Unit 2

The Architecture of DW and BI

BI and DW architectures and its types - Relation between BI and DW - OLAP (Online analytical processing) definitions - Different OLAP Architectures-Data Models-Tools in Business Intelligence-Role of DSS, EIS, MIS and digital Dash boards – Need for Business Intelligence
Difference between OLAP and OLTP - Dimensional analysis - What are cubes? Drill-down and roll-up - slice and dice or rotation - OLAP models - ROLAP versus MOLAP - defining schemas: Stars, snowflakes and fact constellations.

Data Warehouse

- A data warehouse is a large, centralized repository of data that is used to support business decision-making activities.
- Data is extracted from a variety of sources and integrated into the warehouse.
- The data is organized in a way that makes it easy to query and analyze.
- A data warehouse typically contains historical data, which can be used for trend analysis and forecasting.
- Data warehouses are used to support business intelligence (BI) activities, such as reporting, data mining, and online analytical processing (OLAP).
- Data warehouses are designed to support the reporting and analysis of data, rather than the day-to-day transaction processing that occurs in operational systems.
- They are optimized for read operations, and data is structured in a way that makes it easy to access and analyze.
- Data warehouses often use an architecture called Extract, Transform, Load (ETL) to move data from operational systems into the warehouse.
- In addition to ETL, data warehouses often use other techniques to ensure that data is accurate and consistent.
- For example, they may use data profiling to identify data quality issues or data cleansing to correct errors.
- Data warehouses may also use metadata to provide additional information about the data in the warehouse, such as when it was loaded or who is responsible for maintaining it.

- Overall, data warehouses play a critical role in supporting the data-driven decision-making that is essential for modern businesses.
- By providing a centralized repository of data that is optimized for analysis, data warehouses enable organizations to gain insights into their operations and make more informed decisions.

Difference between a data warehouse and a database

- A database is a collection of data that is organized and stored in a structured manner, with the goal of efficiently storing and retrieving data. Databases can be used for a variety of purposes, from simple record-keeping to complex data analysis and management.
- A data warehouse, on the other hand, is a large, centralized repository of data that is specifically designed for supporting business intelligence activities, such as reporting, data analysis, and data mining. Data warehouses are often used to store historical data from various sources in a format that is optimized for querying and analysis.

While both databases and data warehouses store data, they differ in several key ways:

1. Purpose: Databases are designed to efficiently store and retrieve data for operational purposes, such as transaction processing or record-keeping. Data warehouses are designed to support business intelligence activities, such as data analysis and reporting.
2. Schema: Databases typically use a normalized schema, which means that data is stored in multiple tables with relationships between them. Data warehouses, on the other hand, often use a denormalized schema, which means that data is consolidated into fewer tables to simplify querying and analysis.
3. Data volume: Databases typically store data for current or recent transactions, and may only contain a few months or years of data. Data warehouses, on the other hand, may store large volumes of historical data, sometimes spanning many years.
4. Querying and analysis: Databases are designed to support transaction processing and simple queries, while data warehouses are designed to support complex queries and analysis. Data warehouses often use specialized tools and techniques, such as OLAP (Online Analytical Processing) and data mining, to analyze large volumes of data.

Key features of DW

1. Historical data: A data warehouse stores historical data over a long period of time, typically spanning several years. This enables trend analysis and comparison of data over time.
2. Subject-oriented: A data warehouse is designed to be subject-oriented, which means that it organizes data by subject or topic rather than by application or department. This enables better integration and analysis of data across the organization.
3. Integrated: A data warehouse integrates various heterogeneous data sources like RDBMS, flat files, and online transaction records. It requires performing data cleaning and integration during data warehousing to ensure consistency in naming conventions, attribute types, etc., among different data sources.
4. Non-volatile: A data warehouse is non-volatile, which means that once data is loaded into the data warehouse, it is not changed. This enables better data consistency and accuracy and supports historical analysis.
5. Optimized for querying: A data warehouse is optimized for querying and analysis, and typically uses a denormalized schema to simplify queries and improve performance. It may also use specialized techniques such as OLAP (Online Analytical Processing) and data mining to analyze data.
6. Metadata: A data warehouse includes metadata, which describes the structure, content, and origin of the data. This enables better understanding and management of the data in the data warehouse.

Need for a dw

1. Data integration: Organizations often have data scattered across multiple systems and applications, which can make it difficult to get a complete picture of the business. A data warehouse consolidates data from various sources into a single, centralized repository, making it easier to integrate and analyze data.
2. Historical analysis: Many organizations need to analyze data over time to identify trends, patterns, and changes in their business. A data warehouse stores historical data over a long period of time, making it easier to analyze trends and changes.
3. Business intelligence: A data warehouse is a key component of a business intelligence (BI) system, which enables organizations to make better decisions

based on data. A data warehouse provides a foundation for BI by providing a single, consistent view of the organization's data.

4. Performance: Operational databases are optimized for transaction processing, which can make querying and analysis slow and resource-intensive. A data warehouse is optimized for querying and analysis, which can improve performance and reduce the load on operational systems.
5. Data quality: Data quality is a critical issue for many organizations, as inaccurate or inconsistent data can lead to poor decisions and lost opportunities. A data warehouse can improve data quality by integrating and consolidating data from various sources and providing tools for data cleansing and validation.

Benefits of dw

1. Improved decision-making: A data warehouse provides a single, consistent view of an organization's data, which can help decision-makers to make better informed and more timely decisions.
2. Increased efficiency: By consolidating and integrating data from various sources, a data warehouse can improve the efficiency of data analysis and reporting.
3. Better data quality: A data warehouse can improve the quality of an organization's data by providing tools for data cleansing and validation, as well as ensuring that data is consistent across all systems.
4. Historical analysis: A data warehouse stores historical data over a long period of time, which enables trend analysis and comparison of data over time.
5. Cost savings: By reducing the need for data redundancy and improving data quality, a data warehouse can help an organization to save costs.
6. Competitive advantage: With better decision-making and more efficient data analysis, a data warehouse can provide a competitive advantage by enabling an organization to respond more quickly and effectively to changing market conditions.
7. Improved data governance: A data warehouse can provide a framework for data governance, ensuring that data is managed effectively and securely across the organization.

Properties of dw architectures

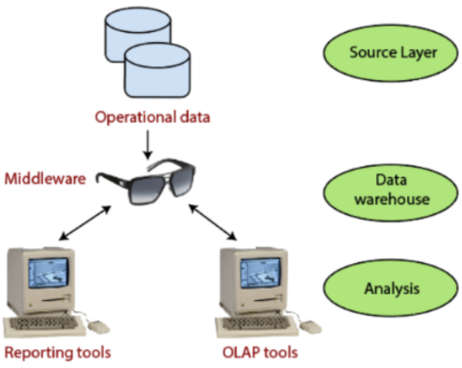
1. Separation: Analytical and transactional processing should be kept apart as much as possible.

