Data Mining and Warehouse

Unit-I

Introduction to Data Warehouse and Dimensional Modelling

By:- Dr. D.R.Patil

Vision of the Institute

To achieve excellence in engineering education with strong ethical values.

Mission of the Institute

To impart high quality Technical Education through:

- Innovative and Interactive learning process and high quality, internationally recognized instructional programs.
- Fostering a scientific temper among students by the means of a liaison with the Academia, Industries and Government.
- Preparing students from diverse backgrounds to have aptitude for research and spirit of Professionalism.
- Inculcating in students a respect for fellow human beings and responsibility towards the society.

Vision of the Department

To provide prominent computer engineering education with socio-moral values.

Mission of the Department

M1 : To provide state-of-the-art ICT based teaching-learning process.

M2: To groom the students to become professionally sound computer engineers to meet growing needs of industry and society.

M3: To make the students responsible human being by inculcating ethical values.

Data Mining and Warehouse (PCCO5010T)

Teaching Scheme
Lectures : 03 Hrs./week
Credits : 03

Examination Scheme
Term Test : 15 Marks
Teacher Assessment : 20 Marks
End Sem Exam : 65 Marks
Total Marks : 100 Marks

Pre-requisite: Basic database concepts, Concepts of algorithm design and analysis.

Course Objectives:

1. To identify the scope and essentiality of Data Mining and Warehouse.
2. To analyze data, choose relevant models and algorithms for respective applications.
3. To develop research interest towards advances in data mining.

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Understand Data Warehouse fundamentals and data mining principles.	L2	Understand
CO2	Design data warehouse with dimensional modelling.	L6	Create
CO3	Understand ETL process and apply OLAP operations.	L2	Understand
CO4	Apply appropriate pre-processing techniques.	L3	Apply
CO5	Identify appropriate data mining algorithms to solve real world problems.	L3	Apply
CO6	Compare and evaluate different data mining techniques like classification, clustering and association rule mining.	L5	Evaluate

Course Contents

Unit-I Introduction to Data Warehouse and Dimensional Modelling 08 Hrs.

Introduction to Strategic Information, Need for Strategic Information, Features of Data Warehouse, Data Warehouse versus Data Marts, Data Warehouse versus Data Lake, Top Down versus Bottom Up Approach. Data Warehouse Architecture, Metadata, E-R Modelling versus Dimensional Modelling, Information Package Diagram, STAR Schema, STAR Schema keys, Snowflake Schema, Fact Constellation Schema, Factless Fact Tables, Update to the Dimension Tables, Aggregate Fact Tables.

Unit-II ETL Process and OLAP 06 Hrs.

Major steps in ETL Process, Data Extraction Techniques, Data Transformation: Basic tasks, Major transformation type.

Data Loading: Applying Data, OLTP Vs OLAP, OLAP Definition, Dimensional Analysis, Hypercubes.

OLAP Operations: Drill Down, Roll Up, Slice, Dice and Rotation, OLAP models: MOLAP, ROLAP.

Unit-III Introduction to Data Mining, Data Exploration and Preprocessing 06 Hrs.

Data Mining Task and Techniques, KDD Process, Issues in Data Mining, Applications of Data Mining.

Data Exploration: Types of Attributes, Statistical Description of Data, Data Visualization, Measuring data similarity and dissimilarity.

Data Preprocessing: Major tasks in Preprocessing, Data Cleaning: Missing Values, Noisy data; Data Integration: Entity Identification Problem, Redundancy and Correlation Analysis, Tuple Duplication, Data Value Conflict Detection and Resolution.

Data Reduction: Attribute Subset Selection, Histograms, Clustering and Sampling.

Data Transformation & Data Discretization: Data Transformation by Normalization, Discretization by Binning, Discretization by Histogram Analysis, Concept hierarchy generation for Nominal data.

Unit-IV Classification and Prediction 06 Hrs.

Basic Concepts of Classification, Decision Tree Induction, Attribute Selection Measures using Information Gain, Tree pruning.

Bayes Classification Methods: Bayes' Theorem, Naïve Bayesian Classification.

Rule Based Classification: Using IF THEN Rules for Classification, Rule Extraction from a Decision Tree, Rule Quality Measures, Rule Pruning.

Model Evaluation & Selection: Metrics for Evaluating Classifier Performance, Holdout Method and

Random Subsampling, Cross Validation, Bootstrap, Model Selection Using Statistical Tests of Significance, Comparing Classifiers Based on Cost–Benefit and ROC Curves Improving Classification Accuracy: Ensemble classification, Bagging, Boosting and AdaBoost, Random Forests, Improving Classification Accuracy in Class Imbalance Data Prediction: Simple Linear regression.

Unit-V Clustering 05 Hrs.

Cluster Analysis and Requirements of Cluster Analysis. Partitioning Methods: k-Means, k-Medoids. Hierarchical Methods: Agglomerative, Divisive. Density Based Methods: DBScan Evaluation of Clustering: Assessing Clustering Tendency, Determining Number of Clusters and Measuring Cluster Quality: Intrinsic and Extrinsic methods.

Unit-VI Mining Frequent Patterns and Association Rules 06 Hrs.

Market Basket Analysis, Frequent Itemsets, Closed Itemsets, and Association Rule. Frequent Itemset Mining Methods: Apriori Algorithm, Association Rule Generation, Improving the Efficiency of Apriori.

FP Growth, Mining Frequent Itemsets using Vertical Data Format. Introduction to Mining Multilevel Association Rules and Multidimensional Association Rules.

Unit-VII Spatial and Web Mining 05 Hrs.

Spatial Data, Spatial Vs. Classical Data Mining, Spatial Data Structures, Mining Spatial Association and Co-location Patterns, Spatial Clustering Techniques: CLARANS Extension, Web Mining: Web Content Mining, Web Structure Mining, Web Usage mining, Applications of Web Mining.

Text Books:

1. Paulraj Ponniah, —Data Warehousing: Fundamentals for IT Professionals, Wiley India.

2. Reema Theraja —Data warehousing, Oxford University Press.

3. M.H. Dunham, "Data Mining Introductory and Advanced Topics", Pearson Education.

Reference Books:

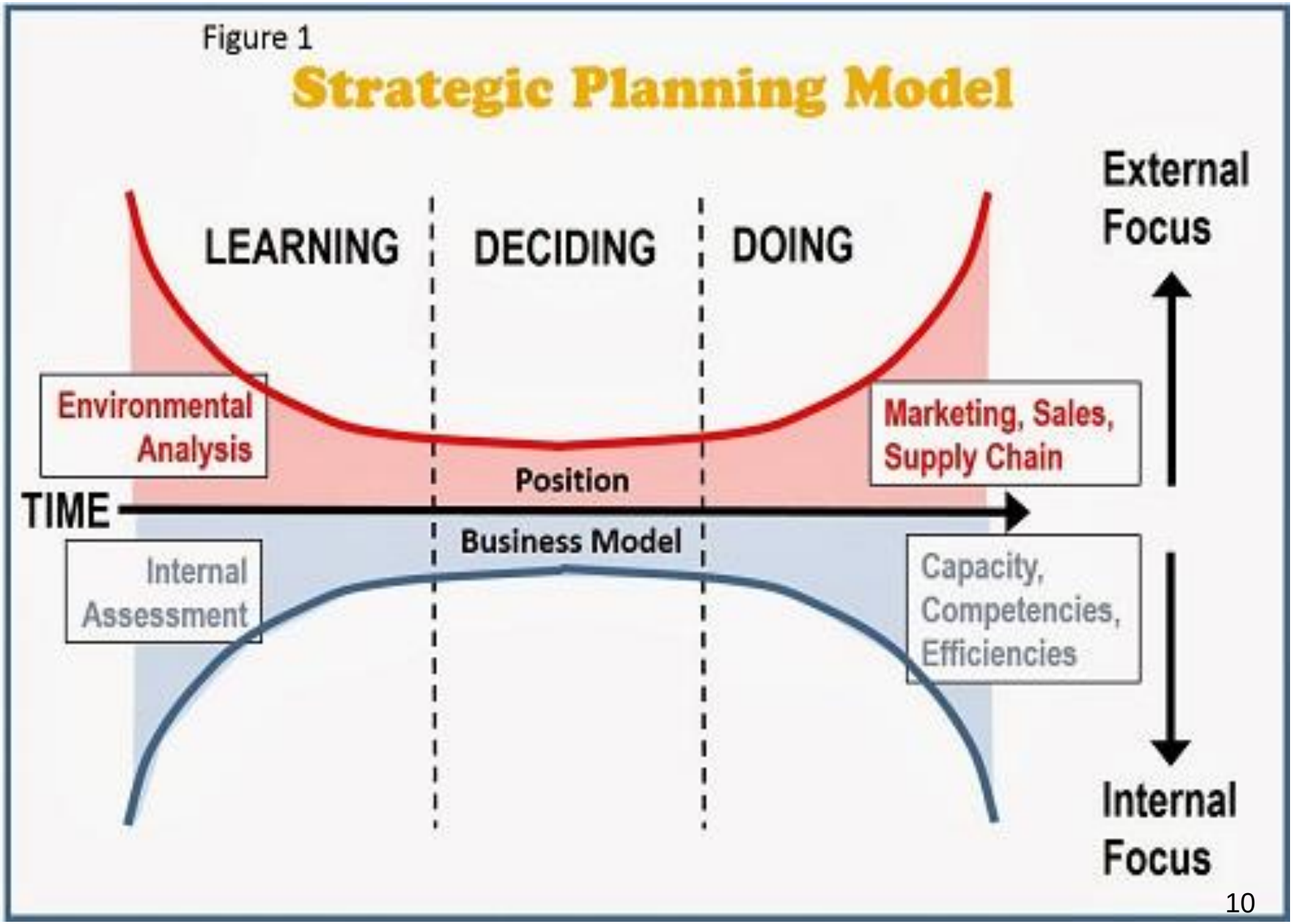
1. Han, Jiawei, Jian Pei, and Micheline Kamber. Data mining: concepts and techniques. Elsevier, 3rd Edition, 2011.

Outline

- Introduction to Strategic Information, Need for Strategic Information, Features of Data Warehouse,
- Data Warehouse versus Data Marts, Data Warehouse versus Data Lake, Top Down versus Bottom Up Approach.
- Data Warehouse Architecture, Metadata, E-R Modelling versus Dimensional Modelling,
- Information Package Diagram, STAR Schema, STAR Schema keys, Snowflake Schema,
- Fact Constellation Schema, Factless Fact Tables, Update to the Dimension Tables, Aggregate Fact Tables.

- **Introduction to Strategic Information:**
  - All organizations plan their strategy according to their needs and requirements.
  - The strategic information refers to what an organization wants to achieve in the short or long term.
  - The following is the input to formulate the strategic information of an organization:
  - **External input:** Macroeconomic environment, what competitors are doing, change in government policies, etc.
  - **Internal input:** Company vision and mission, top management input, audits and feedback, learning from the past, future challenges, etc.

Figure 1


Strategic Planning Model
LEARNING
DECIDING
DOING
External Focus
Environmental Analysis
Marketing, Sales, Supply Chain
Position
TIME
Business Model
Internal Assessment
Capacity, Competencies, Efficiencies
Internal Focus

- **Introduction to Strategic Information:**
  - Strategy refers to the “what” and “why” a company plans to do in the future.
  - Strategy formulation involves consideration of all type of external and internal input while tactics is actually the actions to implement the strategy.
  - Strategic information is needed for long-term planning and directions.
  - Strategic information involves a period generally up to five years while tactical information involves a period of up to a year.

- **Need for Strategic Information**
  - The strategic information is information that plays a vital role while taking decisions which helps the company in staying competitive in the market.
  - The business executives and managers of the company require this strategic information as they are involved in making decisions to keep the company competitive.
  - They utilize this strategic information for setting up the goals and objectives, analyzing the results, formulate new business strategies etc.
  - Strategic information briefs the executives and managers about the operations of the company.

- **Need for Strategic Information**
  - Strategic information let the executives and managers know what does the customer need and how their preferences change over time.
  - They should also consider emerging technologies as every company must implement new technologies to proceed in the competitive market.
  - The strategic information is not required to perform the daily routine operation like billing payment, preparing a claim, settling a claim, preparing an invoice, managing account.
  - It is used to make a decision that keeps the company competitive in the market.

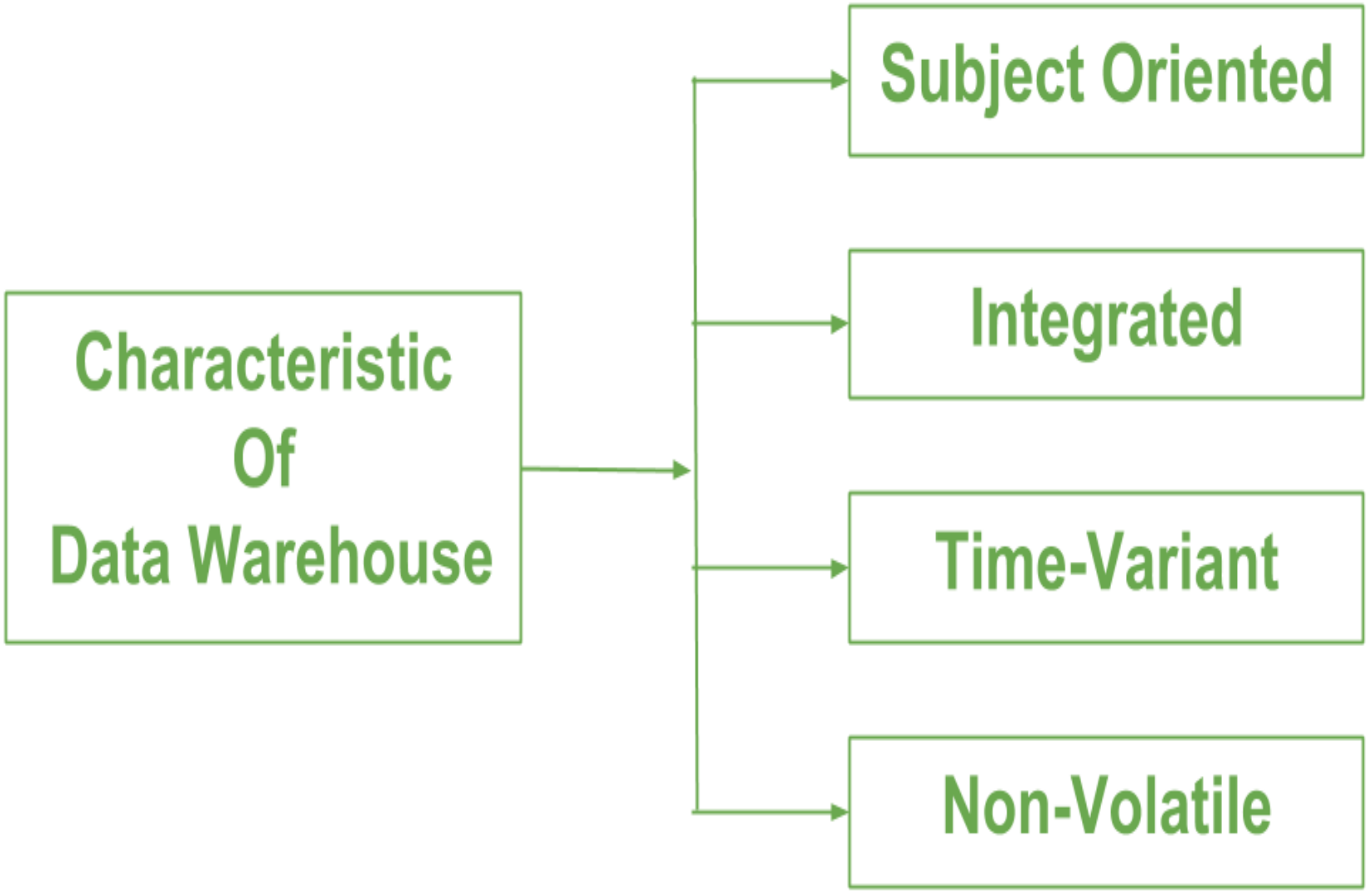
- **Need for Strategic Information**
  - If your strategic information is not properly integrated or analyzed it would lead to making an improper decision which may not be fruitful for the company.

- **Features of Data Warehouse**
  - A **data warehouse** is a relational or multidimensional database that is designed for query and analysis.
  - Data warehouses are not optimized for transaction processing, which is the domain of OLTP systems.
  - Data warehouses usually consolidate historical and analytic data derived from multiple sources.
  - Data warehouses separate analysis workload from transaction workload and enable an organization to consolidate data from several sources.
  - A data warehouse usually stores many months or years of data to support historical analysis.

- **Features of Data Warehouse**
  - The data in a data warehouse is typically loaded through an extraction, transformation, and loading (ETL) process from one or more data sources such as OLTP applications, mainframe applications, or external data providers.
  - Users of the data warehouse perform data analyses that are often time-related.
  - Examples include consolidation of last year's sales figures, inventory analysis, and profit by product and by customer.
  - More sophisticated analyses include trend analyses and data mining, which use existing data to forecast trends or predict futures.

- **Features of Data Warehouse**
  - The data warehouse typically provides the foundation for a business intelligence environment.

- **The Key Characteristics of a Data Warehouse**
  - Some data is deformalized for simplification and to improve performance.
  - Large amounts of historical data are used.
  - Queries often retrieve large amounts of data.
  - Both planned and ad hoc queries are common.
  - The data load is controlled.
  - In general, fast query performance with high data throughput is the key to a successful data warehouse.


Characteristic
Of
Data Warehouse
Subject Oriented
Integrated
Time-Variant
Non-Volatile

- **The Key Characteristics of a Data Warehouse**
  - **Subject-oriented**
  - A data warehouse is always a subject oriented as it delivers information about a theme instead of organization's current operations.
  - It can be achieved on specific theme.
  - That means the data warehousing process is proposed to handle with a specific theme which is more defined.
  - These themes can be sales, distributions, marketing etc.

- **The Key Characteristics of a Data Warehouse**
  - **Subject-oriented**
  - A data warehouse is always a subject oriented as it delivers information about a theme instead of organization's current operations.
  - It can be achieved on specific theme.
  - That means the data warehousing process is proposed to handle with a specific theme which is more defined.
  - These themes can be sales, distributions, marketing etc.
  - A data warehouse never put emphasis only current operations. Instead, it focuses on demonstrating and analysis of data to make various decision.

- **The Key Characteristics of a Data Warehouse**
  - **Integrated –**
  - It is somewhere same as subject orientation which is made in a reliable format.
  - Integration means founding a shared entity to scale the all similar data from the different databases.
  - The data also required to be resided into various data warehouse in shared and generally granted manner.
  - A data warehouse is built by integrating data from various sources of data such that a mainframe and a relational database.

