UNIT-2

What is Data Warehousing –

- ➔ It is a repository of data.
- ➔ It serves s a physical implementation of decision support data model.
- ➔ It stores information which is used to make decisions.
- ➔ A data warehouse is a relational database that is designed for query and analysis but not for transaction processing.
- ➔ The construction of Data Warehouse involves data cleaning and data integration.
- ➔ A data warehouse environment involves the process of- extraction, transformation, loading solution, OLAP engine, client analysis tools and other applications that manage the process of gathering data and delivering it to business users.

According to W.H.Inmon

> "Data Warehouse is a - subject-oriented,
> - integrated,
> - time variant,
> - non volatile,
> Collection of data which help to manage decision making process."

Subject oriented –

1. A data warehouse is focused on modeling and analysis of data for decision making.
2. The data warehouse is organized around some subjects of data such as customer, product & sales.
3. That's why data warehouse provides a view on subject which is useful for taking decisions.

Integrated –

1. A data warehouse is constructed by integrating multiple sources such as relational databases, flat files and online transaction records.
2. Data cleaning and data integration techniques are applied to ensure consistency in naming conflicts, attributes, and attribute measures and so on.

Time Variant–

1. The data which is stored in data warehouse is to provide information from a historical perspective.
2. To discover trends or any updates in business, we need large amount of data which changes from time to time.

Non Volatile–

1. Data once entered into data warehouse, it should not change which means data warehouse does not need to do transaction processing, recovery & concurrency control mechanisms.

Difference between OLTP & OLAP

OLTP

- It is an **operational database** systems.

- It performs online transactions & query processing.

- It performs operations like purchasing, banking, registration, inventory, and manufacturing.

OLAP

-These are **data warehouse** systems.

- It helps for "knowledge workers" to perform data analysis and decision making.

The main feature Differences between OLTP & OLAP:

Features	OLTP	OLAP
1. System Orientation	It is Customer Oriented	It is Market Oriented
2. Users	Users are clerks, clients & IT professionals.	Users are knowledge workers like Executives, managers & analysts.
3. Used for	Transaction & Query Processing	Data Analysis
4. Data contents	It manages current data which is used for decision making.	It manages historical data & by providing facilities for summarization, aggregation of data, can take decisions.
5. Data base Design	It uses ER models which is an application oriented database design.	It uses star, snowflake models which is a subject oriented design.

6. Views	This system focuses mainly on current data within an enterprise.	This system focuses on historical data which occurred due to evolution process of organization.
7. Access Patterns	It performs short atomic transactions which require concurrency control & recovery mechanisms.	It performs read only operations like complex queries.

A multi dimensional data model

Data warehouses and OLAP tools are designed based on a multi dimensional data model.

Terms to be known here- Data cube, Dimension table, Fact table

Data cube -A data cube allows data to be modeled & viewed in multiple dimensions.

-It is defined by dimensions & facts. (dimensions means attributes)

Dimension table

- It holds set of dimensions/attributes like sales of items, branches & locations.
- For example a dimension table for item contains attribute like item name, brand and type.

Fact table

-It holds keys of each dimension tables and also some attributes.

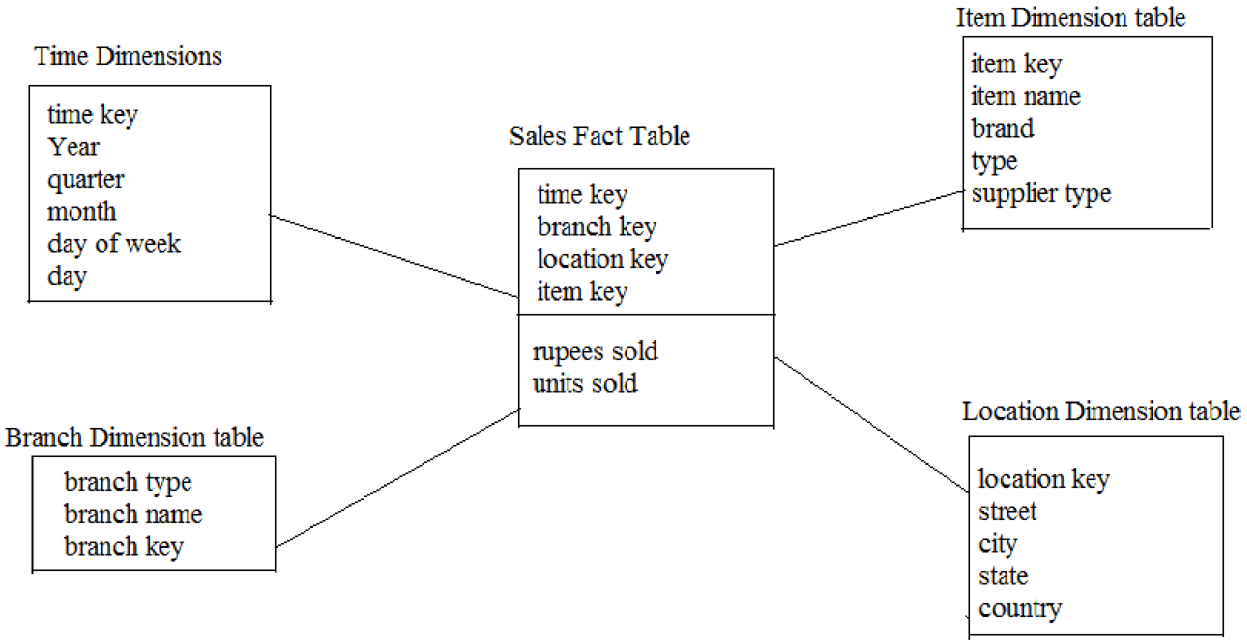
A Data warehouse can be modeled in the form of

-Star schema, Snowflake schema, Fact constellation schema.