2. Scalability: A data warehouse architecture must be scalable to handle large volumes of data, especially as the size of the organization grows. This requires a distributed architecture that can handle high volumes of data, as well as clustering and load balancing to ensure optimal performance
3. Extensibility: The architecture should be able to perform new operations and technologies without redesigning the whole system.
4. Security: A data warehouse architecture must be secure to protect the confidentiality, integrity, and availability of the data. This requires access control mechanisms, authentication and authorization procedures, and encryption to protect data both in transit and at rest.
5. Administrability: Data Warehouse management should not be complicated.

Types of dw architectures

1. Single-tier architecture:

- A single-tier architecture is the simplest type of data warehouse architecture, where all the components of the data warehouse are installed on a single server.
- This approach is suitable for small organizations with limited data volumes and low query complexity.
- The objective of a single layer is to minimize the amount of data stored.
- The goal is to remove data redundancy.
- This architecture is not frequently used in practice.
- 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.



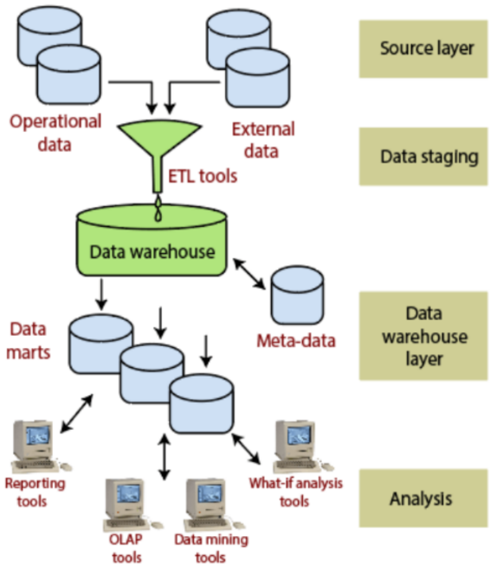
Single-Tier Data Warehouse Architecture

- The vulnerability of this architecture lies in its failure to meet the requirement for separation between analytical and transactional processing.
- Analysis queries are agreed to operational data after the middleware interprets them. In this way, queries affect transactional workloads.

2. Two-tier architecture:

- In a two-tier architecture, the data warehouse is divided into two tiers: the database tier and the client tier.
- The database tier stores the data and the client tier includes tools for data analysis and reporting.
- This architecture provides better scalability and performance than a single-tier architecture.
- Two-layer architecture is one of the Data Warehouse layers which separates physically available sources and data warehouse.
- This architecture is not expandable and also does not support a large number of end-users.
- It also has connectivity problems because of network limitations.
- 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 in corporate relational databases or legacy databases, or it may come from an information system outside the corporate walls.

- Data Staging: The data stored in 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.
- Data Warehouse layer: Information is saved to one logically centralized individual repository: a data warehouse. The data warehouses can be directly accessed, but they 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 flexibly 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.



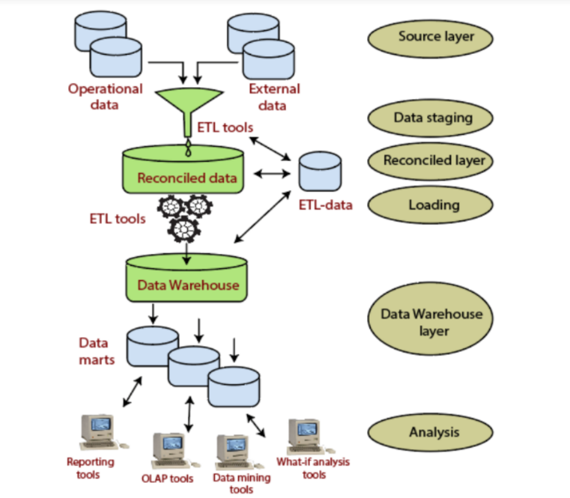
Two-Tier Data Warehouse Architecture

3. Three-tier architecture:

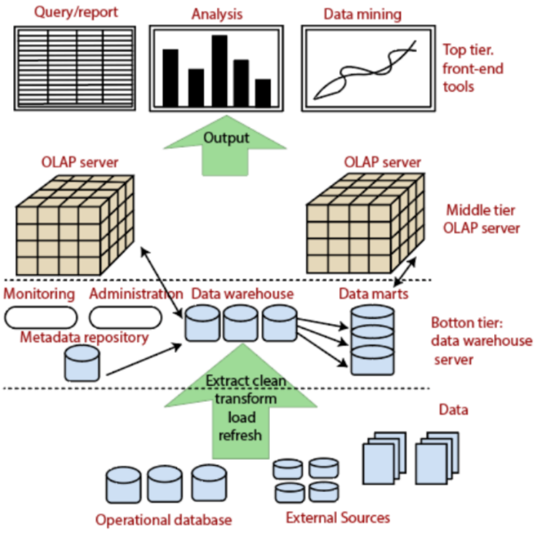
- A three-tier architecture is similar to a two-tier architecture but includes an additional layer, the application server tier, which acts as an intermediary between the client tier and the database tier.
- This architecture provides better scalability, flexibility, and security than a two-tier architecture.

- It consists of the Bottom Tier (Data Warehouse Server), Middle Tier (OLAP Server), and Top Tier (Front end Tools).
- Bottom Tier: The database of the Datawarehouse servers is the bottom tier. It is usually a relational database system. Data is cleansed, transformed, and loaded into this layer using back-end tools.
- Middle Tier: The middle tier in the Data warehouse is an OLAP server which is implemented using either ROLAP or MOLAP model.
- A Relational OLAP (ROLAP) model, i.e., an extended relational DBMS that maps functions on multidimensional data to standard relational operations.
- A Multidimensional OLAP (MOLAP) model, i.e., a particular purpose server that directly implements multidimensional information and operations.
- For a user, this application tier presents an abstract view of the database. This layer also acts as a mediator between the end user and the database.
- Top-Tier: The top tier is a front-end client layer. The top tier is the tools and API that you connect and get data out from the data warehouse. It could be Query tools, reporting tools, managed query tools, Analysis tools and Data mining tools.
- The three-tier architecture consists of the source layer (containing multiple source systems), 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 the data warehouse population.
- In some cases, the reconciled layer is also directly used to accomplish 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 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 for a data warehouse system



Three-Tier Data Warehouse Architecture

4. Hybrid architecture: A hybrid architecture combines two or more types of architectures, depending on the specific needs of the organization. For example, an organization might use a two-tier architecture for its core data warehouse, but add data marts using a single-tier architecture to provide additional functionality and scalability.

5. Cloud-based architecture: A cloud-based data warehouse architecture uses cloud computing resources to store and manage data. This architecture offers the benefits of scalability, flexibility, and cost-effectiveness, and can be easily integrated with other cloud-based services. The main cloud data warehouses are: Amazon Redshift, Google BigQuery, Microsoft Azure SQL Data Warehouse, Snowflake

Relation between BI and DW

Business Intelligence (BI) and Data Warehousing (DW) are closely related concepts and are often used together to support data-driven decision-making in organizations. Here are some key notes on the relationship between BI and DW:

1. DW is the foundation: A data warehouse is a central repository of data that is designed for efficient querying, analysis, and reporting. BI tools rely on the data warehouse as their primary data source and are used to extract insights and intelligence from the data stored in the data warehouse.

