Data Analytics

A. Data Warehouse

- A **data warehouse** is a database designed to enable and support business intelligence (BI) activities, especially analytics.
  - intended to perform queries and analysis
  - optimized for data retrieval, not for transaction processing
  - centralizes and consolidates large amounts of data from multiple sources
  - allows organizations to derive valuable business insights from their data to improve decision-making
  - can be considered an organization's "single source of truth"

Characteristics of Data Warehouses (DW)

- **Subject-Oriented –** The DW can analyze data about a particular subject or functional area.
  - Subjects can be products, customers, departments, regions, etc.
  - The functional area can be sales, marketing, finance, distribution, etc.
  - Focuses on the data rather than on the processes that modify the data
- **Integrated –** The DW creates consistency among different data types from different sources.
  Example: A student's level in the database might be defined as "freshman", "sophomore", "junior", or "senior" in the accounting department, and "FR", "SO", "JR", "SR" in the computer information systems department.
  - DW must conform to a common format that is acceptable throughout the organization.
- **Time-variant –** Data in DW represents the flow of data through time. It can be organized weekly, monthly, or annually, etc.
  Example: When data for previous weekly sales is uploaded to the data warehouse, the weekly, monthly, yearly, and other time-dependent subjects such as products, customers, stores, etc. are also updated.
- **Non-volatile –** Once data is in a data warehouse, it is stable and does not change.

Components of Data Warehouse (DW)

- **Data Warehouse Database –** This is a databank that stocks all enterprise data and makes it manageable for reporting.
  - always implemented on the relational database management system (RDBMS) technology like SQL
- **Extraction, Transformation, and Loading Tools (ETL) –** These tools are used for performing all the conversions, summarizations, and all the changes needed to transform data into a unified format in the data warehouse. These include:
  - In case of missing data, populating them with defaults
  - Calculating summaries and derived data
  - Eliminating unwanted data in operational databases from loading into the data warehouse
  - Converting to common data names and definitions
- **Metadata –** is data about data that describes the data warehouse. It provides the source, transformation, integration, storage, usage, relationships, and history of each data element.
  Example: A line in a sales department database contains:
  SNY-JP-0010-15000
  This is meaningless data until we consult the meta that tell us the following:
    - Brand Model: Sony
    - Country Manufactured: Japan
    - Product ID: 0010
    - Price: ₱15,000

  Metadata can be classified into two (2) categories:
  1. **Technical Metadata –** contains information about the warehouse, which is used by data warehouse designers and administrators.
  2. **Business Metadata –** contains details that give end-users an easy way to understand the information stored in the data warehouse.
- **Data Warehouse Access Tools –** Corporate users generally cannot work with databases directly. They use the assistance of the following tools:
  - **Query and reporting tool –** help users produce corporate reports for analysis that can be in the form of spreadsheets, calculations, or interactive visuals.
  - **Application development tools –** In such cases, custom reports are developed using application development tools

when built-in graphical and analytical tools do not satisfy the analytical needs of an organization.
  - **Data mining –** a process of discovering meaningful new correlations, patterns, and trends by mining a large amount of data. Data mining tools are used to make this process automatic.
  - **OLAP tools –** allow users to analyze the data using elaborate and complex multi-dimensional views.
- **Data Marts –** a small, single-subject data warehouse subset that provides decision support for the particular user group.

4. Helps to reduce total turnaround time for analysis and reporting
5. Restructuring and integration make it easier for the user to use for reporting and analysis.
6. Stores a large amount of historical data. This helps users to analyze different time periods and trends to make future predictions.