Star Schema

It is with dimension tables and fact table and here fact table is located at the center.

Both fact and dimension tables are with set of attributes.



Multidimensional schema is defined using Data Mining Query Language (DMQL).

cube definition and dimension definition, can be used for defining the data warehouses.

Syntax for Cube Definition

define cube < cube_name > [< dimension-list > }: < measure_list >

Syntax for Dimension Definition

define dimension < dimension_name > as (< attribute_or_dimension_list >)

Star schema definition in DMQL

define cube sales star [time, item, branch, location]:

rupees sold = sum(total sales), units sold = count(*)

define dimension time as (time key, day, day of week, month, quarter, year)

define dimension item as (item key, item name, brand, type, supplier type)

define dimension branch as (branch key, branch name, branch type)

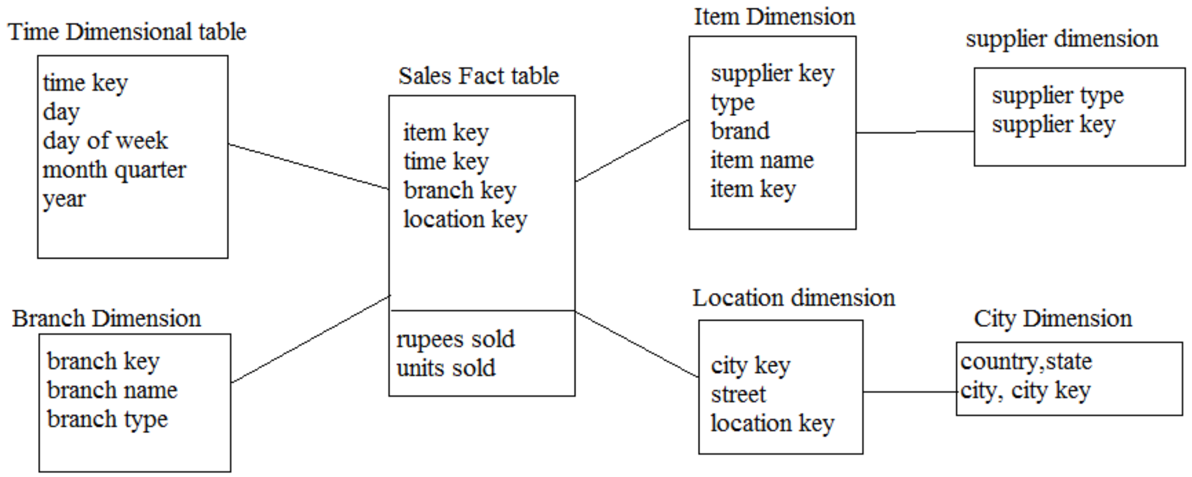
define dimension location as (location key, street, city, state, country)

Snowflake schema

In this the dimension tables are normalized.

So that the normalization splits up the data into additional tables.

Due to normalization the redundancy is reduced and therefore, it is easy to maintain and the save storage space.



Snowflake schema definition in DMQL

define cube sales snowflake [time, item, branch, location]:

rupees sold = sum(sales in rupees), units sold = count(*)

define dimension time as (time key, day, day of week, month, quarter, year)

define dimension item as (item key, item name, brand, type, supplier (supplier key, supplier type))

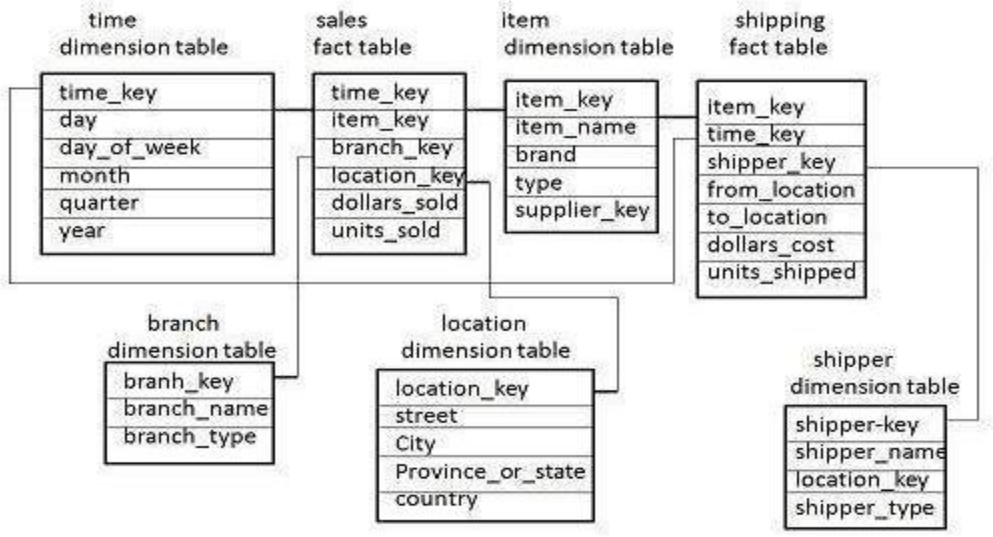
define dimension branch as (branch key, branch name, branch type)

define dimension location as (location key, street, city (city key, city, province or state, country))

Fact Constellation Schema

A fact constellation has multiple fact tables. It is also known as galaxy schema.