2. BI enables analysis and reporting: BI tools provide a set of tools and technologies that enable users to analyze and report on the data stored in the data warehouse. BI tools can be used to create dashboards, reports, visualizations, and other forms of data analysis that help users understand trends, patterns, and insights in the data.
3. DW provides data integration: One of the key benefits of a data warehouse is that it integrates data from multiple sources into a single repository, providing a unified view of the organization's data. This makes it easier for BI tools to access and analyze the data, without having to worry about data integration or data quality issues.
4. BI drives business value: The primary goal of BI is to drive business value by providing insights and intelligence that help organizations make better decisions. By combining BI tools with a data warehouse, organizations can achieve a higher level of data-driven decision-making, and gain a competitive advantage in their industry.

Difference between BI and DW

Category	Business Intelligence (BI)	Data Warehousing (DW)
Purpose	Provide tools and technologies for data analysis and visualization	Provide a central repository of integrated data optimized for querying, reporting, and analysis
Usage	Generating business insights	Storing data from several sources
Data Volume	Works with relatively smaller data sets	Handles large volumes of data
Querying and Reporting	Supports ad-hoc and interactive queries, and provides reporting and dashboard tools	Optimized for complex querying and large-scale reporting
Data Structure	Relies on structured data sources, such as relational databases	Can handle both structured and unstructured data
Performance	Designed for fast and interactive analysis, with results delivered in near real-time	Optimized for efficient querying and reporting, with fast response times and high scalability
Output	Data visualization, dashboards, and reports	Unified data for upstream BI applications
Audience	C-level executives, managers, and data analysts	DW engineers, back-end developers
User Interface	Provides user-friendly tools for data	Typically accessed via SQL queries

	analysis and visualization, often with drag-and-drop interfaces	or specialized reporting tools
Tools	Datapine, Power BI, Tableau, Micro Strategy, SAS business intelligence, Yellowfin BI, QIikSense	Amazon Redshift, Google BigQuery, Microsoft Azure SQL data warehouse, Snowflake
Technology	Relies on a variety of tools and technologies, including data visualization tools, analytics software, and reporting tools	Uses specialized database management software and technologies, such as ETL tools, to manage and optimize data

OLAP (Online Analytical Processing)

- OLAP (Online Analytical Processing) is a type of software technology that is used to manage and analyze large amounts of data in a multidimensional and interactive manner.
- OLAP is based on the concept of a cube, which is a multidimensional representation of data, and it enables users to quickly and easily view and analyze data from different perspectives.
- OLAP is a classification of software technology that authorizes analysts, managers, and executives to gain insight into information through fast, consistent, interactive access to a wide variety of possible views of data that have been transformed from raw information to reflect the real dimensionality of the enterprise as
  understood by the clients.
- OLAP implements the multidimensional analysis of business information and supports the capability for complex estimations, trend analysis, and sophisticated data modeling.
- It is rapidly enhancing the essential foundation for Intelligent Solutions containing Business Performance Management, Planning, Budgeting, Forecasting, Financial Documenting, Analysis, Simulation-Models, Knowledge Discovery, and Data Warehouses Reporting.
- OLAP enables end clients to perform ad hoc analysis of records in multiple dimensions, providing the insight and understanding they require for better decision-making.
- OLAP cubes have two main purposes:

1. To provide business users with a data model more
   intuitive to them than a tabular model. This model is called a Dimensional Model.

2. To enable fast query response that is usually difficult to achieve using tabular models.

- **OLAP applications are used by:**

1. Finance and accounting: Budgeting, Activity-based costing, Financial performance analysis, and financial modeling
2. Sales and marketing: Sales analysis and forecasting, Market research analysis, Promotion analysis, Customer analysis, Market and customer segmentation
3. Production: Production planning, Defect analysis
4. Healthcare: analyze patient data and identify patterns and trends.

- Fundamentally, OLAP has a very simple concept. It pre-calculates most of the queries that are typically very hard to execute over tabular databases, namely aggregation, joining, and grouping.
- These queries are calculated during a process that is usually called 'building' or 'processing' of the OLAP cube.
- This process happens overnight, and by the time end users get to work - data will have been updated.
- **Main characteristics of OLAP:**

1. Multidimensional conceptual view: OLAP systems let business users have a dimensional and logical view of the data in the data warehouse. It helps in carrying slice and dice operations.
2. Multi-User Support: Since the OLAP techniques are shared, the OLAP operation should provide normal database operations, containing retrieval, update, adequacy control, integrity, and security.
3. Accessibility: OLAP acts as a mediator between data warehouses and front end. The OLAP operations should be sitting between data sources (e.g., data warehouses) and an OLAP front end.
4. Storing OLAP results: OLAP results are kept separate from data sources.
5. Uniform documenting performance: Increasing the number of dimensions or database size should not significantly degrade the reporting performance of the OLAP system.
6. OLAP provides for distinguishing between zero values and missing values so that aggregates are computed correctly.

7. OLAP system should ignore all missing values and compute correct aggregate values.
8. OLAP facilitates interactive queries and complex analysis for the users.
9. OLAP allows users to drill down for greater details or roll up for aggregations of metrics along a single business dimension or across multiple dimensions.
10. OLAP provides the ability to perform intricate calculations and comparisons.
11. OLAP presents results in a number of meaningful ways, including charts and graphs.

- **Benefits of OLAP:**

1. OLAP helps managers in decision-making through the multidimensional record views that it is efficient in providing, thus increasing their productivity.
2. OLAP functions are self-sufficient owing to the inherent flexibility that supports organized databases.
3. It facilitates the simulation of business models and problems, through extensive management of analysis capabilities.
4. In conjunction with a data warehouse, OLAP can be used to support a reduction in the application backlog, faster data retrieval, and a reduction in query drag.

- OLAP tools can be divided into two main categories: multidimensional OLAP (MOLAP) and relational OLAP (ROLAP).
- **Comparison between MOLAP and ROLAP:**

	MOLAP	**ROLAP**
Data Storage	Stores data in a multidimensional cube format	Stores data in a relational database format
Data Aggregation	Pre-aggregates data in the cube	Aggregates data on the fly, as needed
Performance	Generally faster than ROLAP due to the pre-aggregation of data	Can be slower than MOLAP due to the need to aggregate data on the fly
Scalability	Limited scalability due to the need to pre-calculate and store data in the cube	Can handle larger amounts of data due to the use of relational databases
Flexibility	Less flexible than ROLAP,	More flexible than MOLAP, as it allows for ad

