BI – Unit 2 (The Architecture of DW and BI) - IN-SEM PYQ Answers

Q1. Differences between OLTP and OLAP systems. [5]

OLTP (Online Transaction Processing)	OLAP (Online Analytical Processing)
Designed for **real-time transaction processing** supporting day-to-day operations (insert/update/delete).	Designed for **complex analysis and querying** over historical and aggregated data.
Handles **high volume of simple, short transactions** with fast response times.	Handles **complex analytical queries** on large datasets; slower response times.
Uses **normalized schemas** to ensure data integrity and reduce redundancy.	Uses **denormalized/multidimensional schemas** (e.g., star/snowflake) for fast analysis.
Data is **current and operational**, updated frequently.	Data is **historical and aggregated**, updated periodically after ETL
Focuses on **transaction throughput** and maintaining consistency (ACID).	Focuses on **analytical insights** and supporting decision-making tasks.

Q2. Explain the dice or rotation operation in OLAP. [5]

In OLAP, analysts explore data stored in a **multidimensional cube** by applying operations that reshape the view of the data. Two important operations are **Dice** and **Rotation (Pivot)**:

1. Dice Operation

- **Definition:**
  Dice is an OLAP operation that creates a **smaller sub-cube** by applying **filters on two or more dimensions** of the original data cube.
- **Purpose:**
  It allows users to focus on a specific subset of data across multiple dimensions simultaneously.
- **Example:**
  Suppose a sales OLAP cube has dimensions: *Product, Region, Time*. Using Dice, one might extract only the data for
  - Region = {“Delhi”, “Mumbai”},
  - Time = {“Q1”, “Q2”},
  - Products = {“Car”, “Bus”}.
    This results in a smaller cube containing only those combinations.
- **Use in BI:**
  Helps compare performance across selected dimensions for finer analysis.

2. Rotation (Pivot) Operation

- **Definition:**
  Rotation, also called **Pivot**, **reorients the axes** of the OLAP cube to provide a **different perspective** of the same data. It changes how dimensions are displayed without altering the underlying data.
- **Purpose:**
  It allows analysts to **view the same information from different dimensional arrangements**, aiding comparison and insight discovery.
- **Example:**
  If the original cube shows *Product* on the X-axis, *Region* on the Y-axis, and *Time* on the Z-axis, rotation might swap *Region* and *Time* so analysts can view sales by quarter across regions.
- **Use in BI:**
  Useful for visual exploration, dashboards and reports where different dimensional views highlight trends or patterns.

Q3. Compare with diagram ROLAP versus MOLAP [5]

ROLAP (Relational OLAP)	MOLAP (Multidimensional OLAP)
Stores and manages data in **relational databases (tables)**; uses SQL for queries.	Stores data in **multidimensional cube structures (arrays)** optimized for fast retrieval.
More scalable; suitable for very large datasets and high cardinality.	Handles **limited data volumes** efficiently; performance decreases with very large data
Query performance is **slower** due to on-the-fly calculations and joins.	Query performance is **faster** due to pre-aggregation and cube storage.
Works directly on data warehouse tables without pre-aggregation.	Uses **pre-summarized, pre-computed data** in cube storage.
Flexible for ad-hoc queries and detailed drill-down.	**Optimized for analytical operations** like slice, dice, roll-up, drill-down
ROLAP Relational DB ↓ SQL/OLAP Engine ↓ OLAP Queries/Reports	MOLAP Multidimensional Cube ↓ Pre-aggregated Data ↓ OLAP Queries/Reports

Q4. What is the Star Schema? Explain the advantages and disadvantages of it. [5]

A **Star Schema** is a **dimensional data warehouse schema** where a central **fact table** (holding measures) is directly connected to multiple **dimension tables** (holding descriptive attributes) in a star-like structure. Fact tables contain numeric metrics and foreign keys to dimensions, while dimension tables provide context such as customer, product, time or location.