It is also possible to share dimension tables between fact tables.



Fact Constellation Schema definition in DMQL

define cube sales [time, item, branch, location]:

rupees sold = sum(sales in rupees), units sold = count(*)

define dimension time as (time key, day, day of week, month, quarter, year)

define dimension item as (item key, item name, brand, type, supplier type)

define dimension branch as (branch key, branch name, branch type)

define dimension location as (location key, street, city, state, country)

define cube shipping [time, item, shipper, from location, to location]:

rupees cost = sum(cost in rupees), units shipped = count(*)

define dimension time as time in cube sales

define dimension item as item in cube sales

define dimension shipper as (shipper key, shipper name, location as location in cube sales, shipper type)

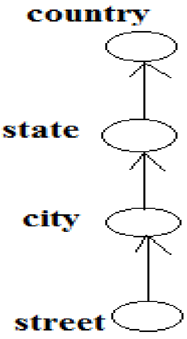
define dimension from location as location in cube sales

define dimension to location as location in cube sales

Concept Hierarchy-

Mapping low level concepts to high level concepts is said to be concept hierarchy.

This hierarchy is common in most of the applications in data mining. For example hierarchy of a location dimensions is as follows.



OLAP Operations in multidimensional data

OLAP servers are based on multidimensional view of data, here is the list of OLAP operations:

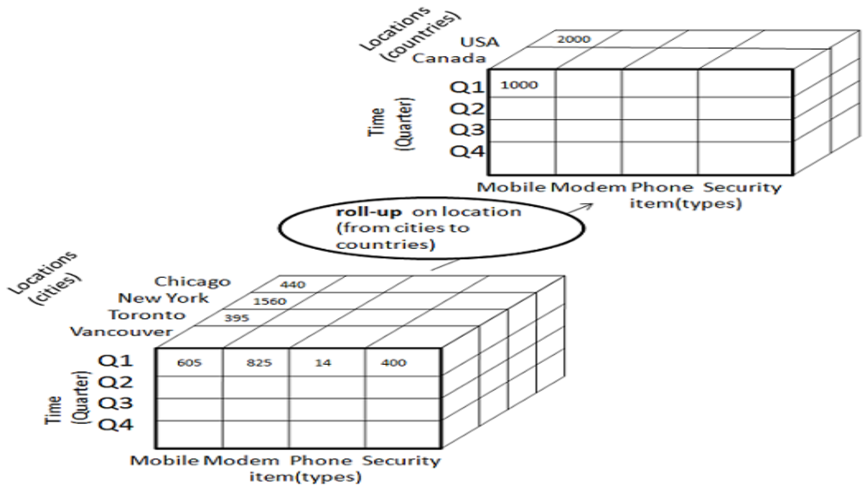
- Roll-up
- Drill-down
- Slice and dice
- Pivot (rotate)

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.

On rolling up, the data is aggregated by ascending the location hierarchy from the level of city to the level of country.

When roll-up is performed, one or more dimensions from the data cube are removed.

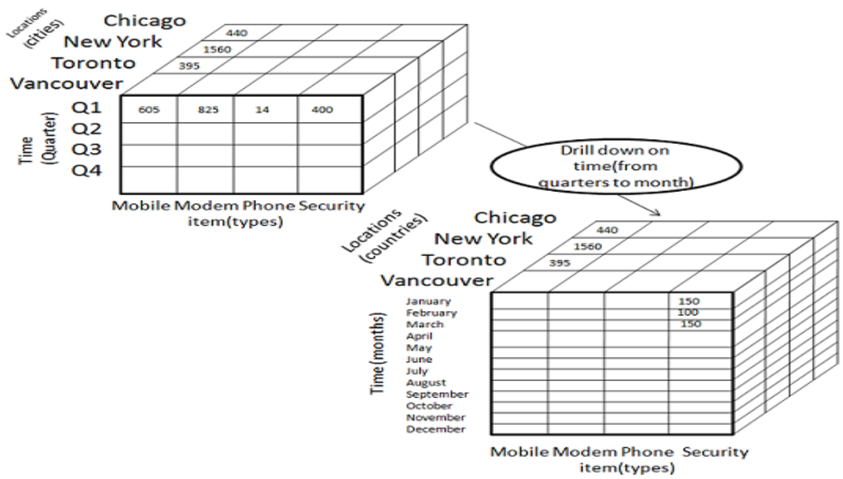
Drill down

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:



When drill-down is performed, one or more dimensions from the data cube are added.