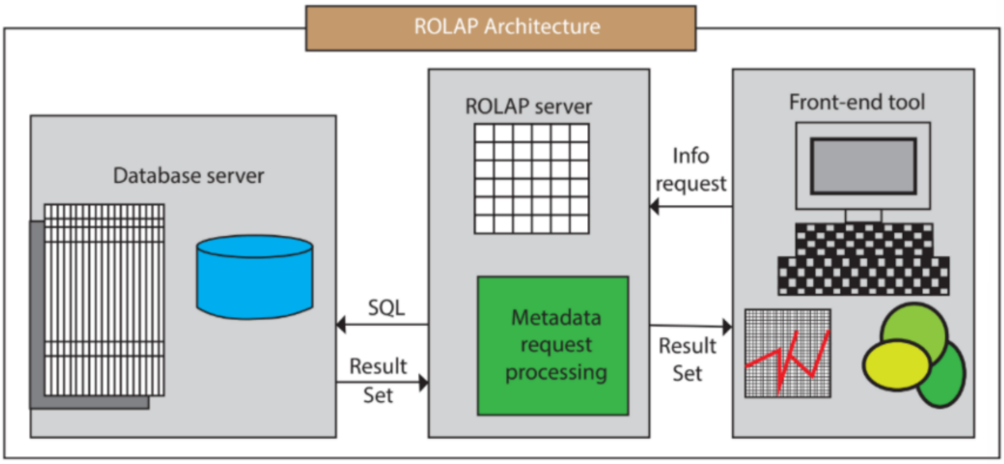
	as the structure of the cube is pre-defined	hoc querying and the ability to change the structure of the data
Complexity	Less complex than ROLAP due to the use of a pre-defined cube structure	Can be more complex than MOLAP due to the need to create and manage relationships between data in relational databases
Examples	Microsoft SQL Server Analysis Services, Essbase, Oracle OLAP	SAP BusinessObjects, Oracle BI, IBM Cognos

- There are three main types of OLAP servers: ROLAP, MOLAP, HOLAP (Hybrid OLAP)

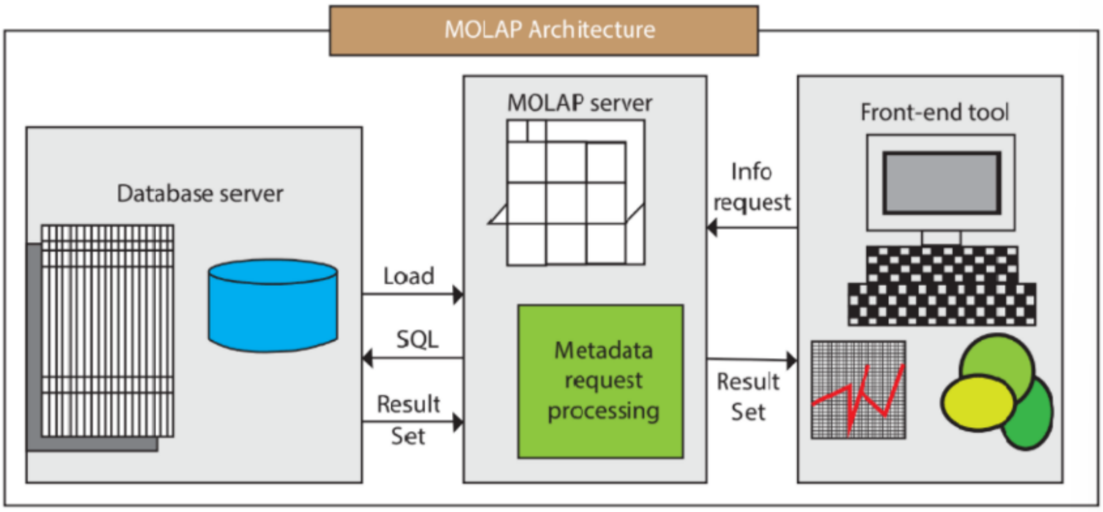
	MOLAP	**ROLAP**	**HOLAP**
Data Storage	Stores data in a multidimensional cube format	Stores data in a relational database format	Combines features of both MOLAP and ROLAP, storing summary data in a cube and detailed data in a relational database
Data Aggregation	Pre-aggregates data in the cube	Aggregates data on the fly, as needed	Pre-aggregates summary data in the cube, and accesses detailed data from the relational database as needed
Performance	Generally faster than ROLAP due to the pre-aggregation of data	Can be slower than MOLAP due to the need to aggregate data on the fly	Faster than ROLAP for aggregated data, and faster than MOLAP for detailed data
Scalability	Limited scalability due to the need to pre-calculate and store data in the cube	Can handle larger amounts of data due to the use of relational databases	Can handle both large and complex data sets, using the cube for summary data and the relational database for detailed data
Flexibility	Less flexible than ROLAP, as the structure of the cube is pre-defined	More flexible than MOLAP, as it allows for ad hoc querying and the ability to change the structure of the data	Offers flexibility for both aggregated and detailed data
Complexity	Less complex than ROLAP	Can be more complex than MOLAP due to the	Can be more complex than MOLAP, due to the

	due to the use of a pre-defined cube structure	need to create and manage relationships between data in relational databases	need to manage both cube and relational database structures
Examples	Microsoft SQL Server Analysis Services, Essbase, Oracle OLAP	SAP BusinessObjects, Oracle BI, IBM Cognos	SAP HANA, Microsoft Analysis Services, IBM Cognos TM1

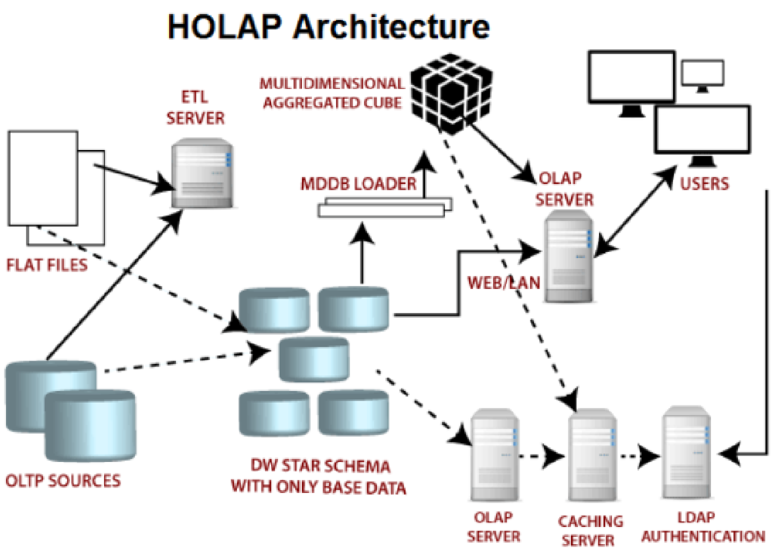
- ROLAP architecture:



- MOLAP architecture:

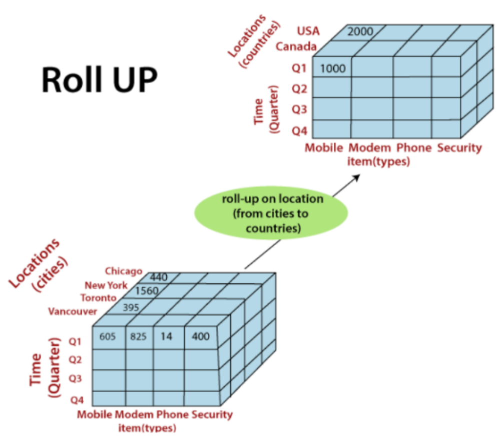


- HOLAP architecture:

