UNIT I

DATA WAREHOUSING AND ONLINE ANALYTICAL PROCESSING

Data Warehouse

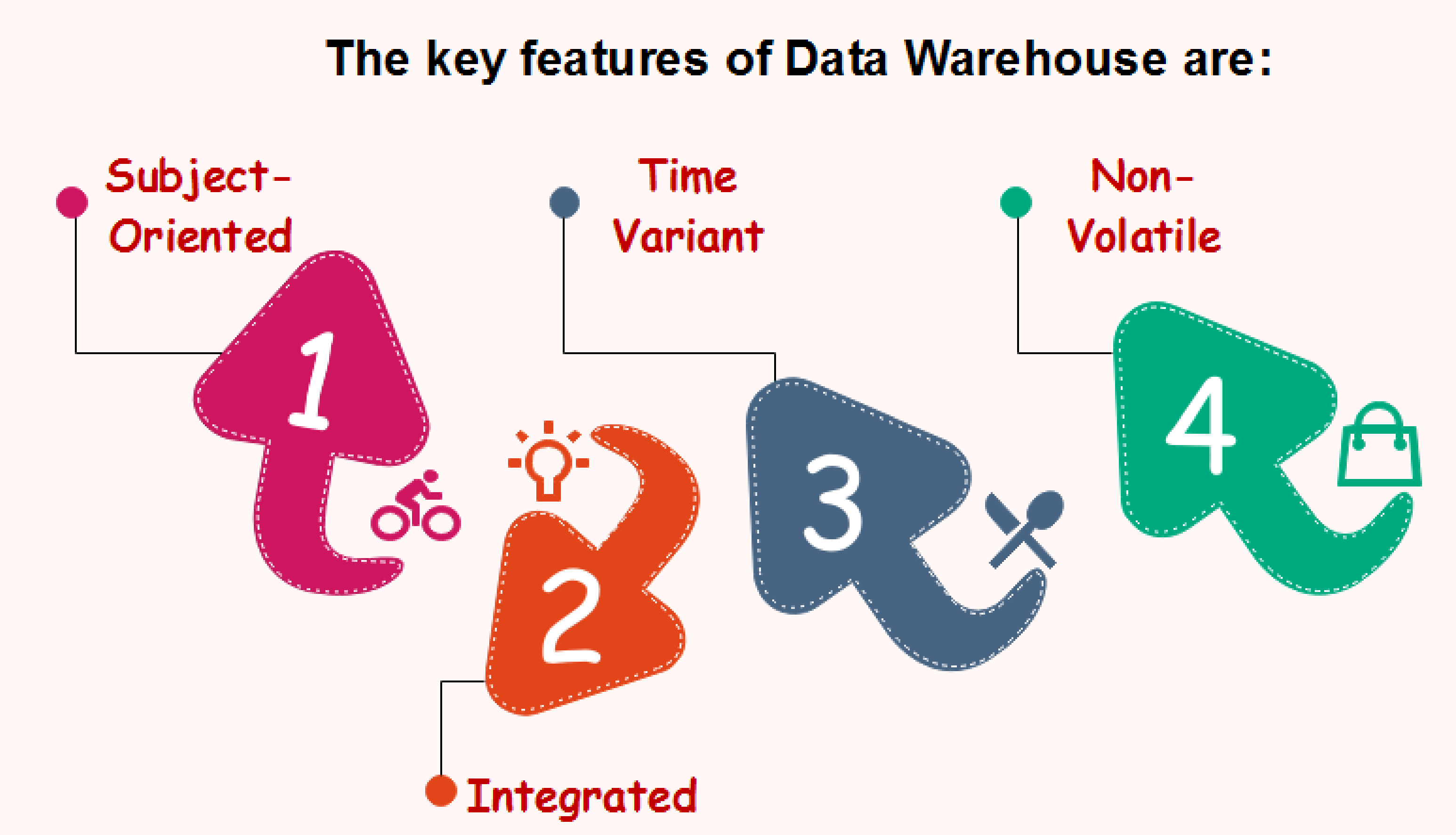
- A Data Warehouse (DW) is a relational database that is designed for query and analysis rather than transaction processing. It includes historical data derived from transaction data from single and multiple sources.

- A Data Warehouse provides integrated, enterprise-wide, historical data and focuses on providing support for decision-makers for data modeling and analysis.

- A Data Warehouse is a group of data specific to the entire organization, not only to a particular group of users.

- It is not used for daily operations and transaction processing but used for making decisions.

Characteristics of Data Warehouse

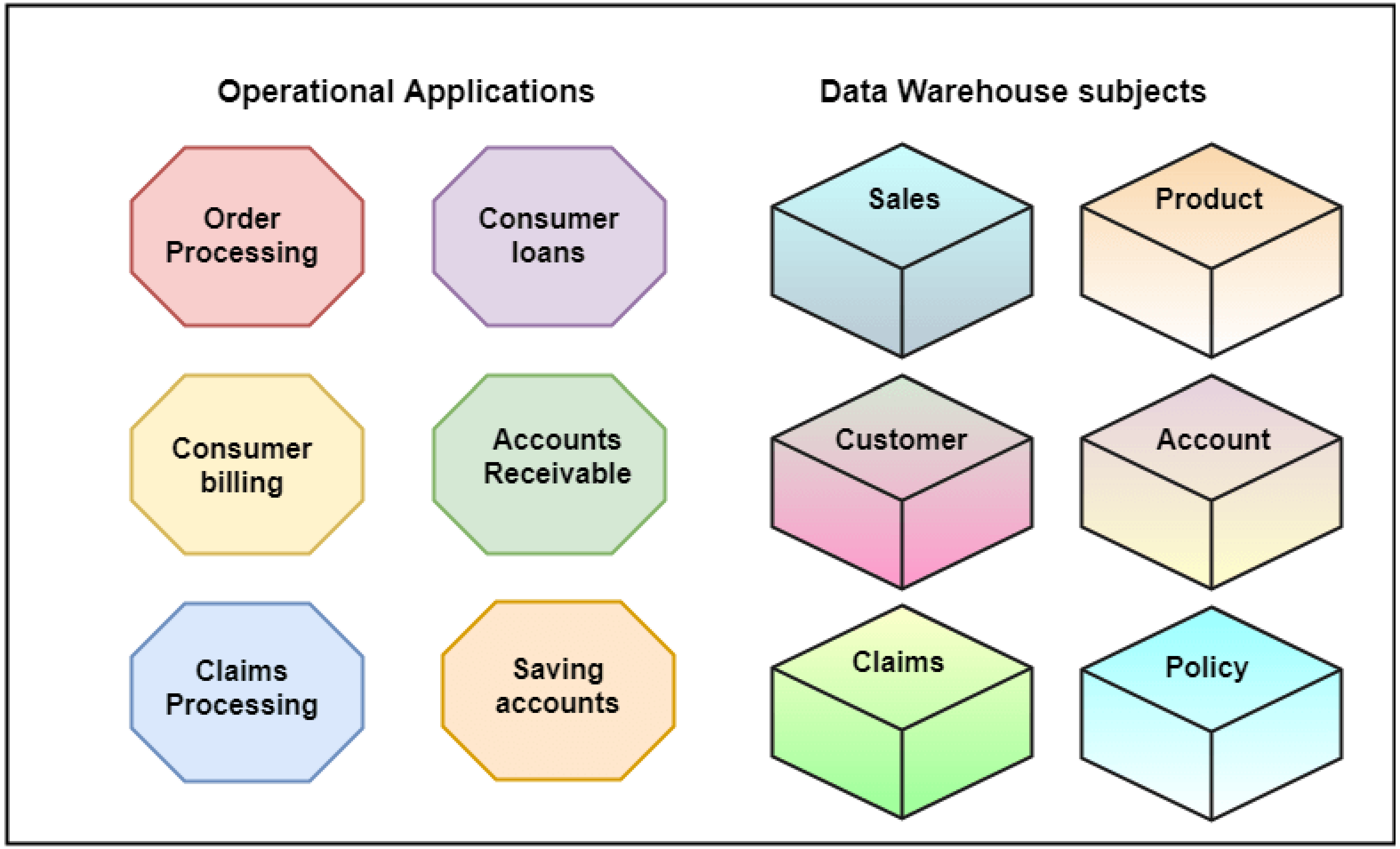


Subject-Oriented

- A data warehouse target on the modeling and analysis of data for decision-makers.