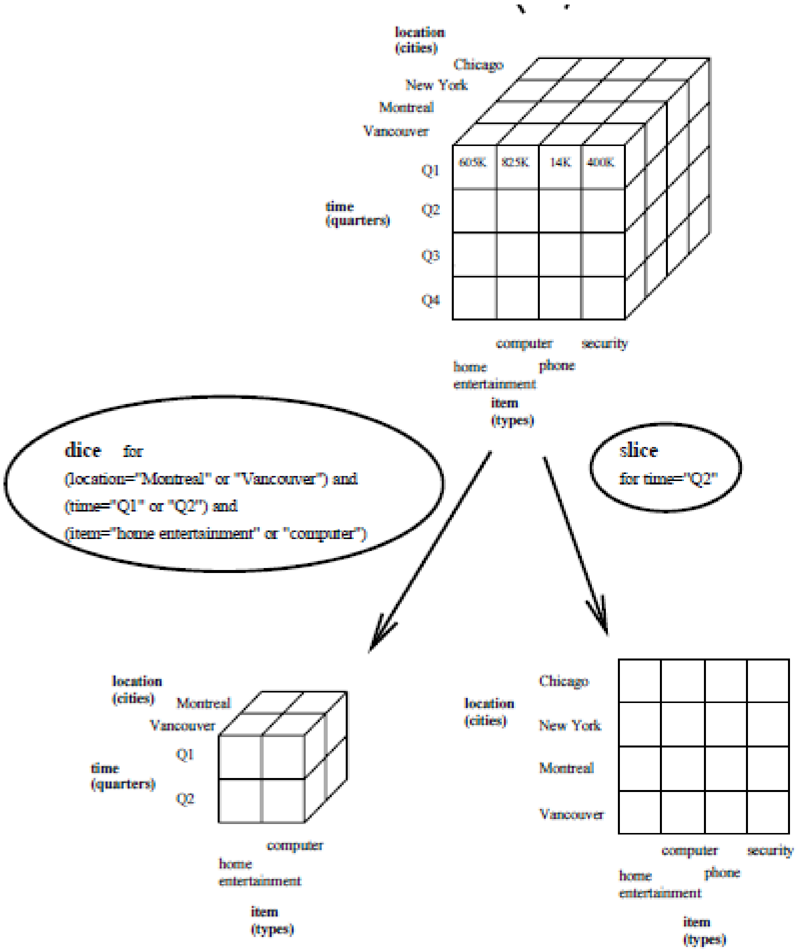
It navigates the data from less detailed data to highly detailed data.

Slice and Dice

The slice operation selects **one particular dimension** from a given cube and provides a new sub-cube.

The 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 illustrated in below figure.

Dice selects two or **more dimensions** from a given cube and provides a new sub-cube.

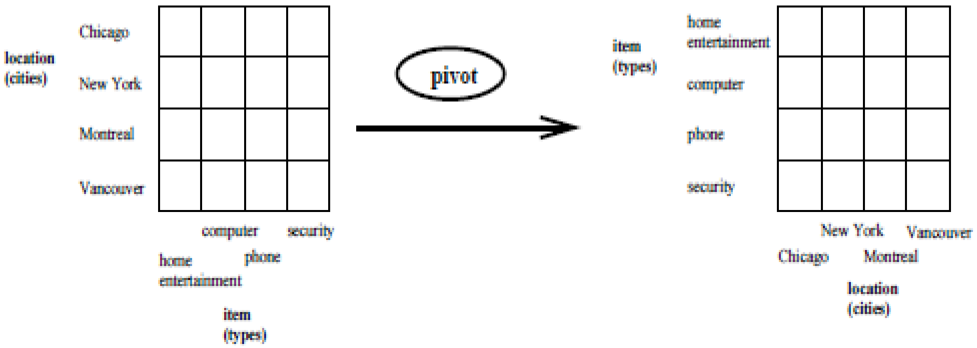


Pivot

The pivot operation is also known as rotation.

It rotates the data axes in view in order to provide an alternative presentation of data.

The figure below illustrates it-



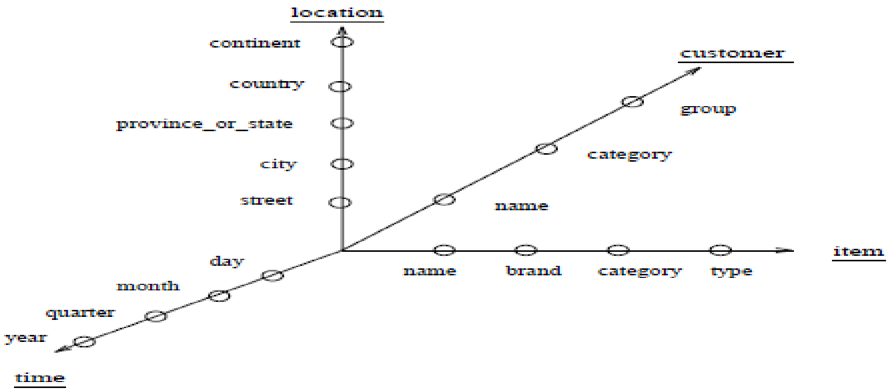
A Starnet model-

To query multidimensional databases it can be based on a starnet model.