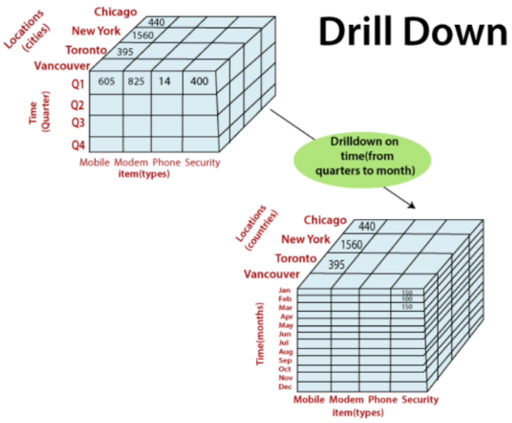


- **Five basic OLAP operations:**

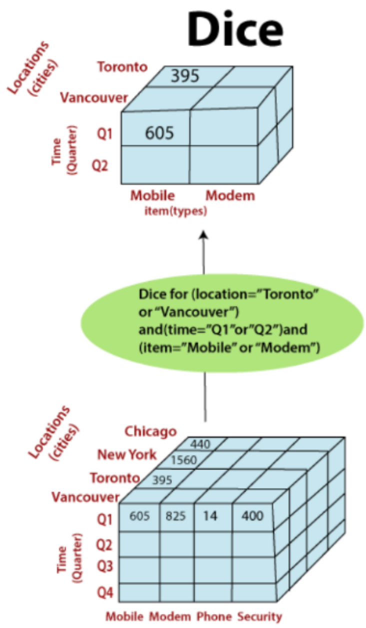
1. Roll-up (Drill-up): This operation allows users to view summarized data at higher levels of aggregation. For example, a user might drill up from monthly sales figures to quarterly or yearly sales figures.



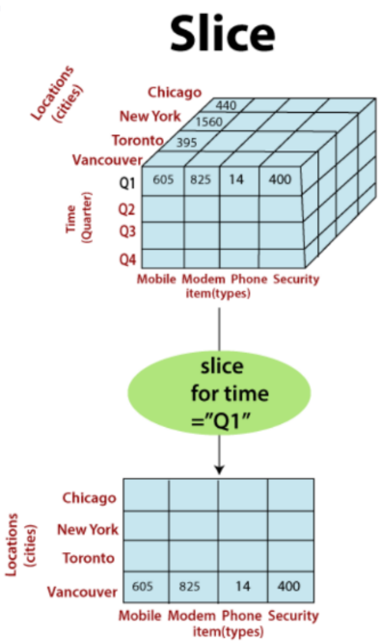
2. Drill-down (Roll-down): This operation allows users to view detailed data at lower levels of aggregation. For example, a user might drill down from yearly sales figures to monthly or daily sales figures.



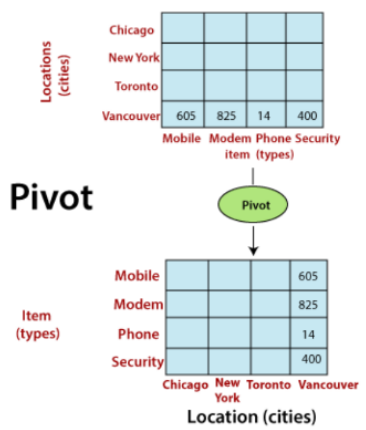
3. Dice: It selects a sub-cube from the OLAP cube by selecting two or more dimensions. In the cube given in the overview section, a sub-cube is selected by selecting the following dimensions with criteria: Location = “Delhi” or “Kolkata”, Time = “Q1” or “Q2”, Item = “Car” or “Bus”



4. Slice: It selects a single dimension from the OLAP cube which results in a new sub-cube creation. In the cube given in the overview section, Slice is performed on the dimension Time = “Q1”



5. Pivot (Rotate): This operation allows users to rotate or transpose the data, changing the orientation of the axes. It rotates the current view to get a new view of the representation. In the sub-cube obtained after the slice operation, performing the pivot operation gives a new view of it. For example, a user might pivot sales data so that product types are listed along the X-axis, and time periods are listed along the Y-axis.



- **Differences between OLTP and OLAP:**

Aspect	OLTP	OLAP
Purpose	Handles operational or transactional data	Handles analytical data
Users	Clerk, IT professional	Knowledge worker

Function	Day-to-day operations	Decision support
DB design	Application-oriented	Subject oriented
Data	Current, up-to-date, detailed, flat relational isolated	Historical, summarized, multidimensional integrated, consolidated
Database Structure	Normalized databases	Denormalized databases
Data Volume	Handles high volumes of transactions	Handles large volumes of data for analysis
Data Latency	Low latency, real-time or near real-time	May have higher latency due to data aggregation
Time dimension	Implicit and current	Explicit and variant
Updating	Continuous and irregular	Periodic and regular
Flexibility	Low	High
Performance	High, a few seconds per query	Maybe low for complex queries
Priority	High performance	High flexibility
Size	mb to gb	gb to tb
Volatility	Dynamic data	Static data

Role of DSS, EIS, MIS, and digital dashboards

Decision Support System (DSS)

- DSS is an interactive computer-based system designed to help decision-makers utilize data and models to identify and solve problems.
- It is specially developed for supporting the solution of a non-structured management problem for improved decision-making.
- In BI, DSS helps to:

1. Gather and integrate data from various sources: DSS can integrate data from different sources, including data warehouses, spreadsheets, and databases, and help to organize it in a way that makes it easy to analyze and make decisions.
2. Analyze and visualize data: DSS can help decision-makers analyze and visualize data in a way that makes it easy to understand trends, patterns, and insights.
3. Model and forecast: DSS can help to model and forecast future scenarios based on historical data, providing decision-makers with the information they need to make informed decisions.

4. Support decision-making: DSS provides decision-makers with the tools they need to analyze data and make informed decisions. This includes providing access to data, reports, dashboards, and analytics tools that can be used to explore data and identify trends and patterns.

- Advantages:

1. It saves time.
2. Enhances efficiency.
3. Reduces the cost.
4. It improves personal efficiency.
5. It increases the decision-makers' satisfaction.

- Disadvantages:

1. Information Overload.
2. Status reduction.
3. Over-emphasize decision-making.

Executive Information System (EIS)

- EIS is defined as a system that helps the high-level executives to take policy decisions.
- Provides executives with quick access to key performance indicators (KPIs) and other critical information, enabling them to make informed decisions.
- This system uses high-level data, analytical models, and user-friendly software for taking decisions.
- It is a structured, automated tracking system that operates continuously to keep everything managed.
- It provides exception and status reporting capabilities.
- In BI, EIS helps to:

1. Monitor organizational performance: EIS provides senior executives with real-time access to critical KPIs (key performance indicators) and other performance metrics, allowing them to monitor the organization's performance and identify potential problems.
2. Analyze data and identify trends: EIS can help executives analyze data and identify trends and patterns, providing them with the information they need to

make informed decisions.

3. Provide insights and forecasts: EIS can provide executives with insights and forecasts based on historical data and trend analysis, allowing them to anticipate future trends and make strategic decisions.
4. Support collaboration and communication: EIS can facilitate collaboration and communication between different departments and teams, providing executives with a comprehensive view of the organization's performance and enabling them to work together to achieve common goals.

- Advantages:

1. Easy to use.
2. Ability to analyze the trends.
3. Time management.
4. Efficiency.
5. Enhances business problem-solving.

