional view. Along with generalized and consolidated view of data, a data warehouses also provides us Online Analytical Processing (OLAP) tools. These tools help us in interactive and effective analysis of data in a multidimensional space. This analysis results in data generalization and data mining.

Data mining functions such as association, clustering, classification, prediction can be integrated with OLAP operations to enhance the interactive mining of knowledge at multiple level of abstraction. That's why data

warehouse has now become an important platform for data analysis and online analytical processing.

Key points on a Data Warehouse

- A data warehouse is a database, which is kept separate from the organization's operational database.
- There is no frequent updating done in a data warehouse.
- It possesses consolidated historical data, which helps the organization to analyze its business.
- A data warehouse helps executives to organize, understand, and use their data to take strategic decisions.
- Data warehouse systems help in the integration of diversity of application systems.
- A data warehouse system helps in consolidated historical data analysis.

Why a Data Warehouse is Separated from Operational Databases

A data warehouses is kept separate from operational databases due to the following reasons −

- An operational database is constructed for well-known tasks and workloads such as searching particular records, indexing, etc. In contrast, data warehouse queries are often complex and they present a general form of data.
- Operational databases support concurrent processing of multiple transactions. Concurrency control and recovery mechanisms are required for operational databases to ensure robustness and consistency of the database.
- An operational database query allows to read and modify operations, while an OLAP query needs only **read only** access of stored data.
- An operational database maintains current data. On the other hand, a data warehouse maintains historical data.

8.4 Data Warehouse Features

The key features of a data warehouse are discussed below −

- **Subject Oriented** − A data warehouse is subject oriented because it provides information around a subject rather than the organization's ongoing operations. These subjects can be product, customers, suppliers, sales, revenue, etc. A data warehouse does not focus on the ongoing operations, rather it focuses on modelling and analysis of data for decision making.
- **Integrated** − A data warehouse is constructed by integrating data from heterogeneous sources such as relational databases, flat files, etc. This integration enhances the effective analysis of data.
- **Time Variant** − The data collected in a data warehouse is identified with a particular time period. The data in a data warehouse provides information from the historical point of view.
- **Non-volatile** − Non-volatile means the previous data is not erased when new data is added to it. A data warehouse is kept separate from the operational database and therefore frequent changes in operational database is not reflected in the data warehouse.

Take Note

A data warehouse does not require transaction processing, recovery, and concurrency controls, because it is physically stored and separate from the operational database.

8.5 Data Warehouse Applications

As discussed before, a data warehouse helps business executives to organize, analyze, and use their data for decision making. A data warehouse serves as a sole part of a plan-execute-assess "closed-loop" feedback system for the enterprise management. Data warehouses are widely used in the following fields −

- Financial services
- Banking services
- Consumer goods
- Retail sectors
- Controlled manufacturing

8.6 Types of Data Warehouse

Information processing, analytical processing, and data mining are the three types of data warehouse applications that are discussed below −

- **Information Processing** − A data warehouse allows to process the data stored in it. The data can be processed by means of querying, basic statistical analysis, reporting using crosstabs, tables, charts, or graphs.
- **Analytical Processing** − A data warehouse supports analytical processing of the information stored in it. The data can be analyzed by means of basic OLAP operations, including slice-and-dice, drill down, drill up, and pivoting.
- **Data Mining** − Data mining supports knowledge discovery by finding hidden patterns and associations, constructing analytical models, performing classification and prediction. These mining results can be presented using the visualization tools.

The table below shows the differences between a data warehouse (i.e., Online Analytical Processing-OLAP systems) and operational databases (commonly known as Online Transaction Processing – OLTP systems)

Sr.No.	Data Warehouse (OLAP)	Operational Database (OLTP)
1	It involves historical processing of information.	It involves day-to-day processing.
2	OLAP systems are used by knowledge workers such as executives, managers, and analysts.	OLTP systems are used by clerks, DBAs, or database professionals.
3	It is used to analyze the business.	It is used to run the business.
4	It focuses on Information out.	It focuses on Data in.
5	It is based on Star Schema, Snowflake Schema, and Fact Constellation Schema.	It is based on Entity Relationship Model.

6	It focuses on Information out.	It is application oriented.
7	It contains historical data.	It contains current data.
8	It provides summarized and consolidated data.	It provides primitive and highly detailed data.
9	It provides summarized and multidimensional view of data.	It provides detailed and flat relational view of data.
10	The number of users is in hundreds.	The number of users is in thousands.
11	The number of records accessed is in millions.	The number of records accessed is in tens.
12	The database size is from 100GB to 100 TB.	The database size is from 100 MB to 100 GB.
13	These are highly flexible.	It provides high performance.

8.7 Data Warehousing - Concepts

8.7.1 Data Warehousing

Data warehousing is the process of constructing and using a data warehouse. A data warehouse is constructed by integrating data from multiple heterogeneous sources that support analytical reporting, structured and/or ad hoc queries, and decision making. Data warehousing involves data cleaning, data integration, and data consolidations.

8.7.2 Uses of Data Warehouse Information

There are decision support technologies that help utilize the data available in a data warehouse. These technologies help executives to use the warehouse quickly and effectively. They can gather data, analyze it, and take decisions based on the information present in the warehouse. The information gathered in a warehouse can be used in any of the following domains −

- **Tuning Production Strategies** − The product strategies can be well tuned by repositioning the products and managing the product portfolios by comparing the sales quarterly or yearly.

- **Customer Analysis** − Customer analysis is done by analyzing the customer's buying preferences, buying time, budget cycles, etc.
- **Operations Analysis** − Data warehousing also helps in customer relationship management, and making environmental corrections. The information also allows us to analyze business operations.

8.7.3 Integrating Heterogeneous Databases

To integrate heterogeneous databases, we have two approaches −

- Query-driven Approach
- Update-driven Approach

8.7.3.1 Query-Driven Approach

This is the traditional approach to integrate heterogeneous databases. This approach was used to build wrappers and integrators on top of multiple heterogeneous databases. These integrators are also known as mediators.

The process of query-driven approach involves the following steps:

- When a query is issued to a client side, a metadata dictionary translates the query into an appropriate form for individual heterogeneous sites involved.
- Now these queries are mapped and sent to the local query processor.
- The results from heterogeneous sites are integrated into a global answer set.

Disadvantages of Query-driven approach

i. Query-driven approach needs complex integration and filtering processes.
ii. This approach is very inefficient.
iii. It is very expensive for frequent queries.
iv. This approach is also very expensive for queries that require aggregations.

8.7.3.2 Update-Driven Approach

This is an alternative to the traditional approach. Today's data warehouse systems follow update-driven approach rather than the traditional approach discussed earlier. In update-driven approach, the information from multiple heterogeneous sources are integrated in advance and are stored in a warehouse. This information is available for direct querying and analysis.

Advantages of update-driven approach

This approach has the following advantages −

i. This approach provides high performance.
ii. The data is copied, processed, integrated, annotated, summarized and restructured in semantic data store in advance.
iii. Query processing does not require an interface to process data at local sources.

8.7.4 Functions of Data Warehouse Tools and Utilities

The following are the functions of data warehouse tools and utilities −

a. **Data Extraction** − Involves gathering data from multiple heterogeneous sources.
b. **Data Cleaning** − Involves finding and correcting the errors in data.
c. **Data Transformation** − Involves converting the data from legacy format to warehouse format.
d. **Data Loading** − Involves sorting, summarizing, consolidating, checking integrity, and building indices and partitions.
e. **Refreshing** − Involves updating from data sources to warehouse.

Take Note That:

Data cleaning and data transformation are important steps in improving the quality of data and data mining results.

8.8 Data Warehousing - Terminologies

Some of the most commonly used terms in data warehousing are:

8.8.1 Metadata

Metadata is simply defined as data about data. The data that are used to represent other data is known as metadata. For example, the index of a book serves as a metadata for the contents in the book. In other words, we can say that metadata is the summarized data that leads us to the detailed data.

In terms of data warehouse, we can define metadata as following −

- Metadata is a road-map to data warehouse.
- Metadata in data warehouse defines the warehouse objects.
- Metadata acts as a directory. This directory helps the decision support system to locate the contents of a data warehouse.

Metadata Repository

Metadata repository is an integral part of a data warehouse system. It contains the following metadata −

- **Business metadata** − It contains the data ownership information, business definition, and changing policies.
- **Operational metadata** − It includes currency of data and data lineage. Currency of data refers to the data being active, archived, or purged. Lineage of data means history of data migrated and transformation applied on it.
- **Data for mapping from operational environment to data warehouse** − Its metadata that includes source databases and their contents, data extraction, data partition, cleaning, transformation rules, data refresh and purging rules.
- **The algorithms for summarization** − It includes dimension algorithms, data on granularity, aggregation, summarizing, etc.