- Data warehouses typically provide a concise and straightforward view around a particular subject, such as customer, product, or sales, instead of the global organization's ongoing operations.

Data Warehouse is Subject-Oriented

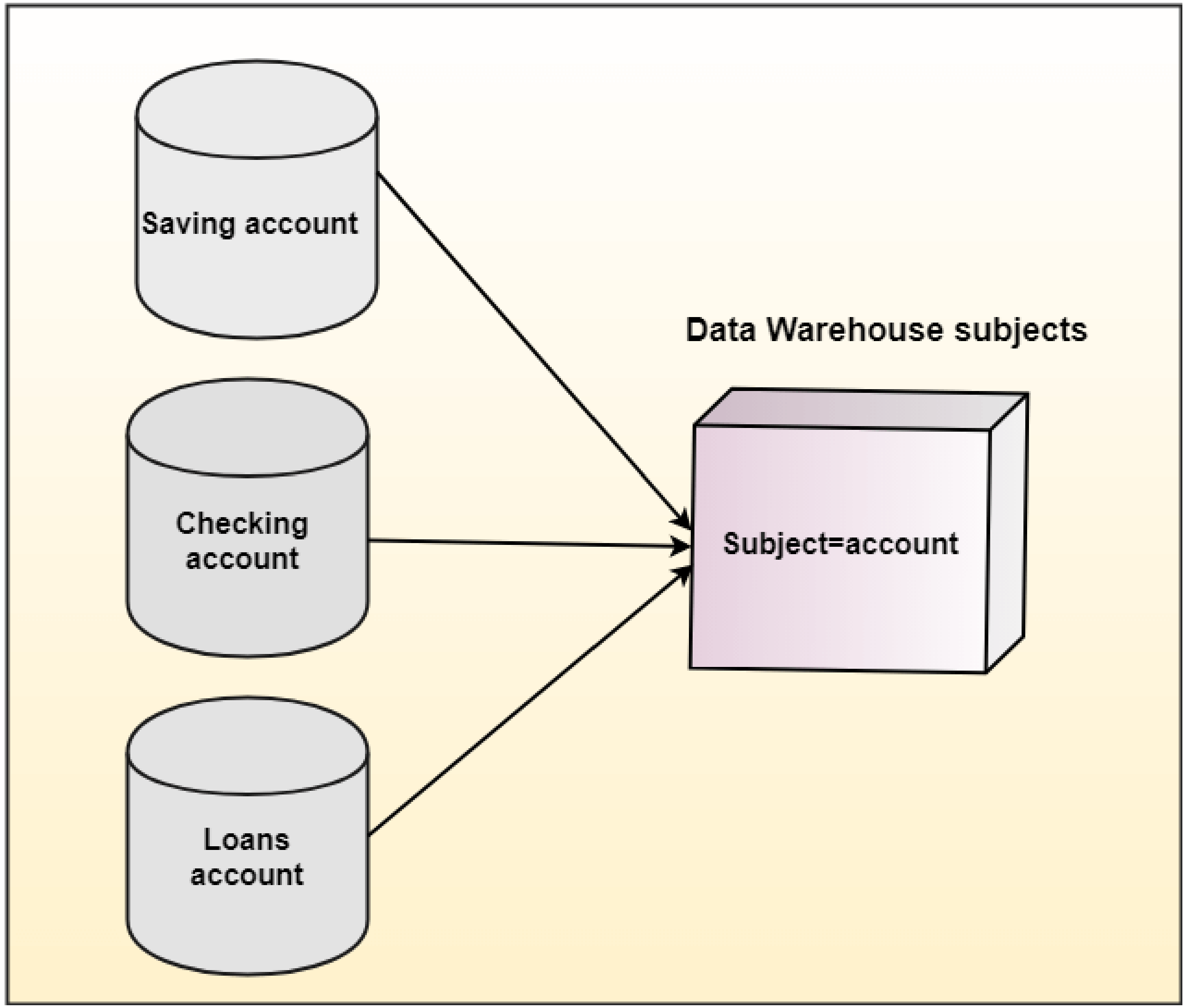


Integrated

- A data warehouse integrates various heterogeneous data sources like RDBMS, flat files, and online transaction records.

- It requires performing data cleaning and integration during data warehousing to ensure consistency in naming conventions, attributes types, etc., among different data sources.

Data Warehouse is Integrated



Time-Variant

- Historical information is kept in a data warehouse. For example, one can retrieve files from 3 months, 6 months, 12 months, or even previous data from a data warehouse.

- These variations with a transactions system, where often only the most current file is kept.

Time Variant
HISTORY

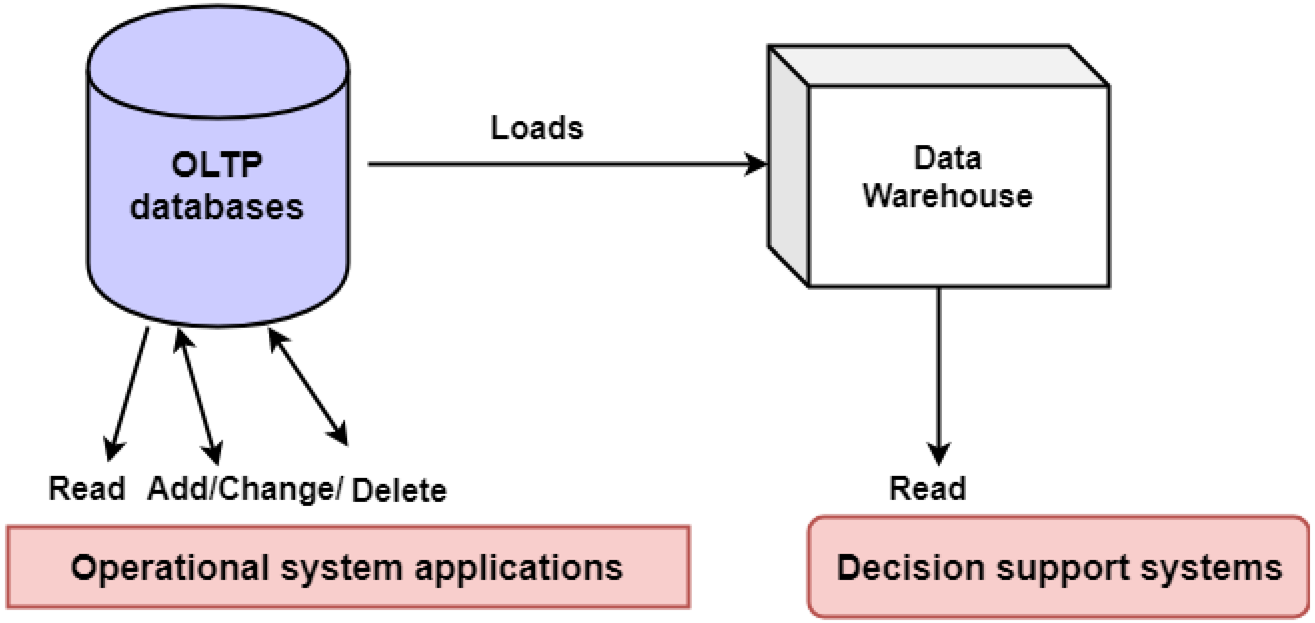
Non-Volatile

- The data warehouse is a physically separate data storage, which is transformed from the source operational RDBMS.