- Disadvantages:

1. Functions are limited.
2. Difficult to keep current data.
3. The system can run slowly.
4. Less reliable.

Management Information System (MIS)

- MIS is a computer-based system that provides managers with reports and other information related to the day-to-day operations of the organization.
- MIS consists of the following three pillars: Management, Information, and System.

1. Management: The art of getting things done through and with the people of formally organized groups. Managerial functions:

(i) Planning (ii) Organizing (iii) Staffing (iv) Directing

(v) Controlling

2. Information: Data that has a meaning with a context, where data is raw facts about an entity (entity is the object of interest).

3. System: A set of interrelated components with a clearly defined boundary working together to achieve a common goal.

- In BI, MIS helps to:

1. Collect and manage data: MIS can collect and manage data from different sources, including internal and external databases, spreadsheets, and other systems, making it easy for managers to access and use the information.
2. Generate reports and dashboards: MIS can generate reports and dashboards that provide managers with an overview of key performance indicators (KPIs), allowing them to monitor performance and identify potential problems.
3. Analyze data and identify trends: MIS can analyze data and identify trends and patterns, providing managers with the information they need to make informed decisions.
4. Streamline processes: MIS can help managers streamline processes, reducing duplication and errors, and improving overall efficiency.

- Advantages:

1. Improves the quality of an organization or information content by providing relevant information for sound decision-making.
2. MIS changes a large amount of data into a summarized form and thereby avoids confusion
3. MIS facilitates the integration of specialized activities by keeping each department aware of the problems and requirements of other departments.
4. MIS serves as a link between managerial planning and control. It improves the ability of management to evaluate and improve performance.

- Disadvantages:

1. Too rigid and difficult to adapt.
2. Resistance to sharing internal information between departments can reduce effectiveness.
3. Hard to quantify benefits to justify the implementation of MIS.
4. The quality of output of an MIS is directly proportional to the quality of input and processes.

- **Difference between dss, eis and mis:**

Features/Functions	DSS (Decision Support System)	EIS (Executive Information System)	MIS (Management Information System)
User	Analysts, managers	Senior executives	Middle, lower-level managers
Purpose	To support semi-structured and unstructured decisions	To provide information for strategic decision-making	To support day-to-day operational decision-making
Application	Diversified areas where managerial decisions are made	Environmental scanning, performance evaluation, identifying problems and opportunities	Production control, sales forecasts, financial analysis, HR management
Principal use	Planning, organizing, staffing and control	Tracking and control	Control
Focus	Analysis and decision-making	Monitoring and control	Data collection and reporting
Data	Internal and external	Internal	Internal
Time frame	Short to medium term	Long term	Short term
Scope	Departmental or functional	Enterprise-wide	Departmental or functional
Functionality	What-if analysis, goal-seeking, optimization	Analysis, forecasting, performance measurement	Reporting, data query, exception reporting
Outputs	Ad-hoc reports, dashboards, charts, graphs	Executive summaries, key performance indicators (KPIs), alerts	Standard reports, pre-formatted reports, exception reports
Hardware	Mainframes, micros or distributed	Distributed system	Mainframes, micros or distributed
Example	Market research, financial analysis	Competitive analysis, market trends	Inventory control, sales analysis

Digital Dashboard

- A digital dashboard is a business intelligence tool that allows business leaders to track, analyze and report on KPIs and metrics.
- Modern, interactive dashboards make it easy to combine data from multiple sources and deeply explore and analyze the data directly within the dashboard itself.
- In informational systems, a Digital Dashboard is usually a a single page that:

1. Contains plain and easy-to-understand data
2. Includes a real-time UI
3. Is integrated with a database for dynamic updating
4. Features graphic elements that reveal the change of key indicators and current trends of the system.

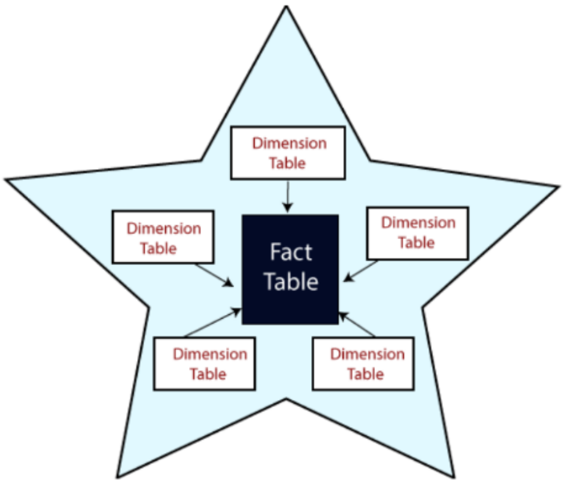
Star Schema

- A star schema is a common type of data model used in BI (Business Intelligence) to organize data into a set of fact tables and dimension tables.
- 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.
- The star schema is called a "star" because the fact table is located in the center, surrounded by the dimension tables, which are arranged in a star-like pattern.
- Here are the key components of a star schema: (same for other two schemas)

1. Fact table: This table contains the quantitative data that is being analyzed, such as sales revenue or customer orders. It typically includes foreign keys that link to the dimension tables.
2. Dimension tables: These tables contain descriptive data that provides context for the measures in the fact table. For example, a dimension table for customers might include attributes such as customer ID, name, address, and demographics.
3. Foreign keys: These are fields in the fact table that reference the primary keys in the dimension tables. They create a relationship between the fact table and the dimension tables.

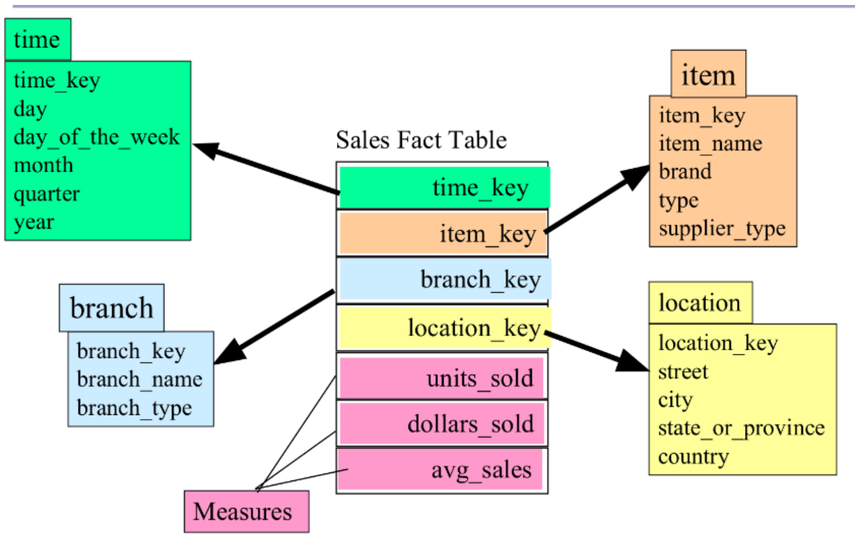
4. Measures: These are the quantitative values that are being analyzed, such as sales revenue, units sold, or profit margin.