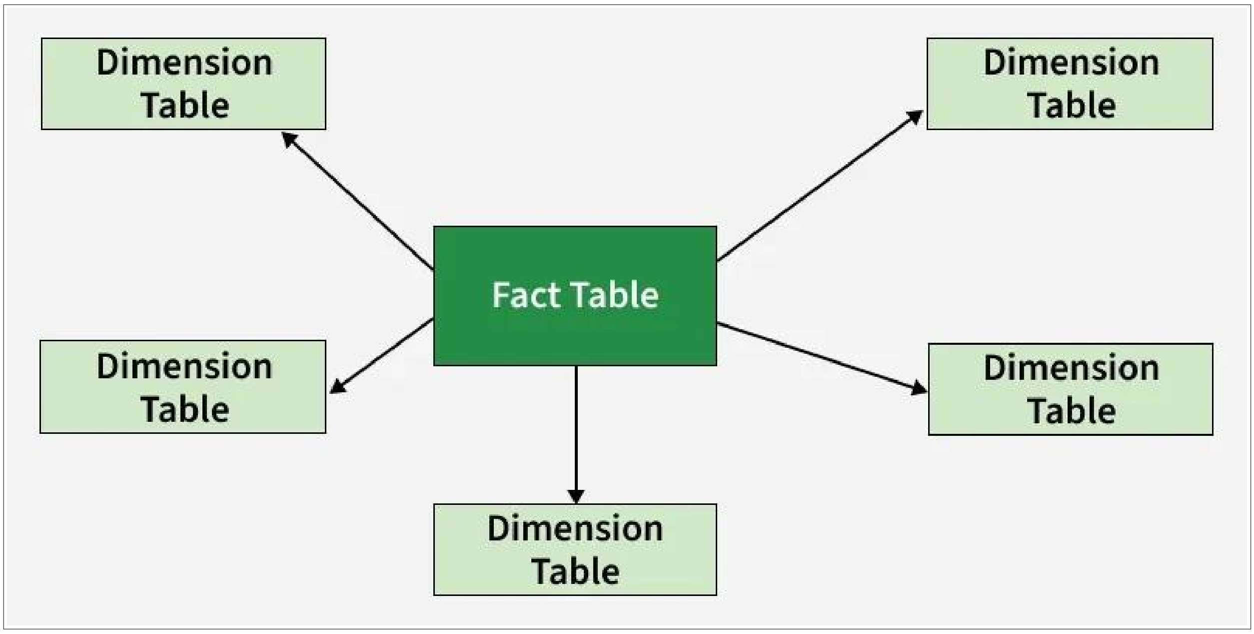
Advantages

1. **Simple & Easy to Understand:** Clear structure with a central fact table and surrounding dimensions makes it intuitive for users and developers.
2. **Fast Query Performance:** Fewer joins are required (fact to dimensions), resulting in faster query execution for analytical workloads.
3. **Efficient for BI Tools & OLAP:** Works well with BI reporting and OLAP tools that optimize multidimensional queries.
4. **Good for Historical and Trend Analysis:** Denormalized dimensions support efficient aggregations over large datasets.
5. **Scalable for Large Datasets:** Can handle growing analytical workloads with simple schema design.

Disadvantages

1. **Data Redundancy:** Denormalized dimension tables lead to repeated data, increasing storage and potential inconsistency.
2. **Less Data Integrity:** Redundancy can cause integrity issues as updates must be applied in multiple places.
3. **Limited Flexibility for Changes:** Schema is less adaptable when business requirements evolve frequently.
4. **Not Ideal for Complex Relationships:** Harder to represent many-to-many relationships or very detailed hierarchies without additional structures.
5. **Higher Storage Costs:** Denormalization increases storage requirements compared with normalized designs.

Q5. Draw and Explain Star Schema. Also state which is Better: Snowflake Schema Or Star Schema? [5]



1. Fact Table: The fact table sits at the center of the schema and stores the measurable, quantitative data used for analysis. Examples include:

- Sales amount

- Units sold
- Discount
- Profit

Each record in a fact table represents a business event (e.g., a sales transaction)

2. Dimension Tables: Dimension tables surround the fact table and contain descriptive attributes that add context to the facts. Common dimensions include:

- Product details
- Customer details
- Time attributes
- Employee or store information

These tables allow users to slice, dice, filter, and group the fact data for analysis (e.g., sales by region, by month, by product category).

Star Schema vs Snowflake Schema — Which is Better?

A Star Schema and a Snowflake Schema are two common dimensional modeling approaches used in data warehouses. A star schema has a central fact table connected to denormalized dimension tables, while a snowflake schema normalizes dimension tables into multiple related tables.