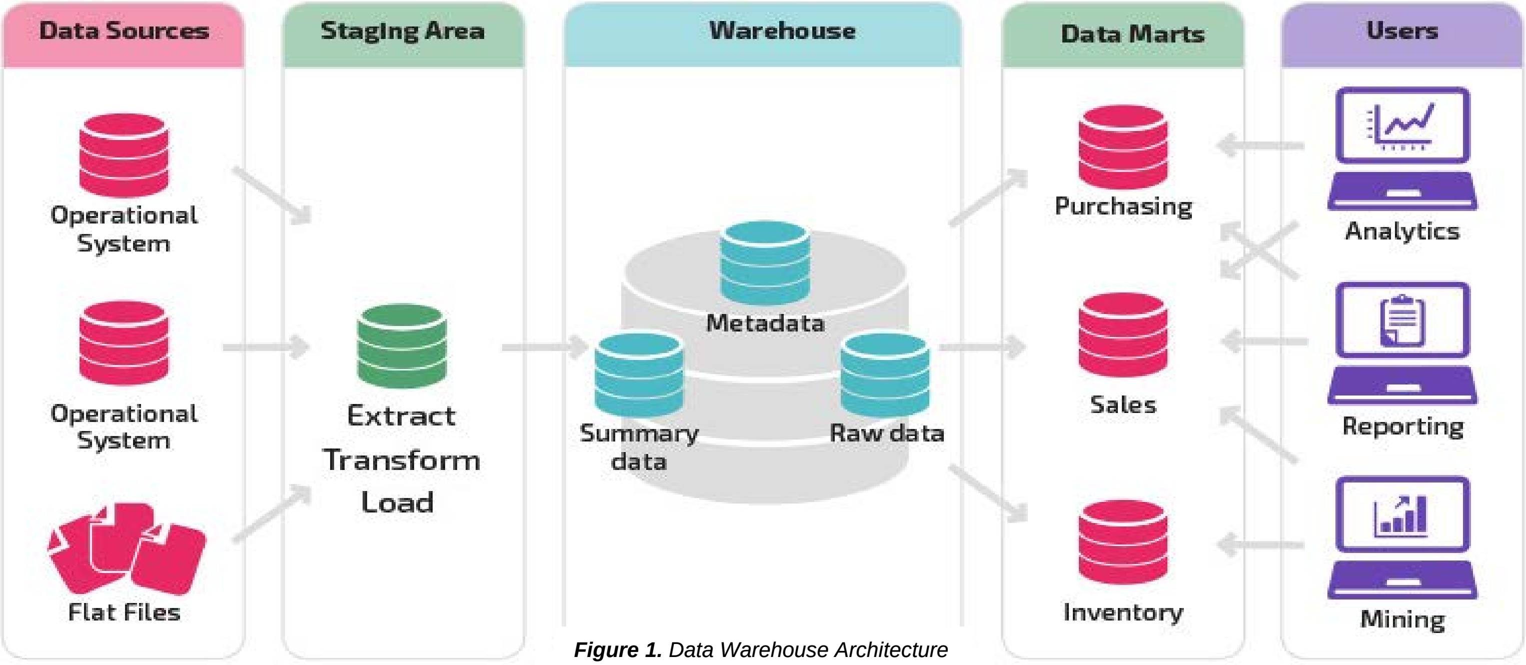


Figure 1. *Data Warehouse Architecture*

Benefits of a Data Warehouse

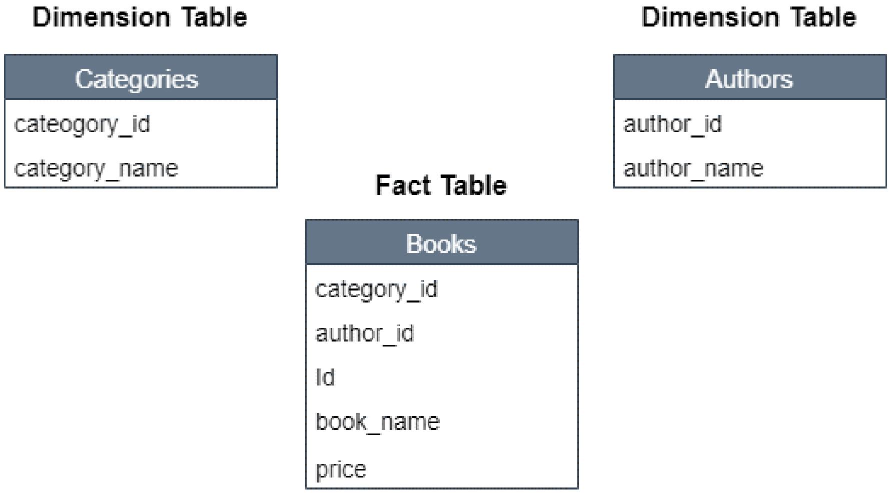
1. Allows business users to quickly access critical data from some sources all in one place. Therefore, it saves the user's time of retrieving data from multiple sources
2. Provides consistent information on various cross-functional activities. It is also supporting **ad-hoc** reporting and query.
3. Helps to integrate many sources of data to reduce stress on the production system