8.8.2 Data Cube

A data cube helps us represent data in multiple dimensions. It is defined by ***dimensions*** and ***facts***. The dimensions are the entities with respect to which an enterprise preserves the records.

Illustration of Data Cube

Suppose a company wants to keep track of sales records with the help of sales data warehouse with respect to time, item, branch, and location. These dimensions allow to keep track of monthly sales and at which branch the

items were sold. There is a table associated with each dimension. This table is known as **dimension table**. For example, "item" dimension table may have attributes such as item_name, item_type, and item_brand.

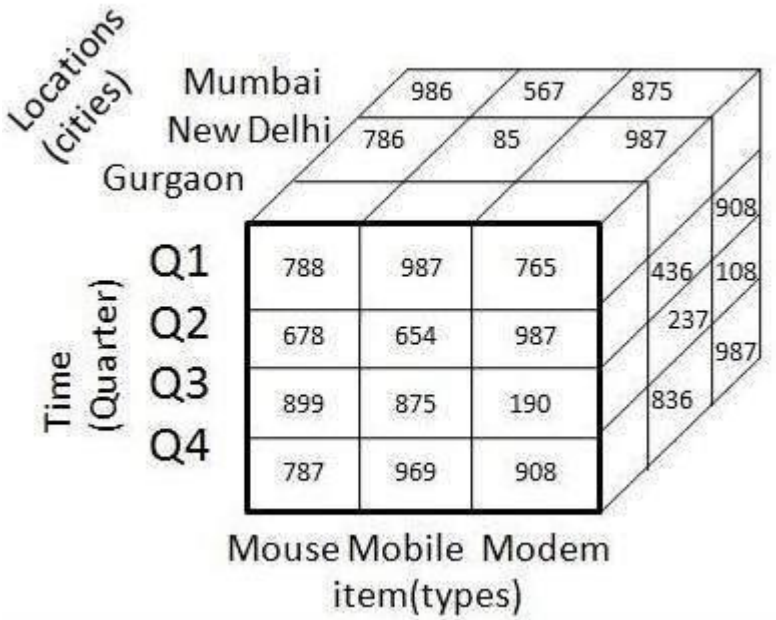
The following table represents the 2-D view of Sales Data for a company with respect to time, item, and location dimensions.

Location="New Delhi"				
Time(quarter)	Item(type)			
	Entertainment	Keyboard	Mobile	Locks
Q1	500	700	10	300
Q2	769	765	30	476
Q3	987	489	18	659
Q4	666	976	40	539

But here in this 2-D table, we have records with respect to time and item only. The sales for New Delhi are shown with respect to time, and item dimensions according to type of items sold. If we want to view the sales data with one more dimension, say, the location dimension, then the 3-D view would be useful. The 3-D view of the sales data with respect to time, item, and location is shown in the table below −

Time	Location="Gurgaon"			Location="New Delhi"			Location="Mumbai"		
	Item			Item			Item		
	Mouse	Mobile	Modem	Mouse	Mobile	Modem	Mouse	Mobile	Modem
Q1	788	987	765	786	85	987	986	567	875
Q2	678	654	987	659	786	436	980	876	908
Q3	899	875	190	983	909	237	987	100	1089
Q4	787	969	908	537	567	836	837	926	987

The above 3-D table can be represented as ***3-D data cube*** as shown in the following figure −


Locations (cities)
Mumbai
New Delhi
Gurgaon
986
567
875
786
85
987
908
436
108
237
987
836
Time (Quarter)
Q1
Q2
Q3
Q4
788
987
765
678
654
987
899
875
190
787
969
908
Mouse
Mobile
Modem
item(types)

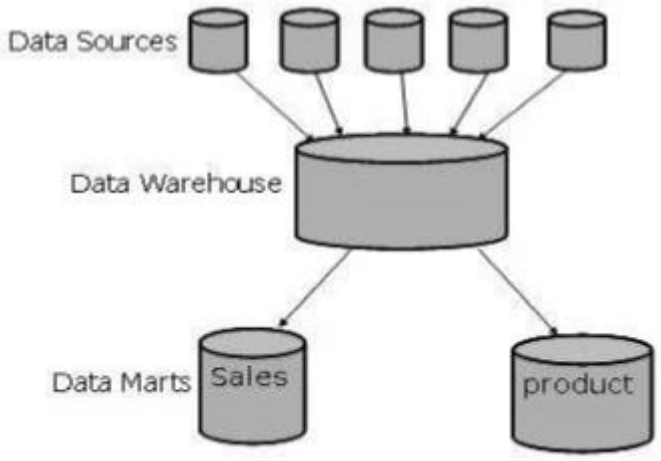
8.8.3 Data Mart

Data marts contain a subset of organization-wide data that is valuable to specific groups of people in an organization. In other words, a data mart contains only those data that is specific to a particular group. For example, the marketing data mart may contain only data related to items, customers, and sales. Data marts are confined to subjects.

Some important points to remember about data marts are:

- Windows-based or Unix/Linux-based servers are used to implement data marts. They are implemented on low-cost servers.
- The implementation cycle of a data mart is measured in short periods of time, i.e., in weeks rather than months or years.
- The life cycle of data marts may be complex in the long run, if their planning and design are not organization-wide.
- Data marts are small in size.
- Data marts are customized by department.
- The source of a data mart is departmentally structured data warehouse.
- Data marts are flexible.

The following figure shows a graphical representation of data marts.



8.8.4 Virtual Warehouse

The view over an operational data warehouse is known as virtual warehouse. It is easy to build a virtual warehouse. Building a virtual warehouse requires excess capacity on operational database servers.

Definition

A virtual data warehouse is a set of separate databases, which can be queried together, so a user can effectively access all the data as if it was stored in one data warehouse

In a data warehouse model, data is aggregated from a range of source systems relevant to a specific business area, such as sales or finance.

8.9 Data Warehousing - System Processes

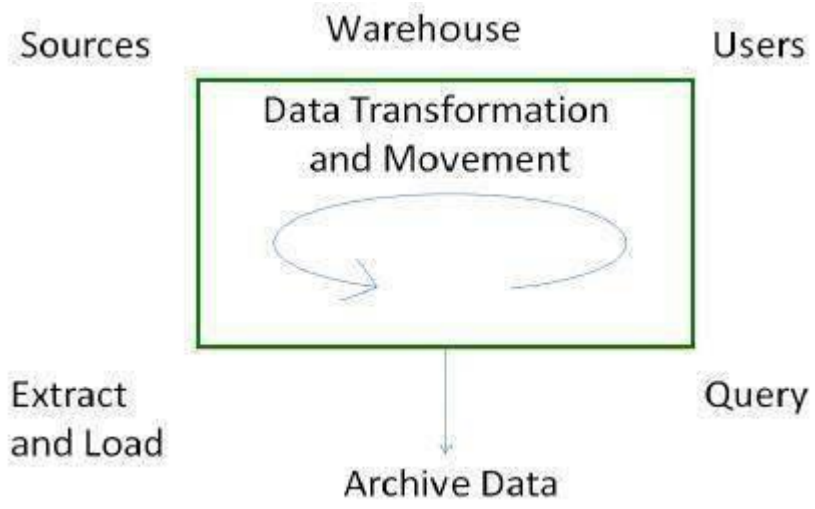
We have a fixed number of operations to be applied on the operational databases and we have well-defined techniques such as **use normalized data**, **keep table small**, etc. These techniques are suitable for delivering a solution. But in case of decision-support systems, we do not know what query and operation needs to be executed in future. Therefore, techniques applied on operational databases are not suitable for data warehouses.

Under data warehousing - system processes, we will discuss how to build data warehousing solutions on top open-system technologies like Unix and relational databases.

8.9.1 Process Flow in Data Warehouse

There are four major processes that contribute to a data warehouse −

- Extract and load the data.
- Cleaning and transforming the data.
- Backup and archive the data.
- Managing queries and directing them to the appropriate data sources.



Extract and Load Process

Data extraction takes data from the source systems. Data load takes the extracted data and loads it into the data warehouse. Before loading the data into the data warehouse, the information extracted from the external sources must be reconstructed.

i. Controlling the Process

Controlling the process involves determining when to start data extraction and the consistency check on data. Controlling process ensures that the tools, the logic modules, and the programs are executed in correct sequence and at correct time.

ii. When to Initiate Extract

Data needs to be in a consistent state when it is extracted, i.e., the data warehouse should represent a single, consistent version of the information to the user.

For example, in a customer profiling data warehouse in telecommunication sector, it is illogical to merge the list of customers at 8 pm on Wednesday from a customer database with the customer subscription events up to 8 pm on Tuesday. This would mean that we are finding the customers for whom there are no associated subscriptions.

iii. Loading the Data

After extracting the data, it is loaded into a temporary data store where it is cleaned up and made consistent (Consistency checks are executed only when all the data sources have been loaded into the temporary data store).