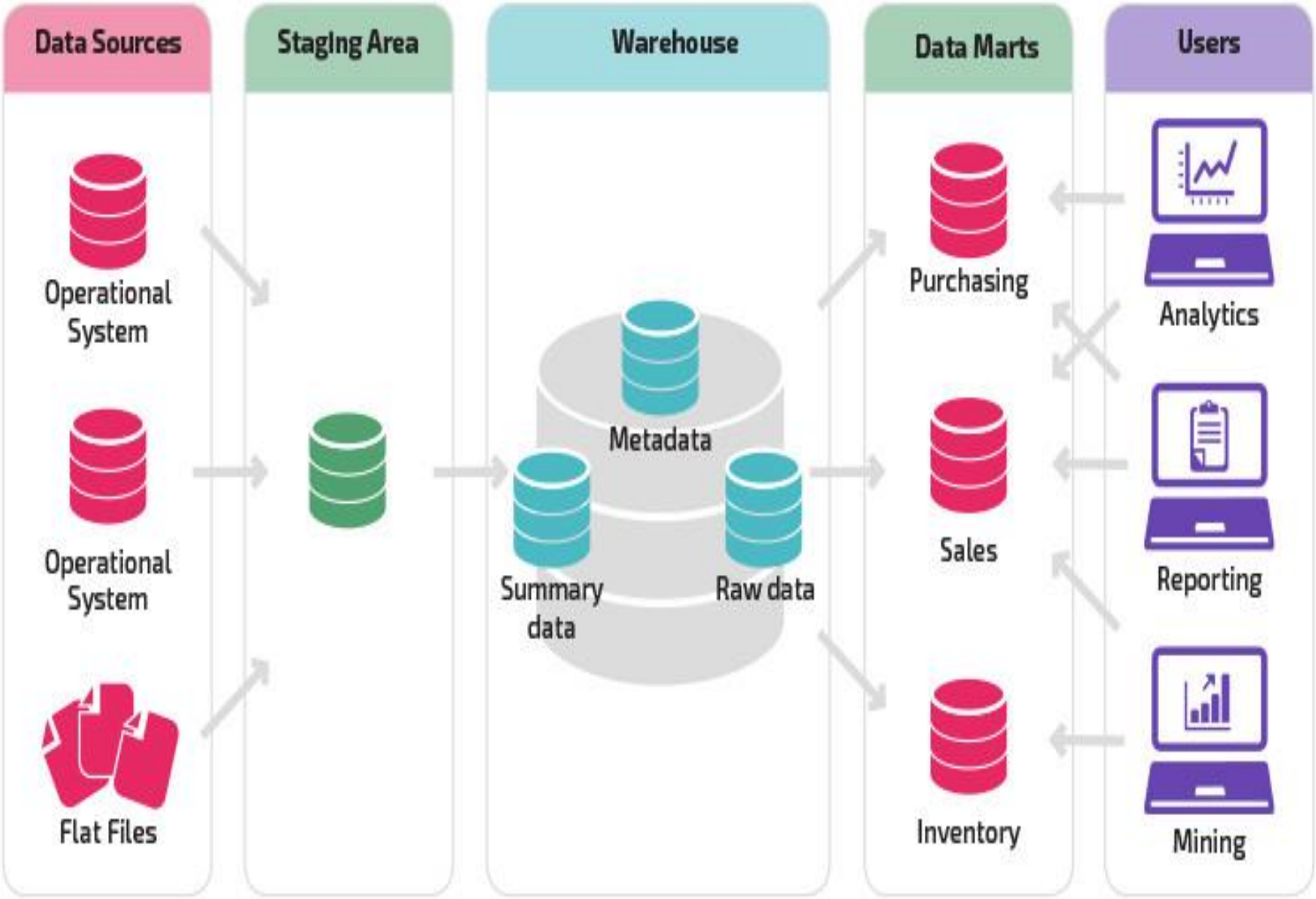
- **The Key Characteristics of a Data Warehouse**
  - **Time-Variant** –

    In this data is maintained via different intervals of time such as weekly, monthly, or annually etc.
  - It founds various time limit which are structured between the large datasets and are held in online transaction process (OLTP).
  - The time limits for data warehouse is wide-ranged than that of operational systems.
  - The data resided in data warehouse is predictable with a specific interval of time and delivers information from the historical perspective.

- **The Key Characteristics of a Data Warehouse**
  - **Non-Volatile –**
  - As the name defines the data resided in data warehouse is permanent.
  - It also means that data is not erased or deleted when new data is inserted.
  - It includes the mammoth quantity of data that is inserted into modification between the selected quantity on logical business.
  - It evaluates the analysis within the technologies of warehouse.

- **Functions of Data warehouse**
  - It works as a collection of data and here is organized by various communities that endures the features to recover the data functions.
  - It has stocked facts about the tables which have high transaction levels which are observed so as to define the data warehousing techniques and major functions which are involved in this are mentioned below,
    - Data consolidation
    - Data Cleaning
    - Data Integration

- **Data Warehouse versus Data Marts**
  - A **data mart** is a subset of a data warehouse oriented to a specific business line.
  - Data marts contain repositories of summarized data collected for analysis on a specific section or unit within an organization, for example, the sales department.
  - A **data warehouse** is a large centralized repository of data that contains information from many sources within an organization.
  - The collated data is used to guide business decisions through analysis, reporting, and data mining tools.


Data Sources
Staging Area
Warehouse
Data Marts
Users
Operational System
Operational System
Flat Files
Metadata
Summary data
Raw data
Purchasing
Sales
Inventory
Analytics
Reporting
Mining

- **Data Warehouse versus Data Marts**
  - **Data Mart**
  - **Focus**: A single subject or functional organization area
  - **Data Sources**: Relatively few sources linked to one line of business
  - **Size**: Less than 100 GB
  - **Normalization**: No preference between a normalized and denormalized structure
  - **Decision Types**: Tactical decisions pertaining to particular business lines and ways of doing things
  - **Cost**: Typically from $10,000 upwards
  - **Setup Time**: 3-6 months

- **Data Warehouse versus Data Marts**
  - **Data Warehouse**
  - **Focus**: Enterprise-wide repository of disparate data sources
  - **Data Sources**: Many external and internal sources from different areas of an organization
  - **Size**: 100 GB minimum but often in the range of terabytes for large organizations
  - **Normalization**: Modern warehouses are mostly denormalized for quicker data querying and read performance
  - **Decision Types**: Strategic decisions that affect the entire enterprise
  - **Cost**: Varies but often greater than $100,000; for cloud solutions costs can be dramatically lower as organizations pay per use
  - **Setup Time**: At least a year for on-premise warehouses; cloud data warehouses are much quicker to set up
  - **Data Held**: Raw data, metadata, and summary data

- **Data Warehouse versus Data Marts**
  - **Data Warehouse**
  - **Focus**: Enterprise-wide repository of disparate data sources
  - **Data Sources**: Many external and internal sources from different areas of an organization
  - **Size**: 100 GB minimum but often in the range of terabytes for large organizations
  - **Normalization**: Modern warehouses are mostly denormalized for quicker data querying and read performance
  - **Decision Types**: Strategic decisions that affect the entire enterprise
  - **Cost**: Varies but often greater than $100,000; for cloud solutions costs can be dramatically lower as organizations pay per use
  - **Setup Time**: At least a year for on-premise warehouses; cloud data warehouses are much quicker to set up
  - **Data Held**: Raw data, metadata, and summary data

- Data Warehouse versus Data Marts

S.NO	Data Warehouse	Data Mart
1.	Data warehouse is a Centralised system.	While it is a decentralised system.
2.	In data warehouse, lightly denormalization takes place.	While in Data mart, highly denormalization takes place.
3.	Data warehouse is top-down model.	While it is a bottom-up model.
4.	To built a warehouse is difficult.	While to build a mart is easy.
5.	In data warehouse, Fact constellation schema is used.	While in this, Star schema and snowflake schema are used.
6.	Data Warehouse is flexible.	While it is not flexible.

- Data Warehouse versus Data Marts

9.	In Data Warehouse, Data are contained in detail form.	While in this, data are contained in summarized form.
10.	Data Warehouse is vast in size.	While data mart is smaller than warehouse.
11.	It collects data from various data sources.	It generally stores data from a data warehouse.
12.	Long time for processing the data because of large data.	Less time for processing the data because of handling only a small amount of data.

- **Data Warehouse versus Data Lake**
  - **What is Data Warehouse?**
  - Data Warehouse is a blend of technologies and components for the strategic use of data.
  - It collects and manages data from varied sources to provide meaningful business insights.
  - It is the electronic storage of a large amount of information designed for query and analysis instead of transaction processing.
  - It is a process of transforming data into information.

- **Data Warehouse versus Data Lake**
  - **What is Data Lake?**
  - A **Data Lake** is a storage repository that can store a large amount of structured, semi-structured, and unstructured data.
  - It is a place to store every type of data in its native format with no fixed limits on account size or file.
  - It offers a large amount of data quantity for increased analytical performance and native integration.
  - Data Lake is like a large container which is very similar to real lake and rivers.
  - Just like in a lake, you have multiple tributaries coming in; similarly, a data lake has structured data, unstructured data, machine to machine, logs flowing through in real-time.

- **Data Warehouse versus Data Lake**
  - **Key Difference**
  - Data Lake stores all data irrespective of the source and its structure whereas Data Warehouse stores data in quantitative metrics with their attributes.
  - Data Lake is a storage repository that stores huge structured, semi-structured and unstructured data while Data Warehouse is blending of technologies and component which allows the strategic use of data.
  - Data Lake defines the schema after data is stored whereas Data Warehouse defines the schema before data is stored.

- **Data Warehouse versus Data Lake**
  - **Key Difference**
  - Data Lake uses the ELT(Extract Load Transform) process while the Data Warehouse uses ETL(Extract Transform Load) process.
  - Comparing Data lake vs Warehouse, Data Lake is ideal for those who want in-depth analysis whereas Data Warehouse is ideal for operational users.

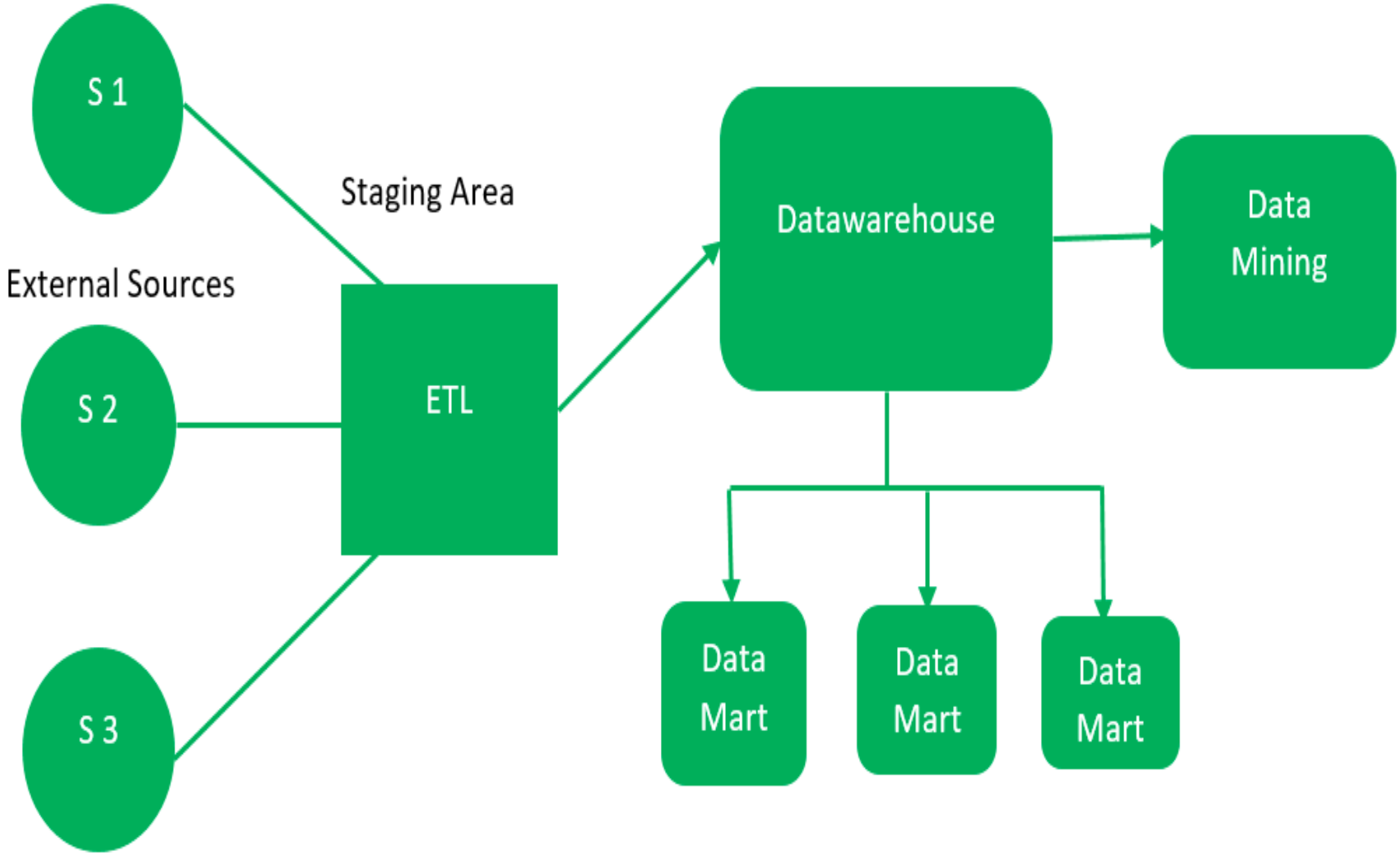
Parameters	Data Lake	Data Warehouse
Storage	In the data lake, all data is kept irrespective of the source and its structure. Data is kept in its raw form. It is only transformed when it is ready to be used.	A data warehouse will consist of data that is extracted from transactional systems or data which consists of quantitative metrics with their attributes. The data is cleaned and transformed
History	Big data technologies used in data lakes is relatively new.	Data warehouse concept, unlike big data, had been used for decades.

Data Timeline	Data lakes can retain all data. This includes not only the data that is in use but also data that it might use in the future. Also, data is kept for all time, to go back in time and do an analysis.	In the data warehouse development process, significant time is spent on analyzing various data sources.
Users	Data lake is ideal for the users who indulge in deep analysis. Such users include data scientists who need advanced analytical tools with capabilities such as predictive modeling and statistical analysis.	The data warehouse is ideal for operational users because of being well structured, easy to use and understand.

Task	Data lakes can contain all data and data types; it empowers users to access data prior the process of transformed, cleansed and structured.	Data warehouses can provide insights into pre-defined questions for pre-defined data types.
Processing time	Data lakes empower users to access data before it has been transformed, cleansed and structured. Thus, it allows users to get to their result more quickly compares to the traditional data warehouse.	Data warehouses offer insights into pre-defined questions for pre-defined data types. So, any changes to the data warehouse needed more time.

Data processing	Data Lakes use of the ELT (Extract Load Transform) process.	Data warehouse uses a traditional ETL (Extract Transform Load) process.
Complain	Data is kept in its raw form. It is only transformed when it is ready to be used.	The chief complaint against data warehouses is the inability, or the problem faced when trying to make change in in them.
Key Benefits	They integrate different types of data to come up with entirely new questions as these users not likely to use	Most users in an organization are operational. These type of

- **Top Down versus Bottom Up Approach.**
  - A data-warehouse is a heterogeneous collection of different data sources organized under a unified schema.
  - There are 2 approaches for constructing data-warehouse: Top-down approach and Bottom-up approach are explained as below.
  - **Top-down approach:**


S 1
Staging Area
External Sources
S 2
ETL
S 3
Datawarehouse
Data
Mining
Data
Mart
Data
Mart
Data
Mart

- **Top Down versus Bottom Up Approach.**
  - The essential components are discussed below:
  - **External Sources –**
  - External source is a source from where data is collected irrespective of the type of data.
  - Data can be structured, semi structured and unstructured as well.
  - **Stage Area –**
  - Since the data, extracted from the external sources does not follow a particular format, so there is a need to validate this data to load into datawarehouse.
  - For this purpose, it is recommended to use ETL tool.

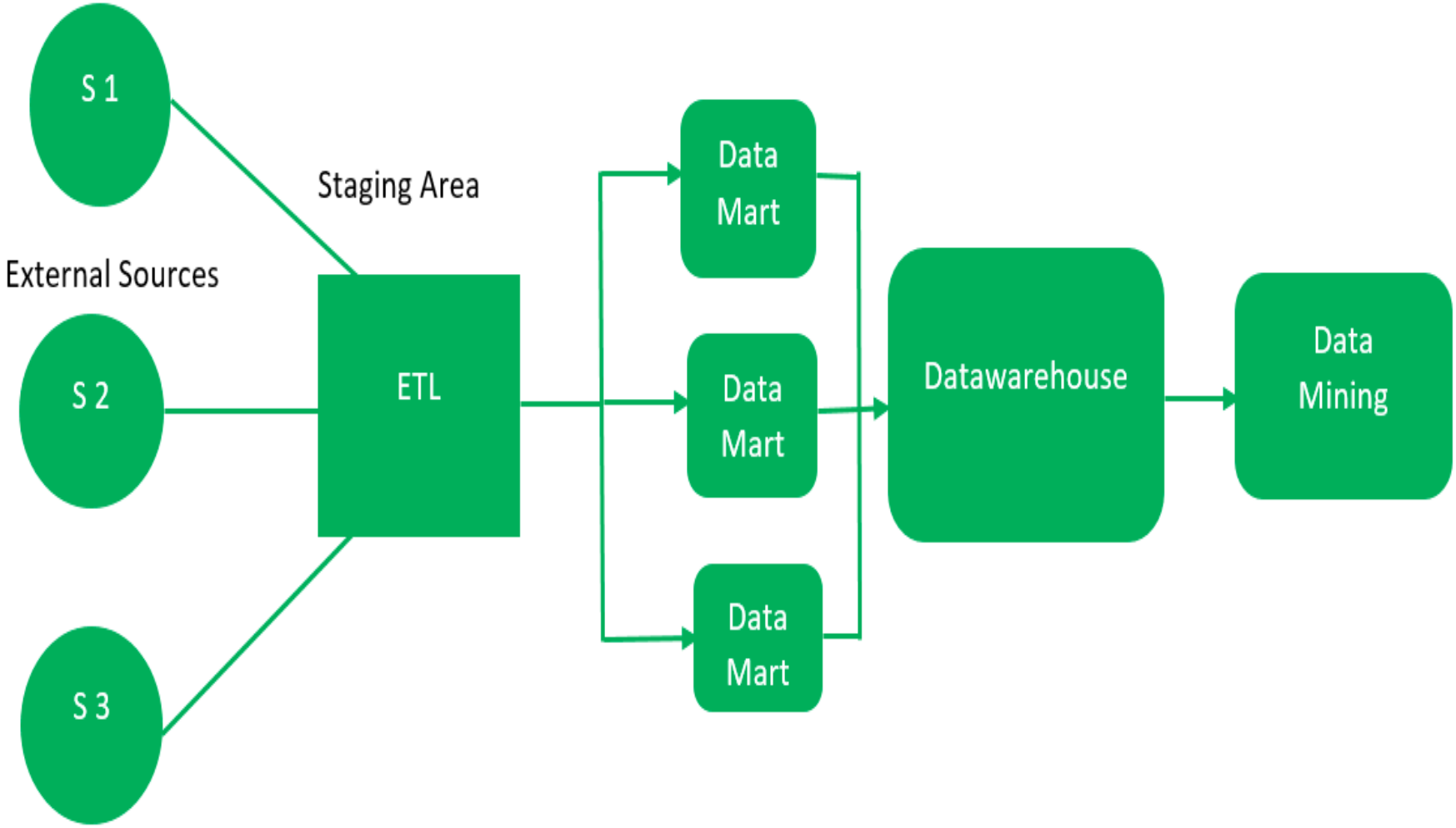
- **Top Down versus Bottom Up Approach.**
  - E(Extracted): Data is extracted from External data source.
  - T(Transform): Data is transformed into the standard format.
  - L(Load): Data is loaded into datawarehouse after transforming it into the standard format.
  - **Data-warehouse –**
  - After cleansing of data, it is stored in the datawarehouse as central repository.
  - It actually stores the meta data and the actual data gets stored in the data marts.
  - Note that datawarehouse stores the data in its purest form in this top-down approach.

- **Top Down versus Bottom Up Approach.**
  - **Data Marts –**
  - Data mart is also a part of storage component.
  - It stores the information of a particular function of an organization which is handled by single authority.
  - There can be as many number of data marts in an organization depending upon the functions.
  - We can also say that data mart contains subset of the data stored in datawarehouse.

- **Top Down versus Bottom Up Approach.**
  - **Data Mining –**
  - The practice of analyzing the big data present in datawarehouse is data mining.
  - It is used to find the hidden patterns that are present in the database or in datawarehouse with the help of algorithm of data mining.
  - This approach is defined by Inmon as – datawarehouse as a central repository for the complete organization and data marts are created from it after the complete datawarehouse has been created.

- **Top Down versus Bottom Up Approach.**
  - **Advantages of Top-Down Approach –**
  - Since the data marts are created from the datawarehouse, provides consistent dimensional view of data marts.
  - Also, this model is considered as the strongest model for business changes. That's why, big organizations prefer to follow this approach.
  - Creating data mart from datawarehouse is easy.
  - **Disadvantages of Top-Down Approach –**
  - The cost, time taken in designing and its maintenance is very high.