- The operational updates of data do not occur in the data warehouse, i.e., update, insert, and delete operations are not performed. It usually requires only two procedures in data accessing: Initial loading of data and access to data.

- Non-Volatile defines that once entered into the warehouse, and data should not change.

Non-Volatile



OLTP databases

Loads

Data Warehouse

Read

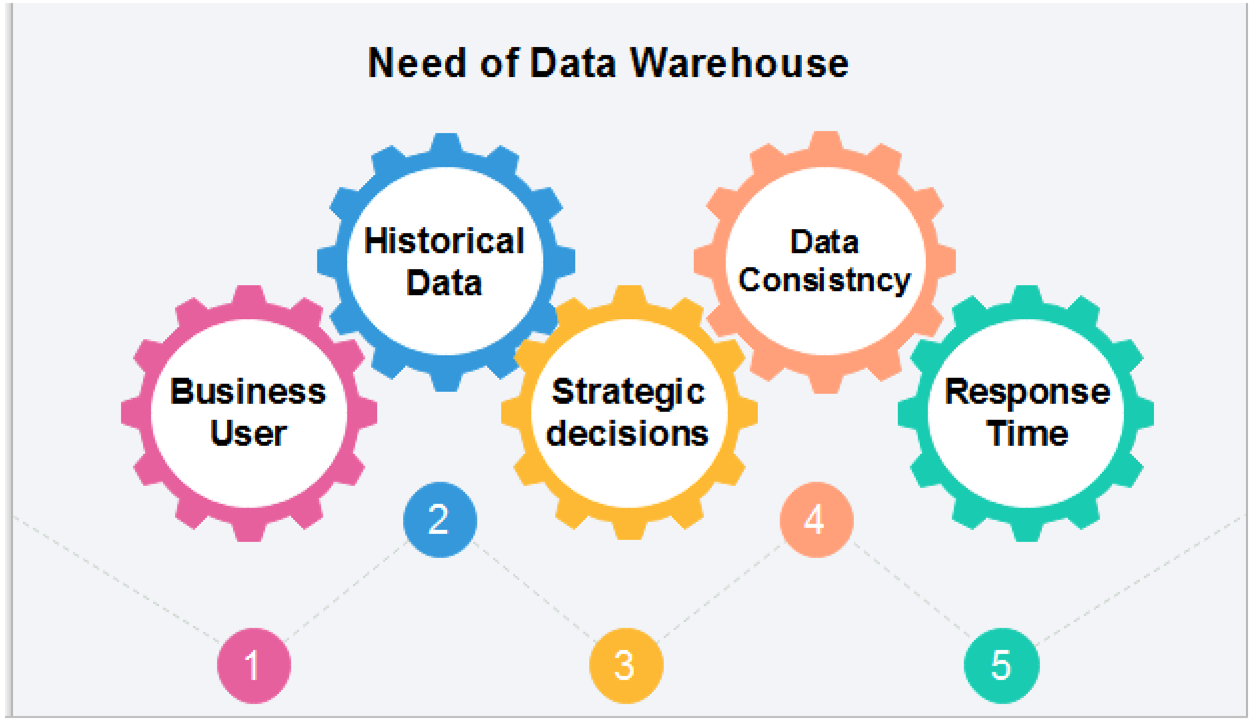
Add/Change/ Delete

Read

Operational system applications

Decision support systems

Need for Data Warehouse



1) **Business User:** Business users require a data warehouse to view summarized data from the past. Since these people are non-technical, the data may be presented to them in an elementary form.

2) **Store historical data:** Data Warehouse is required to store the time variable data from the past. This input is made to be used for various purposes.

3) **Make strategic decisions:** Some strategies may be depending upon the data in the data warehouse. So, data warehouse contributes to making strategic decisions.

4) **For data consistency and quality:** Bringing the data from different sources at a commonplace, the user can effectively undertake to bring the uniformity and consistency in data.

5) **High response time:** Data warehouse has to be ready for somewhat unexpected loads and types of queries, which demands a significant degree of flexibility and quick response time.

Benefits of Data Warehouse

1.Understand business trends and make better forecasting decisions.

2.Data Warehouses are designed to perform well enormous amounts of data.

3.The structure of data warehouses is more accessible for end-users to navigate, understand, and query.

4.Queries that would be complex in many normalized databases could be easier to build and maintain in data warehouses.

5.Data warehousing is an efficient method to manage demand for lots of information from lots of users.

6.Data warehousing provide the capabilities to analyze a large amount of historical data.

Data Warehouse Modeling

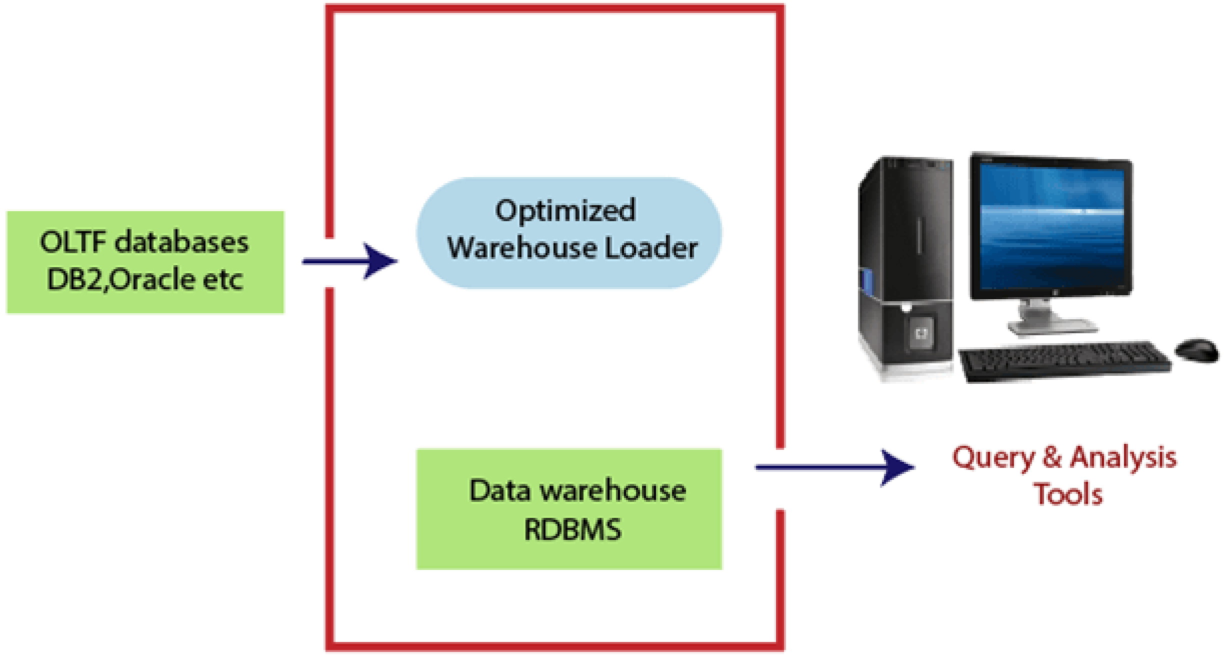
- Data warehouse modeling is the process of designing the schemas of the detailed and summarized information of the data warehouse.

- The goal of data warehouse modeling is to develop a schema describing the reality, or at least a part of the fact, which the data warehouse is needed to support.