Clean and Transform Process

Once the data is extracted and loaded into the temporary data store, it is time to perform Cleaning and Transforming. Here is the list of steps involved in Cleaning and Transforming −

- Clean and transform the loaded data into a structure
- Partition the data
- Aggregation

i. Clean and transform the Loaded Data into a Structure

Cleaning and transforming the loaded data helps speed up the queries. It can be done by making the data consistent −

- within itself.
- with other data within the same data source.
- with the data in other source systems.
- with the existing data present in the warehouse.

Transforming involves converting the source data into a structure. Structuring the data increases the query performance and decreases the operational cost. The data contained in a data warehouse must be transformed to support performance requirements and control the ongoing operational costs.

ii. Partition the Data

It will optimize the hardware performance and simplify the management of data warehouse. Here we partition each fact table into multiple separate partitions.

iii. Aggregation

Aggregation is required to speed up common queries. Aggregation relies on the fact that most common queries will analyze a subset or an aggregation of the detailed data.

Backup and Archive the Data

In order to recover the data in the event of data loss, software failure, or hardware failure, it is necessary to keep regular back ups. Archiving involves

removing the old data from the system in a format that allow it to be quickly restored whenever required.

For example, in a retail sales analysis data warehouse, it may be required to keep data for 3 years with the latest 6 months data being kept online. In such as scenario, there is often a requirement to be able to do month-on-month comparisons for this year and last year. In this case, we require some data to be restored from the archive.

Query Management Process

This process performs the following functions −

- manages the queries.
- helps speed up the execution time of queries.
- directs the queries to their most effective data sources.
- ensures that all the system sources are used in the most effective way.
- monitors actual query profiles.

The information generated in this process is used by the warehouse management process to determine which aggregations to generate. This process does not generally operate during the regular load of information into data warehouse.

8.10 Data Warehousing - Architecture

Under data warehousing – architecture, we will discuss the business analysis framework for the data warehouse design and architecture of a data warehouse.

8.10.1 Business Analysis Framework

The business analyst gets 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. Having a data warehouse offers the following advantages −

- Since a data warehouse can gather information quickly and efficiently, it can enhance business productivity.

- A data warehouse provides us a consistent view of customers and items; hence, it helps us manage customer relationship.
- A data warehouse also helps in bringing down the costs by tracking trends, patterns over a long period in a consistent and reliable manner.

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.

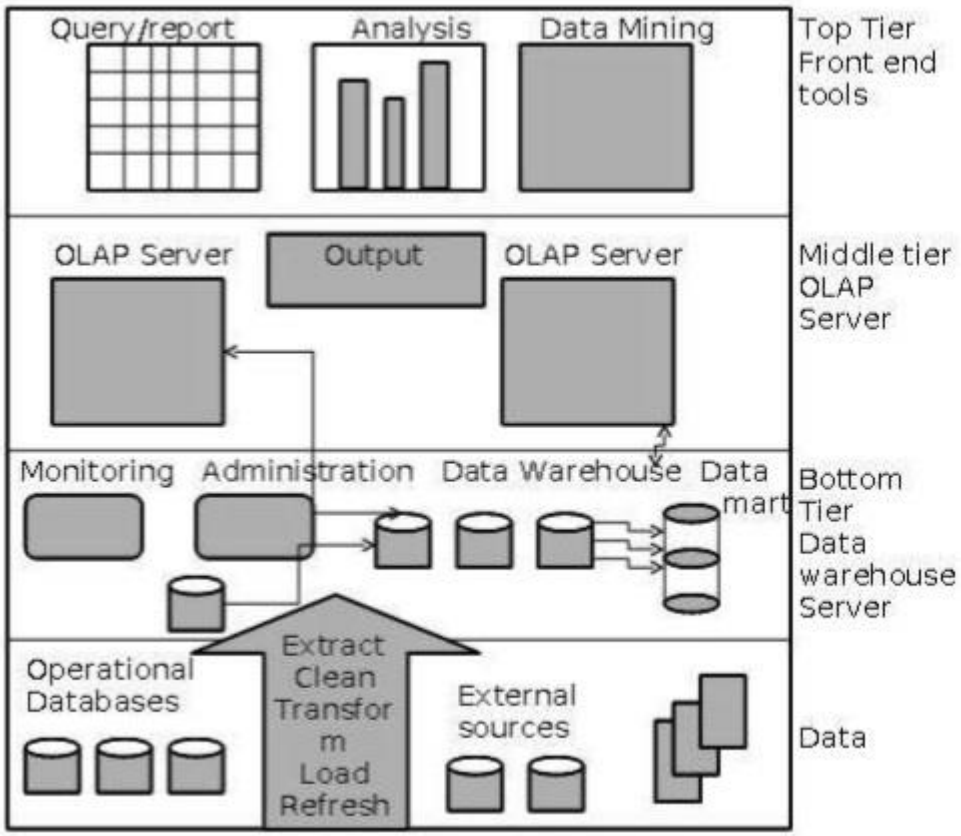
8.10.2 Three-Tier Data Warehouse Architecture

Generally, a data warehouses adopts a three-tier architecture. Following are the three tiers of the data warehouse architecture.

- **Bottom Tier** − The bottom tier of the architecture is the data warehouse database server. It is the relational database system. We use the back end tools and utilities to feed data into the bottom tier. These back-end tools and utilities perform the Extract, Clean, Load, and refresh functions.
- **Middle Tier** − In the middle tier, we have the OLAP Server that can be implemented in either of the following ways.
    - By Relational OLAP (ROLAP), which is an extended relational database management system. The ROLAP maps the operations on multidimensional data to standard relational operations.
    - By Multidimensional OLAP (MOLAP) model, which directly implements the multidimensional data and operations.

- **Top-Tier** − This tier is the front-end client layer. This layer holds the query tools and reporting tools, analysis tools and data mining tools.

The following diagram depicts the three-tier architecture of data warehouse −



8.10.3 Data Warehouse Models

From the perspective of data warehouse architecture, we have the following data warehouse models −

- Virtual Warehouse
- Data mart
- Enterprise Warehouse

i. Virtual Warehouse

The view over an operational data warehouse is known as a virtual warehouse. It is easy to build a virtual warehouse. Building a virtual warehouse requires excess capacity on operational database servers.

ii. Data Mart

Data mart contains a subset of organization-wide data. This subset of data is valuable to specific groups of an organization.

In other words, we can claim that data marts contain data specific to a particular group. For example, the marketing data mart may contain data related to items, customers, and sales. Data marts are confined to subjects.

Points to remember about data marts −

- Window-based or Unix/Linux-based servers are used to implement data marts. They are implemented on low-cost servers.
- The implementation data mart cycles is measured in short periods of time, i.e., in weeks rather than months or years.
- The life cycle of a data mart may be complex in long run, if its planning and design are not organization-wide.
- Data marts are small in size.
- Data marts are customized by department.
- The source of a data mart is departmentally structured data warehouse.
- Data mart are flexible.

a. Enterprise Warehouse

- An enterprise warehouse collects all the information and the subjects spanning an entire organization
- It provides us enterprise-wide data integration.
- The data is integrated from operational systems and external information providers.
- This information can vary from a few gigabytes to hundreds of gigabytes, terabytes or beyond.

Load Manager

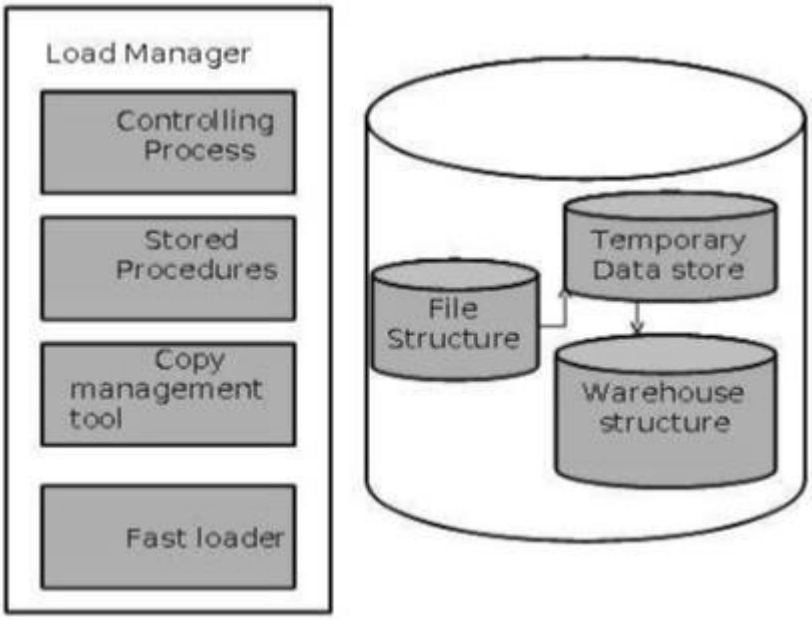
This component performs the operations required to extract and load process. The size and complexity of the load manager varies between specific solutions from one data warehouse to other.