Which is Better?

- **Star Schema is generally considered better** for most Business Intelligence and OLAP environments because of its **simplicity, faster query performance and ease of use** for end users and BI tools.
- The **Snowflake Schema** may be preferred when **storage efficiency, data integrity and highly normalized relationships** are critical, but this often comes at the cost of increased query complexity and slower performance.

Why Star Schema is Often Better for BI

1. **Faster Query Performance:** Denormalized dimensions reduce join operations, speeding up analytical queries.
2. **Simpler Design and Understanding:** Easy for business users and analysts to interpret the schema.
3. **Better for BI Tools & Reporting:** Most OLAP tools optimize star schemas directly.
4. **Easier Maintenance:** Fewer tables and simpler relationships reduce ETL and maintenance effort.
5. **Ad-hoc Analysis Ready:** Ideal when quick, ad-hoc querying is needed.

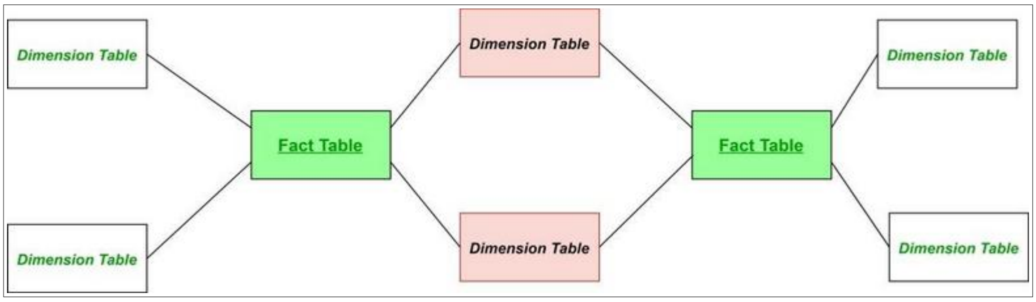
When Snowflake Schema Might Be Better

- **Normalized dimension hierarchies** reduce redundancy and storage use.
- **Higher data integrity** and easier updates for complex hierarchies.
- Suitable when data models have deep hierarchies or frequent updates.

Q6. Explain the Fact Constellation Schema with a diagram. [5]

A **Fact Constellation Schema**, also called a **Galaxy Schema**, is a multidimensional data warehouse schema that contains **multiple fact tables** sharing one or more **common dimension tables**. It extends the star schema concept to support complex analytical requirements across multiple business processes.

Diagram

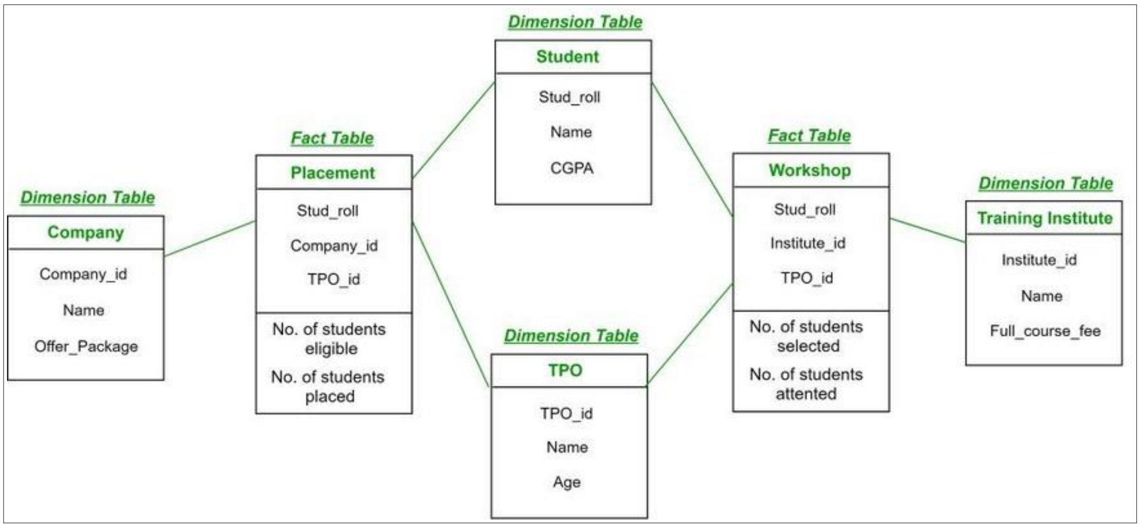


The pink colored Dimension tables are the common ones among both the star schemas. Green colored fact tables are the fact tables of their respective star schemas.