Star Schema

- A **star schema** is a data-modeling technique used to map multi-dimensional decision support data into a relational database.
- It has two (2) common components:
  - **Facts table –** are data that will be included in reports and used as the basis of business decisions. It contains measurement or facts to the data and foreign key to dimension table.

- **Dimension table –** are attributes that qualify and provide more information about facts. It contains dimensions of a fact and they are joined to fact table via foreign key.

Example:



B. Online Analytical Processing (OLAP)

- **Online Analytical Processing (OLAP)**
  - a software tool that is used for data analysis and reporting purposes for business decisions
  - used by business analysts, managers, and executives.
    Example: In Netflix, OLAP was used for movie recommendations based on watch history.
- **Online Transaction Processing (OLTP)**
  - an operational system that manages the day-to-day transactions of an organization
  - used by the Database Administrator (DBA) and Database Professionals
    Example: In ATM centers, OLTP is used for money withdrawals, transfers, deposits, and inquiries.

OLAP vs. OLTP

OLAP	OLTP
Provides historical data for reporting and planning	Manages day to day operations
Uses complex queries for retrieving a large amount of data	Uses standard queries for data such as inserting, deleting, and updating

Characteristics of OLAP

- **Multi-dimensional data analysis techniques –** Data is processed and viewed as part of a multi-dimensional structure.
- **Advanced Database support –** To deliver efficient decision support, OLAP tools must have the following:
  - Access to many kinds of DBMSs, flat files, and internal and external data sources
  - Rapid and consistent query response times
  - Support for very large databases because the data warehouse could easily and quickly grow to multiple terabytes in size
- **Easy-to-use end-user interfaces –** permit the user to navigate the data in a way that simplifies and accelerates decision making or data analysis with easy-to-use graphical interfaces

Types of OLAP

- **Relational OLAP (ROLAP)**
  - Works directly with relational databases
  - Fact and dimension tables are stored as relations.
- **Multi-dimensional OLAP (MOLAP)**
  - extends OLAP functionality to multi-dimensional database management systems (MDBMS)
  - best suited to manage, store, and analyze multi-dimensional data

ROLAP vs. MOLAP

Characteristic	ROLAP	MOLAP
Schema	Uses Star Schema	Uses data cubes
Speed	Good with small data sets	Faster for large data sets
Access	Unlimited dimensions	Limited to predefined dimensions