Load Manager Architecture

The load manager performs the following functions −

- Extract the data from source system.

- Fast Load the extracted data into temporary data store.
- Perform simple transformations into structure similar to the one in the data warehouse.



Extract Data from Source

The data is extracted from the operational databases or the external information providers. Gateways is the application programs that are used to extract data. It is supported by underlying DBMS and allows client program to generate SQL to be executed at a server. Open Database Connection (ODBC), Java Database Connection (JDBC), are examples of gateway.

Fast Load

- In order to minimize the total load window, the data need to be loaded into the warehouse in the fastest possible time.
- The transformations affect the speed of data processing.
- It is more effective to load the data into relational database prior to applying transformations and checks.
- Gateway technology proves to be not suitable, since they tend not be performant when large data volumes are involved.

Simple Transformations

While loading it may be required to perform simple transformations. After this has been completed, we are in position to do the complex checks. Suppose we

are loading the EPOS sales transaction we need to perform the following checks:

- Strip out all the columns that are not required within the warehouse.
- Convert all the values to required data types.

Warehouse Manager

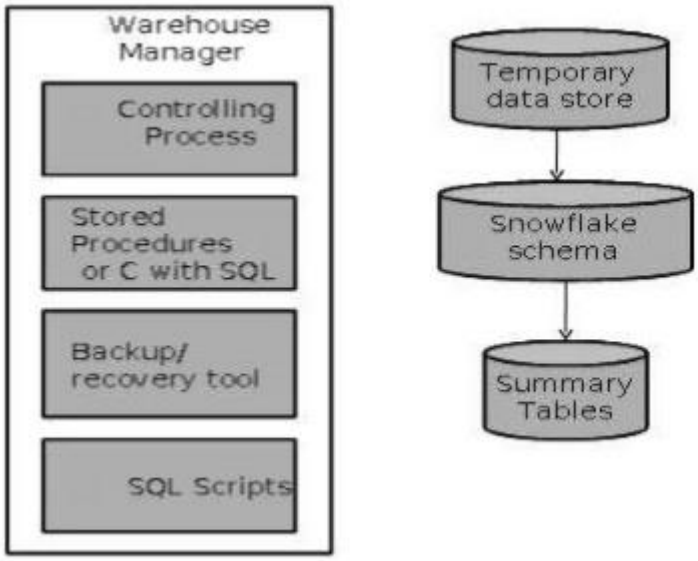
A warehouse manager is responsible for the warehouse management process. It consists of third-party system software, C programs, and shell scripts.

The size and complexity of warehouse managers varies between specific solutions.

Warehouse Manager Architecture

A warehouse manager includes the following −

- The controlling process
- Stored procedures or C with SQL
- Backup/Recovery tool
- SQL Scripts



Operations Performed by Warehouse Manager

- A warehouse manager analyses the data to perform consistency and referential integrity checks.
- Creates indexes, business views, partition views against the base data.
- Generates new aggregations and updates existing aggregations. Generates normalizations.

- Transforms and merges the source data into the published data warehouse.
- Backup the data in the data warehouse.
- Archives the data that has reached the end of its captured life.

Note − A warehouse Manager also analyses query profiles to determine index and aggregations are appropriate.

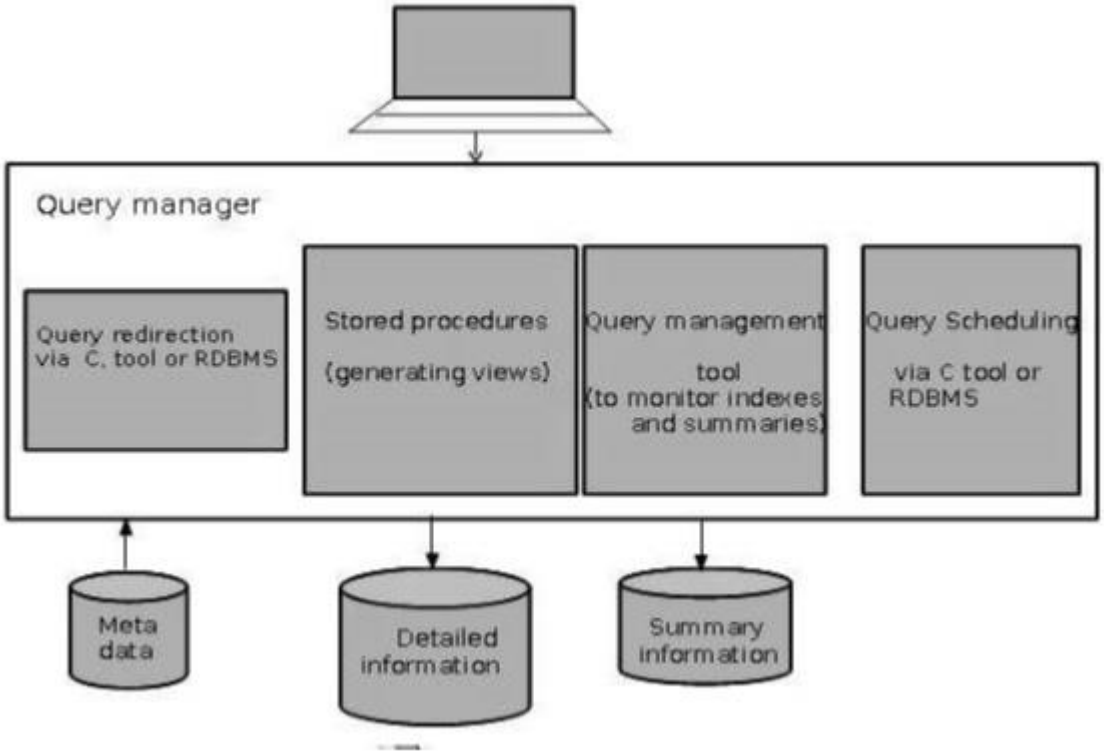
Query Manager

- Query manager is responsible for directing the queries to the suitable tables.
- By directing the queries to appropriate tables, the speed of querying and response generation can be increased.
- Query manager is responsible for scheduling the execution of the queries posed by the user.

Query Manager Architecture

The following screenshot shows the architecture of a query manager. It includes the following:

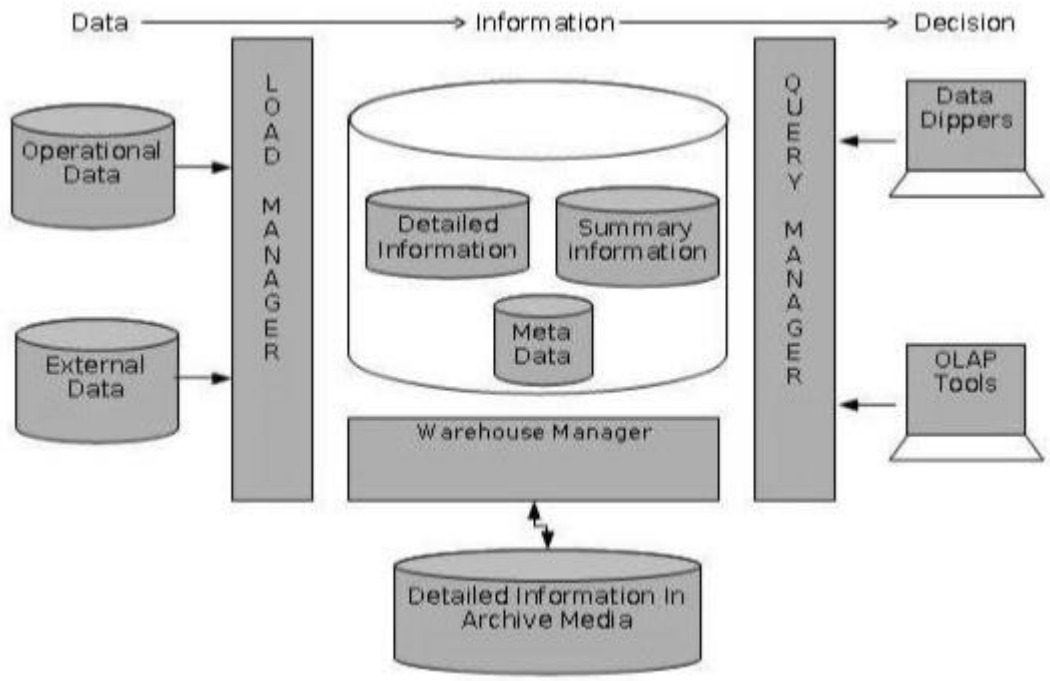
- Query redirection via C tool or RDBMS
- Stored procedures
- Query management tool
- Query scheduling via C tool or RDBMS
- Query scheduling via third-party software



Detailed Information

Detailed information is not kept online, rather it is aggregated to the next level of detail and then archived to tape. The detailed information part of data warehouse keeps the detailed information in the starflake schema. Detailed information is loaded into the data warehouse to supplement the aggregated data.