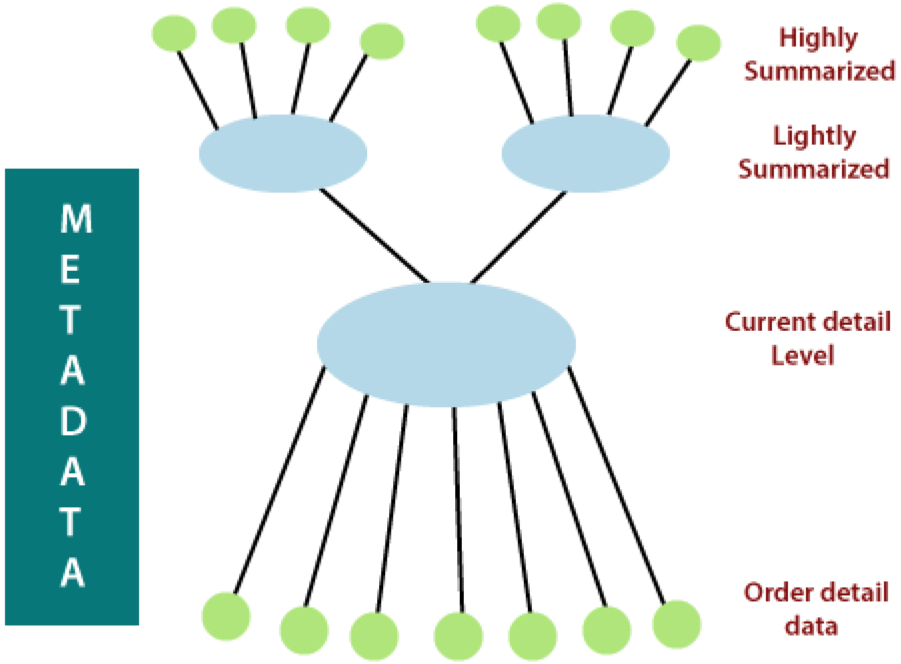
Data warehouse modeling is an essential stage of building a data warehouse for two main reasons.

Firstly, through the schema, data warehouse clients can visualize the relationships among the warehouse data, to use them with greater ease.

Secondly, a well-designed schema allows an effective data warehouse structure to emerge, to help decrease the cost of implementing the warehouse and improve the efficiency of using it.



Data Warehouse Model



The Structure of data inside the data warehouse

The current detail record is central in importance as it:

- Reflects the most current happenings, which are commonly the most stimulating.

- It is numerous as it is saved at the lowest method of the Granularity.

- It is always (almost) saved on disk storage, which is fast to access but expensive and difficult to manage.

Older detail data is stored in some form of mass storage, and it is infrequently accessed and kept at a level detail consistent with current detailed data.

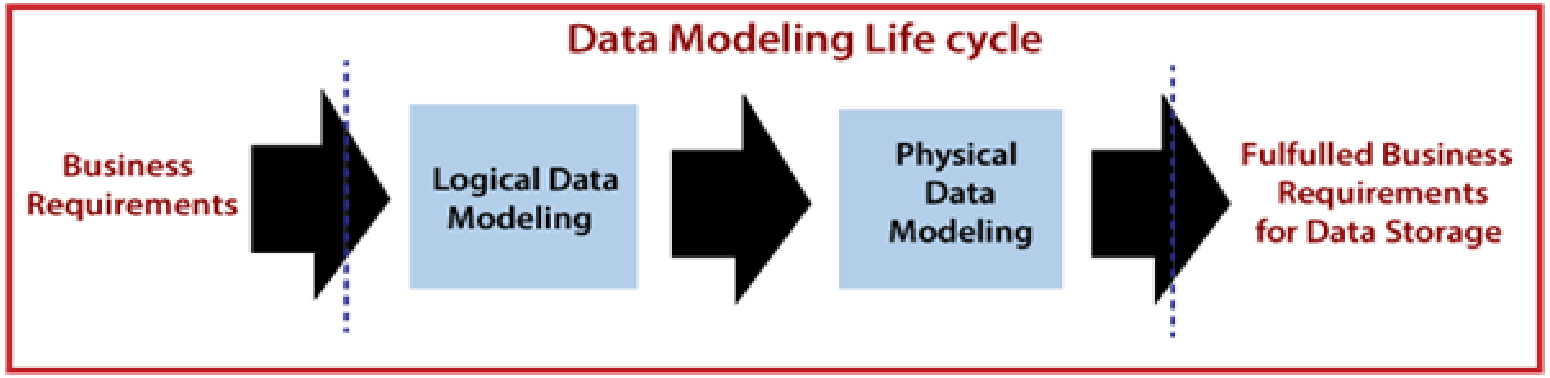
Lightly summarized data is data extract from the low level of detail found at the current, detailed level and usually is stored on disk storage. When building the data warehouse have to remember what unit of time is summarization done over and also the components or what attributes the summarized data will contain.

Highly summarized data is compact and directly available and can even be found outside the warehouse.

Metadata is the final element of the data warehouses and is really of various dimensions in which it is not the same as file drawn from the operational data, but it is used as:-

•A directory to help the DSS investigator locate the items of the data warehouse.
•A guide to the mapping of record as the data is changed from the operational data to the data warehouse environment.
•A guide to the method used for summarization between the current, accurate data and the lightly summarized information and the highly summarized data, etc.

Data Modeling Life Cycle



A generic data modeling life cycle

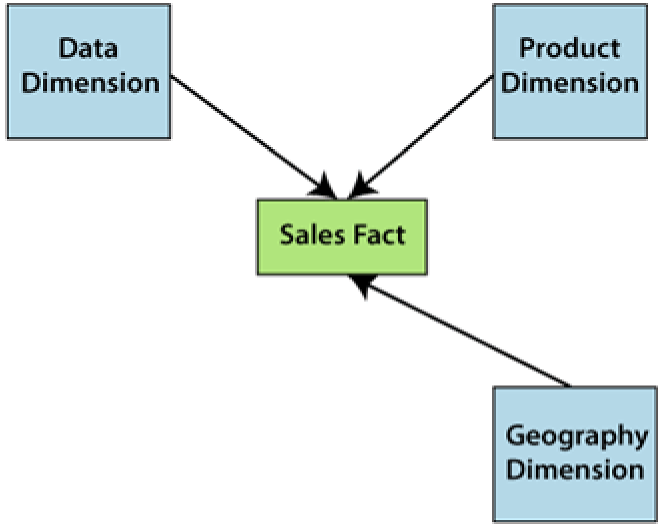
Conceptual Data Model

A conceptual data model recognizes the highest-level relationships between the different entities.
Characteristics of the conceptual data model

•It contains the essential entities and the relationships among them.

•No attribute is specified.

•No primary key is specified.



Example of Conceptual Data Model

Logical Data Model

A logical data model defines the information in as much structure as possible, without observing how they will be physically achieved in the database.