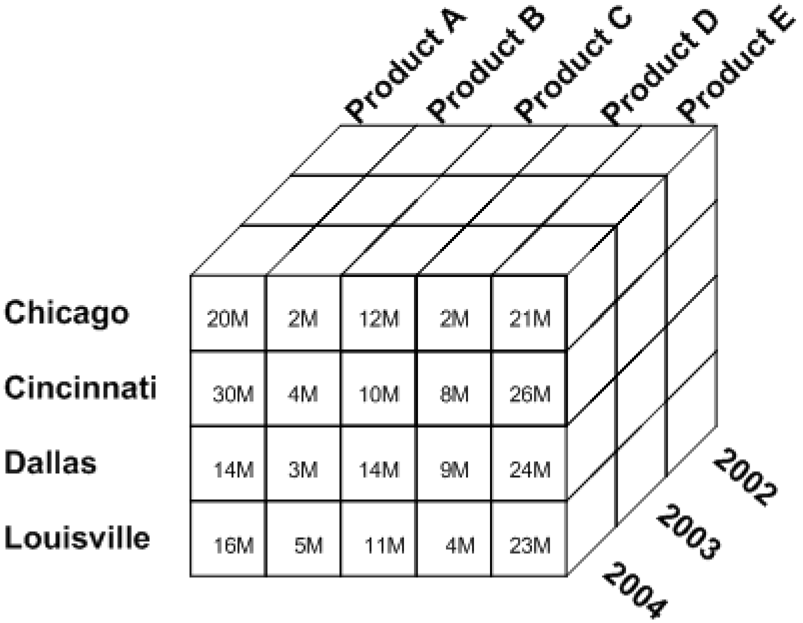


Figure 2. *Multi-dimensional data/data cubes*

OLAP operations

SQL has been enhanced with analytic functions that support OLAP-type processing. This includes:

- **ROLLUP operator –** an extension of the GROUP BY clause that is used to create subtotals and grand totals for a set of columns
- **CUBE operator –** Like ROLLUP, this generates subtotals for all the combinations of grouping column s specified in the GROUP BY clause.
- **PIVOT operator –** allows you to write a cross-tabulation, which means you can aggregate your results and rotate rows into columns

Example:

Assume that we have a table named Enrolled_Students (see Table 1) having columns *Campus*, *NumberOfStudents*, and *Program*.

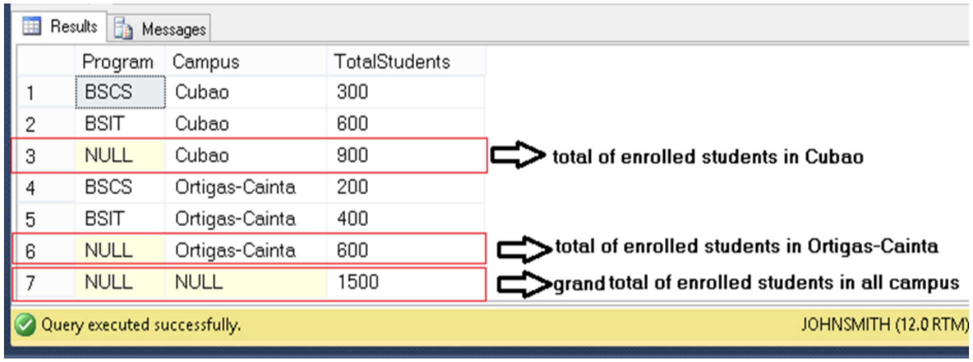
Campus	NumberOfStudents	Program
Ortigas-Cainta	400	BSIT
Ortigas-Cainta	200	BSCS
Cubao	600	BSIT
Cubao	300	BSCS

Table 1. *Enrolled_Students*

Using the **ROLLUP** operator, we will display the total number of students enrolled in specific campuses and the grand total of students enrolled in all campuses.

```
SELECT Program, Campus,
           SUM(NumberOfStudents) AS 'TotalStudents'
FROM Enrolled_Students
GROUP BY ROLLUP (Campus, Program)
```

Output:



```
   Program  Campus          TotalStudents
1  BSCS     Cubao           300
2  BSIT     Cubao           600
3  NULL     Cubao           900    ⇨ total of enrolled students in Cubao
4  BSCS     Ortigas-Cainta  200
5  BSIT     Ortigas-Cainta  400
6  NULL     Ortigas-Cainta  600    ⇨ total of enrolled students in Ortigas-Cainta
7  NULL     NULL            1500   ⇨ grand total of enrolled students in all campus
Query executed successfully.                                   JOHNSMITH (12.0 RTM)
```