A starnet model consists of radial lines from a central point where each line represents a concept hierarchy for dimensions.

Each abstract level in hierarchy is called a footprint.

The figure below is a starnet model with dimensions.



Data warehouse Architecture-

To design an effective data warehouse we have to understand and analyze business needs and then construct a business analysis framework.

The construction of a data warehouse can be viewed in 4 different ways

1. Top down view- In this view we need to see whether it is possible to allows selection of relevant information necessary for the data warehouse.
2. Data Source view- In this view we need to see whether it presents the information being captured, stored & managed by the operational system.
3. Data warehouse view- In this view we need to see whether it includes fact tables and dimension tables that which is required for historical context.
4. Business query view- In this view we need to see the perspective of data in data warehouse from end user view point.

The process of data warehouse design consists the below steps-

1. We need to choose a business process model-

    If this process is organizational→ then the design should focuses on complex object collections (i.e., Data warehouse model will be chosen)

    If this process is departmental→ then the design should focuses on analysis (i.e., Data mart model will be chosen)
2. We need to choose the grain of the business process-

    Grain is the fundamental, atomic level of data which is represented in fact table.
3. We need to choose the dimensions that are apply to each fact table record-

    Dimensions means attributes.
4. We need to choose measures that populate to each fact tables-

    Measures mean numeric additive quantities in a table they are like units sold, rupees sold.

Since, Constructing a Data warehouse is a difficult / long term task

↓

Goal of data warehouse implementation should be specific, achievable, and measurable.

↓

Once designed & constructed then go for deployment includes installation, training & orientation.

↓

And then platform maintenance to be considered.

↓

Data warehouse administration includes data refreshment, planning for disaster recovery, manages access control, security, managing data growth, managing database performance.

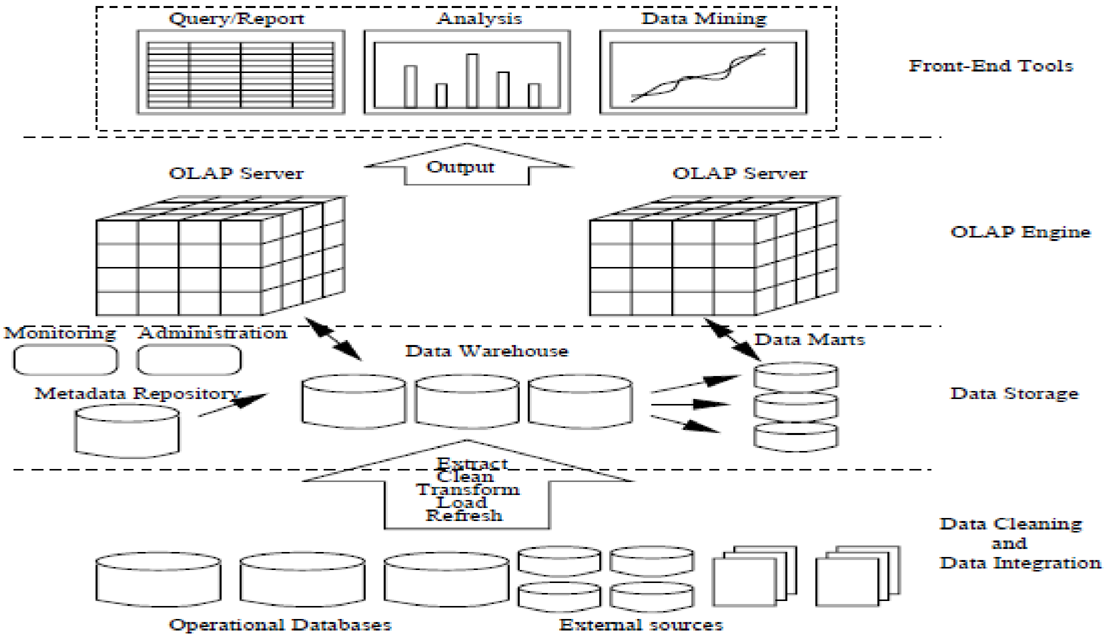
↓

Scope management deals with controlling number of queries, limiting size of data warehouse.

There exists several design tools, such as

Data warehouse development tool And Planning & Analysis tool.

Three tier data warehouse architecture.



Bottom tier - It is a warehouse DB server which is a relational Database system.

Middle tier - It is a OLAP server.

-It is implemented using relational OLAP(ROLAP) it deals with multidimensional data to standard relational operations.

Top tier - It is a client layer.

-It holds query tools, analysis tools, data mining tools.

From the architecture point of view there are three data warehouse models-

1. Enterprise warehouse-
    - Collects information about subjects related to an organization.
    - It provides enterprise wide data integration.
    - It contains detailed data within a range of hundreds of gigabytes to hundreds of terabytes.
    - It will take years to design.
2. Data mart-
    - It contains subset of enterprise wide data.
    - This data can be related to specific groups.
    - This data is related to a department of an enterprise.
    - Ex- market data contains items, customers, sales.
3. Virtual Warehouse-
    - Set of views over operational databases.
    - Easy to build but takes more capacity for efficient query processing.