- Top Down versus Bottom Up Approach.
  - Bottom-up approach:



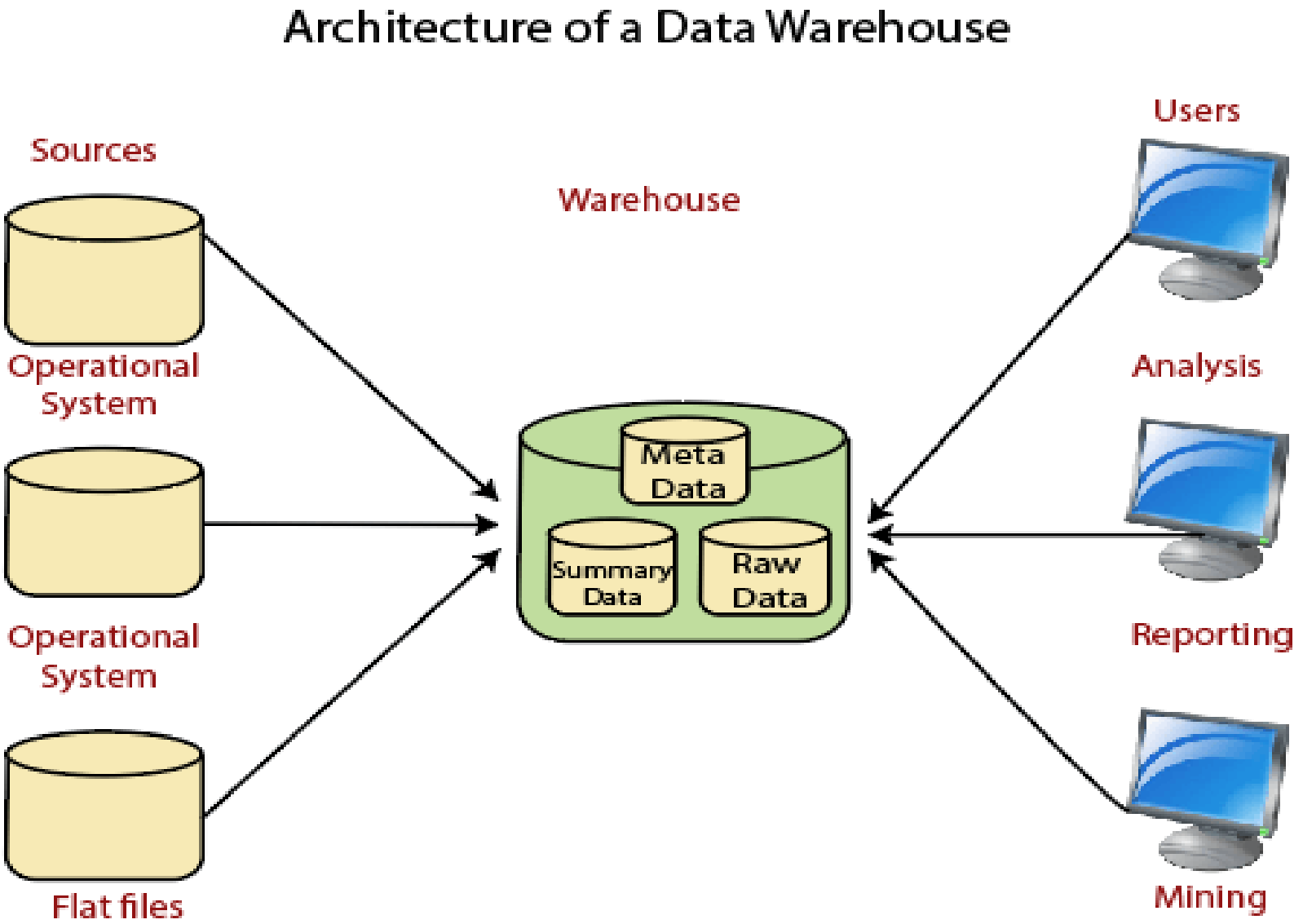
- **Top Down versus Bottom Up Approach.**
  - First, the data is extracted from external sources (same as happens in top-down approach).
  - Then, the data go through the staging area (as explained above) and loaded into data marts instead of datawarehouse.
  - The data marts are created first and provide reporting capability. It addresses a single business area.
  - These data marts are then integrated into datawarehouse.
  - This approach is given by **Kinball** as – data marts are created first and provides a thin view for analyses and datawarehouse is created after complete data marts have been created.

- **Top Down versus Bottom Up Approach.**
  - **Advantages of Bottom-Up Approach**
  - As the data marts are created first, so the reports are quickly generated.
  - We can accommodate more number of data marts here and in this way datawarehouse can be extended.
  - Also, the cost and time taken in designing this model is low comparatively.
  - **Disadvantage of Bottom-Up Approach**
  - This model is not strong as top-down approach as dimensional view of data marts is not consistent as it is in above approach.

- **Data Warehouse Architecture**
  - A data warehouse architecture is a method of defining the overall architecture of data communication processing and presentation that exist for end-clients computing within the enterprise.
  - Each data warehouse is different, but all are characterized by standard vital components.
  - Production applications such as payroll accounts payable product purchasing and inventory control are designed for online transaction processing (OLTP).
  - Such applications gather detailed data from day to day operations.

- **Data Warehouse Architecture**
  - Data Warehouse applications are designed to support the user ad-hoc data requirements, an activity recently dubbed online analytical processing (OLAP).
  - These include applications such as forecasting, profiling, summary reporting, and trend analysis.
  - Data warehouses and their architectures very depending upon the elements of an organization's situation.
  - Three common architectures are:
    - Data Warehouse Architecture: Basic
    - Data Warehouse Architecture: With Staging Area
    - Data Warehouse Architecture: With Staging Area and Data Marts

- Data Warehouse Architecture: Basic



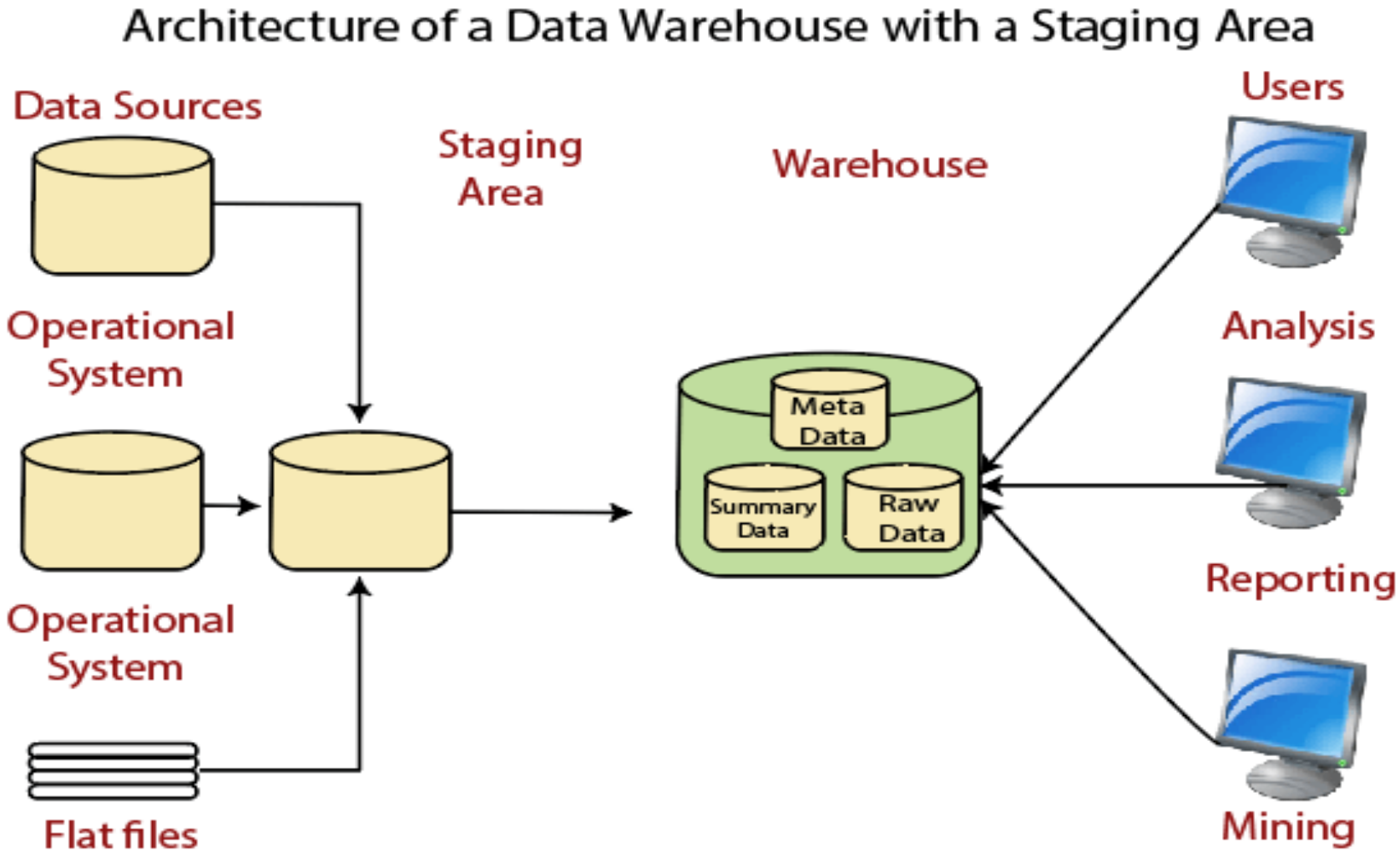
- **Data Warehouse Architecture**
  - **Operational System**
  - An operational system is a method used in data warehousing to refer to a system that is used to process the day-to-day transactions of an organization.
  - **Flat Files**
  - A Flat file system is a system of files in which transactional data is stored, and every file in the system must have a different name.
  - **Meta Data**
  - A set of data that defines and gives information about other data.

- **Data Warehouse Architecture**
  - Meta Data used in Data Warehouse for a variety of purpose, including:
  - Meta Data summarizes necessary information about data, which can make finding and work with particular instances of data more accessible.
  - For example, author, data build, and data changed, and file size are examples of very basic document metadata.
  - Metadata is used to direct a query to the most appropriate data source.

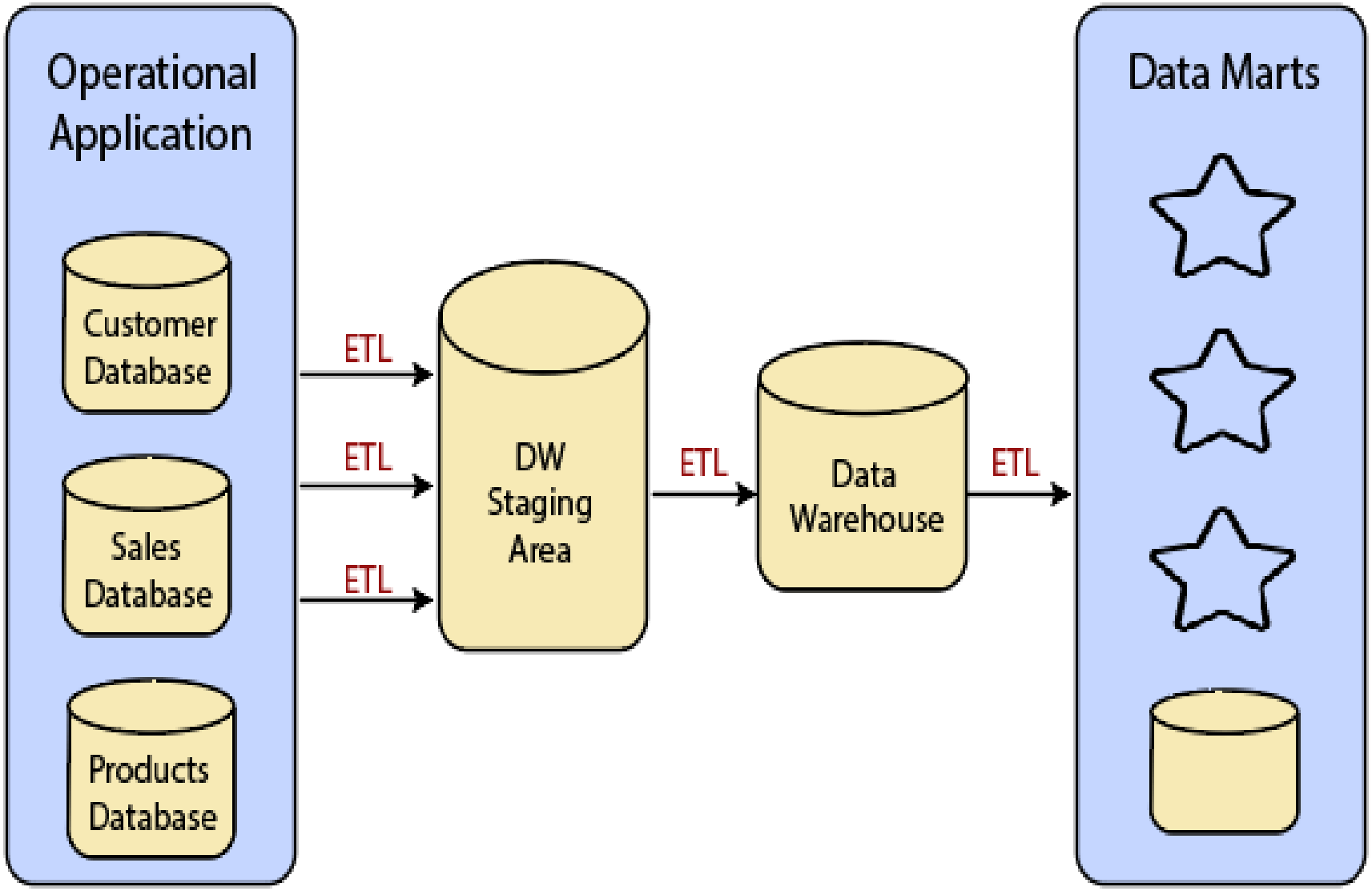
- **Data Warehouse Architecture**
  - **Lightly and highly summarized data**
  - The area of the data warehouse saves all the predefined lightly and highly summarized (aggregated) data generated by the warehouse manager.
  - The goals of the summarized information are to speed up query performance.
  - The summarized record is updated continuously as new information is loaded into the warehouse.

- **Data Warehouse Architecture**
  - **End-User access Tools**
  - The principal purpose of a data warehouse is to provide information to the business managers for strategic decision-making.
  - These customers interact with the warehouse using end-client access tools.
  - The examples of some of the end-user access tools can be:
    - Reporting and Query Tools
    - Application Development Tools
    - Executive Information Systems Tools
    - Online Analytical Processing Tools
    - Data Mining Tools

- **Data Warehouse Architecture: With Staging Area**
  - We must clean and process your operational information before put it into the warehouse.
  - We can do this programmatically, although data warehouses uses a staging area (A place where data is processed before entering the warehouse).
  - A staging area simplifies data cleansing and consolidation for operational method coming from multiple source systems, especially for enterprise data warehouses where all relevant data of an enterprise is consolidated.

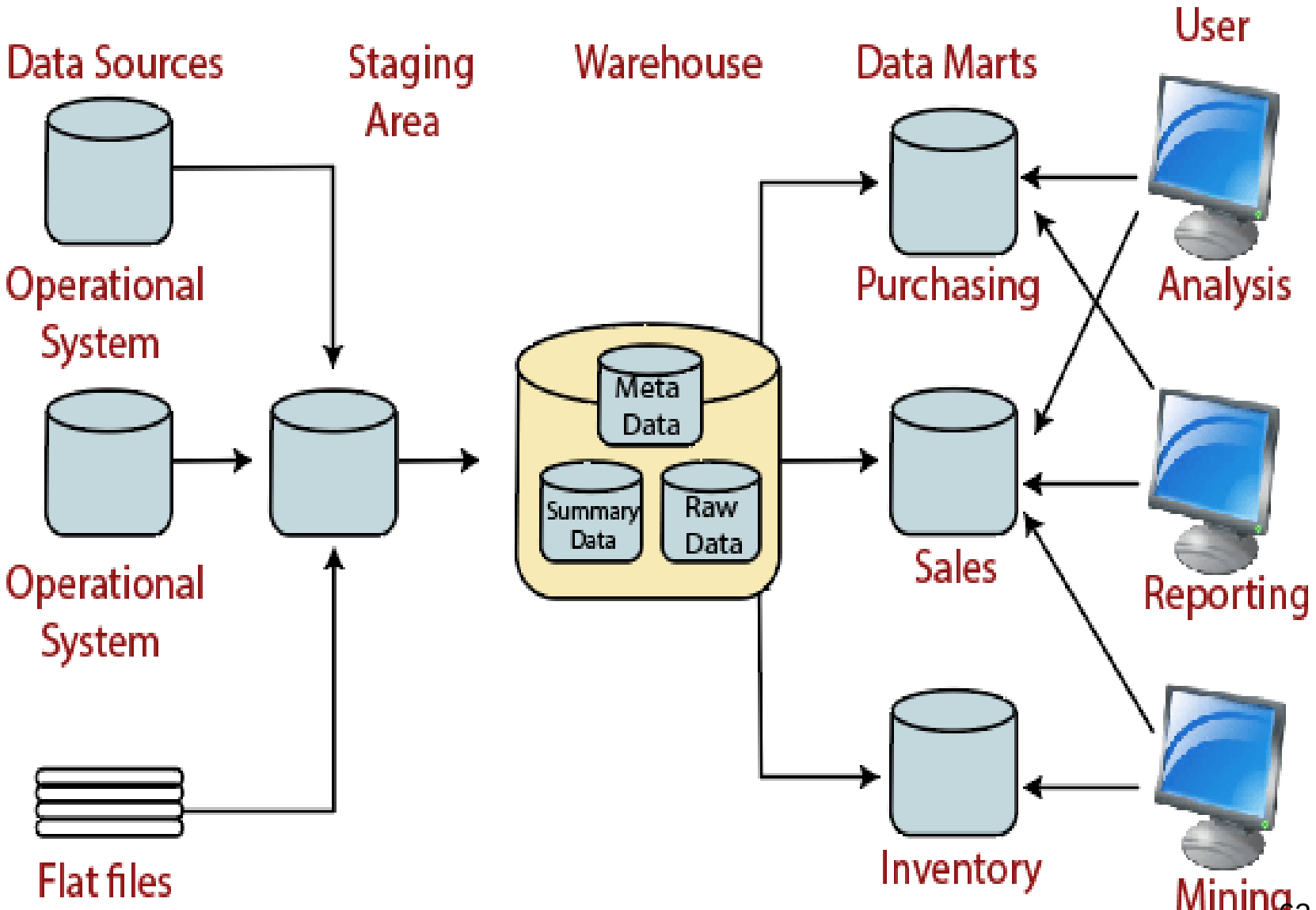


Data Warehouse Staging Area is a temporary location where a record from source systems is copied.