Explanation

1. **Multiple Fact Tables:**
    - Each fact table represents a different business process or measurement area (e.g., Sales, Inventory).
    - Fact tables store measurable quantitative data (e.g., quantity sold, stock levels).
2. **Common Dimension Tables:**
    - Dimension tables such as Time, Product, Location are shared across fact tables.
    - Shared dimensions are called **conformed dimensions** and ensure consistent analysis across measures.
3. **Flexible Analysis:**
    - The schema enables querying across different facts using common dimensions—e.g., comparing sales trends and inventory levels by product and time.
4. **Complexity and Scalability:**
    - This schema is **more complex than star or snowflake schemas** due to multiple fact–dimension relationships and greater number of joins.
    - It is suitable for **large analytical systems** with diverse business processes.

Q7. Write down example of Fact Constellation. [5]



In the diagram below, we have two fact tables: Placement and Workshop, each associated with their own star schemas. These star schemas share common dimension tables, forming the Fact Constellation schema, also referred to as the Galaxy Schema.

Fact Tables

- Placement:
    - Attributes: (Stud_roll, Company_id, TPO_id)
    - Facts: (Number of students eligible, Number of students placed)

  The Placement fact table records data related to student placements. It connects to the Company, Student, and TPO dimension tables.
- Workshop:
    - Attributes: (Stud_roll, Institute_id, TPO_id)
    - Facts: (Number of students selected, Number of students attended)

  The Workshop fact table captures information related to workshops attended by students. It is connected to the Student, TPO, and Training Institute dimension tables.

Dimension Tables

- Company: Contains attributes like (Company_id, Name, Offer_package).
- Student: Contains attributes like (Stud_roll, Name, CGPA).
- TPO (Training and Placement Officer): Contains attributes like (TPO_id, Name, Age).
- Training Institute: Contains attributes like (Institute_id, Name, Full_course_fee).

1. The Placement fact table connects to the Company, Student, and TPO dimension tables, forming a star schema.
2. The Workshop fact table connects to the Student, TPO, and Training Institute dimension tables, forming another star schema.
3. Both star schemas share the Student and TPO dimension tables, thus combining into a Fact Constellation Schema.
4. This structure allows for complex analysis across different aspects of the student data, such as placement and workshop participation, using shared dimensions.
5. The fact tables (Placement and Workshop) represent different business processes, and by sharing common dimensions (like Student and TPO), this schema offers a powerful way to analyze and process data in a data warehouse.

Q8. What is a digital dashboard? What are its types? Discuss anyone with an example. [5]

A **digital dashboard** is an interactive **data visualization interface** that aggregates information from multiple sources and displays key metrics, performance indicators (KPIs) and data in a consolidated, easy-to-read format to support **monitoring and decision making**. It provides a *snapshot view* of business performance at a glance, often in real-time.

Types of Digital Dashboards:

1. **Operational Dashboards**
    - Focus on **real-time monitoring** of daily operations.
    - Tracks live data to help frontline users manage processes.

    - Example metrics: current sales, inventory levels, system uptime.
2. **Strategic Dashboards**
    - Designed for **high-level executives**.
    - Focuses on KPIs linked to long-term objectives and goals.
    - Uses summarized, periodic data.
3. **Analytical Dashboards**
    - Used for **deeper data exploration and trend analysis**.
    - Includes features like filtering, trend lines, drill-downs.
    - Helps understand *why* data behaved a certain way.
4. **Tactical Dashboards** *(sometimes included)*
    - Used by middle managers to monitor projects or specific functions.
    - Focuses on short-term performance and targets.

Example – Operational Dashboard:

- **Purpose:** Monitor daily sales performance in a retail store.
- **Displayed Metrics:**
    - *Real-time sales totals* for the current day
    - *Number of transactions per hour*
    - *Top-selling products*
    - *Inventory alerts* for low stock
- **Use:** The store manager sees live data and can act immediately if sales drop unexpectedly or stock runs low — improving responsiveness and operational efficiency.