The following diagram shows a pictorial impression of where detailed information is stored and how it is used.



Note − If detailed information is held offline to minimize disk storage, we should make sure that the data has been extracted, cleaned up, and transformed into starflake schema before it is archived.

Summary Information

Summary Information is a part of data warehouse that stores predefined aggregations. These aggregations are generated by the warehouse manager. Summary Information must be treated as transient. It changes on-the-go in order to respond to the changing query profiles.

The points to note about summary information are as follows −

- Summary information speeds up the performance of common queries.
- It increases the operational cost.
- It needs to be updated whenever new data is loaded into the data warehouse.
- It may not have been backed up, since it can be generated fresh from the detailed information.

8.11 Data Warehousing - OLAP

Online Analytical Processing Server (OLAP) is based on the multidimensional data model. It allows managers, and analysts to get an insight of the

information through fast, consistent, and interactive access to information. We will discuss the different types of OLAP, operations on OLAP, difference between OLAP, and statistical databases and OLTP.

Types of OLAP Servers

We have four types of OLAP servers −

- Relational OLAP (ROLAP)
- Multidimensional OLAP (MOLAP)
- Hybrid OLAP (HOLAP)
- Specialized SQL Servers

8.11.1 Relational OLAP

ROLAP servers are placed between relational back-end server and client front-end tools. To store and manage warehouse data, ROLAP uses relational or extended-relational DBMS.

ROLAP includes the following −

- Implementation of aggregation navigation logic.
- Optimization for each DBMS back end.
- Additional tools and services.

8.11.2 Multidimensional OLAP

MOLAP uses array-based multidimensional storage engines for multidimensional views of data. With multidimensional data stores, the storage utilization may be low if the data set is sparse. Therefore, many MOLAP server use two levels of data storage representation to handle dense and sparse data sets.

8.11.3 Hybrid OLAP

Hybrid OLAP is a combination of both ROLAP and MOLAP. It offers higher scalability of ROLAP and faster computation of MOLAP. HOLAP servers allows to store the large data volumes of detailed information. The aggregations are stored separately in MOLAP store.

8.11.4 Specialized SQL Servers

Specialized SQL servers provide advanced query language and query processing support for SQL queries over star and snowflake schemas in a read-only environment.

OLAP Operations

Since OLAP servers are based on multidimensional view of data, we will discuss OLAP operations in multidimensional data.

Here is the list of OLAP operations −

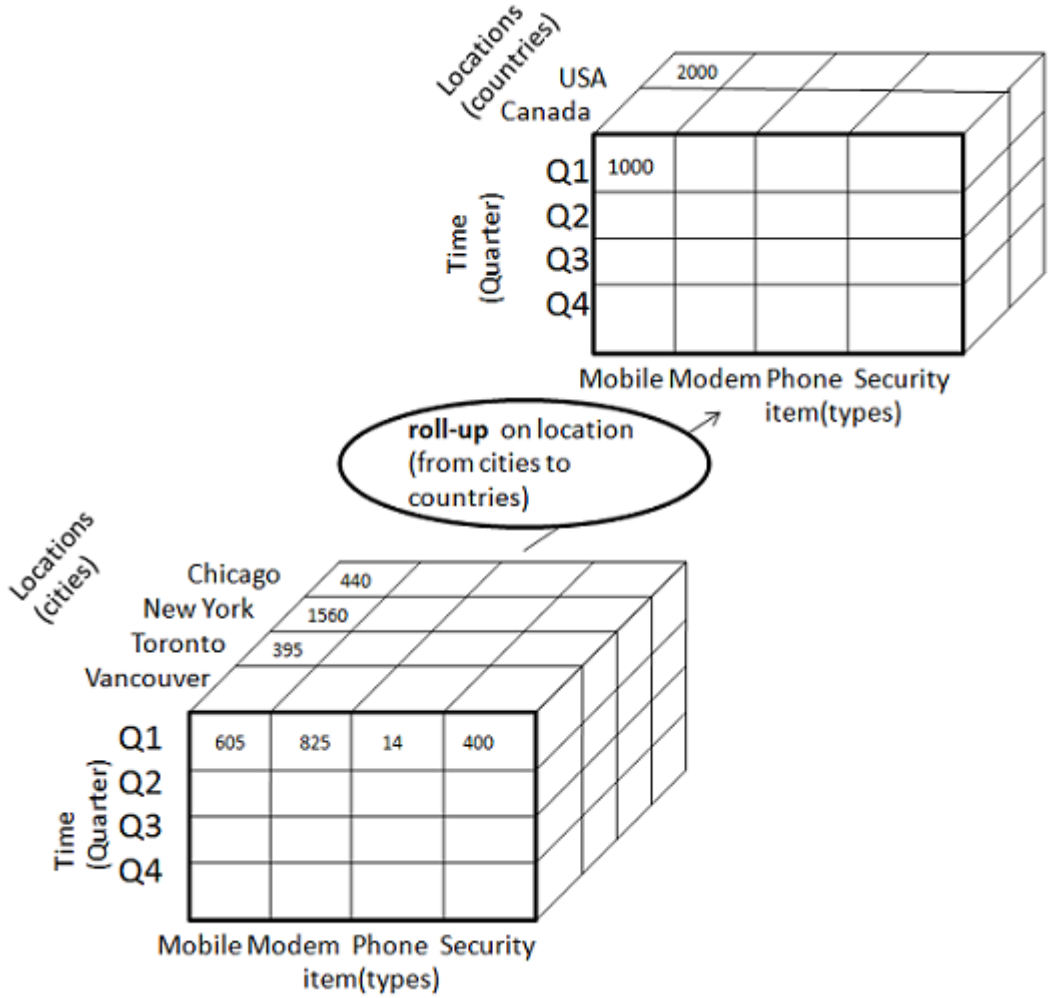
- Roll-up
- Drill-down
- Slice and dice
- Pivot (rotate)

i. Roll-up

Roll-up performs aggregation on a data cube in any of the following ways −

- By climbing up a concept hierarchy for a dimension
- By dimension reduction

The following diagram illustrates how roll-up works.



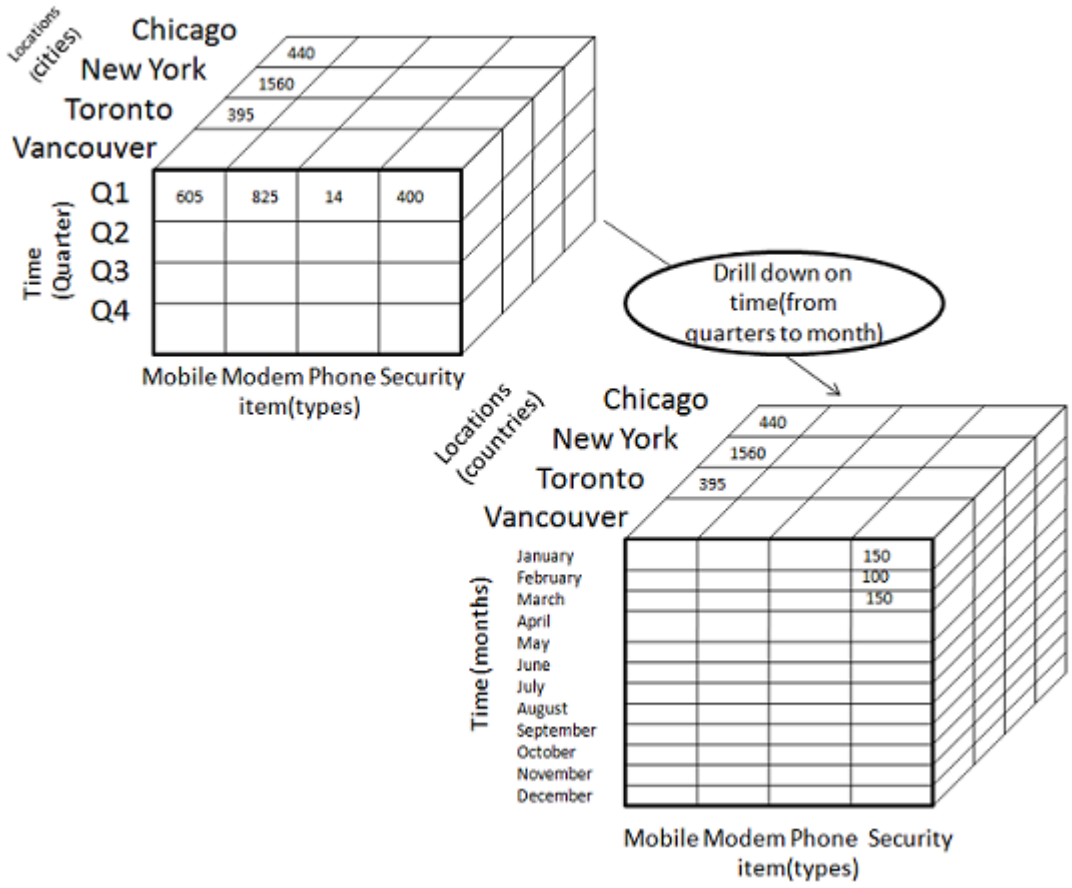
- Roll-up is performed by climbing up a concept hierarchy for the dimension location.
- Initially the concept hierarchy was "street < city < province < country".
- On rolling up, the data is aggregated by ascending the location hierarchy from the level of city to the level of country.
- The data is grouped into cities rather than countries.
- When roll-up is performed, one or more dimensions from the data cube are removed.

ii. Drill-down

Drill-down is the reverse operation of roll-up. It is performed by either of the following ways −

- By stepping down a concept hierarchy for a dimension
- By introducing a new dimension.