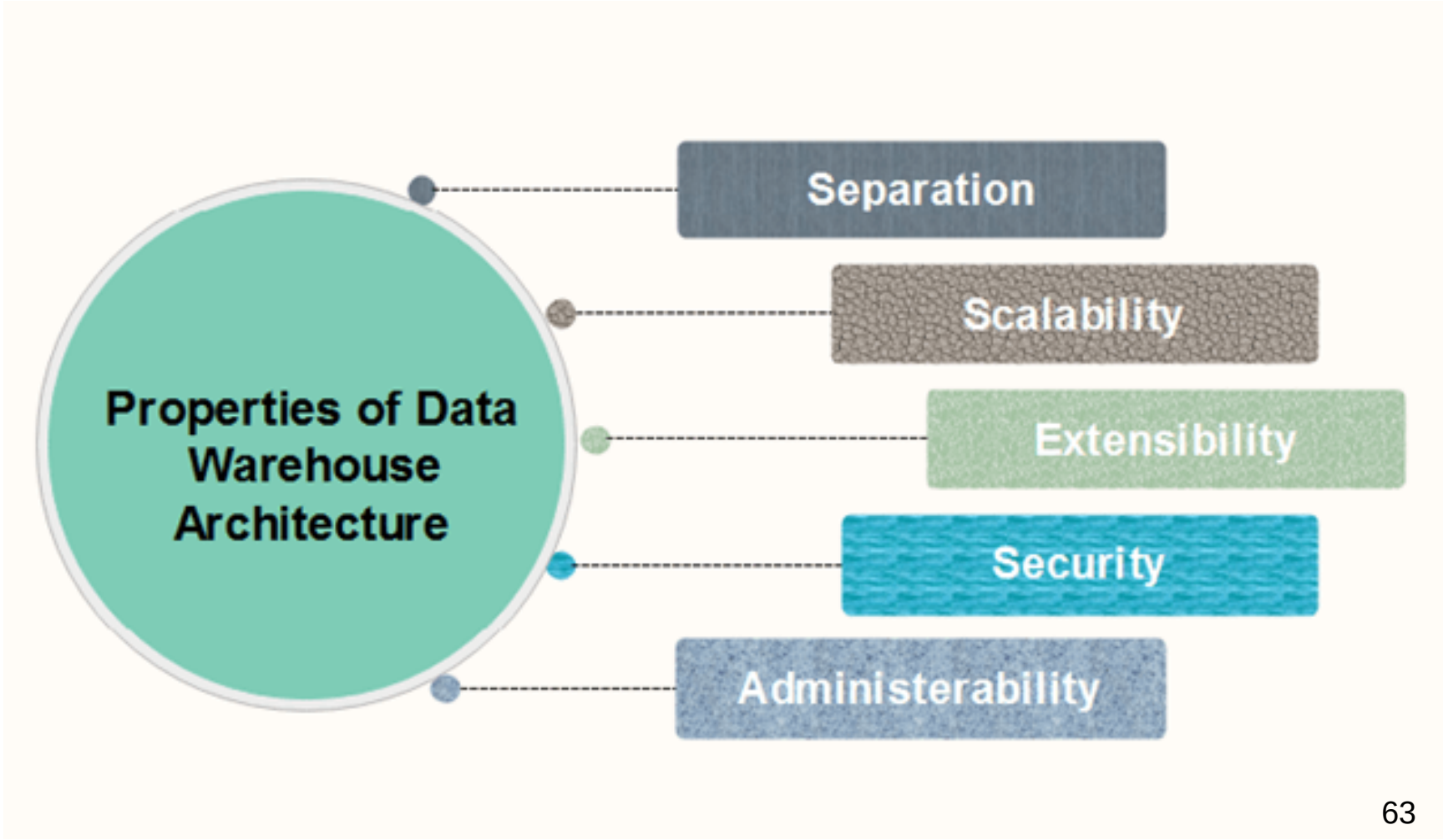

Operational Application
Customer Database
Sales Database
Products Database
ETL
ETL
ETL
DW Staging Area
ETL
Data Warehouse
ETL
Data Marts

- **Data Warehouse Architecture: With Staging Area and Data Marts**
  - We may want to customize our warehouse's architecture for multiple groups within our organization.
  - We can do this by adding **data marts**.
  - A data mart is a segment of a data warehouses that can provided information for reporting and analysis on a section, unit, department or operation in the company, e.g., sales, payroll, production, etc.
  - The figure illustrates an example where purchasing, sales, and stocks are separated. In this example, a financial analyst wants to analyze historical data for purchases and sales or mine historical information to make predictions about customer

Architecture of a Data Warehouse with a Staging Area and Data Marts

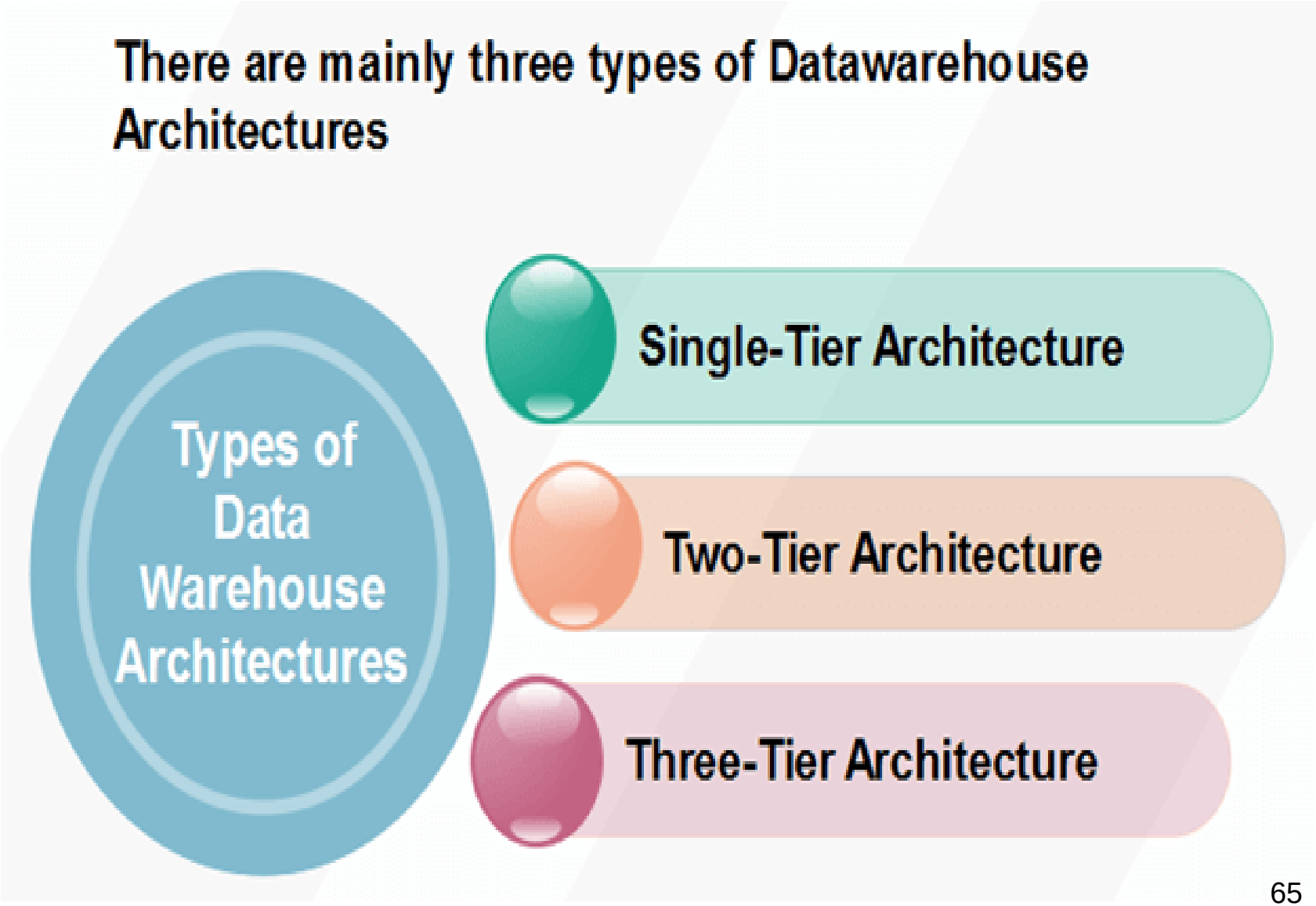


- Properties of Data Warehouse Architectures
  - The following architecture properties are necessary for a data warehouse system:



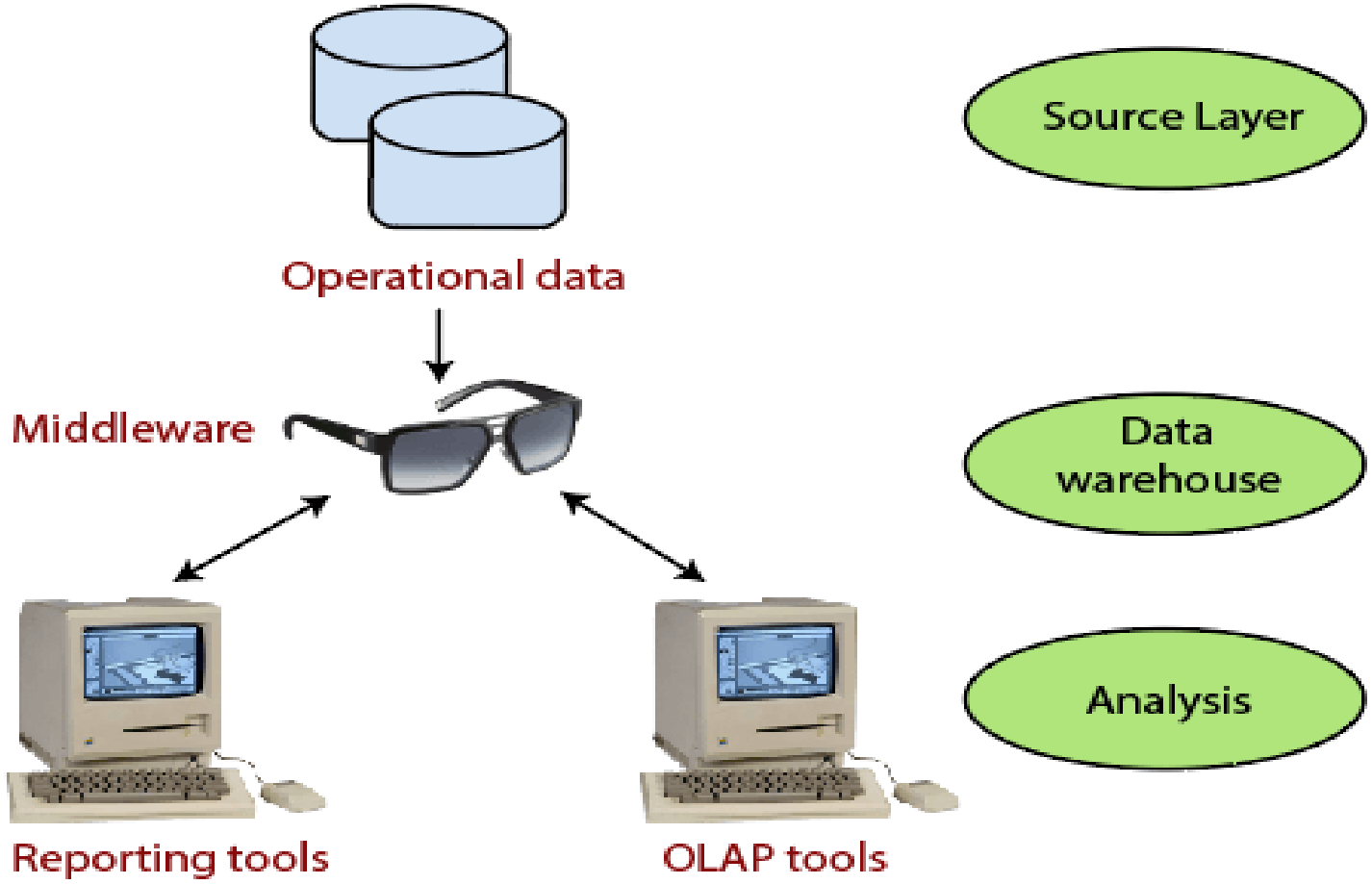
- **Properties of Data Warehouse Architectures**
  - **1. Separation:** Analytical and transactional processing should be keep apart as much as possible.
  - **2. Scalability:** Hardware and software architectures should be simple to upgrade the data volume, which has to be managed and processed, and the number of user's requirements, which have to be met, progressively increase.
  - **3. Extensibility:** The architecture should be able to perform new operations and technologies without redesigning the whole system.
  - **4. Security:** Monitoring accesses are necessary because of the strategic data stored in the data warehouses.
  - 5. **Administerability:** Data Warehouse management should not be complicated.

- Types of Data Warehouse Architectures



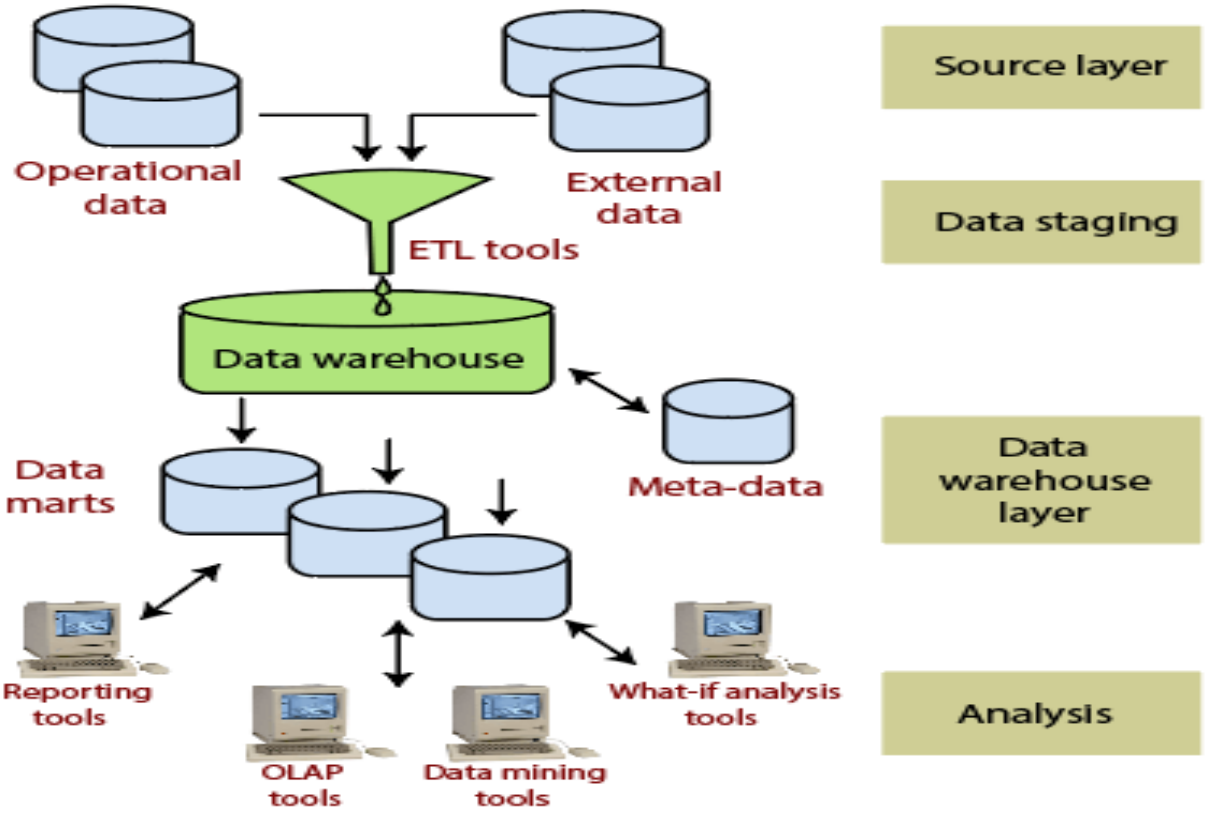
- **Single-Tier Architecture**
  - Single-Tier architecture is not periodically used in practice. Its purpose is to minimize the amount of data stored to reach this goal; it removes data redundancies.
  - The figure shows the only layer physically available is the source layer.
  - In this method, data warehouses are virtual.
  - This means that the data warehouse is implemented as a multidimensional view of operational data created by specific middleware, or an intermediate processing layer.
  - The vulnerability of this architecture lies in its failure to meet the requirement for separation between analytical and transactional processing.

- Single-Tier Architecture



Single-Tier Data Warehouse Architecture

- Two-Tier Architecture
  - The requirement for separation plays an essential role in defining the two-tier architecture for a data warehouse system, as shown in fig:



Two-Tier Data Warehouse Architecture

- **Two-Tier Architecture**
  - Although it is typically called two-layer architecture to highlight a separation between physically available sources and data warehouses, in fact, consists of four subsequent data flow stages:
  - **Source layer:** A data warehouse system uses a heterogeneous source of data. That data is stored initially to corporate relational databases or legacy databases, or it may come from an information system outside the corporate walls.

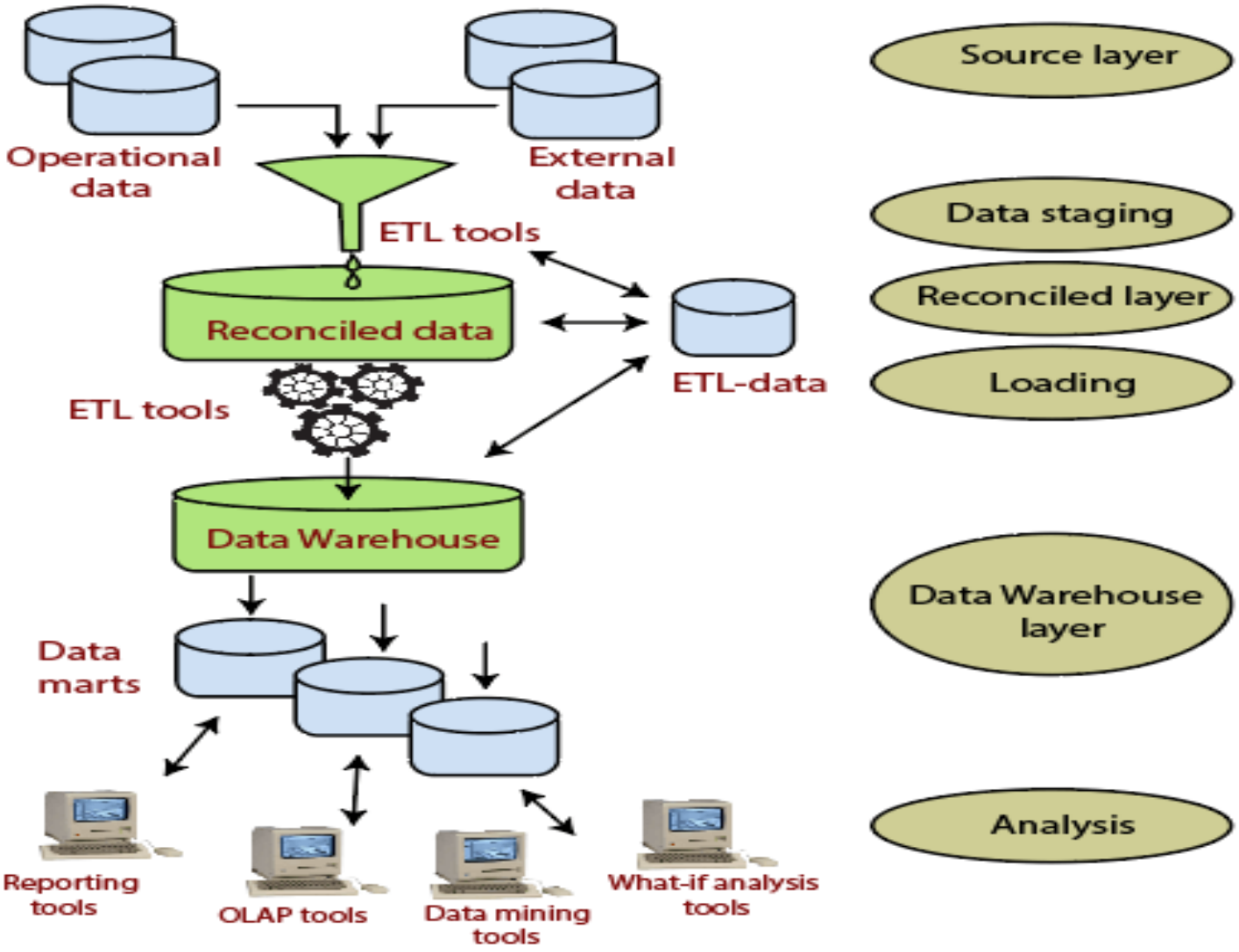
- **Two-Tier Architecture**
  - **Data Staging:** The data stored to the source should be extracted, cleansed to remove inconsistencies and fill gaps, and integrated to merge heterogeneous sources into one standard schema.
  - The so-named Extraction, Transformation, and Loading Tools (ETL) can combine heterogeneous schemata, extract, transform, cleanse, validate, filter, and load source data into a data warehouse.

- **Two-Tier Architecture**
  - **Data Warehouse layer:** Information is saved to one logically centralized individual repository: a data warehouse.
  - The data warehouses can be directly accessed, but it can also be used as a source for creating data marts, which partially replicate data warehouse contents and are designed for specific enterprise departments.
  - Meta-data repositories store information on sources, access procedures, data staging, users, data mart schema, and so on**.**
  - **Analysis:** In this layer, integrated data is efficiently, and flexible accessed to issue reports, dynamically analyze information, and simulate hypothetical business scenarios. It should feature aggregate information navigators, complex query optimizers, and customer-friendly GUIs.

- **Three-Tier Architecture**
  - The three-tier architecture consists of the source layer (containing multiple source system), the reconciled layer and the data warehouse layer (containing both data warehouses and data marts).
  - The reconciled layer sits between the source data and data warehouse.
  - The main advantage of the **reconciled layer** is that it creates a standard reference data model for a whole enterprise.
  - At the same time, it separates the problems of source data extraction and integration from those of data warehouse population.

- **Three-Tier Architecture**
  - In some cases, **the reconciled layer** is also directly used to accomplish better some operational tasks, such as producing daily reports that cannot be satisfactorily prepared using the corporate applications or generating data flows to feed external processes periodically to benefit from cleaning and integration.
  - This architecture is especially useful for the extensive, enterprise-wide systems.
  - A disadvantage of this structure is the extra file storage space used through the extra redundant reconciled layer. It also makes the analytical tools a little further away from being real-time.

- Three-Tier Architecture



Three-Tier Architecture for a data warehouse system

- **Metadata**
  - A crucial area of data warehousing is metadata, which is a kind of data that describes the data warehouse itself.
  - Within a data warehouse, metadata describes and locates data components, their origins (which may be either the
  - operational systems or the data warehouse), and their movement through the data warehouse process.
  - The data access, data stores and processing information will have associated descriptions about the data and processing the inputs, calculations and outputs – documented in the

metadata.