Q9. Discuss the three main functions of ETL tools. [5]

Q10. Explain Extraction, Transformation, and Loading (ETL) [5]

1. **Extraction**: Extraction is the first stage of ETL where data is **collected from diverse source systems** such as transactional databases, flat files, APIs, spreadsheets, cloud storage etc., without altering its original form. The extracted data is often staged temporarily for further processing.

Purpose**:**

- To pull raw data from **multiple heterogeneous sources**.
- To ensure data needed for BI or analytics is available in one place for processing.

Example**:**
A retail company extracts sales records from their POS database, customer profiles from the CRM, and website click logs from server files for further consolidation.

2. **Transformation**: Transformation is the middle phase where the extracted data is **cleaned, validated, formatted and structured** to meet the target system's requirements and business rules. This ensures consistency, correctness and usability of data before loading.

Purpose**:**

- To **cleanse data** (remove duplicates, correct errors).
- To **standardize and integrate** data into a uniform format.
- To apply business logic such as joins, aggregations and calculations.

Example**:**
Transforming sales dates into a common format, removing duplicate records, and calculating total monthly revenue for each store before analysis.

3. **Loading**: Loading is the final ETL phase where transformed data is **inserted into the target system** (data warehouse, data mart, or analytics database) so it becomes available for reporting, dashboards, and decision making.

Purpose**:**

- To **store transformed and quality-assured data** in the final repository.
- To make consolidated, usable data accessible for BI, OLAP, and analytics.

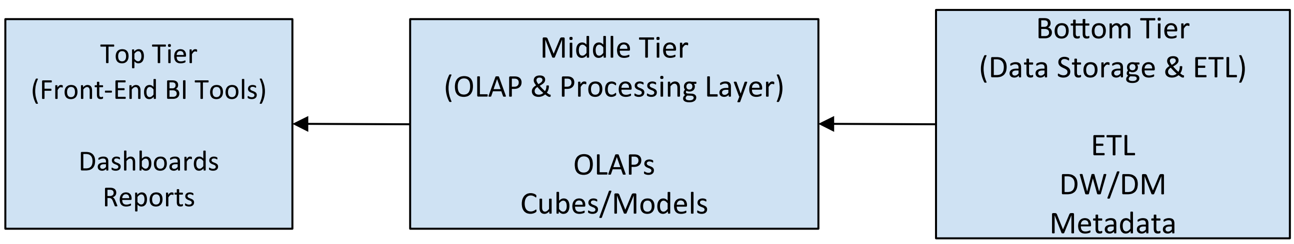
Example**:**
After transformation, the cleaned sales and customer data is loaded into a centralized data warehouse so BI tools can generate trend reports and dashboards.

Summary Table:

ETL Function	What It Does	Example
Extraction	Pulls data from source systems	Extracting POS, CRM & logs
Transformation	Cleans and standardizes data	Formatting dates, removing duplicates
Loading	Loads data into a warehouse	Saving into DW for BI

Q11. Draw and Explain Data Warehouse Tier Architecture. [5]

A **three-tier data warehouse architecture** is the most widely used framework to build scalable, efficient and easily accessible data warehousing systems. It divides the warehouse into three distinct layers that support data flow from raw sources to end-user analytics.



Explanation of Tiers

1. **Bottom Tier – Data Storage Layer**
    - Consists of **source systems** (operational databases, flat files, external sources) and the **data warehouse repository**.
    - The **ETL process** (Extract, Transform, Load) collects raw data from multiple sources, cleans it, transforms it into a consistent format, and loads it into the warehouse.
    - This layer is the foundation where **integrated, historical data** is stored for analysis.

2. **Middle Tier – Data Processing and OLAP Layer**
    - Acts as the **processing backbone** of the architecture.
    - Contains the **OLAP server** or analytical engines that organize, aggregate and model data into multidimensional structures (cubes) for faster analytical queries.
    - It enables functions like **drill-down, roll-up, slice and dice** to support deep analysis.
3. **Top Tier – Presentation and Access Layer**
    - This is the **front-end layer** interacting with business users.
    - Includes **BI tools**, reporting interfaces, dashboards, and query tools that allow users to visualize, explore and extract insights from data.
    - Supports decision making through reports, charts and interactive analytics.