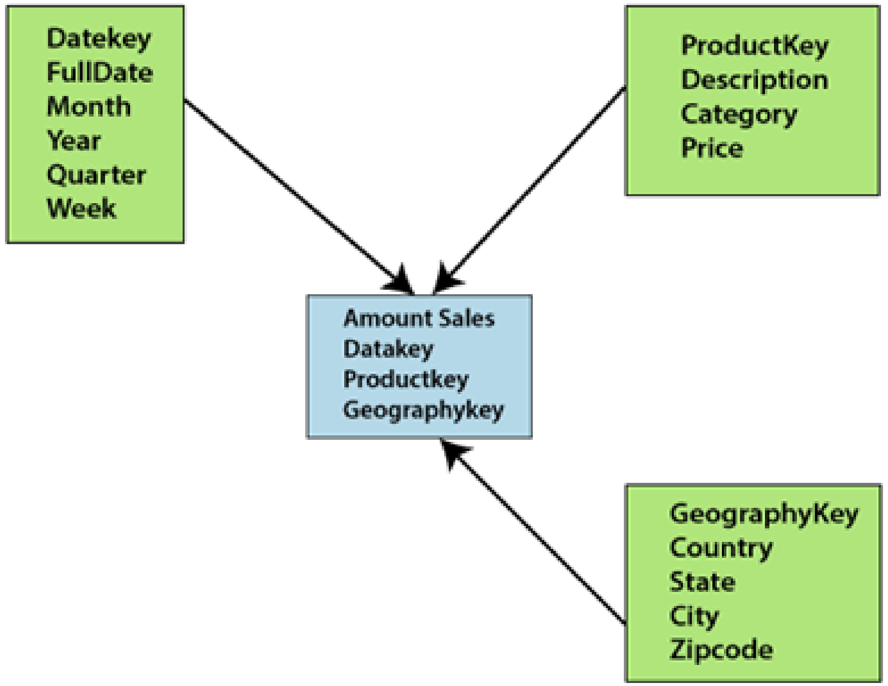
The primary objective of logical data modeling is to document the business data structures, processes, rules, and relationships by a single view - the logical data model.

Features of a logical data model

- It involves all entities and relationships among them.
- All attributes for each entity are specified.
- The primary key for each entity is stated.
- Referential Integrity is specified (FK Relation).

The phase for designing the logical data model which are as follows:

- Specify primary keys for all entities.
- List the relationships between different entities.
- List all attributes for each entity.
- Normalization.
- No data types are listed



Example of Logical Data Model

Physical Data Model

Physical data model describes how the model will be presented in the database.

A physical database model demonstrates all table structures, column names, data types, constraints, primary key, foreign key, and relationships between tables.

The purpose of physical data modeling is the mapping of the logical data model to the physical structures of the RDBMS system hosting the data warehouse

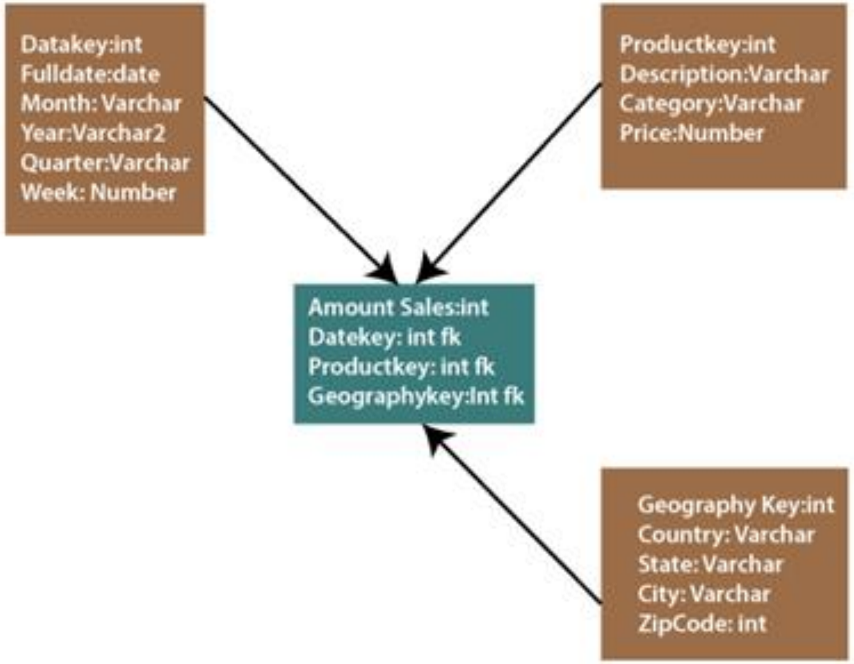
Characteristics of a physical data model

- Specification all tables and columns.

- Foreign keys are used to recognize relationships between tables.

The steps for physical data model design which are as follows:

- Convert entities to tables.
- Convert relationships to foreign keys.
- Convert attributes to columns.



Datakey:int
Fulldate:date
Month: Varchar
Year:Varchar2
Quarter:Varchar
Week: Number

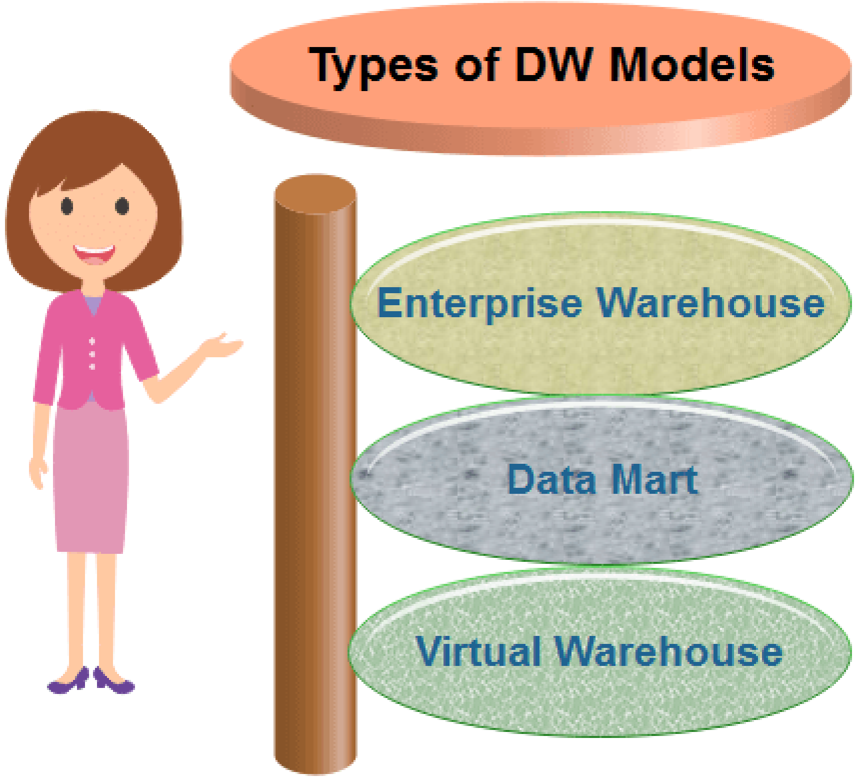
Productkey:int
Description:Varchar
Category:Varchar
Price:Number

Amount Sales:int
Datekey: int fk
Productkey: int fk
Geographykey:Int fk

Geography Key:int
Country: Varchar
State: Varchar
City: Varchar
ZipCode: int

Example of Physical Data Model

Types of Data Warehouse Models



Enterprise Warehouse

- An Enterprise warehouse collects all of the records about subjects spanning the entire organization. It supports corporate-wide data integration, usually from one or more operational systems or external data providers, and it's cross-functional in scope.

- It generally contains detailed information as well as summarized information and can range in estimate from a few gigabyte to hundreds of gigabytes, terabytes, or beyond.

Data Mart

A data mart includes a subset of corporate-wide data that is of value to a specific collection of users. The scope is confined to particular selected subjects.

For example, a marketing data mart may restrict its subjects to the customer, items, and sales. The data contained in the data marts tend to be summarized.

Data Marts is divided into two parts:

Independent Data Mart: Independent data mart is sourced from data captured from one or more operational systems or external data providers, or data generally locally within a different department or geographic area.
Dependent Data Mart: Dependent data marts are sourced exactly from enterprise data-warehouses.