- **Metadata**
  - This metadata should be captured within the data architecture and managed from the beginning of a data warehouse project.
  - The metadata repository should contain information such as that listed below:
    - Description of the data model.
    - Description of the layouts used in the database design.
    - Definition of the primary system managing the data items.
    - A map of the data from the system of record to the other locations in the data warehouse, including the descriptions of transformations and aggregations.
    - Specific database design definitions.
    - Data element definitions, including rules for derivations and summaries.

- **Metadata**
  - It is through metadata that a data warehouse becomes an effective tool for an overall enterprise.
  - This repository of information will tell the story of the data: where it originated, how it has been transformed, where it went and how often that is, its genealogy or artefacts.
  - Technically, the metadata will also improve the maintainability and manageability of a warehouse by making impact analysis information and entity life histories available to the support staff.
  - Equally important, metadata provides interactive access to users to help understand content and find data. Thus, there is a need to create a metadata interface for users.

- **Metadata**
  - One important functional component of the metadata repository is the information directory.
  - The content of the information directory is the metadata that helps users exploit the power of data warehousing.
  - This directory helps integrate, maintain and view the contents of the data warehousing system.
  - From a technical requirements point of view, the information directory and the entire metadata repository should:
    - Be a gateway to the data warehouse environment, and therefore, should be accessible from any platform via transparent and seamless connections.

- **Metadata**
  - Support an easy distribution and replication of its content for high performance and availability.
  - Be searchable by business-oriented keywords,
  - Act as a launch platform for end-user data access and analysis tools.
  - Support the sharing of information objects such as queries, reports, data collections and subscriptions between users.
  - Support a variety of scheduling options for requests against the data warehouse, including on-demand, one-time, repetitive, event-driven and conditional delivery (in conjunction with the information delivery system).

- **Metadata**
  - Support the distribution of query results to one or more destinations in any of the user-specified formats (in conjunction with the information delivery system).
  - Support and provide interfaces to other applications such as e-mail, spreadsheet and schedules.
  - Support end-user monitoring of the status of the data warehouse environment.
  - These requirements define a very sophisticated repository of metadata information.
  - In reality, however, existing products often come up short when implementing these requirements.

- **E-R Modelling versus Dimensional Modelling**
  - **ER model** is used for logical representation or the conceptual view of data.
  - It is a high level of the conceptual data model. It forms a virtual representation of data that describes how all the data are related to each other.
  - It is a complex diagram that is used to represent multiple processes.
  - It helps to describe entities, attributes, and relationships.
  - It helps to analyze data requirements systematically to produce a well-designed database. At the view level, the ER model is considered a good option for designing databases.

- **E-R Modelling versus Dimensional Modelling**
  - Data in a warehouse are usually in the multidimensional form.
  - **Dimensional modeling** prefers keeping the table denormalized.
  - The primary purpose of dimensional modeling is to optimize the database for faster retrieval of the data.
  - The concept of Dimensional Modelling was developed by Ralph Kimball and consists of “fact” and “dimension” tables.
  - The primary purpose of dimensional modeling is to enable business intelligence (BI) reporting, query, and analysis

- **E-R Modelling versus Dimensional Modelling**
  - **Dimensional modeling** is a form of modeling of data that is more flexible from the perspective of the user.
  - These dimensional and relational models have their unique way of data storage that has specific advantages.
  - Dimensional models are built around business processes.
  - They need to ensure that dimension tables use a surrogate key.
  - Dimension tables store the history of the dimensional information.

S.No	ER Modeling	Dimensional Modeling
1	It is transaction-oriented.	It is subject-oriented.
2	Entities and Relationships.	Fact Tables and Dimension Tables.
3	Few levels of granularity.	Multiple levels of granularity.
4	Real-time information.	Historical information.
5	It eliminates redundancy.	It plans for redundancy.

7	Highly Volatile data.	Non-volatile data.
8	Physical and Logical Model.	Physical Model.
9	Normalization is suggested.	De-Normalization is suggested.
10	OLTP Application.	OLAP Application.
Ex	The application is used for buying products from e-commerce websites like Amazon	Application to analyze buying patterns of the customer of the various cities over the past 10 years

- **Information Package Diagram**
  - The presence of information package diagrams in the requirements definition document is the major and significant difference between operational systems and data warehouse systems.
  - Remember that information package diagrams are the best approach for determining requirements for a data warehouse.
  - The information package diagrams crystallize the information requirements for the data warehouse.

- **Information Package Diagram**
  - They contain the critical metrics measuring the performance of the business units, the business dimensions along which the metrics are analyzed, and the details of how drill-down and roll-up analyses are done.
  - Spend as much time as needed to make sure that the information package diagrams are complete and accurate.
  - Your data design for the data warehouse will be totally dependent on the accuracy and adequacy of the information package diagrams.

- Information Packages – novel idea for determining and recording information requirements for a data warehouse.
- Determining requirements for a data warehouse is based on business dimensions
- The relevant dimension and measurements in that dimension are captured and kept in a data warehouse
- This creates an information package for a specific subject

An information Package

Dimensions

Hierarchies

Time Periods	Locations	Products	Age Groups		
Year	Country	Class	Group 1		

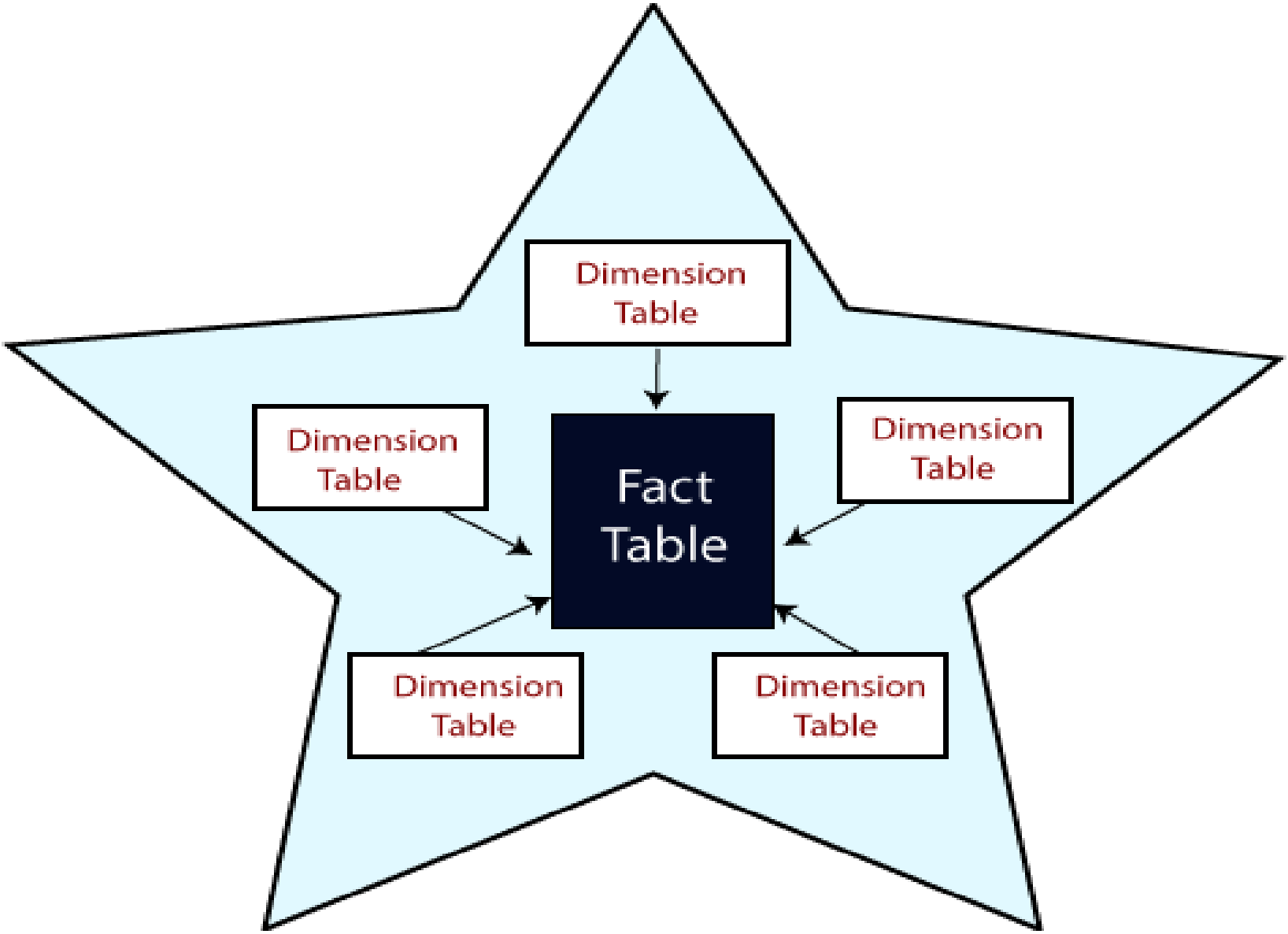
Measured Facts: Forecast Sales, Budget Sales, Actual Sales

IPD enables you to....

- Define the common subject areas
- Design key business metrics
- Decide how data must be presented
- Determine how users will aggregate or roll up
- Decide the data quantity for user analysis or query
- Decide how data will be accessed
- Establish data granularity
- Estimate data warehouse size
- Determine the frequency for data refreshing
- Ascertain how information must be packaged

- **STAR Schema**
  - A star schema is the elementary form of a dimensional model, in which data are organized into facts and dimensions.
  - A **fact** is an event that is counted or measured, such as a sale or log in.
  - A **dimension** includes reference data about the fact, such as date, item, or customer.
  - A star schema is a relational schema where a relational schema whose design represents a multidimensional data model.

- **STAR Schema**
  - The star schema is the explicit data warehouse schema.
  - It is known as star schema because the entity-relationship diagram of this schemas simulates a star, with points, diverge from a central table.
  - The center of the schema consists of a **large fact table, and the points of the star are the dimension tables.**


Dimension Table
Dimension Table
Fact Table
Dimension Table
Dimension Table
Dimension Table

Star Schema

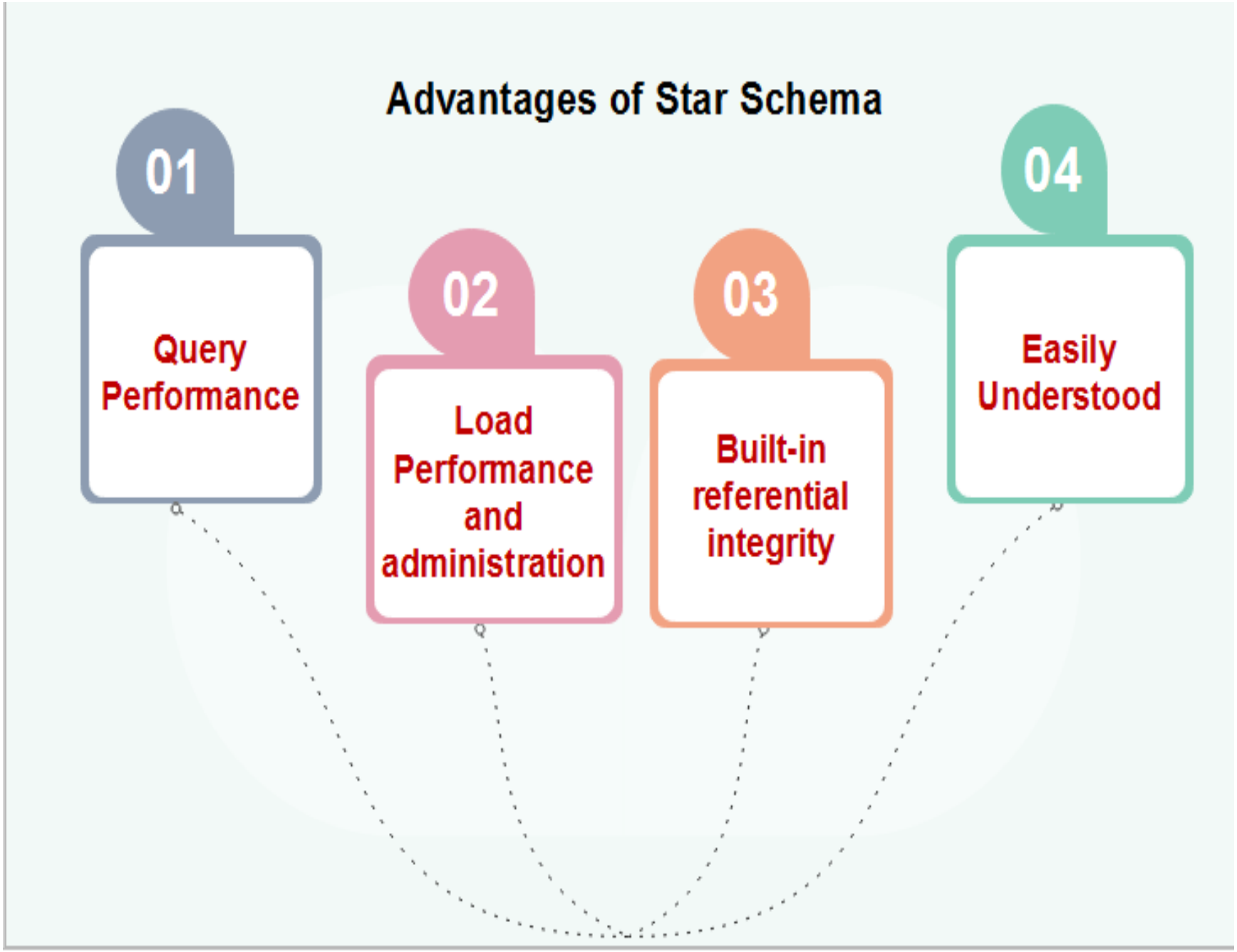
- **STAR Schema**
  - Fact Tables
  - A table in a star schema which contains facts and connected to dimensions.
  - A fact table has two types of columns: those that include fact and those that are foreign keys to the dimension table.
  - The primary key of the fact tables is generally a composite key that is made up of all of its foreign keys.
  - A fact table might involve either detail level fact or fact that have been aggregated (fact tables that include aggregated fact are often instead called summary tables). A fact table generally contains facts with the same level of aggregation.

- **STAR Schema**
  - Dimension Tables
  - A dimension is an architecture usually composed of one or more hierarchies that categorize data.
  - If a dimension has not got hierarchies and levels, it is called a flat dimension or list.
  - The primary keys of each of the dimensions table are part of the composite primary keys of the fact table.
  - Dimensional attributes help to define the dimensional value.
  - They are generally descriptive, textual values.
  - Dimensional tables are usually small in size than fact table.

- **STAR Schema**
  - Dimension Tables
  - Fact tables store data about sales while dimension tables data about the geographic region (markets, cities), clients, products, times, channels.

- **STAR Schema**
  - Characteristics of Star Schema
  - The star schema is intensely suitable for data warehouse database design because of the following features:
    - It creates a DE-normalized database that can quickly provide query responses.
    - It provides a flexible design that can be changed easily or added to throughout the development cycle, and as the database grows.
    - It provides a parallel in design to how end-users typically think of and use the data.
    - It reduces the complexity of metadata for both developers and end-users.

- STAR Schema
  - Advantages of Star Schema
  - Star Schemas are easy for end-users and application to understand and navigate.
  - With a well-designed schema, the customer can instantly analyze large, multidimensional data sets.
  - The main advantage of star schemas in a decision-support environment are:


Advantages of Star Schema
01
Query Performance
02
Load Performance and administration
03
Built-in referential integrity
04
Easily Understood

- **STAR Schema**
  - **Query Performance**
  - A star schema database has a limited number of table and clear join paths, the query run faster than they do against OLTP systems.
  - Small single-table queries, frequently of a dimension table, are almost instantaneous.
  - Large join queries that contain multiple tables takes only seconds or minutes to run.
  - In a star schema database design, the dimension is connected only through the central fact table.

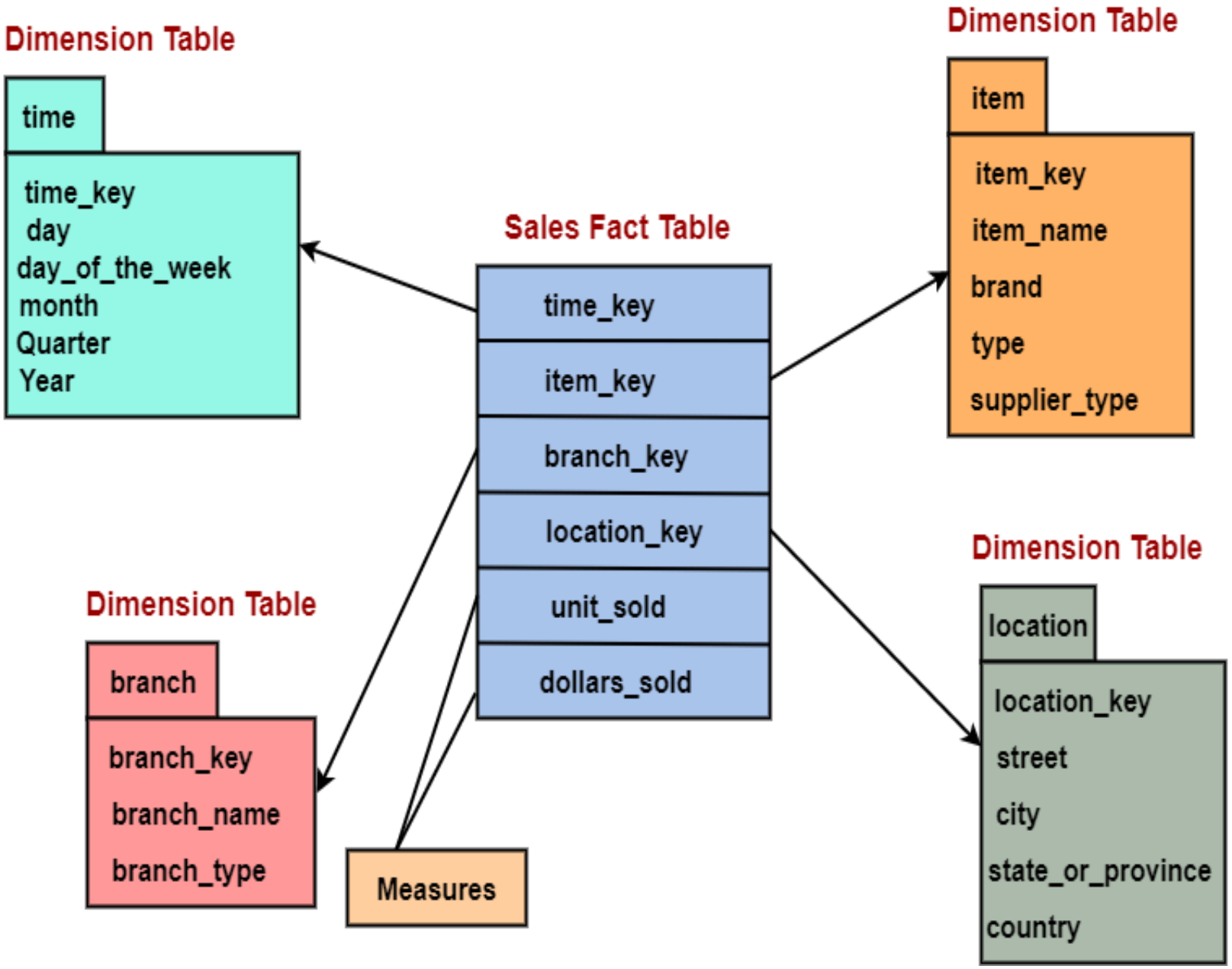
- **STAR Schema**
  - When the two-dimension table is used in a query, only one join path, intersecting the fact tables, exist between those two tables.
  - This design feature enforces authentic and consistent query results.
  - **Load performance and administration**
  - Structural simplicity also decreases the time required to load large batches of record into a star schema database.
  - By describing facts and dimensions and separating them into the various table, the impact of a load structure is reduced.

- **STAR Schema**
  - Dimension table can be populated once and occasionally refreshed.
  - We can add new facts regularly and selectively by appending records to a fact table.
  - **Built-in referential integrity**
  - A star schema has referential integrity built-in when information is loaded.
  - Referential integrity is enforced because each data in dimensional tables has a unique primary key, and all keys in the fact table are legitimate foreign keys drawn from the dimension table.