OLAP server architectures:

For OLAP process implementation of data warehouse, server engine includes-

Relational OLAP (ROLAP) servers

- These servers are placed between relational back end and client front end tools.

- These servers use relational DBMS to store and manage data warehouse.

Multidimensional OLAP (MOLAP) servers

- These servers use array based multidimensional storage engines.

- These servers can store both dense data (time series) and sparse data (document).

Hybrid OLAP (HOLAP) servers

- Here, it combines both ROLAP & MOLAP technology.

- From ROLAP it takes benefit of scalability and from MOLAP it takes benefit of fast computation.

Specialized SQL servers

- It provides advanced query language and query processing to support SQL queries over star & snowflake schemas.

SQL can be extended to support OLAP operations

The following are the operations to be extended in SQL for OLAP operations

1. By extending aggregate functions-
   -SQL has sum(), avg(), count(), min(), max() operations
   -OLAP queries answering requires Extension of SQL has rank(), median(), mode() operations.
2. By adding reporting features-
   -These are like report writing software's which are used to evaluate.
   -Examples cumulative totals, moving averages, break points etc, which are useful in decision support systems.
3. By implementing multiple group by's feature-
   -By grouping set of attribute using "group-by" the search strategy over a period of time will be efficient. For example we can extract data related totals sales occurred over a period 2012-2014 using this "group by".
   -Because through SQL statements it can be done in an inefficient way.

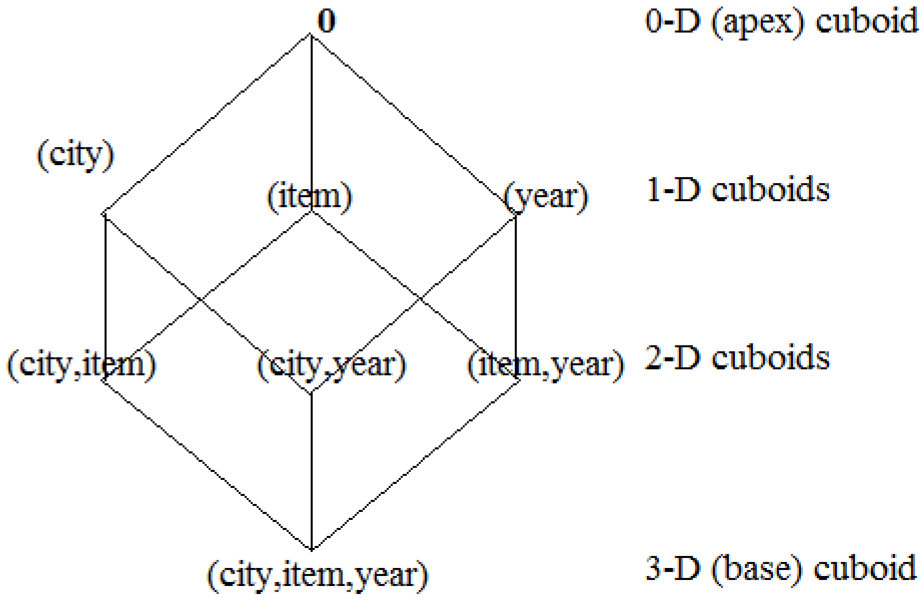


Figure represents lattice (partial order of dimensions) of cuboids

Data warehouse Implementation:

Data warehouse can be implemented efficiently using-

1. Computation techniques,
2. Access methods,
3. Query processing techniques.

Computation of data cubes-

The compute cube operator implementation-

-Aggregation performed on multiple data sets referred to as group-by in SQL.
-To define cube aggregation or grouping different dimensions Compute cube operator is used.

Syntax- compute cube cube_name
Ex- compute cube sum of sales group by item and city.

An SQL query containing no group by then it is a zero dimensional operation.
An SQL query containing one group by then it is a one dimensional operation.
An SQL query containing n group by then it is a n dimensional operation performed by a group by operator.

For a cube with n dimensions then there exists a total of 2^n cuboids including base cuboid.

For detailed explanation refer pageno-25, example 2.11 Data Mining Concepts and Techniques Jiawei Han and Micheline Kamber 1st edition

To pre compute and materialize all of the cuboids which are in data cube is a unrealistic (not an easy task). So by partial materialization i.e., materializing only some of the possible cuboids is possible.

There are 3 choices for data materialization:

1. Pre compute only the base cuboid and none of the remaining cuboids is said to be **no materialization**. It computes expensive multidimensional aggregates which is slower one.

2. Pre compute all of the cuboids is said to be **full materialization**. It requires huge amount of memory to store pre computed cuboids.

3. Selectively compute a proper subset of whole set of possible cuboids is said to be **partial materialization**. It performs with quick response time by consuming less memory.

The partial materialization of cuboids should consider 3 factors:

-Identify the subset of cuboids to materialize.
-Exploit the materialized cuboids during query processing.
-Efficiently update the materialized cuboids during load and refresh.

→The selection of the subset of cuboids to materialize should take the factors to be considered such as queries in the workload, their frequencies and their accessing costs.
→There are several heuristic approaches existed for cuboid selection.
→Once selected cuboids have been materialized it is important to take advantage of them during query processing which involves determining the relevant cuboid from materialized cuboids.