- The star schema is a popular choice for BI applications because it is easy to understand and query, and it provides a flexible structure for analyzing data.
- The fact table and dimension tables can be easily expanded to include new measures or dimensions as needed, making it a scalable solution for BI.



Star Schema

- Example of 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 six columns for time data, five columns for item data, three columns for branch data, and five 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 a single change in the dimension table (surrounding tables), instead of making many changes in the fact table (center table).
- **Characteristics of star schema**

1. It creates a DE-normalized database that can quickly provide query responses.
2. It provides a flexible design that can be changed easily or added to throughout the development cycle, and as the database grows.
3. It provides a parallel in design to how end-users typically think of and use the data.
4. It reduces the complexity of metadata for both developers and end-users.

- **Advantages of star schema**

1. Simplified queries: The star schema makes it easy to query data, even with complex queries that involve multiple tables. Because the schema is highly denormalized, there are fewer joins required, which speeds up query processing time.
2. Improved performance: The simplified queries and reduced number of joins in the star schema lead to faster query processing times, making it possible to analyze large volumes of data in real time.
3. Easier to understand: The star schema is easy to understand and interpret, even for non-technical users. It provides a clear structure for organizing data and makes it easy to identify relationships between tables.
4. Flexibility: The star schema is highly flexible and can be easily modified to accommodate changes in the data or business requirements. It is also scalable, making it suitable for use with large volumes of data.
5. Better analytics: The star schema provides a clear structure for organizing data, making it easier to analyze and interpret. This, in turn, leads to better insights and decision-making.
6. Reduced data redundancy: The star schema is highly denormalized, which means there is less data redundancy. This results in more efficient use of storage space and faster query processing times.

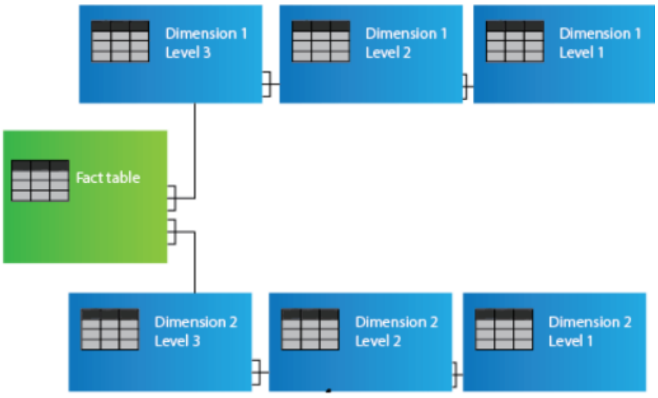
- **Disadvantages of star schema**

1. Complexity: The star schema can become complex as the number of dimensions and measures increases. This can make it more difficult to understand and maintain the schema, especially for non-technical users.
2. There are some conditions which cannot be met by star schemas like the relationship between the user and bank account cannot be described as a star schema as the relationship between them is many to many.

Snowflake schema

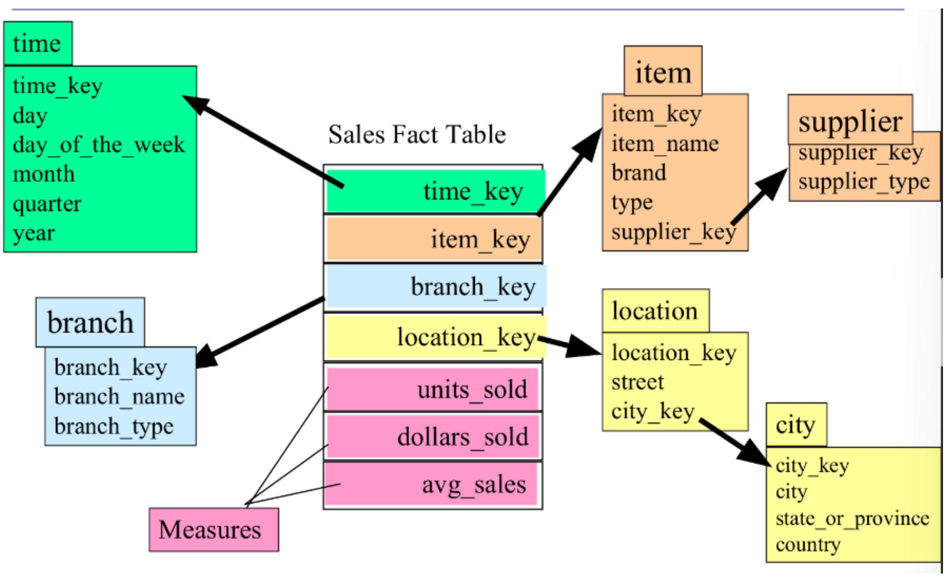
- Similar to the star schema
- Normalized form of star schema designed to further reduce data redundancy
- A schema is known as a snowflake if one or more dimension tables do not connect directly to the fact table but must join through other dimension tables.
- The snowflake schema is an expansion of the star schema where each point of the star explodes into more points.
- The fact table remains at the center of the schema and is connected to the dimension tables using foreign keys, as in the star schema.
- 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.
- Tables in a snowflake schema are generally normalized to the third normal form.
- Each dimension table performs exactly one level in a hierarchy.
- It results in better performance and reduced storage requirements.
- A snowflake schema is designed for flexible querying across more complex dimensions and relationships.
- It is suitable for many-to-many and one-to-many relationships between dimension levels.
- However, it can be more complex to query and maintain due to the additional tables and relationships.
- Overall, the choice between the star schema and the snowflake schema will depend on the specific needs of the organization and the nature of the data being analyzed.

- The following diagram shows a snowflake schema with two dimensions, each having three levels. A snowflake schema can have any number of dimensions, and each dimension can have any number of levels.



Snowflake Schema

- Example of snowflake schema:



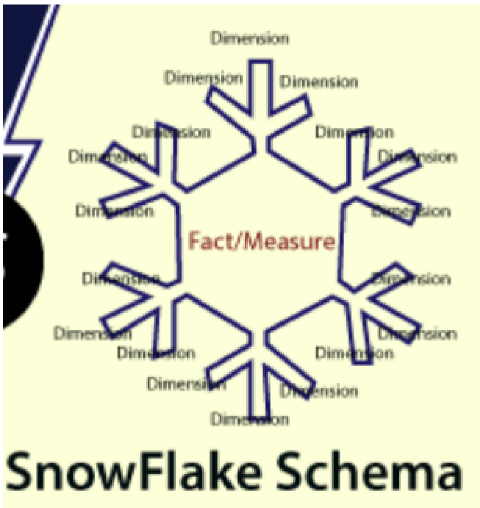
- Advantages

1. Reduced data redundancy: The snowflake schema reduces data redundancy even further than the star schema, which can result in better performance and reduced storage requirements.
2. Improved data quality: Because the dimension tables in the snowflake schema are normalized into multiple related tables, data quality can be improved. This is because each piece of data is stored in only one place, reducing the risk of inconsistencies or errors.