- **STAR Schema**
  - A record in the fact table which is not related correctly to a dimension cannot be given the correct key value to be retrieved.
  - **Easily Understood**
  - A star schema is simple to understand and navigate, with dimensions joined only through the fact table.
  - These joins are more significant to the end-user because they represent the fundamental relationship between parts of the underlying business.
  - Customer can also browse dimension table attributes before constructing a query.

- **STAR Schema**
  - Disadvantage of Star Schema
  - There is some condition which cannot be meet by star schemas **like the relationship between the user**, and **bank account cannot describe as star schema as the relationship between them is many to many.**
  - Example:
  - Suppose a star schema is composed of a fact table, SALES, and several dimension tables connected to it for time, branch, item, and geographic locations.

- **STAR Schema**
  - The TIME table has a column for each day, month, quarter, and year.
  - The ITEM table has columns for each item_Key, item_name, brand, type, supplier_type.
  - The BRANCH table has columns for each branch_key, branch_name, branch_type.
  - The LOCATION table has columns of geographic data, including street, city, state, and country.


Dimension Table
time
time_key
day
day_of_the_week
month
Quarter
Year
Dimension Table
item
item_key
item_name
brand
type
supplier_type
Sales Fact Table
time_key
item_key
branch_key
location_key
unit_sold
dollars_sold
Dimension Table
branch
branch_key
branch_name
branch_type
Measures
Dimension Table
location
location_key
street
city
state_or_province
country

- **STAR Schema**
  - In this scenario, the SALES table contains only four columns with IDs from the dimension tables, TIME, ITEM, BRANCH, and LOCATION, instead of four columns for time data, four columns for ITEM data, three columns for BRANCH data, and four columns for LOCATION data.
  - Thus, the size of the fact table is significantly reduced.
  - When we need to change an item, we need only make a single change in the dimension table, instead of making many changes in the fact table.

- **STAR Schema keys**
  - **Primary Key**
  - Dimension table’s every row is identified by a unique value which is generally known as primary key of the table.
  - Extraordinary type of one of a kind limitation which can be utilized as the essential method to recover an interesting record from the table.
  - Tables can have numerous interesting records, however it can have just a single Primary Key Constraint.
  - This requirement is actualized by means of a one of a kind file and is accessible to be referenced by a Foreign Key.

- **STAR Schema keys**
  - **Surrogate Key –**
  - These are the keys which are generated by the system and generally does not have any built in meaning.
  - It is UNIQUE since it is consecutively created number for each record being embedded in the table.
  - It is MEANINGLESS since it doesn’t convey any business significance in regards to the record it is connected to in any table.
  - It is SEQUENTIAL since it is doled out in successive request as and when new records are made in the table, beginning with one and going up to the most elevated

- **STAR Schema keys**
  - **Surrogate Key –**
  - **For example**, on the off chance that the data warehouse contains information around 20,000 clients, who on normal made 15 buys, at that point the fact table will contain around 300,000 surrogate key values, though the dimension table will contain 20,000 business key qualities notwithstanding a similar number of surrogate key values.

- **STAR Schema keys**
  - **Foreign Key –**
  - In the fact table the primary key of other dimension table is act as the foreign key.
  - **Alternate key –**
  - It is also a unique value of the table and generally knows as secondary key of the table.

- **STAR Schema keys**
  - **Composite key –**
  - It is the key which consist of two or more attribute.
  - **For example,** the entity has a clientID and a employeeCode as its primary key.
  - Every one of the characteristics that make up the primary key are basic keys on the grounds that each speaks to an exceptional reference while distinguishing a client in one occasion and a employee in the other, so this key is a composite key.

- **STAR Schema keys**
  - **Candidate key –**
  - A substance type in an intelligent information model will have at least zero competitor keys, likewise alluded to just as one of a kind identifiers .
  - For instance, on the off chance that we just interface with American residents, at that point SSN is one up-and-comer key for the Person element type and the mix of name and telephone number (expecting the mix is one of a kind) is possibly a subsequent competitor key.
  - Both of these keys are called up-and-comer keys since they are possibility to be picked as the essential key, a substitute

- **Designing the Star Schema in Data Warehousing**
  - **Problem statement**
  - Consider an order management operational database that tracks order numbers, dates, the requested ship dates, customers and their shipping and billing addresses, products and their quantity and gross dollar amount, sales representatives that take and process orders, the deals (promotions) and discounts proposed/offered to customers.

- **Designing the Star Schema in Data Warehousing**
  - You have to design a data warehouse that will be updated from the above operational database and should support decision making by helping to answer analytical questions about the net order dollar amounts per customer, products, promotions or deals, and the performance of their sales representatives or agents.
  - Analysis of requested ship dates is important for analysis as well. It is also important to allow for performing order amount analysis in various currencies: dollars, dirhams, euros.
  - **Draw the star schema(s) showing the main attributes, including primary keys, foreign keys, and facts.**

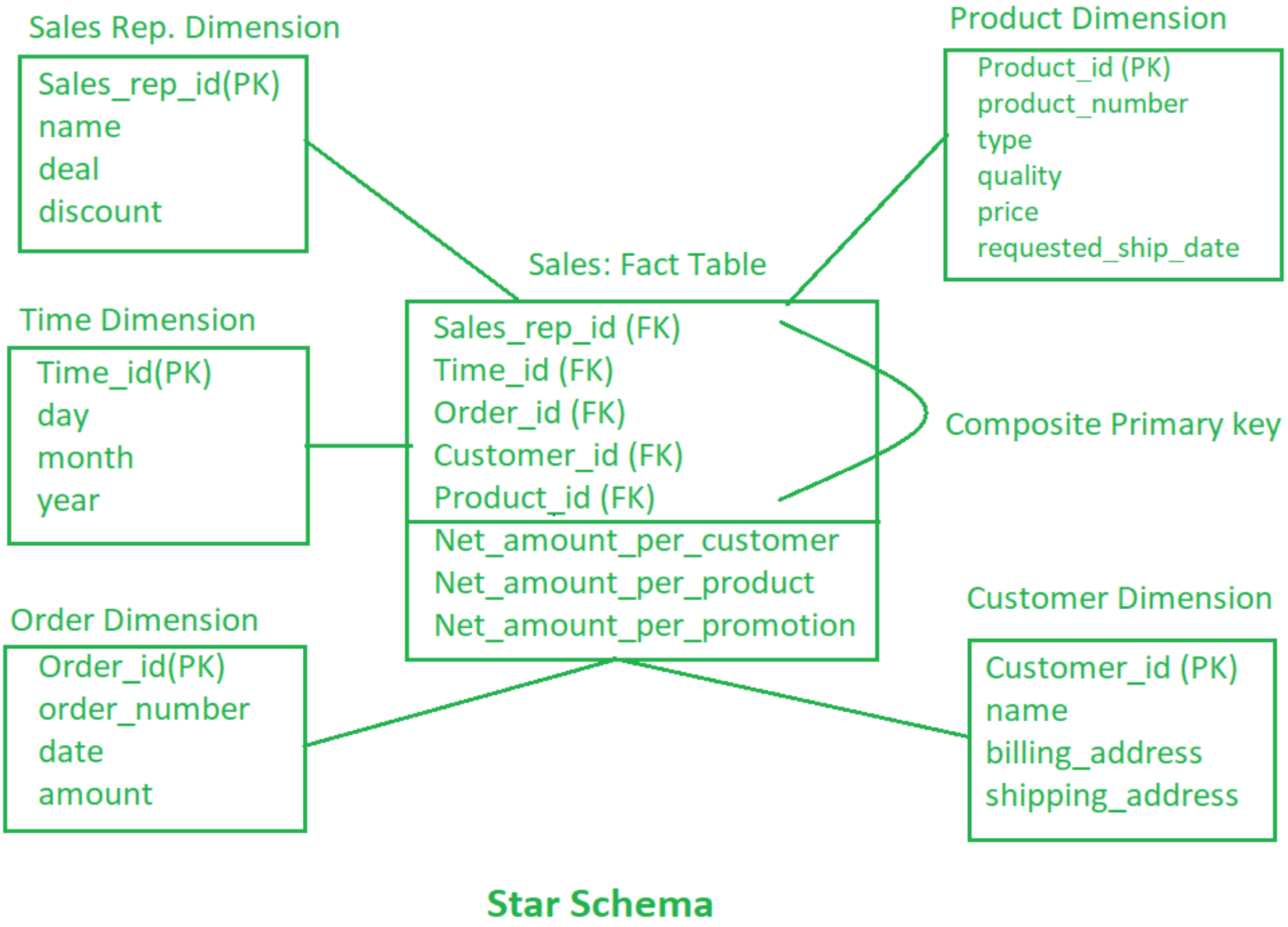
- **Designing the Star Schema in Data Warehousing**
  - **Step 1:** Identify the Business process to model in order to identify the fact table.
  - We are talking about Sales here. Fact table will be named as ‘Sales’. Facts or measures are:
    - Net_amount_per_customer
    - Net_amount_per_product
    - Net_amount_per_promotion

- **Designing the Star Schema in Data Warehousing**
  - **Step 2:** Choose the dimensions for the fact table.
  - Dimensions are
    - Sales Representative
    - Time
    - Customer
    - Product
    - Order

- **Designing the Star Schema in Data Warehousing**
  - **Step 3:** Choose the attributes of dimension tables.
  - **Attributes of SalesRepresentative Dimensio**n:
    - Sales_rep_id (primary and surrogate key)
    - Name
    - Deal
    - Discount
  - **Attributes of Time Dimension**:
    - Time_id (primary and surrogate key)
    - day
    - month

- **Designing the Star Schema in Data Warehousing**
  - **Attributes of Customer Dimension:**
    - Customer_id (primary and surrogate key)
    - name
    - billing_address
    - shipping_address
  - Attributes of Product Dimension:
    - Product_id (primary key and surrogate key)
    - quality
    - price
    - product_number
    - requested_ship_date

- **Designing the Star Schema in Data Warehousing**
  - **Attributes of Order Dimension:**
    - Order_id (primary key and surrogate key)
    - order_number
    - date
    - amount
  - **Step 4**: Draw the star schema.

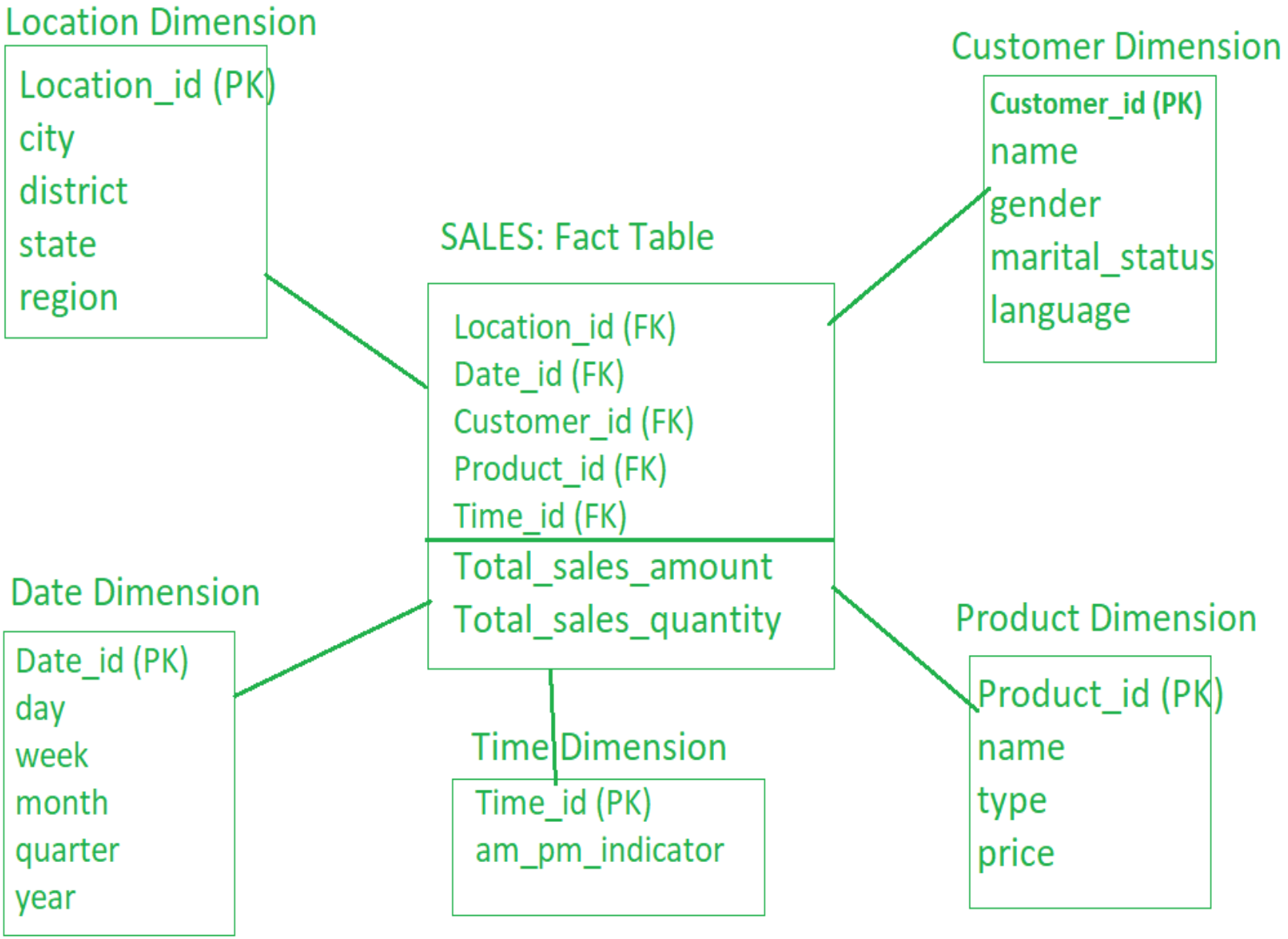

Sales Rep. Dimension
Sales_rep_id(PK)
name
deal
discount
Product Dimension
Product_id (PK)
product_number
type
quality
price
requested_ship_date
Sales: Fact Table
Sales_rep_id (FK)
Time_id (FK)
Order_id (FK)
Customer_id (FK)
Product_id (FK)
Net_amount_per_customer
Net_amount_per_product
Net_amount_per_promotion
Time Dimension
Time_id(PK)
day
month
year
Composite Primary key
Order Dimension
Order_id(PK)
order_number
date
amount
Customer Dimension
Customer_id (PK)
name
billing_address
shipping_address

Star Schema

- **Designing the Star Schema in Data Warehousing**
  - **Problem Statement-2:**
  - Consider a franchise of retail stores having the business setup only in India.
  - The analysis requirements of the franchise include getting to know which items are purchased together by each individual consumer.
  - They wish to know the sales figures in terms of sales amount in Rupees as well as quantity of the individual stores and also for the city, state and region in which they are located.

- **Designing the Star Schema in Data Warehousing**
  - **Problem Statement-2:**
  - They also wish to know how sales differ over different months, quarters and years; how sales figures change with the hour of the day – e.g., how sales of morning hours are different from sales of evening hours, etc.; how buying habits of male consumers are different from that of the female consumers; how buying habits of married consumers are different from that of the unmarried consumers; how buying habits of consumers vary with their native languages (e.g., Kannad, Telugu, Marathi, etc.).

- **Designing the Star Schema in Data Warehousing**
  - **Problem Statement-2:**
  - Design a star schema for such a data warehouse clearly identifying the fact table and dimension tables, their primary keys, and foreign keys.
  - Also, mention which columns in the fact table represent dimensions and which ones represent measures or facts.


Location Dimension
Location_id (PK)
city
district
state
region
Customer Dimension
Customer_id (PK)
name
gender
marital_status
language
SALES: Fact Table
Location_id (FK)
Date_id (FK)
Customer_id (FK)
Product_id (FK)
Time_id (FK)
Total_sales_amount
Total_sales_quantity
Date Dimension
Date_id (PK)
day
week
month
quarter
year
Time Dimension
Time_id (PK)
am_pm_indicator
Product Dimension
Product_id (PK)
name
type
price

Q1 a) Information requirements are recorded for "Hotel occupancy" considering dimensions like Hotel, Room and Time. Few Facts recorded are vacant rooms, occupied rooms, number of occupants, etc. **[10]**

Answer the following questions for this problem:

i. Design the star schema.

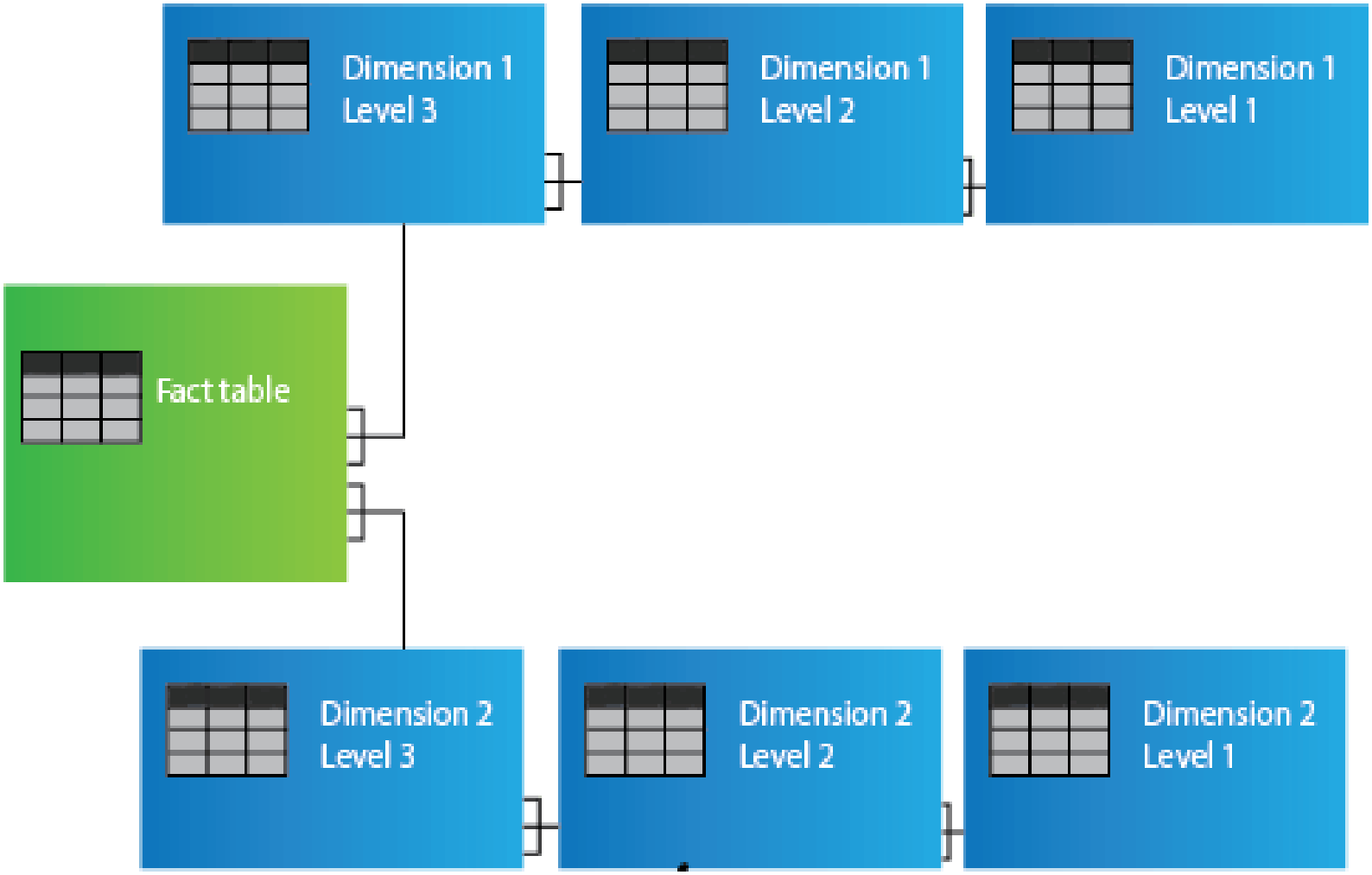
ii. Can you convert this star schema to a snowflake schema? If yes, justify and draw the snowflake schema.