Q12. Distinguish between Business Intelligence and Data Warehouse [5]

Business Intelligence (BI)	**Data Warehouse (DWH)**
BI is a **set of tools, methodologies and processes** used to analyse data and support decision making (reports, dashboards, OLAP).	A Data Warehouse is a **centralized repository** that stores integrated historical data from multiple sources.
BI works as a **decision support system (front-end)** that extracts insights from data.	DWH serves as the **back-end storage system** that organizes and consolidates data for BI.
Focuses on **presentation, visualization and analysis** of data for users (executives, analysts).	Focuses on **data integration, cleaning, storage and preparation** for analysis.
BI tools access and analyse data from the data warehouse (or other sources) to generate insights.	The Data Warehouse provides the BI system with **clean, structured, historical data**.
Output of BI includes **dashboards, reports, KPIs and actionable insights**.	Output of a Data Warehouse is **query-ready structured data** organized for analysis.

List the tools in business Intelligence. Explain any 2 tools. [5]

Common Business Intelligence tools include:

- Microsoft Power BI
- Tableau
- Qlik Sense
- SAP BusinessObjects
- Looker Studio
- Zoho Analytics
- Sisense
- Dundas BI

- Oracle BI
- Domo, etc.

1. **Microsoft Power BI**: Microsoft Power BI is a modern, cloud-based BI platform that enables data analysis, reporting and interactive dashboards from multiple data sources. It transforms raw data into visible insights to support decision making.

Key Features:

- Connects to various data sources (databases, cloud services).
- Rich visualization options (charts, graphs, maps).
- Supports real-time dashboards & interactive reports.
- Integration with Microsoft ecosystem (Excel, Azure).

Use Case Example:
A retail company uses Power BI to monitor daily sales performance, track trends over time, and generate executive dashboards for strategic planning.

2. **Tableau**: Tableau is a leading BI and data visualization tool that allows users to create interactive visual analytics and dashboards easily. It supports drag-and-drop interfaces to turn complex data into clear insights.

Key Features:

- Simple drag-and-drop visuals.
- Live data connection to databases and files.
- Interactive dashboards for deep analysis.
- Strong support for data exploration and pattern discovery.

Use Case Example:
An analyst uses Tableau to visualize customer segmentation data, uncover patterns in buying behavior, and support marketing decisions.

Q13. Explain need for business intelligence in different sectors. [5]

Business Intelligence (BI) is essential because it **transforms raw data into useful insights** that improve performance, decision-making, efficiency, and competitive advantage across industries. BI helps answer core business questions with structured data and analytics rather than guesswork.

1. Healthcare Sector

- BI helps **analyse clinical and operational data** to improve patient outcomes, reduce costs and optimize resources.
- It supports **decision-making in treatment planning, workflow efficiency, claims management and cost control**.
- Example: A hospital uses BI to track patient readmission risks and treatment outcomes to improve care quality.

2. Retail Sector

- BI helps track **sales trends, customer buying behaviour and inventory performance**.

- Enables retailers to **optimize pricing, promotions, stock levels and customer loyalty strategies**.
- Example: A retail chain uses dashboards to monitor best-selling products and forecast demand.

3. Finance & Banking Sector

- BI is used for **risk management, fraud detection, credit evaluation and financial performance monitoring**.
- Delivers faster, data-driven decisions for lending, compliance and customer retention.
- Example: Banks use BI to identify credit card spend patterns to detect potential fraud.

4. Manufacturing Sector

- BI helps optimize **production processes, supply chain, maintenance planning and quality control**.
- Facilitates **real-time monitoring** of plant performance and reduces downtime.
- Example: Manufacturers analyse production line data to reduce defects and waste.

5. Agriculture and Energy Sector

- BI helps analyse **weather patterns, pricing, livestock data and output trends** to guide planning and risk mitigation.
- In energy, BI predicts **supply–demand patterns** and balances grid operations.
- Example: Farmers use BI analytics to plan crop cycles based on climatic trends.

BI is needed across sectors to **enable faster, accurate decision-making, improve operational efficiency, enhance customer outcomes, reduce costs, and gain strategic competitive advantage** by converting data into actionable insights.