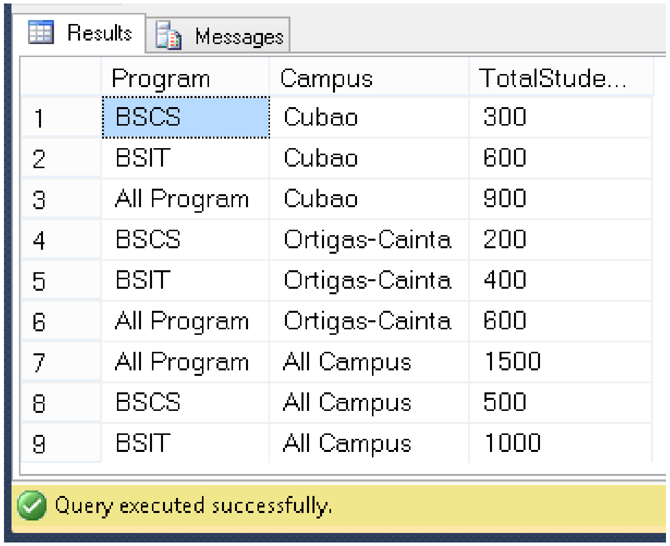
Explanation:

- ROLLUP operator creates an additional row that represents subtotals for each campus. In the last row, it represents the grand total for all values in the NumberOfStudents column.
  *(**Note:** To make the output more readable, you can use the COALESCE() function to substitute the appropriate value representing subtotal and grand total to the NULL values.)*

Using the **CUBE** operator, we will display all possible combinations of columns in the **Enrolled_Students** table (see Table 1).

```
SELECT  COALESCE(Program, 'All Program') AS 'Program',
		COALESCE(Campus, 'All Campus') AS 'Campus',
		SUM(NumberOfStudents) AS 'TotalStudents'
FROM Enrolled_Students
GROUP BY CUBE (Program, Campus)
```

Output:



	Program	Campus	TotalStude...
1	BSCS	Cubao	300
2	BSIT	Cubao	600
3	All Program	Cubao	900
4	BSCS	Ortigas-Cainta	200
5	BSIT	Ortigas-Cainta	400
6	All Program	Ortigas-Cainta	600
7	All Program	All Campus	1500
8	BSCS	All Campus	500
9	BSIT	All Campus	1000

Query executed successfully.

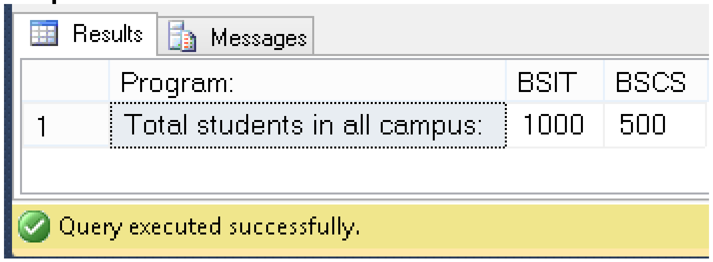
Explanation:

- We use the COALESCE function to specify the returning text of NULL values in a specific column.
- It has similar output to ROLLUP, but it returns two (2) additional rows below the grand total. This is because the ROLLUP operator generates aggregated results for the selected columns like Campus in a hierarchical way, while the CUBE operator generates an aggregated result that contains all the possible combinations for the selected columns.

Using the **PIVOT** operator, we will turn the unique values/rows in the *Program* column into multiple columns.

```
SELECT 'Total students in all campus:' AS 'Program:', [BSIT], [BSCS]
FROM
(
		SELECT NumberOfStudents, Program FROM
Enrolled_Students
) AS SourceTable
PIVOT
(
		SUM(NumberOfStudents)
		FOR Program IN ([BSIT], [BSCS])
) AS PivotTable
```

Output:



	Program:	BSIT	BSCS
1	Total students in all campus:	1000	500

Query executed successfully.