Q1 A) A manufacturing company has a huge sales network. To control the sales, it is divided into regions. Each region has multiple zones. Each zone has different cities. Each sales person is allocated different cities. The objective is to track sales figure at different granularity levels of region and to count no. of products sold. Design a star schema by considering granularity levels for region, sales person and time. Convert the star schema to snowflake schema. [10]

- **Snowflake Schema**
  - A snowflake schema is equivalent to the star schema.
  - "A schema is known as a snowflake if one or more dimension tables do not connect directly to the fact table but must join through other dimension tables."
  - The snowflake schema is an expansion of the star schema where each point of the star explodes into more points.
  - It is called snowflake schema because the diagram of snowflake schema resembles a snowflake.
  - **Snowflaking** is a method of normalizing the dimension tables in a STAR schemas. When we normalize all the dimension tables entirely, the resultant structure resembles

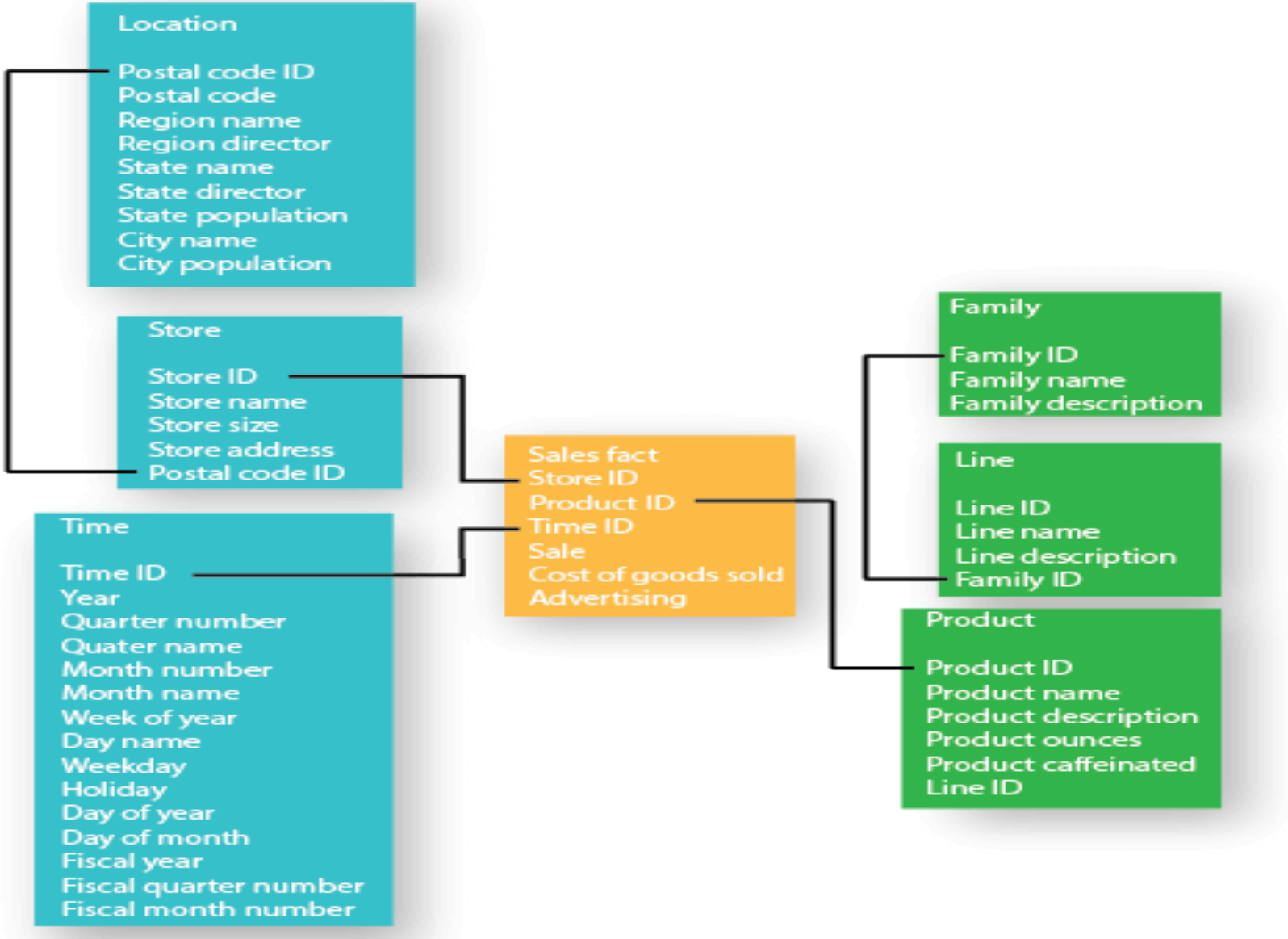
- **Snowflake Schema**
  - **Snowflaking** is used to develop the performance of specific queries.
  - The schema is diagramed with each fact surrounded by its associated dimensions, and those dimensions are related to other dimensions, branching out into a snowflake pattern.
  - The snowflake schema consists of one fact table which is linked to many dimension tables, which can be linked to other dimension tables through a many-to-one relationship.

- **Snowflake Schema**
  - Tables in a snowflake schema are generally normalized to the third normal form.
  - Each dimension table performs exactly one level in a hierarchy.
  - The following diagram shows a snowflake schema with two dimensions, each having three levels.
  - A snowflake schemas can have any number of dimension, and each dimension can have any number of levels.



Snowflake Schema

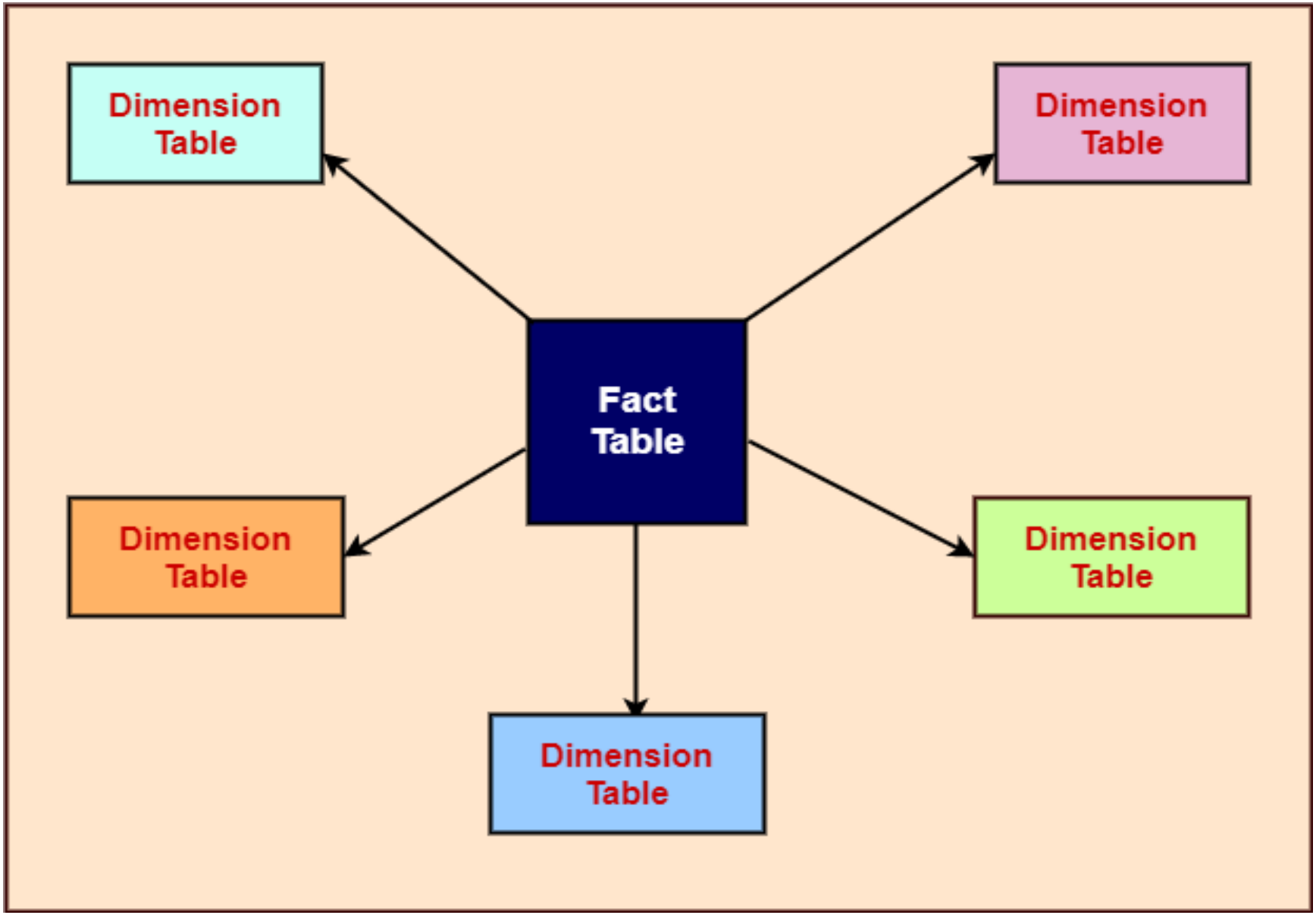
- **Snowflake Schema**
  - **Example**: Figure shows a snowflake schema with a Sales fact table, with Store, Location, Time, Product, Line, and Family dimension tables.
  - The Market dimension has two dimension tables with Store as the primary dimension table, and Location as the outrigger dimension table.
  - The product dimension has three dimension tables with Product as the primary dimension table, and the Line and Family table are the outrigger dimension tables.


Location
Postal code ID
Postal code
Region name
Region director
State name
State director
State population
City name
City population
Store
Store ID
Store name
Store size
Store address
Postal code ID
Time
Time ID
Year
Quarter number
Quater name
Month number
Month name
Week of year
Day name
Weekday
Holiday
Day of year
Day of month
Fiscal year
Fiscal quarter number
Fiscal month number
Sales fact
Store ID
Product ID
Time ID
Sale
Cost of goods sold
Advertising
Family
Family ID
Family name
Family description
Line
Line ID
Line name
Line description
Family ID
Product
Product ID
Product name
Product description
Product ounces
Product caffeinated
Line ID

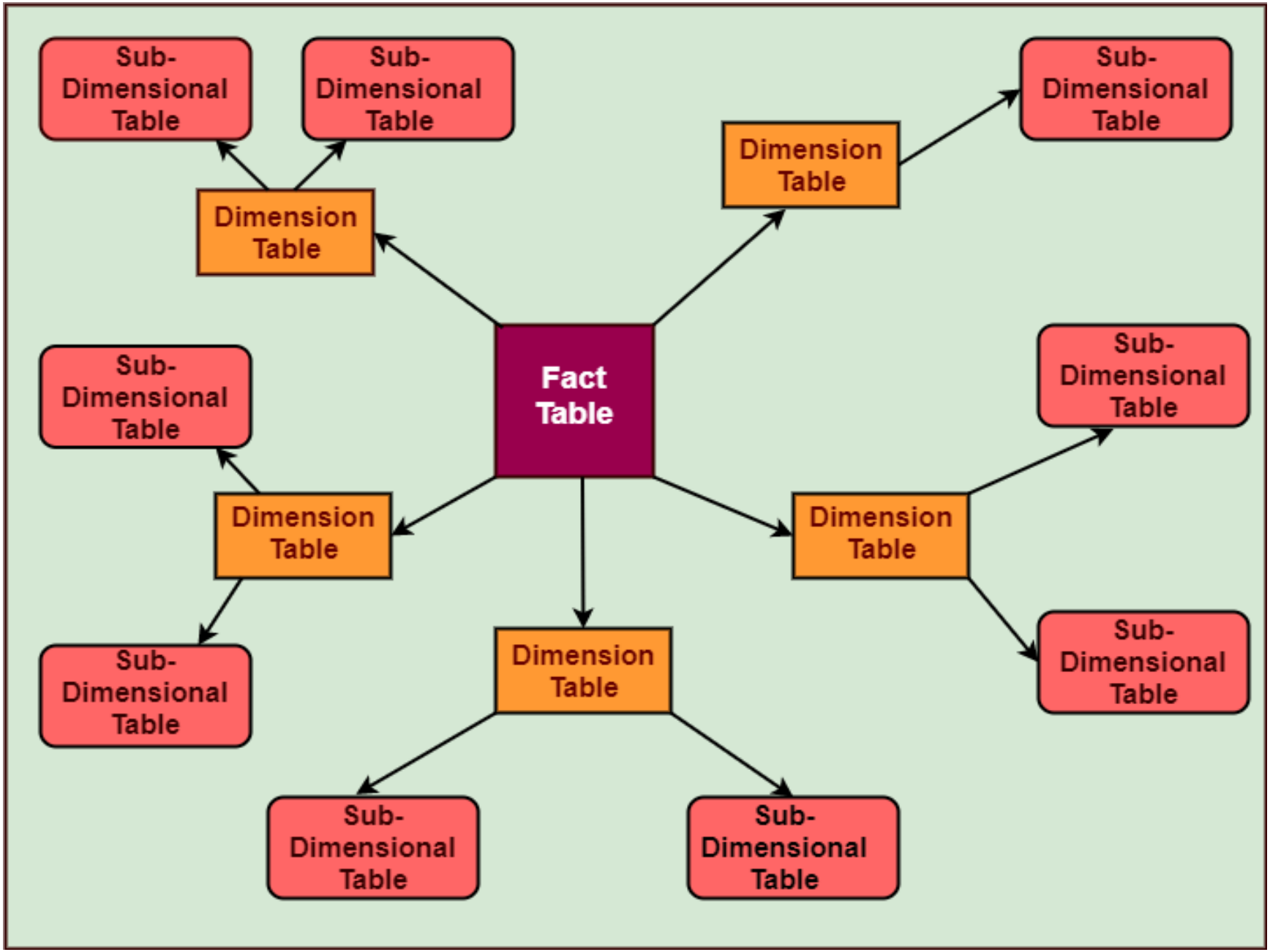
- **Snowflake Schema**
  - **Advantage of Snowflake Schema**
  - The primary advantage of the snowflake schema is the development in query performance due to minimized disk storage requirements and joining smaller lookup tables.
  - It provides greater scalability in the interrelationship between dimension levels and components.
  - No redundancy, so it is easier to maintain.

- **Snowflake Schema**
  - **Disadvantage of Snowflake Schema**
  - The primary disadvantage of the snowflake schema is the additional maintenance efforts required due to the increasing number of lookup tables. It is also known as a multi fact star schema.
  - There are more complex queries and hence, difficult to understand.
  - More tables more join so more query execution time.

- Difference between Star and Snowflake Schemas
  - Star Schema
  - In a star schema, the fact table will be at the center and is connected to the dimension tables.
  - The tables are completely in a denormalized structure.
  - SQL queries performance is good as there is less number of joins involved.
  - Data redundancy is high and occupies more disk space.


Dimension Table
Dimension Table
Fact Table
Dimension Table
Dimension Table
Dimension Table

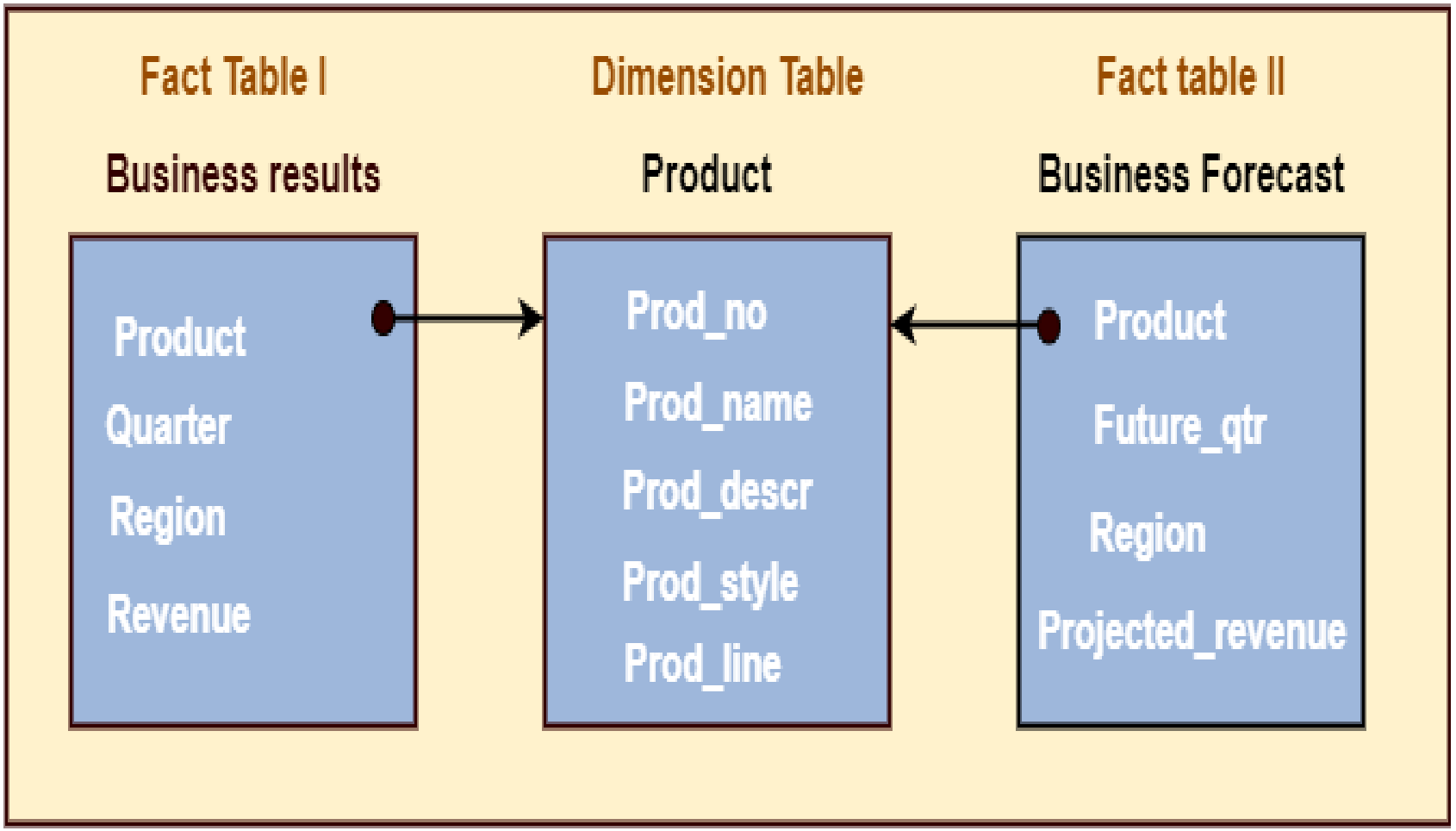
- **Difference between Star and Snowflake Schemas**
  - **Snowflake Schema**
  - A snowflake schema is an extension of star schema where the dimension tables are connected to one or more dimensions.
  - The tables are partially denormalized in structure.
  - The performance of SQL queries is a bit less when compared to star schema as more number of joins are involved.
  - Data redundancy is low and occupies less disk space when compared to star schema.