Q14. What is Cube? Explain the types of schema design. [5]

A **cube** in BI/OLAP is a **multidimensional data structure** that allows data to be viewed and analysed across multiple dimensions (e.g., product, time, region).

- Although called a “cube,” it can have more than three dimensions (often called a *hypercube*).
- Each cell in the cube stores a **measure** (such as sales amount) aggregated at the intersection of dimension values.
- Cubes support fast analytical queries and OLAP operations like *slice*, *dice*, *roll-up* and *pivot*.
- **Simple Example:** A sales cube might have dimensions *Product*, *Time* and *Region*. A cell in the cube could represent *Total Sales for Product X in Region Y during Q1*.

Types of Schema Design in Data Warehousing

Data warehouse schemas define how facts (measures) and dimensions (attributes) are organized. The main types are:

a) Star Schema

- **Structure:** One central **fact table** connected directly to multiple **dimension tables**.
- **Characteristics:** Dimension tables are **denormalized**, making the schema easy to understand and query.
- **Use:** Best for simpler, high-performance analytical queries.

```
  Dim_Product
      |
Dim_Time — Fact_Sales — Dim_Customer
      |
  Dim_Store
```

b) Snowflake Schema

- **Structure:** Similar to star schema but **dimension tables are normalized** into multiple related tables.
- **Characteristics:** Reduces redundancy; more complex relationships and more joins are required in queries.
- **Use:** Useful when dimension hierarchies are deep and normalized design is needed.

```
   Dim_Product
      |
 Dim_Category
      |
Dim_Time — Fact_Sales — Dim_Customer
      |
 Dim_Region
```

c) Fact Constellation Schema (Galaxy Schema)

- **Structure:** Contains **multiple fact tables** that share one or more **dimension tables**.
- **Characteristics:** Also called a galaxy schema; supports complex analytical environments with multiple business processes.
- **Use:** Suitable when different fact tables need to be analyzed using common dimensions.

```
  Dim_Time
      |
Fact_Sales — Dim_Product — Fact_Inventory
      |
  Dim_Store
```

Summary:

- **Cube:** Multidimensional data view for fast analytics.
- **Star Schema:** Simple, denormalized, fast queries.
- **Snowflake Schema:** Normalized dimensions, reduces redundancy.
- **Fact Constellation:** Multiple fact tables sharing dimensions for complex BI scenarios.

Q15. Explain multi-dimensional data cube analysis with OLAP operations.[5]

Multidimensional Data Cube
A **data cube** is a multidimensional array used in OLAP (Online Analytical Processing) that organizes business data across several dimensions (such as *Time, Product, Region*) with measures (like *Sales, Profit*). It enables fast, interactive analysis from multiple perspectives.

In a cube, each dimension is an axis and each cell contains a measure value. The cube supports efficient querying and aggregation for business decision support.

OLAP Operations for Cube Analysis

Multidimensional analysis uses various OLAP operations to explore data at different levels of detail and from different angles.

1. **Slice**
   - **Definition:** Selects a **single dimension value** to create a sub-cube, reducing dimensionality.
   - **Use:** Focus on one specific value, e.g., sales in *Q1* across all products and regions.
   - **Purpose:** Simplifies analysis by isolating one dimension of interest.
2. **Dice**
   - **Definition:** Selects a **subset of values across multiple dimensions** to form a smaller sub-cube.
   - **Use:** Examine sales for specific product categories in selected regions and time periods.
   - **Purpose:** Enables focused analysis across multiple criteria.
3. **Roll-Up (Aggregation)**
   - **Definition:** Aggregates data by moving **up a hierarchy** (lower detail → higher summary).
   - **Use:** Summarize daily sales to monthly or yearly totals.
   - **Purpose:** Provides broader trends and higher-level insights.
4. **Drill-Down**
   - **Definition:** Opposite of roll-up; moves from **higher summary → detailed level**.
   - **Use:** From annual sales to quarterly and monthly breakdowns.
   - **Purpose:** Reveals detailed underlying data and finer insights.
5. **Pivot (Rotation)**
   - **Definition:** Rotates the cube axes to change the **viewing perspective**.
   - **Use:** Switch rows and columns in a report, e.g., view *Product vs Time* instead of *Region vs Time*.
   - **Purpose:** Offers alternative views without changing the data itself for deeper analysis.