Explanation:

- The first query specifies the column for cross-tabulation results. We want to display the first column as the identifier of the remaining column (second and third columns).
- As for the source table, we specify the returning data that will be used for the pivot statement.
- In the pivot statement, we used the SUM() function to get the total number of students that are enrolled.
- We need to specify what rows/values to include from the *Program* column as it will become our column headings in our pivot table.

C. Data Mining

- **Data mining** refers to analyzing massive amounts of data in a data warehouse or other sources to uncover hidden trends, patterns, and relationships. This explains the past and predicting the future for analysis.

Data Mining Implementation Process

1. **Business Understanding**: In this step, the goals of the businesses are set, and the important factors that will help in achieving the goal are discovered.

2. **Data Understanding**: This step will collect the entire data and populate the data in the tool (if using any tool).
3. **Data Preparation**: This step involves selecting the appropriate data, cleaning, constructing attributes from data, integrating data from multiple databases.
4. **Modeling**: Selection of the data mining technique such as decision-tree, generate test design for evaluating the selected model, building models from the dataset, and assessing the built model with experts to discuss the result is done in this step.
5. **Evaluation**: This step will determine the degree to which the resulting model meets the business requirements. The model is reviewed for any mistakes or steps that should be repeated.
6. **Deployment**: In this step, a deployment plan is made. The strategy to monitor and maintain the data mining model results to check for its usefulness is formed. Final reports are also made, and a review of the whole process is done to check any mistake and see if any step is repeated.

Data Mining Techniques

1. **Classification:** used to retrieve important and relevant information about data and metadata.
2. **Clustering**: used to identify data that are like each other. This process helps to understand the differences and similarities between the data.
3. **Regression:** used to identify and analyze the relationship between variables.
4. **Association Rules**: used to help find the association between two or more Items. It discovers a hidden pattern in the data set.
5. **Outer detection:** used to observe data items in the dataset that do not match an expected pattern or expected behavior.
6. **Sequential Patterns:** used to discover or identify similar patterns or trends in transaction data for a certain period.
7. **Prediction:** used to combine other data mining techniques like trends, sequential patterns, clustering, classification, etc. It analyzes past events or instances in the right sequence for predicting a future event.

Benefits of Data Mining

- Helps with the decision-making process
- Helps companies to get knowledge-based information
- Facilitates automated prediction of trends and behaviors as well as the automated discovery of hidden patterns
- The speedy process which makes it easy for the users to analyze a huge amount of data in less time

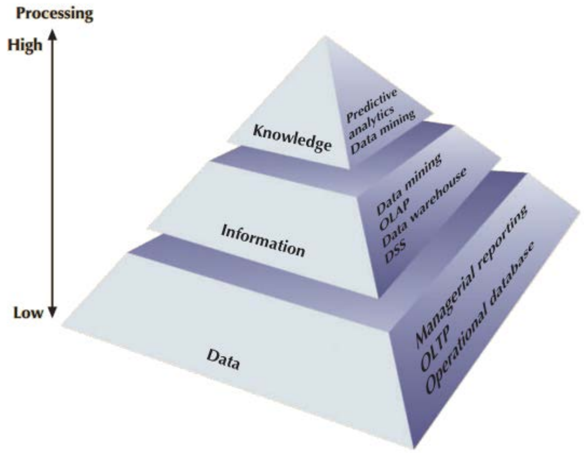


Figure 3. *Extracting knowledge from data*

REFERENCES

Coronel, C. and Morris, S. (2018). Database systems design, implementation, & management (13th ed.). Cengage Learning.

Elmasri, R. & Navathe, S. (2016). Fundamentals of Database Systems (7th ed.). Pearson Higher Education.

Kroenke, D. & Auer, D. Database Processing: Fundamentals, Design, and Implementation (12th ed.). Pearson Higher Education.

Silberschatz A., Korth H.F., & Sudarshan, S. (2019). Database system concepts (7th ed.). McGraw-Hill Education.