Sub-Dimensional Table
Sub-Dimensional Table
Dimension Table
Sub-Dimensional Table
Dimension Table
Fact Table
Sub-Dimensional Table
Sub-Dimensional Table
Dimension Table
Dimension Table
Sub-Dimensional Table
Sub-Dimensional Table
Dimension Table
Sub-Dimensional Table
Sub-Dimensional Table

3.7.1 Star Schema Vs Snowflake Schema

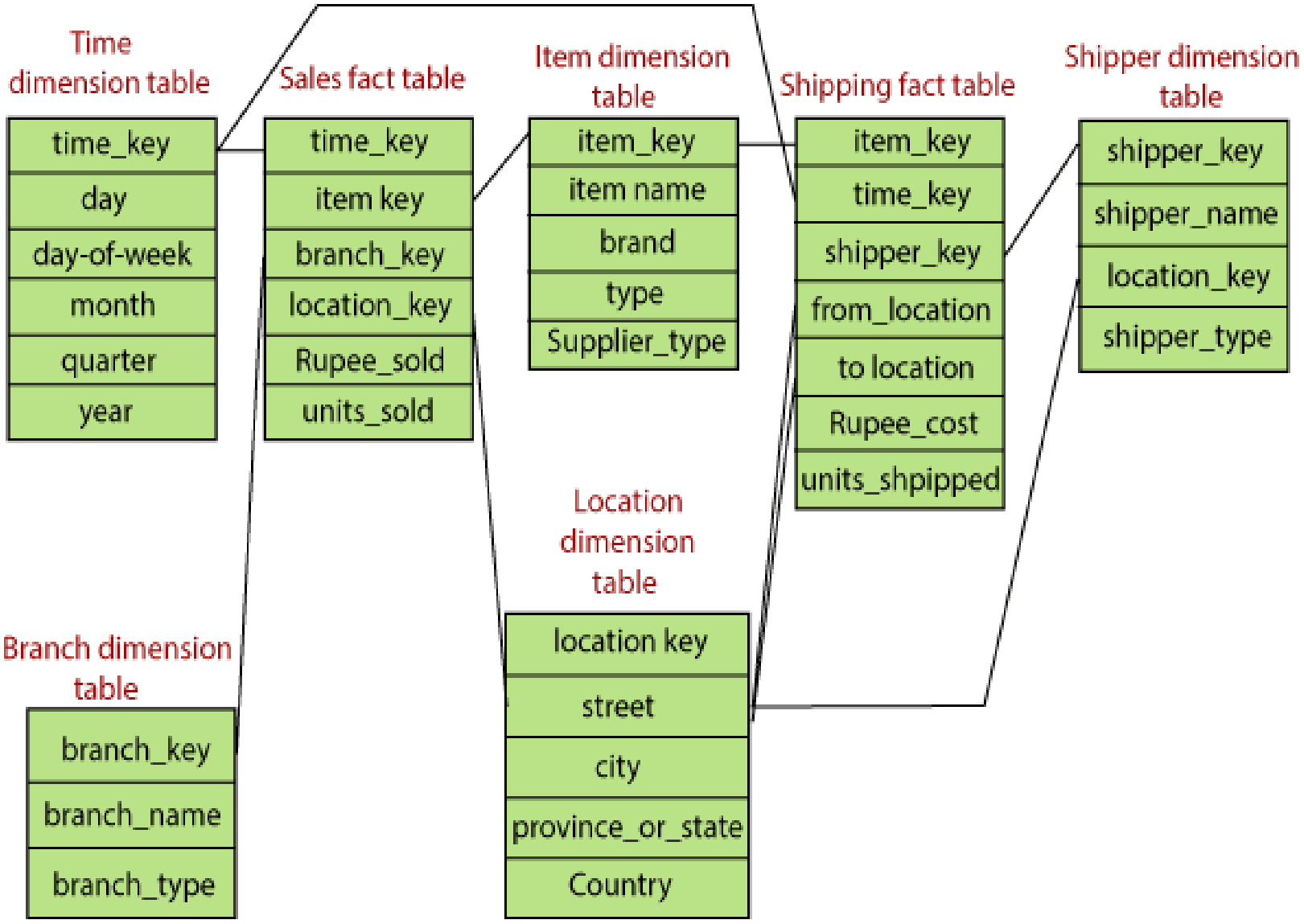
Features	Star Schema	Snowflake Schema
Normalized Dimension Tables	The dimension tables in star schema are not normalized so they may contain redundancies	This schema has normalized dimension tables
Queries	The execution of queries is relatively faster as there are less joins needed in forming a query.	The execution of snowflake schema complex queries is slower than star schema as many joins and foreign key relations are needed to form a query. Thus performance is affected.
Performance	Star schema model has faster execution and response time	It has slow performance as compared to star schema
Storage Space	This type of schema requires more storage space as compared to snowflake due to unnormalised tables.	Snowflake schema tables are easy to maintain and save storage space due to normalized tables.
Usage	Star schema is preferred when the dimension tables have lesser rows	If the dimension table contains large number of rows, snowflake schema is preferred
Type of DW	This schema is suitable for 1:1 or 1: many relationships such as data marts.	It is used for complex relationships such as many: many in enterprise Data warehouses.
Dimension Tables	Star schema has a single table for each dimension	Snowflake schema may have more than one dimension table for each dimension.

- **Fact Constellation Schema**
  - A Fact constellation means two or more fact tables sharing one or more dimensions.
  - **It is also called Galaxy schema.**
  - Fact Constellation Schema describes a logical structure of data warehouse or data mart.
  - Fact Constellation Schema can design with a collection of de-normalized FACT, Shared, and Conformed Dimension tables.



FACT Constellation Schema

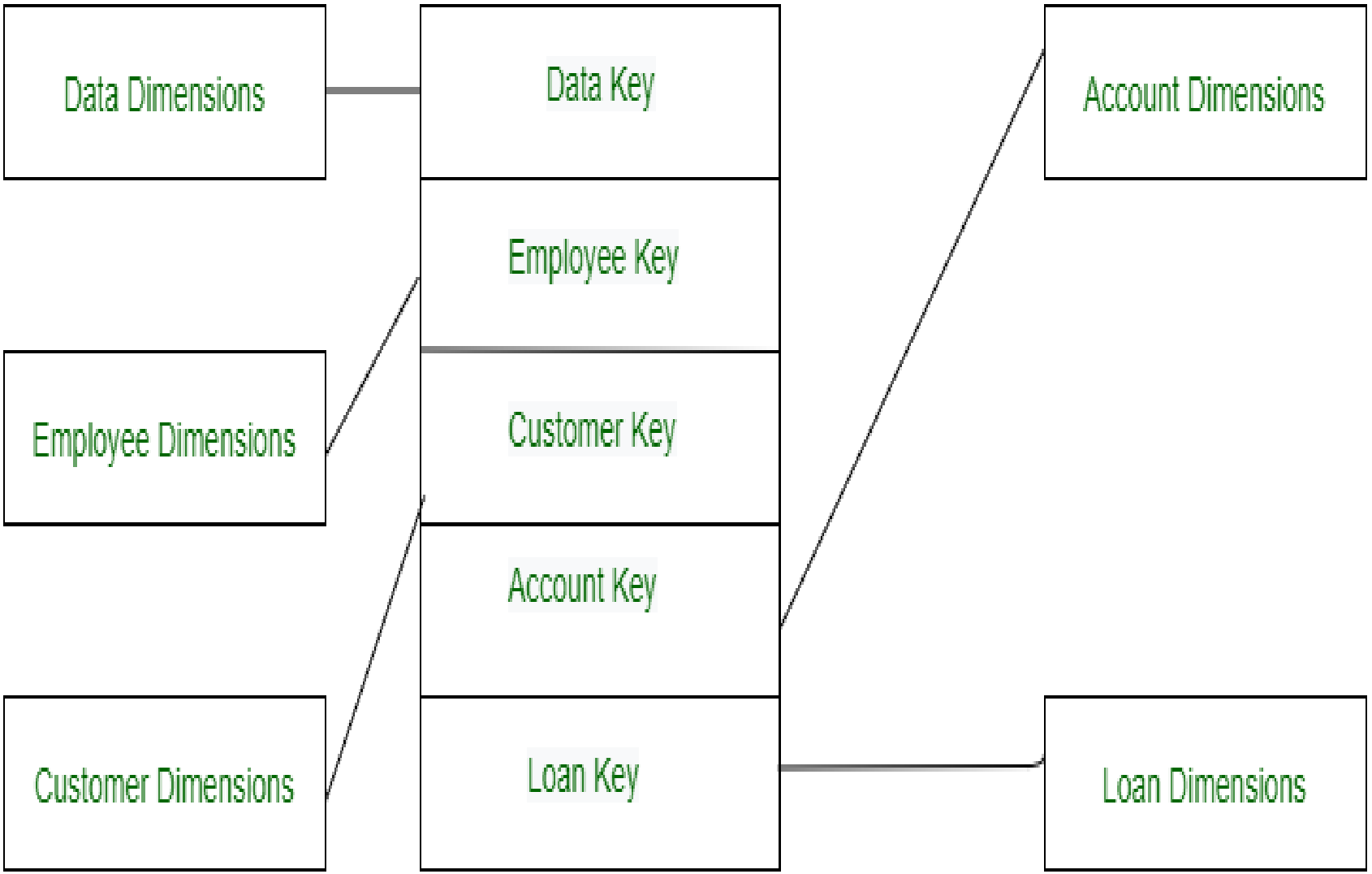
- **Fact Constellation Schema**
  - Fact Constellation Schema is a sophisticated database design that is difficult to summarize information.
  - Fact Constellation Schema can implement between aggregate Fact tables or decompose a complex Fact table into independent simplex Fact tables.
  - **Example:** A fact constellation schema is shown in the figure below.


Time dimension table
time_key
day
day-of-week
month
quarter
year
Sales fact table
time_key
item key
branch_key
location_key
Rupee_sold
units_sold
Item dimension table
item_key
item name
brand
type
Supplier_type
Shipping fact table
item_key
time_key
shipper_key
from_location
to location
Rupee_cost
units_shpipped
Shipper dimension table
shipper_key
shipper_name
location_key
shipper_type
Location dimension table
location key
street
city
province_or_state
Country
Branch dimension table
branch_key
branch_name
branch_type

- Fact Constellation Schema
  - This schema defines two fact tables, sales, and shipping. Sales are treated along four dimensions, namely, time, item, branch, and location.
  - The schema contains a fact table for sales that includes keys to each of the four dimensions, along with two measures: Rupee_sold and units_sold.
  - The shipping table has five dimensions, or keys: item_key, time_key, shipper_key, from_location, and to_location, and two measures: Rupee_cost and units_shipped.

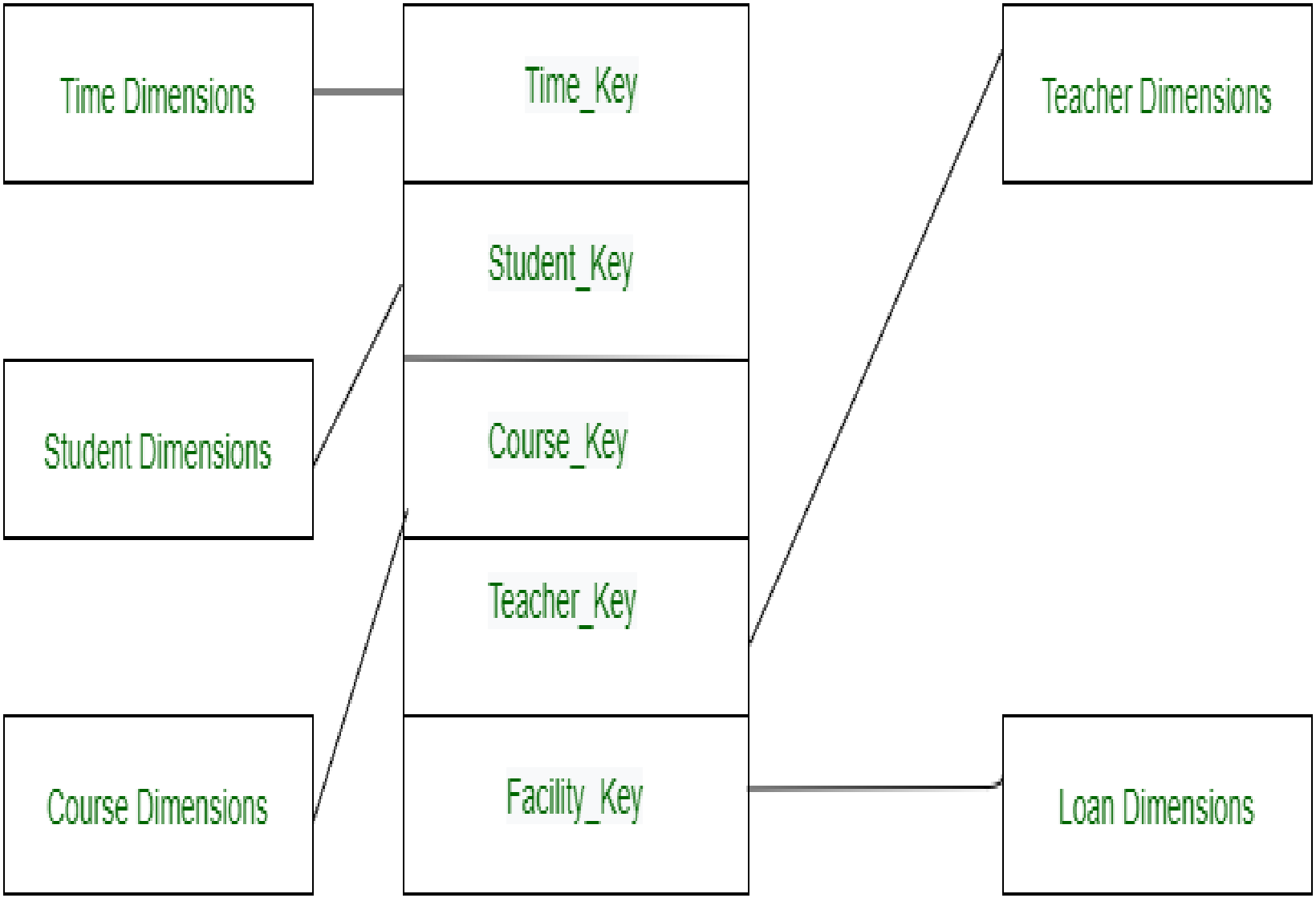
- Fact Constellation Schema
  - **The primary disadvantage** of the fact constellation schema is that it is a more challenging design because many variants for specific kinds of aggregation must be considered and selected.

- Factless Fact Table
  - Factless tables simply mean the key available in the fact that no remedies are available.
  - Factless fact tables are only used to establish relationships between elements of different dimensions.
  - And are also useful for describing events and coverage, meaning tables contain information that nothing has happened. It often represents many-to-many relationships.
  - The only thing they have is an abbreviated key. They still represent a focal phenomenon that is identified by the combination referenced in the dimension tables.


Data Dimensions
Data Key
Account Dimensions
Employee Key
Employee Dimensions
Customer Key
Account Key
Customer Dimensions
Loan Key
Loan Dimensions

- Factless Fact Table
  - There are two types of factless table :
  - **Event Tracking Tables –**
  - Use a factless fact table to track events of interest to the organization.
  - **For example**, attendance at a cultural event can be tracked by creating a fact table that contains the following foreign keys (i.e. links to dimension tables) event identifier speaker/entertainment identifier, participant identifier, event type; Date. This table can then be searched for information, such as the most popular ones. Which cultural program or program type.

- Factless Fact Table
  - **The following example** shows a factless fact table that records each time a student attends a course or which class has the maximum attendance? Or what is the average number of attendance of a given course?
  - All questions are based on COUNT () with group BY questions. So we can first count and then implement other aggregate functions like Aggress, Max, Min.


Time Dimensions
Time_Key
Teacher Dimensions
Student_Key
Student Dimensions
Course_Key
Teacher_Key
Course Dimensions
Facility_Key
Loan Dimensions

- Factless Fact Table
  - Coverage Tables
  - The second type of factless fact table is called a coverage table by Ralph.
  - It is used to support negative analysis reports.
  - For example, to create a report that a store did not sell a product for a certain period of time, you should have a fact table to capture all possible combinations.
  - Then you can find out what is missing.

- Factless Fact Table
  - Common examples of factless fact table:
    - Ex-Visitors to the office.
    - List of people for the web click.
    - Tracking student attendance or registration events.

- **Update to Dimension Table**
  - Every day, more and more sales take place, so more and more rows are added to the fact table.
  - Updating due to the change in fact table happens very rarely.
  - Dimension tables are more stable as compared to the fact tables.
  - Dimension table changes due to the change in attributes themselves, but not because of an increase in the number of rows.

- Aggregate Fact Tables
  - Since, in the data warehouse the data is stored in multidimensional cube.
  - In the information technology industry, there are various tools available to process the queries posted on the data warehouse engine.
  - These tools are called business intelligence (BI) tools.
  - These tools help to answer the complex queries and to take decisions.
  - Aggregate word is very similar to the aggregation of the database schemas of relational tables that you must be familiar with

- **Aggregate Fact Tables**
  - Aggregate fact tables roll up the basic fact tables of the schema to improve the query processing.
  - The business tools smoothly select the level of aggregation to improve the query performance.
  - Aggregate fact tables contain foreign keys referring to dimension tables.

- Aggregate Fact Tables
  - Points to note about Aggregate tables:
    - 1) It is also called summary tables.
    - 2) It contains pre-computed queries of the data warehouse schema.
    - 3) It reduces the dimensionality of the base fact tables.
    - 4) It can be used to respond to the queries of the dimensions that are saved.

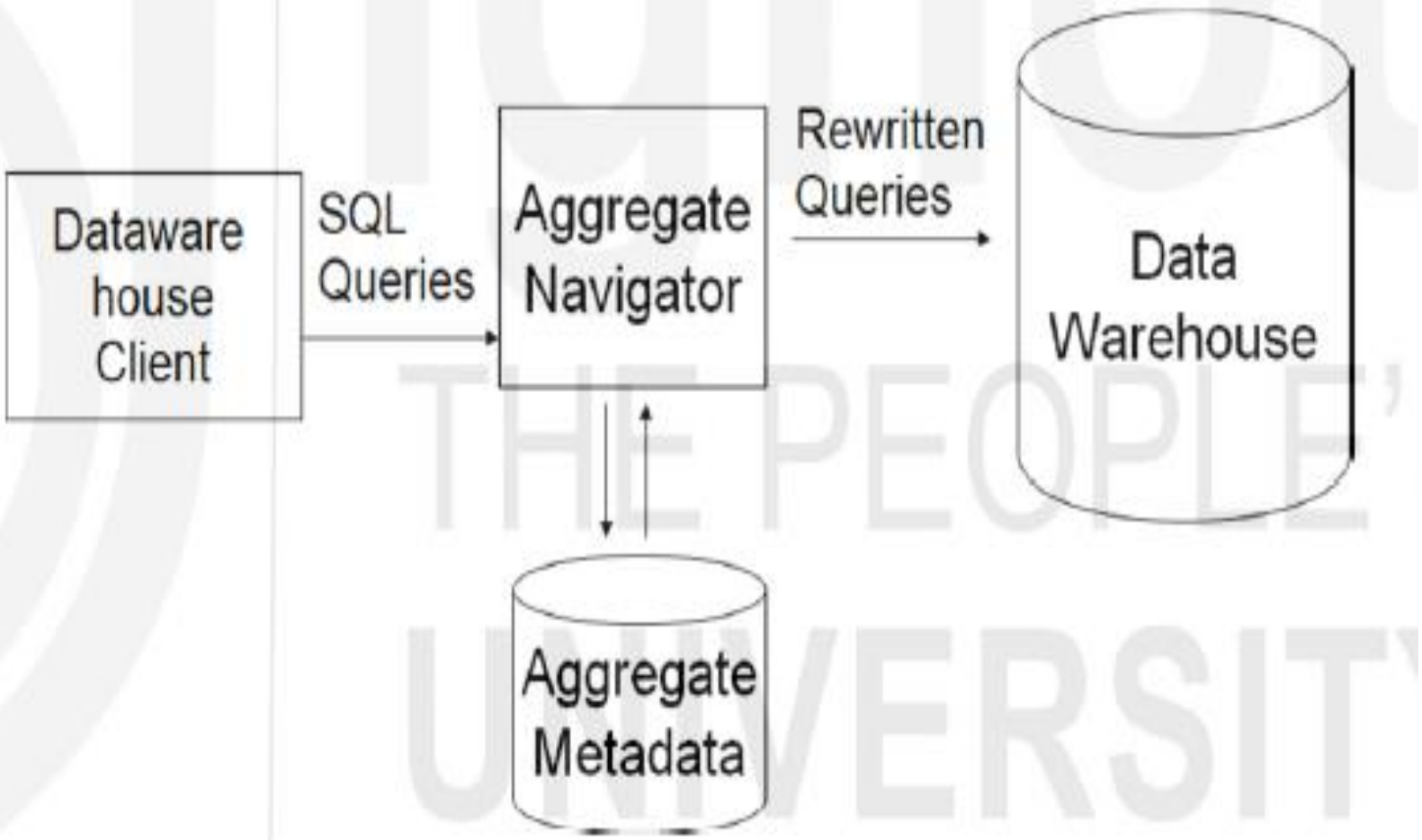


Figure 5: Aggregate Tables

- NEED FOR BUILDING AGGREGATE FACT TABLES
  - Let us understand the need of building aggregate table.
  - Aggregate tables also referred to pre-computed tables having partially summarized data.
    - Simply putting in one word, it's about speed or quick response to queries.
    - This you can understand as an intermediate table which stores the results of the queries on I/O disk space.
    - It uses aggregates functionality.

- NEED FOR BUILDING AGGREGATE FACT TABLES
  - It occupies less space than atomic fact tables.
  - It nearly takes the half time of a general query processing.
  - One of the more popular uses of aggregates is to adjust the granularity of a dimension.
  - When the granularity of a dimension is changed, the fact table must be partially summarized to match the current grain of the new dimension, resulting in the creation of new dimensional and fact tables that fit this new grain standard.

- NEED FOR BUILDING AGGREGATE FACT TABLES
  - The Roll-up OLAP operation of the base fact tables generates aggregate tables.
  - Hence the query performance increases as it reduces the number of rows to be accessed for the retrieval of data of a query.