Virtual Warehouses

Virtual Data Warehouses is a set of perception over the operational database. For effective query processing, only some of the possible summary vision may be materialized.

A virtual warehouse is simple to build but required excess capacity on operational database servers.

Data Warehouse Design

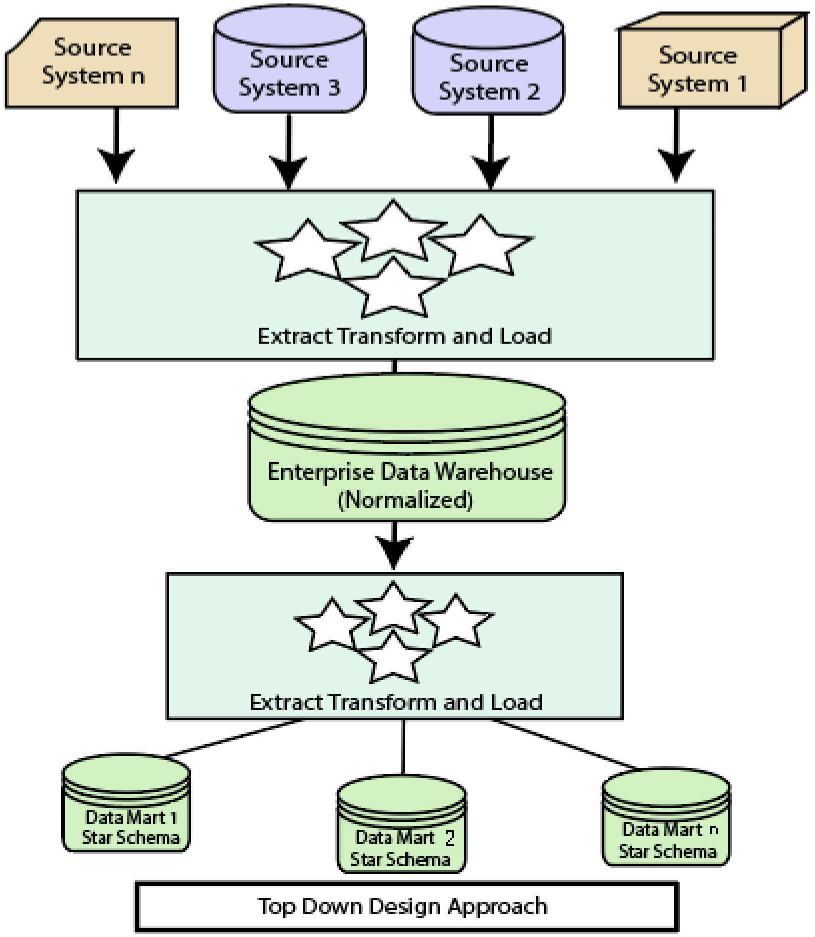
There are two approaches

1. "top-down" approach
2. "bottom-up" approach

Top-down Design Approach

In the "Top-Down" design approach, a data warehouse is described as a subject-oriented, time-variant, non-volatile and integrated data repository for the entire enterprise data from different sources are validated, reformatted and saved in a normalized (up to 3NF) database as the data warehouse.

An approach is a data-driven approach as the information is gathered and integrated first and then business requirements by subjects for building data marts are formulated. The advantage of this method is which it supports a single integrated data source.



Top Down Design Approach

Advantages of top-down design

- Data Marts are loaded from the data warehouses.
- Developing new data mart from the data warehouse is very easy.

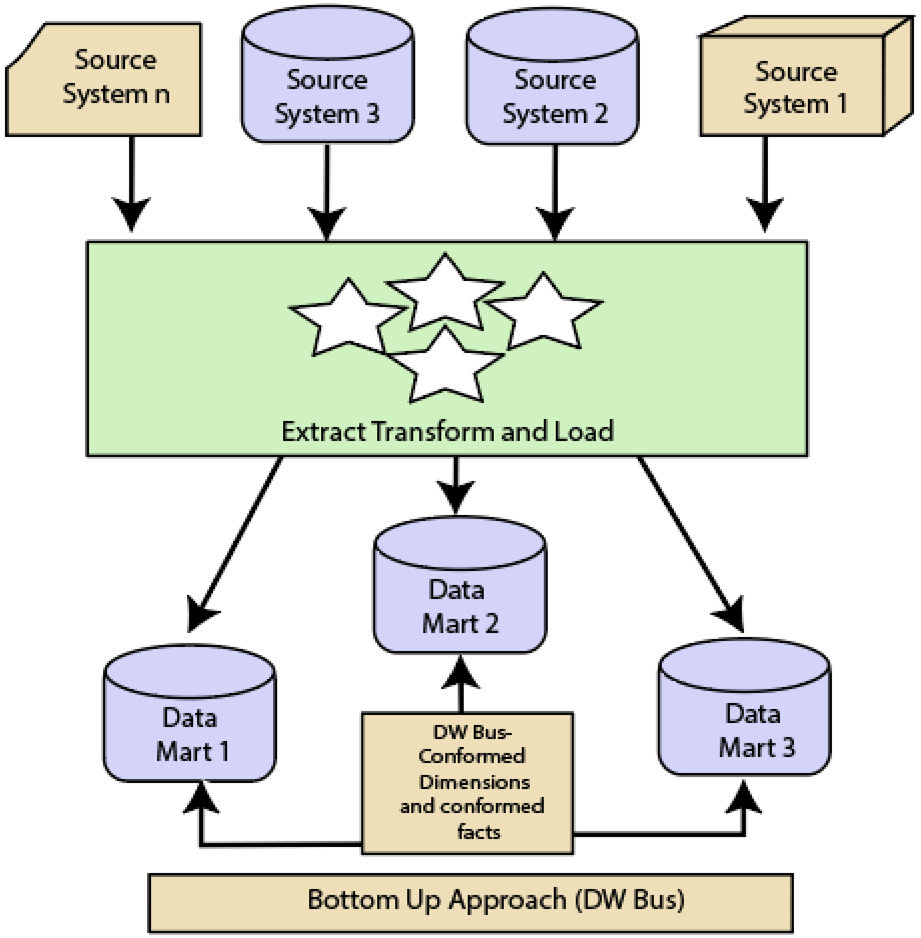
Disadvantages of top-down design

- This technique is inflexible to changing departmental needs.
- The cost of implementing the project is high.

Bottom-Up Design Approach

In the "Bottom-Up" approach, a data warehouse is described as "a copy of transaction data specifical architecture for query and analysis," term the star schema.

In this approach, a data mart is created first to necessary reporting and analytical capabilities for particular business processes (or subjects). Thus it is needed to be a business-driven approach in contrast to Inmon's data-driven approach.



Source System n

Source System 3

Source System 2

Source System 1

Extract Transform and Load

Data Mart 1

Data Mart 2

Data Mart 3

DW Bus- Conformed Dimensions and conformed facts

Bottom Up Approach (DW Bus)

Bottom Up Design Approach

Advantages of bottom-up design

Documents can be generated quickly.
The data warehouse can be extended to accommodate new business units.
It is just developing new data marts and then integrating with other data marts.