Multiway array aggregation in the computation of data cubes-

In order to ensure fast OLAP we need to pre compute all of the cuboids of a given data cube by exploring efficient methods for computing.

-ROLAP uses tuples and relational tables as its basic data structure and acts as a cube computation techniques.
-This ROLAP uses the below optimization techniques:

1. Sorting, hashing and grouping operations are applied to dimension attributes in order to re order and cluster related tuples.
2. Grouping is performed on some sub aggregates as a “partial grouping set”. So that these partial groups used to speed up the computation of other sub aggregates.
3. Aggregates may be computed from previously computed aggregates rather than from the base fact tables.

-MOLAP uses multidimensional array as a data structure and act as a cube computation Technique.
-This MOLAP uses the below optimization techniques:

1. Partition the array into chunks. A chunk is a sub cube that is a smaller one with less space available for computation. Chunking is a method for dividing n diminesional array into small chunks.
2. Compute aggregates by visiting cube cells. So that partial aggregates can be computed simultaneously and unnecessary revisiting of cells is avoided.

Indexing OLAP data

For efficient data access the data warehouse systems support index structures and materialized views.

OLAP data can be indexed in two ways:

Bitmap indexing-

-It allows quick search in data cubes.
-In this values are represented using a bit vectors.
-This is advantageous for low cardinality domains because comparison, join and aggregation operations. And with this string of characters will be represented by a single bit.

Ex- This is bitmap indexing process using bit representation.

Base table is as follows-

RecordID	ITEM	CITY
R1	H	V
R2	C	V
R3	P	V
R4	S	V
R5	H	T
R6	C	T
R7	P	T
R8	S	T

Item bitmap index table

RecordID	H	C	P	S
R1	1	0	0	0
R2	0	1	0	0
R3	0	0	1	0
R4	0	0	0	1
R5	1	0	0	0
R6	0	1	0	0
R7	0	0	1	0
R8	0	0	0	1

City bitmap index table

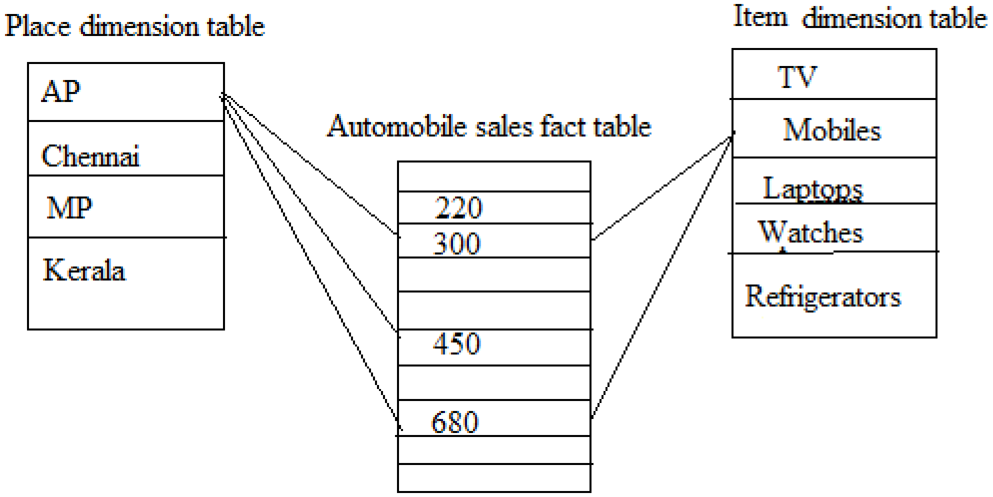
RecordID	V	T
R1	1	0
R2	1	0
R3	1	0
R4	1	0
R5	0	1
R6	0	1
R7	0	1
R8	0	1

Join Indexing-

-It is used in relational database query processing.
-These records can identify joinable tuples without performing costly join operations.
-It is useful for maintaining the relationship between a foreign key and its matching primary keys from the joinable relation.
-The star schema model makes this join indexing more attractive.

For example if two relations R (RID, A) and S (B, SID) join the attributes A & B, then the join index records contains the pair (RID, SID) where RID and SID are record ID's of R & S relations respectively.

For example-



Place	Item
AP	300
AP	450
AP	680

Join index table for Place & Item

Product	Sales
Mobiles	300
Mobiles	680

Join index table for product & sales

To speed up query processing both join & bitmap indexing can be integrated to form bitmapped join indices.

Efficient processing of OLAP queries:

The purpose of materializing cuboids and constructing OLAP indexing is to speed up query processing in data cubes. They should follow-

1. Determining which operations should be performed on the available cuboids-
   -It involves transforming any selection, projection, roll up and drill down operations in the query into corresponding SQL or OLAP operations.
   -For example slicing & dicing a data cube may correspond to selection/projection operations on a materialized cuboid.
2. Determining to which materialized cuboid the relevant operations should be applied-
   -It involves identifying all the materialized cuboids that are used to answer the query, estimating the costs of using the remaining materialized cuboids and selecting the least cost.

Metadata repository:

Metadata means data about data. In data warehouse metadata means data that defines data warehouse objects.