The following diagram illustrates how drill-down works −

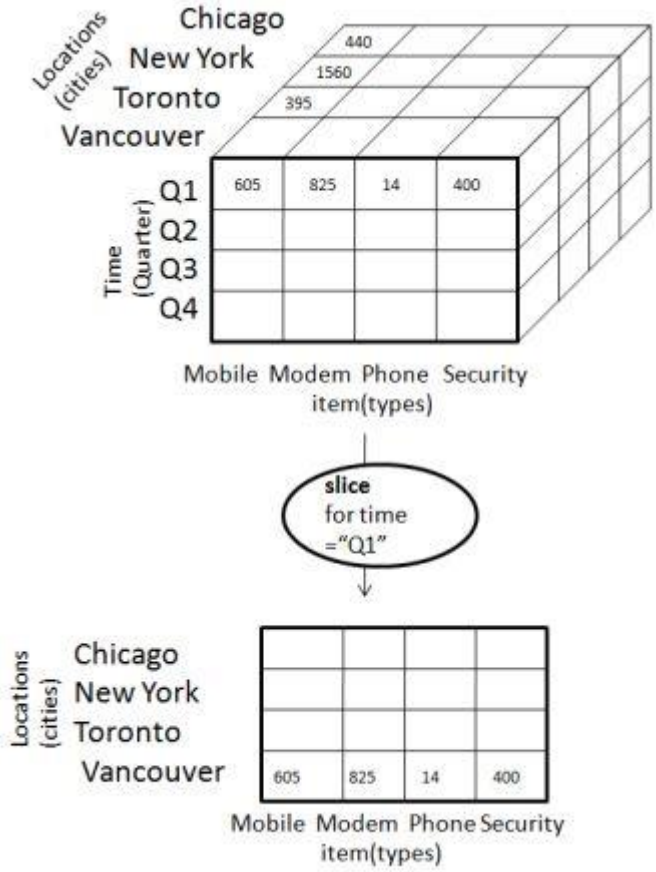


- Drill-down is performed by stepping down a concept hierarchy for the dimension time.

- Initially the concept hierarchy was "day < month < quarter < year."
- On drilling down, the time dimension is descended from the level of quarter to the level of month.
- When drill-down is performed, one or more dimensions from the data cube are added.
- It navigates the data from less detailed data to highly detailed data.

iii. Slice

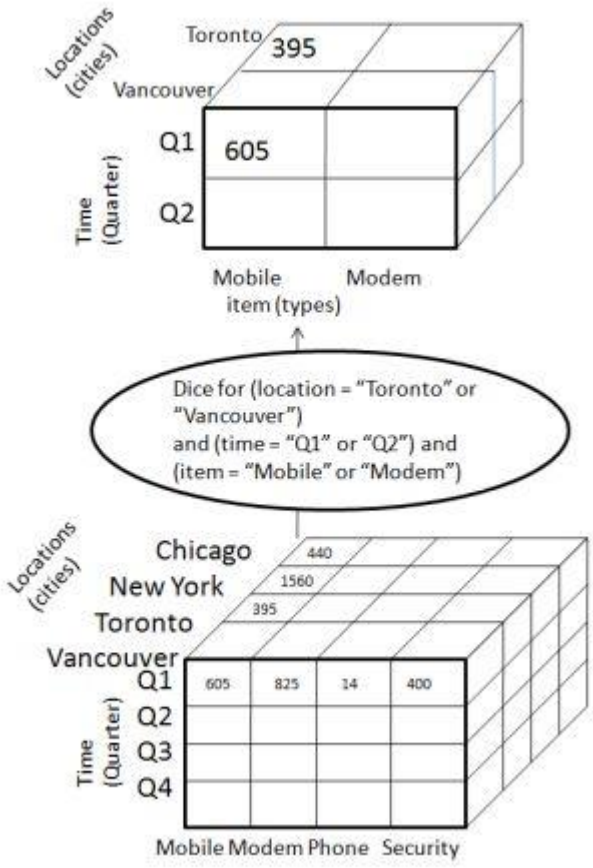
The slice operation selects one particular dimension from a given cube and provides a new sub-cube. Consider the following diagram that shows how slice works.



- Here Slice is performed for the dimension "time" using the criterion time = "Q1".
- It will form a new sub-cube by selecting one or more dimensions.

iv. Dice

Dice selects two or more dimensions from a given cube and provides a new sub-cube. Consider the following diagram that shows the dice operation.

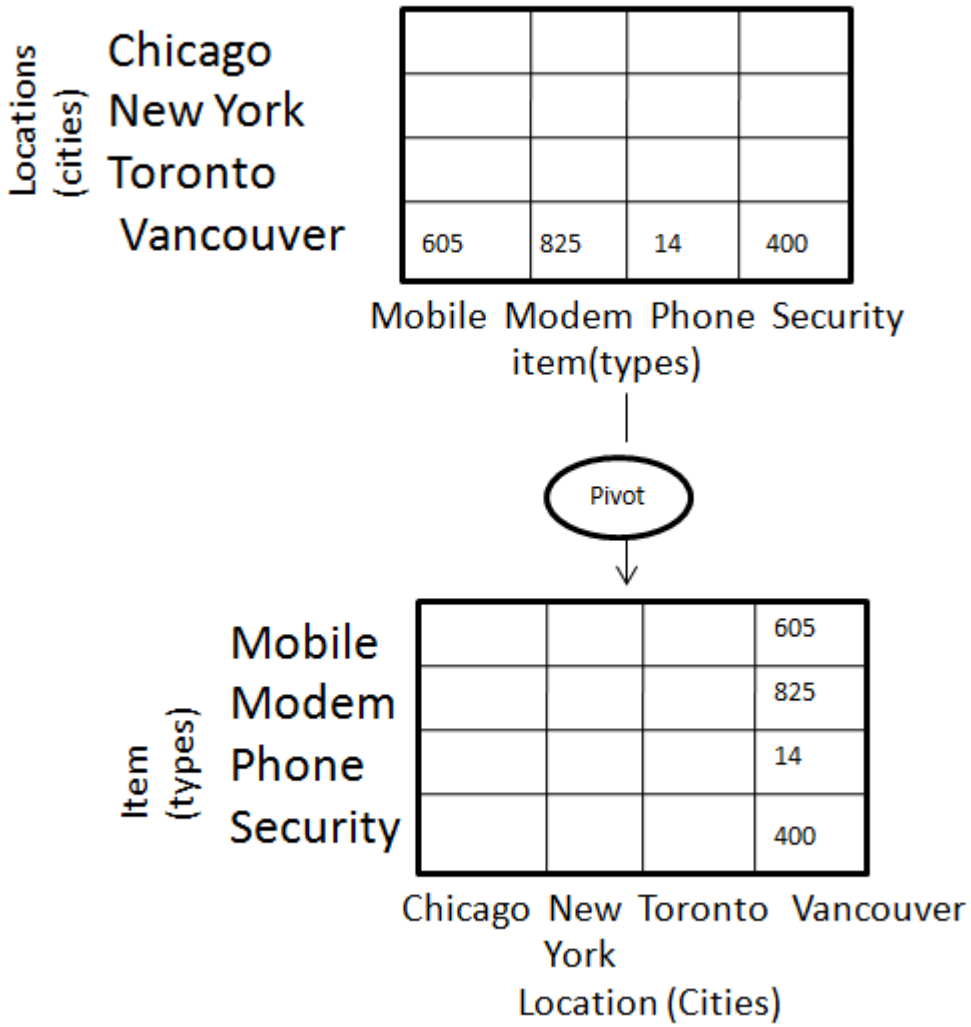


The dice operation on the cube based on the following selection criteria involves three dimensions.