Disadvantages of bottom-up design

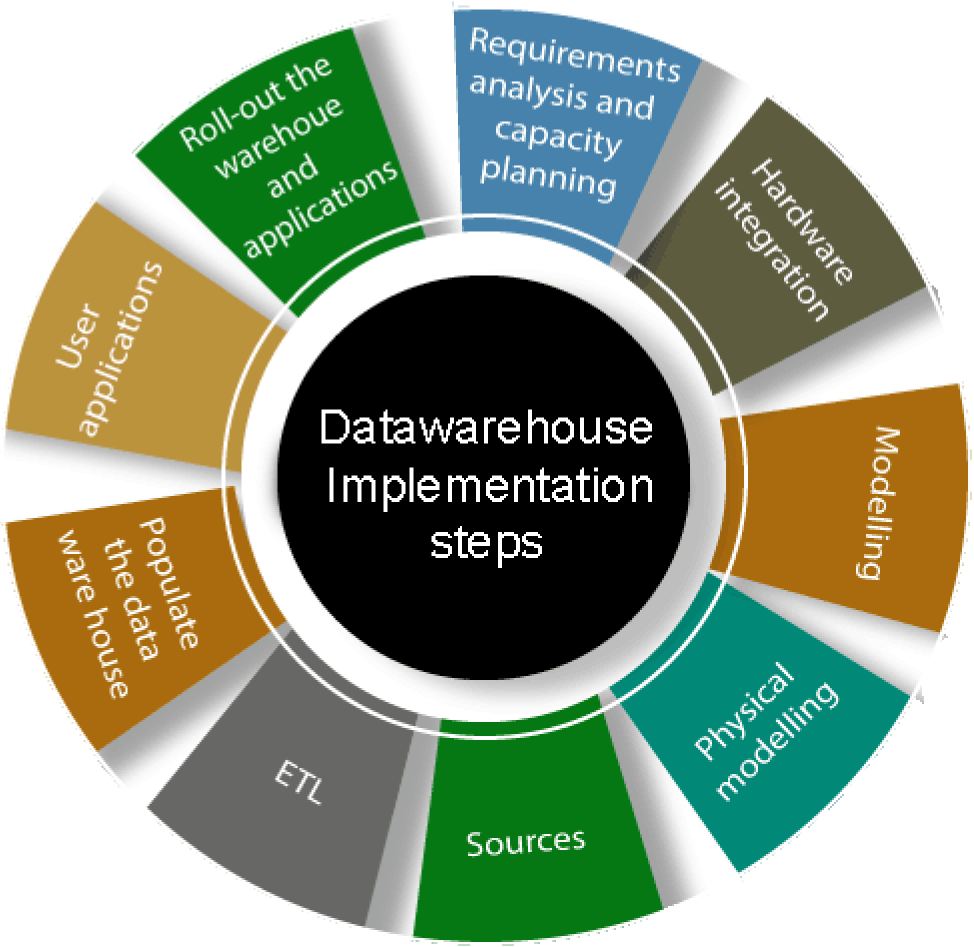
the locations of the data warehouse and the data marts are reversed in the bottom-up approach design.

Differentiate between Top-Down Design Approach and Bottom-Up Design Approach

Top-Down Design Approach	Bottom-Up Design Approach
Breaks the vast problem into smaller subproblems.	Solves the essential low-level problem and integrates them into a higher one.
Inherently architected- not a union of several data marts.	Inherently incremental; can schedule essential data marts first.
Single, central storage of information about the content.	Departmental information stored.
Centralized rules and control.	Departmental rules and control.
It includes redundant information.	Redundancy can be removed.
It may see quick results if implemented with repetitions.	Less risk of failure, favorable return on investment, and proof of techniques.

Data Warehouse Implementation

There are various implementation in data warehouses which are as follows


Datawarehouse Implementation steps
Requirements analysis and capacity planning
Hardware integration
Modelling
Physical modelling
Sources
ETL
Populate the data ware house
User applications
Roll-out the warehoue and applications

1.Requirements analysis and capacity planning: The first process in data warehousing involves defining enterprise needs, defining architectures, carrying out capacity planning, and selecting the hardware and software tools. This step will contain be consulting senior management as well as the different stakeholder.

2. Hardware integration: Once the hardware and software has been selected, they require to be put by integrating the servers, the storage methods, and the user software tools.

3. Modeling: Modelling is a significant stage that involves designing the warehouse schema and views. This may contain using a modeling tool if the data warehouses are sophisticated.

4. Physical modeling: For the data warehouses to perform efficiently, physical modeling is needed. This contains designing the physical data warehouse organization, data placement, data partitioning, deciding on access techniques, and indexing.

5. Sources: The information for the data warehouse is likely to come from several data sources. This step contains identifying and connecting the sources using the gateway, ODBC drives, or another wrapper.

6. ETL: The data from the source system will require to go through an ETL phase. The process of designing and implementing the ETL phase may contain defining a suitable ETL tool vendors and purchasing and implementing the tools. This may contains customize the tool to suit the need of the enterprises.

7. Populate the data warehouses: Once the ETL tools have been agreed upon, testing the tools will be needed, perhaps using a staging area. Once everything is working adequately, the ETL tools may be used in populating the warehouses given the schema and view definition.

8. User applications: For the data warehouses to be helpful, there must be end-user applications. This step contains designing and implementing applications required by the end-users.

9. Roll-out the warehouses and applications: Once the data warehouse has been populated and the end-client applications tested, the warehouse system and the operations may be rolled out for the user's community to use.

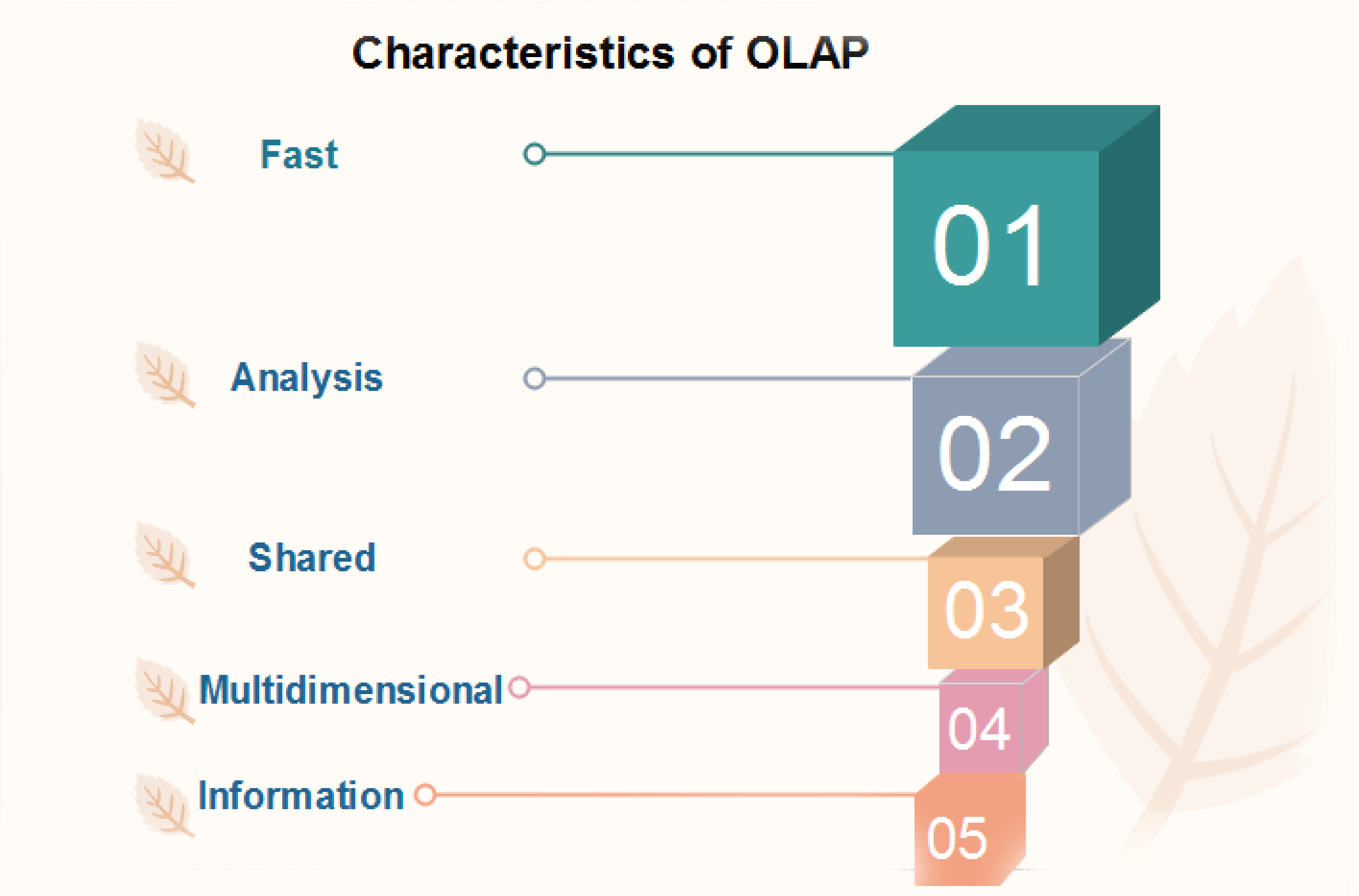
What is OLAP (Online Analytical Processing)?