3. More flexible: The snowflake schema is more flexible than the star schema because it allows for a greater degree of normalization. This means that changes to the data structure can be made more easily and with less impact on the overall schema.
4. Better suited for large and complex data: The snowflake schema is better suited for large and complex data sets, where data redundancy and normalization are important factors in improving performance.
5. Scalability: The snowflake schema is highly scalable, making it suitable for use with large volumes of data. This is because the schema is designed to reduce data redundancy and improve performance.
6. Better for managing data changes: The snowflake schema is better for managing changes to the data structure because it allows for more flexible changes to be made without impacting the overall schema.

- Disadvantages

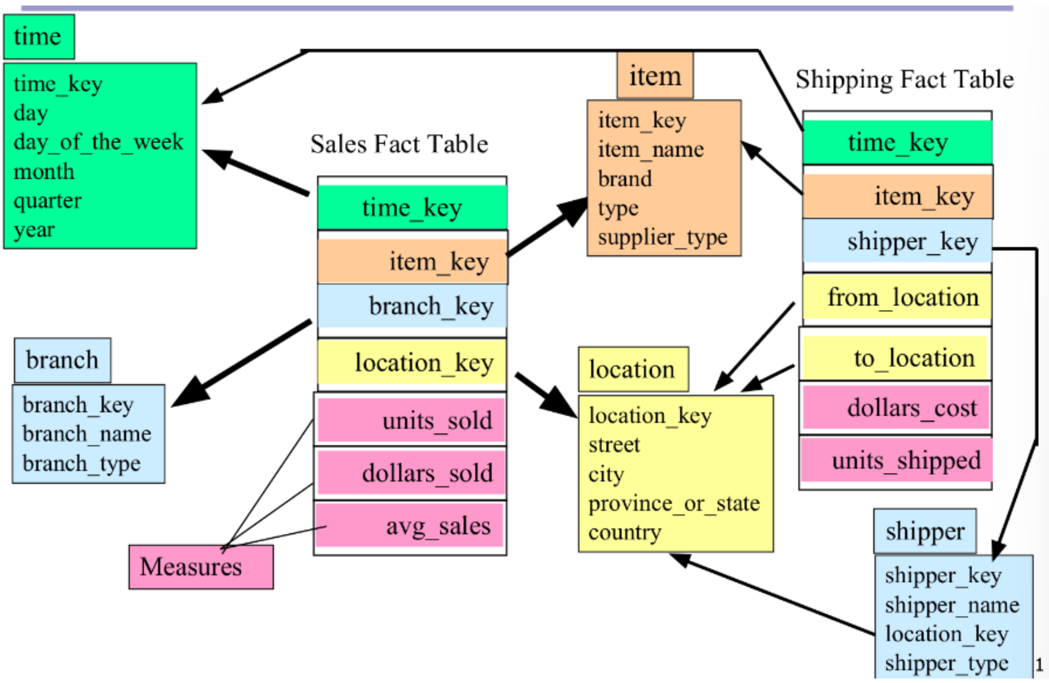
1. The primary disadvantage of the snowflake schema is the additional maintenance efforts required due to the increasing number of lookup tables. It is also known as a multi-fact star schema.
2. There are more complex queries and hence, difficult to understand.
3. More tables, more joins so more query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



Fact constellation schema

- Combines multiple fact tables and dimension tables into a single schema.
- It is also known as the galaxy schema or the fact and dimension bus schema.

- In the fact constellation schema, multiple fact tables are connected to multiple dimension tables, forming a network of relationships.
- The fact tables can share common dimensions, and the dimension tables can be shared among multiple fact tables.
- This creates a flexible and scalable schema that can support complex data analysis.
- The fact constellation schema is designed to provide a flexible and scalable data model for complex BI applications.
- It can handle multiple fact tables and dimension tables and can support a wide range of analysis requirements.
- However, it can be more complex to design and maintain than simpler schema models like the star schema or the snowflake schema.
- Example of fact constellation schema



- Advantages

1. Flexible data analysis: The fact constellation schema supports a wide range of data analysis requirements by allowing multiple fact tables and dimension tables to be combined in a single schema. This allows for more flexible data analysis than simpler schema models like the star schema or snowflake schema.

2. Reduced data redundancy: The fact constellation schema can reduce data redundancy by allowing multiple fact tables to share common dimension tables. This can result in improved performance and reduced storage requirements.

3. Scalability: The fact constellation schema is highly scalable, making it suitable for use with large volumes of data. It can support complex data analysis requirements and is designed to handle changes to the data structure over time.

4. Improved accuracy: The fact constellation schema can improve data accuracy by allowing for the integration of multiple data sources. This can result in more accurate analysis and decision-making.

5. Better suited for complex data: The fact constellation schema is better suited for handling complex data sets that have multiple dimensions and measures. This is because it can handle multiple fact tables and dimension tables with a high degree of flexibility.

6. Improved data quality: The fact constellation schema can improve data quality by allowing for the integration of multiple data sources and ensuring that data is stored in a consistent and standardized format.

- Disadvantages

1. Increased complexity: The fact constellation schema can be more complex than simpler schema models like the star schema or snowflake schema. This complexity can make it more difficult to design, implement, and maintain.

2. Higher storage requirements: The fact constellation schema can result in higher storage requirements than simpler schema models due to the increased number of tables and relationships.

3. Slower query performance: The fact constellation schema can result in slower query performance due to the increased number of tables and relationships. This can make it more difficult to retrieve data in real-time for some applications.

4. More difficult to navigate: The fact constellation schema can be more difficult to navigate and understand than simpler schema models. This can make it more challenging for users to locate and analyze data.

5. Increased maintenance requirements: The fact constellation schema can require more maintenance than simpler schema models due to the increased number of tables and relationships. This can make it more time-consuming and costly to maintain the schema over time.

6. Higher development costs: The fact constellation schema can result in higher development costs due to the increased complexity and maintenance requirements. This can make it more challenging for smaller organizations or projects with limited resources.

- Difference between star, snowflake, and fact constellation schema:

Schema Type	**Star Schema**	**Snowflake Schema**	**Fact Constellation Schema**
Data structure	A single fact table at the center is connected to multiple dimension tables	Multiple normalized dimension tables that form a hierarchy connected to a central fact table	Multiple fact tables and multiple dimension tables interconnected in a complex network
Table Design	Denormalized design with redundant data	Normalized design with shared dimension tables	Hybrid design with both denormalized and normalized tables
Query performance	Faster query performance due to denormalized design	Slower query performance due to the need to join multiple normalized tables	Slower query performance due to the complexity of the schema
Storage efficiency	Less storage is required due to the denormalized design	More storage is required due to the normalized design	More storage is required due to the complex design
Complexity	Simple and easy to understand	More complex than star schema but simpler than fact constellation schema	Most complex and difficult to design and understand
Suitable for	Small to medium-sized data warehouses with simple relationships	Medium to large-sized data warehouses with complex relationships	Large and complex data warehouses with multiple fact tables and dimensions
Maintenance	Easy to maintain due to the denormalized design	Moderate maintenance due to the normalized design	More complex to maintain due to the complex design
Data quality	Data quality can be compromised due to redundant data	Good data quality due to normalized tables	Good data quality due to the hybrid design
Example usage	Sales data with product and	HR data with employee hierarchy and	Healthcare data with patient encounters and

	customer dimensions	department dimensions	medical procedure dimensions