- (location = "Toronto" or "Vancouver")
- (time = "Q1" or "Q2")
- (item =" Mobile" or "Modem")

v. Pivot

The pivot operation is also known as rotation. It rotates the data axes in view in order to provide an alternative presentation of data. Consider the following diagram that shows the pivot operation.



OLAP vs OLTP

Sr.No.	Data Warehouse (OLAP)	Operational Database (OLTP)
1	Involves historical processing of information.	Involves day-to-day processing.
2	OLAP systems are used by knowledge workers such as executives, managers and analysts.	OLTP systems are used by clerks, DBAs, or database professionals.
3	Useful in analyzing the business.	Useful in running the business.
4	It focuses on Information out.	It focuses on Data in.

5	Based on Star Schema, Snowflake, Schema and Fact Constellation Schema.	Based on Entity Relationship Model.
6	Contains historical data.	Contains current data.
7	Provides summarized and consolidated data.	Provides primitive and highly detailed data.
8	Provides summarized and multidimensional view of data.	Provides detailed and flat relational view of data.
9	Number or users is in hundreds.	Number of users is in thousands.
10	Number of records accessed is in millions.	Number of records accessed is in tens.
11	Database size is from 100 GB to 1 TB	Database size is from 100 MB to 1 GB.
12	Highly flexible.	Provides high performance.

Data Warehousing - Relational OLAP

Relational OLAP servers are placed between relational back-end server and client front-end tools. To store and manage the warehouse data, the relational OLAP uses relational or extended-relational DBMS.

ROLAP includes the following −

- Implementation of aggregation navigation logic
- Optimization for each DBMS back-end
- Additional tools and services

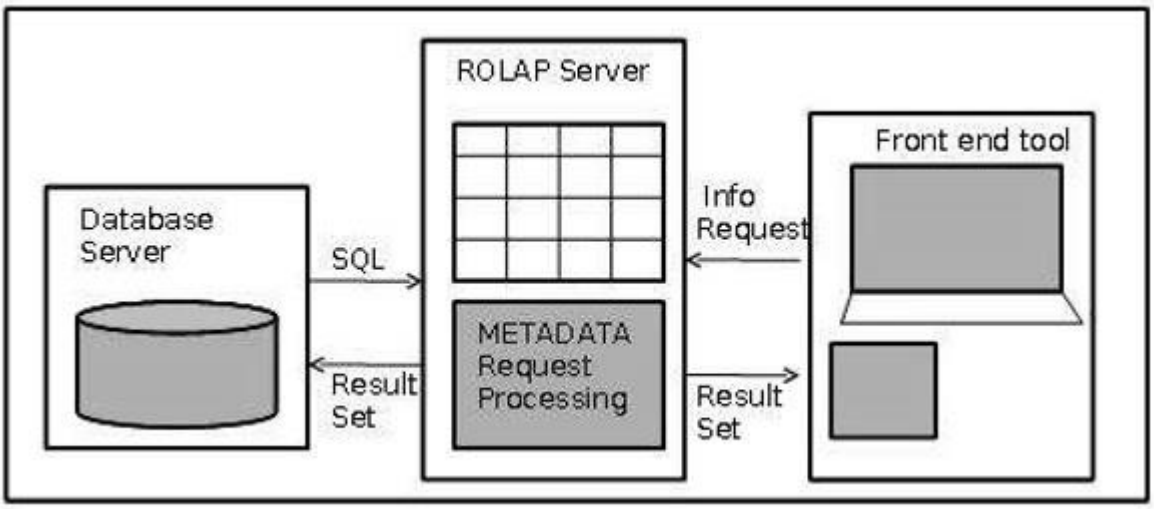
Points to Remember

- ROLAP servers are highly scalable.
- ROLAP tools analyze large volumes of data across multiple dimensions.
- ROLAP tools store and analyze highly volatile and changeable data.

Relational OLAP Architecture

ROLAP includes the following components −

- Database server
- ROLAP server
- Front-end tool.



Advantages

- ROLAP servers can be easily used with existing RDBMS.
- Data can be stored efficiently, since no zero facts can be stored.
- ROLAP tools do not use pre-calculated data cubes.
- DSS server of micro-strategy adopts the ROLAP approach.

Disadvantages

- Poor query performance.
- Some limitations of scalability depending on the technology architecture that is utilized.

Data Warehousing - Multidimensional OLAP

Multidimensional OLAP (MOLAP) uses array-based multidimensional storage engines for multidimensional views of data. With multidimensional data stores, the storage utilization may be low if the dataset is sparse. Therefore, many MOLAP servers use two levels of data storage representation to handle dense and sparse datasets.

Points to Remember –

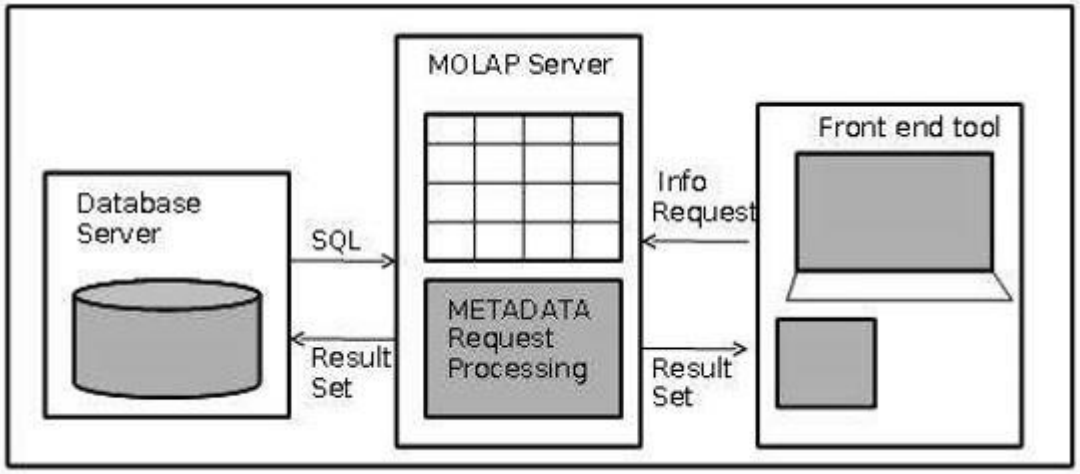
- MOLAP tools process information with consistent response time regardless of level of summarizing or calculations selected.
- MOLAP tools need to avoid many of the complexities of creating a relational database to store data for analysis.
- MOLAP tools need fastest possible performance.

- MOLAP server adopts two level of storage representation to handle dense and sparse data sets.
- Denser sub-cubes are identified and stored as array structure.
- Sparse sub-cubes employ compression technology.

MOLAP Architecture

MOLAP includes the following components −

- Database server.
- MOLAP server.
- Front-end tool.



Advantages

- MOLAP allows fastest indexing to the pre-computed summarized data.
- Helps the users connected to a network who need to analyze larger, less-defined data.
- Easier to use, therefore MOLAP is suitable for inexperienced users.

Disadvantages

- MOLAP are not capable of containing detailed data.
- The storage utilization may be low if the data set is sparse.

MOLAP vs ROLAP

Sr.No.	MOLAP	ROLAP
1	Information retrieval is fast.	Information retrieval is comparatively slow.

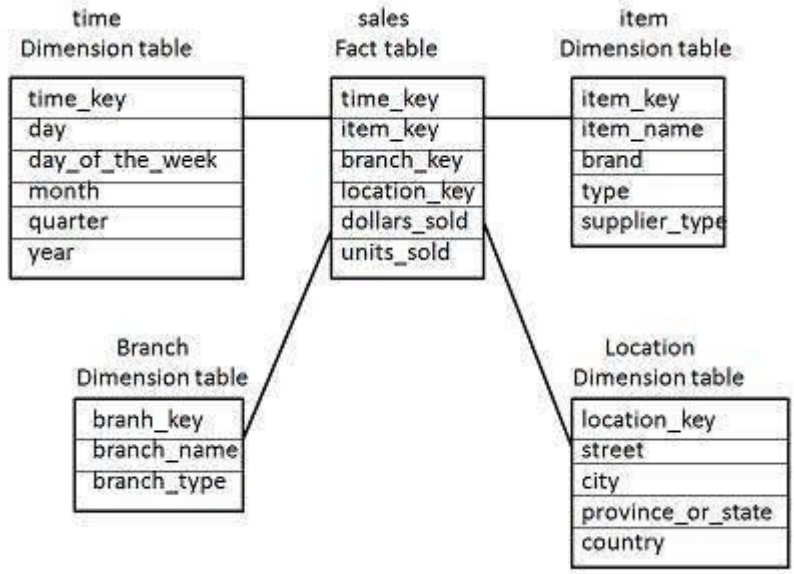
2	Uses sparse array to store data-sets.	Uses relational table.
3	MOLAP is best suited for inexperienced users, since it is very easy to use.	ROLAP is best suited for experienced users.
4	Maintains a separate database for data cubes.	It may not require space other than available in the Data warehouse.
5	DBMS facility is weak.	DBMS facility is strong.