In data warehouse metadata repository contains-

1. A description of the structure of data warehouse- which includes data mart locations, data warehouse schemas, views, dimensions and contents.

2. An operational metadata-which includes-data lineage (historical data),
   -currency data (archived data),
   -monitoring information (warehouse usage, error reports).

3. The algorithms which are used as measures, aggregation, predefined queries for summarization.

4. The mapping information from the operational environment such as data cleaning, transformation rules, data extraction, security control information to the data warehouse.

5. The data related to system performance

6. The business metadata which includes business terms and data ownership information.

Data warehouse back end tools and utilities:

Data warehouse systems use back end tools and utilities to refresh its data.
These tools and utilities includes the below functions-

1. Data extraction-which gathers data from multiple external sources.
2. Data cleaning-which detects errors in the data and resolves it.
3. Data transformation-which converts data from a legacy/host format to warehouse format.
4. Load-which sorts, summarizes, consolidates, computes, checks integrity.
5. Refresh-which propagates the updates from data sources to warehouse.

Discovery driven exploration

- ➔ A user or analyst can search for interesting patterns (required data) in the cube by specifying a number of OLAP operations such as drill down, roll up, slice & dice.
- ➔ By using these tools the user can discover the data by his/her own hypothesis and tries to recognize exceptions or anomalies in the data.
  This is said to **hypothesis driven exploration** which is with many disadvantages.
- → The disadvantages are:
  1. Search space is very large
  2. High level aggregations may not give indication of anomalies at low levels.
  3. When looking at a subset of a cube such as slice, the user faces many data values to examine.
  4. Due to large volume of data values alone, it is easy for users to miss exceptions in the data.

Discovery driven exploration is an approach with pre computed measures that which indicates data exceptions, so that it helps to guide the user in data analysis process at all levels of aggregation. These measures are said to be exception indicators.

Exception means a data cube cell value which is different from expected value based on a statistical model.

The computation of exception indictors can be overlapped with cube construction, so that the overall construction of data cubes for discovery driven exploration is efficient.

Three measures are used as exception indicators help to identify data anomalies. These measures are computed with every cell for all levels of aggregation:

1. SelfExp- Indicates the degree of surprise of the cell value relative to other cells at the same level of aggregation.
2. InExp- Indicates the degree of surprise somewhere beneath the cell, if we were to drill down from it.
3. PathExp- Indicates the degree of surprise for each drill down path from the cell.

   For detailed explanation refer page no-33 example-2.14 in Data Mining Concepts and Techniques Jiawei Han and Micheline Kamber 1st edition.

Multi feature cubes:

*****For this explanation refer page no-36 examples 2.15, 2.16, 2.17 Data Mining Concepts and Techniques Jiawei Han and Micheline Kamber 1st edition*****

From data warehousing to data mining:

Data warehouses and data marts are used in a wide range of applications such as banking, financial services, consumer goods, retail distribution sectors.

The evolution of data warehouses involves few phases-

-Initially used for generating reports & answering predefined queries.
-Progressively used to analyze summarized and detailed data so that we get results in the form of reports & charts.
-Later used for strategic purposes performing multidimensional analysis and to perform slice & dice operations.
-Finally used as knowledge discovery and strategic decision making using data mining tools.

There are three types of data warehouse applications-

1. Information processing-supports querying, basic statistical analysis and reporting using tables graphs or charts.
2. Analytical processing- supports basic OLAP operations includes slice & dice, drill down, roll up, pivoting. It operates on historical data.
3. Data mining- supports knowledge discovery by finding hidden patterns and associations, by performing classification and prediction and presenting the mining results using visualization tools.

The functionalities of OLAP and data mining are different:

OLAP

- It is a data summarization/aggregation tool which helps to simplify data analysis.
-OLAP systems provides general description of data from data warehouses
-It performs data summary and comparison operations (by drilling, pivoting, slicing & dicing).

Data mining

-It allows the automated discovery of implicit patterns (data) and interesting knowledge hidden in large amount of data.
-It covers both data description and data modeling.
-It performs data summary & comparison operations and also performs classification, prediction, clustering, time-series analysis.
-It not only analyze data existed in data warehouse, it may analyze data existing at more detailed granularities other than the data in a data warehouse.

Transformation From OLAP to OLAM (On-line analytical processing to On-line analytical mining)

→Online analytical mining is an integration process of OLAP with data mining.
Why data mining?

Mining or extracting knowledge in multi dimensional databases is important for the following reasons-

1. High quality of data in data warehouses- A data mining tool works on integrated, consistent, and cleaned data which requires costly data cleaning, data transformation and data integration as preprocessing steps.

2. Available information processing infrastructure surrounding data warehouses- Data analysis & information processing (which means based on queries finding useful information which do not reflects data mining) will be constructed systematically includes accessing, integration, heterogeneous databases, ODBC/OLEDB connections, web accessing and OLAP analysis tools.

3. OLAP-based exploratory data analysis- Effective data mining needs exploratory data analysis. OLAM provides facilities for data mining on different subsets of data at different levels of abstraction by OLAP operations on a data cube.

4. On-line selection of data mining functions- By integrating OLAP with multiple data mining functions OLAM provides users, the flexibility to select desired data mining functions and swap data mining tasks dynamically.