OLAP stands for **On-Line Analytical Processing**. OLAP is a classification of software technology which authorizes analysts, managers, and executives to gain insight into information through fast, consistent, interactive access in a wide variety of possible views of data that has been transformed from raw information to reflect the real dimensionality of the enterprise as understood by the clients.

OLAP implement the multidimensional analysis of business information and support the capability for complex estimations, trend analysis, and sophisticated data modeling. It is rapidly enhancing the essential foundation for Intelligent Solutions containing Business Performance Management, Planning, Budgeting, Forecasting, Financial Documenting, Analysis, Simulation-Models, Knowledge Discovery, and Data Warehouses Reporting. OLAP enables end-clients to perform ad hoc analysis of record in multiple dimensions, providing the insight and understanding they require for better decision making.

Characteristics of OLAP

In the **FASMI characteristics of OLAP methods**, the term derived from the first letters of the characteristics are:



Fast

It defines which the system targeted to deliver the most feedback to the client within about five seconds, with the elementary analysis taking no more than one second and very few taking more than 20 seconds.

Analysis

It defines which the method can cope with any business logic and statistical analysis that is relevant for the function and the user, keep it easy enough for the target client.

Share

It defines which the system tools all the security requirements for understanding and, if multiple write connection is needed, concurrent update location at an appropriated level, not all functions need customer to write data back, but for the increasing number which does, the system should be able to manage multiple updates in a timely, secure manner.

Multidimensional
This is the basic requirement. OLAP system must provide a multidimensional conceptual view of the data, including full support for hierarchies, as this is certainly the most logical method to analyze business and organizations.

Information
The system should be able to hold all the data needed by the applications. Data sparsity should be handled in an efficient manner.

The main characteristics of OLAP are as follows:

1. **Multidimensional conceptual view:** OLAP systems let business users have a dimensional and logical view of the data in the data warehouse. It helps in carrying slice and dice operations.

2. **Multi-User Support:** Since the OLAP techniques are shared, the OLAP operation should provide normal database operations, containing retrieval, update, adequacy control, integrity, and security.

3. **Accessibility:** OLAP acts as a mediator between data warehouses and front-end. The OLAP operations should be sitting between data sources (e.g., data warehouses) and an OLAP front-end.
4. **Storing OLAP results:** OLAP results are kept separate from data sources.

5. **Uniform documenting performance:** Increasing the number of dimensions or database size should not significantly degrade the reporting performance of the OLAP system.

Benefits of OLAP

OLAP holds several benefits for businesses: -

1.OLAP helps managers in decision-making through the multidimensional record views that it is efficient in providing, thus increasing their productivity.

2.OLAP functions are self-sufficient owing to the inherent flexibility support to the organized databases.

3.It facilitates simulation of business models and problems, through extensive management of analysis-capabilities.

4.In conjunction with data warehouse, OLAP can be used to support a reduction in the application backlog, faster data retrieval, and reduction in query drag.

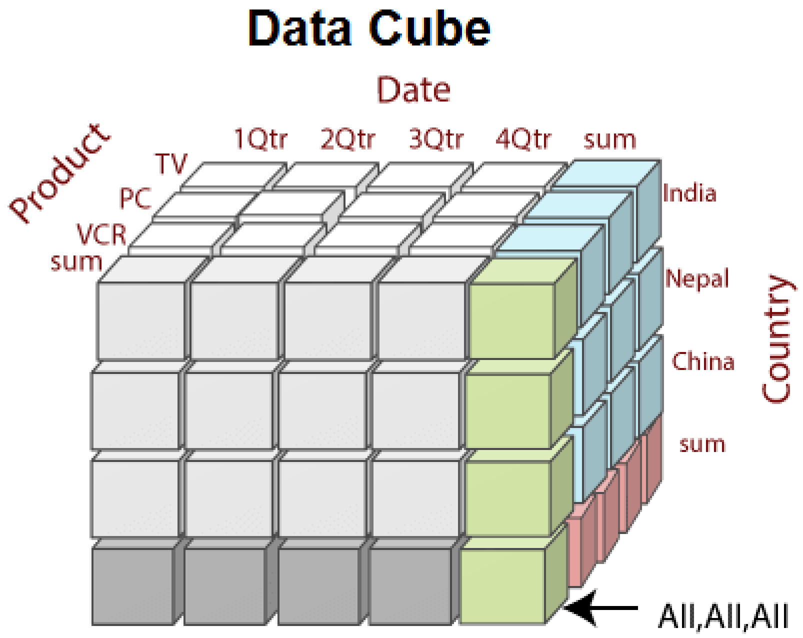
Data Cube

When data is grouped or combined in multidimensional matrices called Data Cubes. The data cube method has a few alternative names or a few variants, such as "Multidimensional databases," "materialized views," and "OLAP (On-Line Analytical Processing)."
The general idea of this approach is to materialize certain expensive computations that are frequently inquired.

Data cube method is an interesting technique with many applications. Data cubes could be sparse in many cases because not every cell in each dimension may have corresponding data in the database.

A data cube enables data to be modeled and viewed in multiple dimensions.

Dimensions are a fact that defines a data cube. Facts are generally quantities, which are used for analyzing the relationship between dimensions.



Types of Data Cube

There are two types of Data cubes which are used mostly in business or enterprises:

1. Multidimensional Data Cube (MOLAP)

As its name suggests Multidimensional Data cube is used mostly in the business requirement where there are huge sets of data. Products developed and follow involves the structure of MOLAP which has a multidimensional array format.

This structure helps in improving the huge data set with a sparser and an increased level of MOLAP. From this, we can come into a fact that this will not represent any specific data or clustered data value from a data set.

2. Relational Data Cube (ROLAP)

It is also another category of data analysis data cube which religiously follows the relational database model.

If we compared to the Multi-dimensional data cube, then it possesses double the number of relational tables to specify the dimensions with data sets and requirements.

Each of these tables contains a specific view which is called as a cuboid.

Benefits

•Increases the productivity of an enterprise.

•Improves the overall performance and efficiency.

•Representation of huge and complex data sets get simplified and streamlined.

•Huge database and complex SQL queries are also manageable.

•Indexing and ordering provides the best set of data for analysis and data mining techniques.

•Faster and easily accessible as It will posses pre-defined and pre-calculated data sets or data cubes.

•Aggregation of data makes access to all data very fast at each micro-level which ultimately leads to easy and efficient maintenance and reduced development time.

•OLAP will help in getting Fast Response time, Fast curve of Learning, versatile environment, reach to a wide range of reach to all applications, need of resources for deployment and less wait time with a quality result.