Data Warehousing - Schemas

Schema is a logical description of the entire database. It includes the name and description of records of all record types including all associated data-items and aggregates. Much like a database, a data warehouse also requires to maintain a schema. A database uses relational model, while a data warehouse uses Star, Snowflake, and Fact Constellation schema.

i. Star Schema

- Each dimension in a star schema is represented with only one-dimension table.
- This dimension table contains the set of attributes.
- The following diagram shows the sales data of a company with respect to the four dimensions, namely time, item, branch, and location.

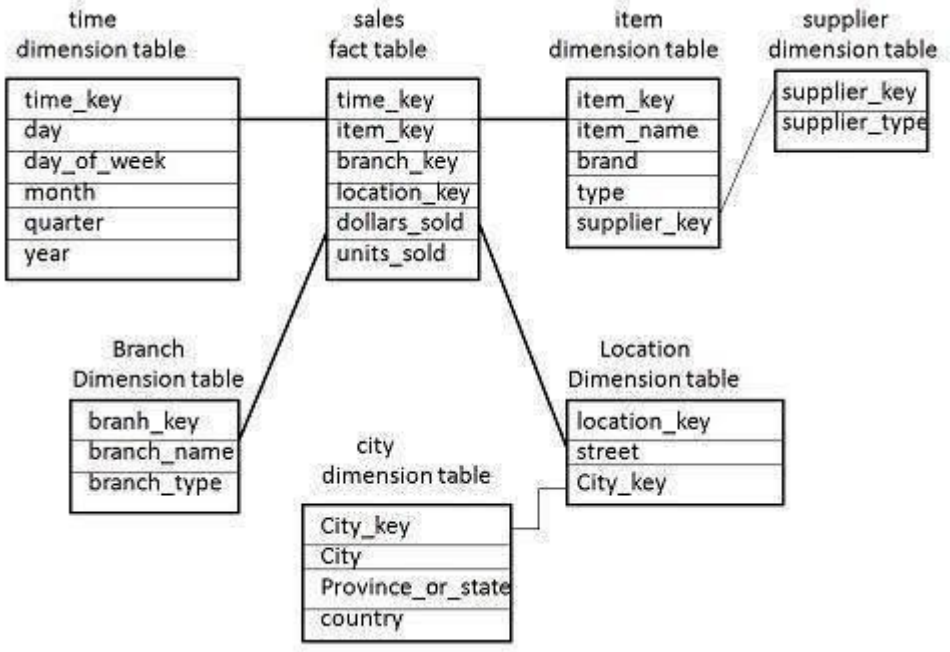


- There is a fact table at the center. It contains the keys to each of four dimensions.
- The fact table also contains the attributes, namely dollars sold and units sold.

Note − Each dimension has only one dimension table and each table holds a set of attributes. For example, the location dimension table contains the attribute set {location_key, street, city, province_or_state,country}. This constraint may cause data redundancy. For example, "Vancouver" and "Victoria" both the cities are in the Canadian province of British Columbia. The entries for such cities may cause data redundancy along the attributes province_or_state and country.

ii. Snowflake Schema

- Some dimension tables in the Snowflake schema are normalized.
- The normalization splits up the data into additional tables.
- Unlike Star schema, the dimensions table in a snowflake schema are normalized. For example, the item dimension table in star schema is normalized and split into two dimension tables, namely item and supplier table.



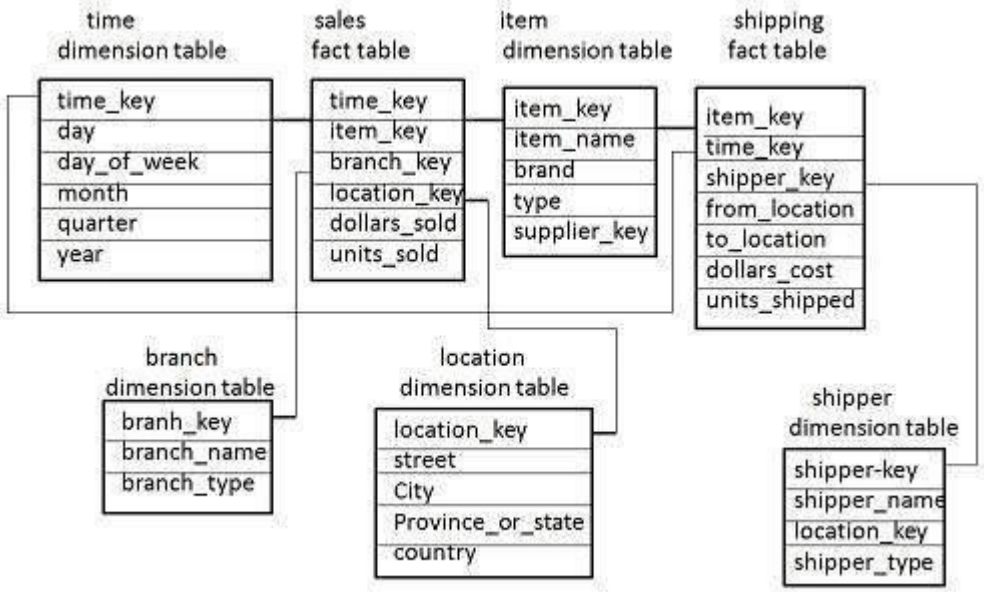
- Now the item dimension table contains the attributes item_key, item_name, type, brand, and supplier-key.

- The supplier key is linked to the supplier dimension table. The supplier dimension table contains the attributes supplier_key and supplier_type.

Note – Due to normalization in the Snowflake schema, the redundancy is reduced and therefore, it becomes easy to maintain and the save storage space.

iii. Fact Constellation Schema

- A fact constellation has multiple fact tables. It is also known as galaxy schema.
- The following diagram shows two fact tables, namely sales and shipping.



- The sales fact table is same as that in the star schema.
- The shipping fact table has the five dimensions, namely item_key, time_key, shipper_key, from_location, to_location.
- The shipping fact table also contains two measures, namely dollars sold and units sold.
- It is also possible to share dimension tables between fact tables. For example, time, item, and location dimension tables are shared between the sales and shipping fact table.

8.12 Summary

In this lecture session we defined we learnt about data warehousing particularly the motivation towards data warehousing, benefits of data

warehousing, architecture of data warehousing, types of data in a data warehouse and information flow, the problems of data warehousing, data warehouse design, data marts, data analysis tools and finally the various types of OLAP servers. I hope you enjoyed the class. Wishing you a great week ahead.

8.13 Student Activity

i. Define the term data warehouse.
ii. Briefly discuss the features of a data warehousing.
iii. What does ETL stand for in data warehousing? Explain each process.
iv. Enumerate the main differences between OLAP and OLTP.
v. How does a Data mart differ from a data warehouse?
vi. Using examples, discuss the main operations of OLAP servers.

8.14 Reference Materials

Core Books

i. Coronel, C., & Morris, S. (2017). *Database Systems: Design, Implementation, & Management* (12th ed.). Boston, MA: Cengage Learning. ISBN: 1305627482.
ii. Elmasri, R., & Navathe, S. B. (2017). *Fundamentals of Database Systems* (7th ed.). Hoboken, NJ: Pearson Education Ltd. ISBN: 0133970779.
iii. Connolly, T. M., & Begg, C. E. (2015). *Database Systems: A Practical Approach to Design, Implementation, and Management* (6th ed.). Boston, MA: Pearson Education Ltd. ISBN: 0132943263.

Core Journals

i. Journal of Database Management. ISSN: 1063-8016.
ii. Database Management & Information Retrieval. ISSN: 1862-5347.
iii. International Journal of Information Technology and Database Systems. ISSN: 2231-1807.

Recommended Text Books

i. Hernandez, M. J. (2013). *Database Design for Mere Mortals: A Hands-On Guide to Relational Database Design* (3rd ed.). Harlow, UK: Addison-Wesley. ISBN: 0321884493.

ii. Rankins, R., Bertucci, P., Gallelli, C., & Silverstein, A. (2015). *Microsoft SQL Server 2014 Unleashed*. Indianapolis, IN: Sams Publishing. ISBN: 0672337290.

iii. Comeau, A. (2016). *MySQL Explained: Your Step By Step Guide to Database Design*. Bradenton, FL: OSTraining. ISBN: 151942437X.

Recommended Journals

i. International Journal of Intelligent Information and Database Systems. ISSN: 1751-5858.

ii. Database Systems Journal. ISSN: 2069-3230.

iii. Distributed and Parallel Databases. ISSN: 0926-8782.

iv. International Journal of Database Management Systems. ISSN: 0975 - 5985.

Journal of Database Manageme