

# DTS207TC Database Development and Design

## Lecture 7

### Chap 11 Data Analysis

Di Zhang, Autumn 2025

*Page titles with \* will not be assessed*

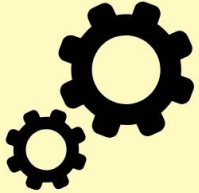
- Data Warehouse
- OLAP
- Data Mining

# What is a Data Silo?

An isolated repository containing data that isn't connected with any other system or application, such as folder structures or independent software applications.



# Why do data silos exist?



## **Legacy systems & technical debt**

Organizations continue using databases and systems put in place years ago



## **Organizational silos & politics**

Organizational politics, leadership problems, and lack of data governance create barriers to change



## **Inconsistent data standards**

Inconsistent schemas, formats, definitions, and rules around data create barriers to integration



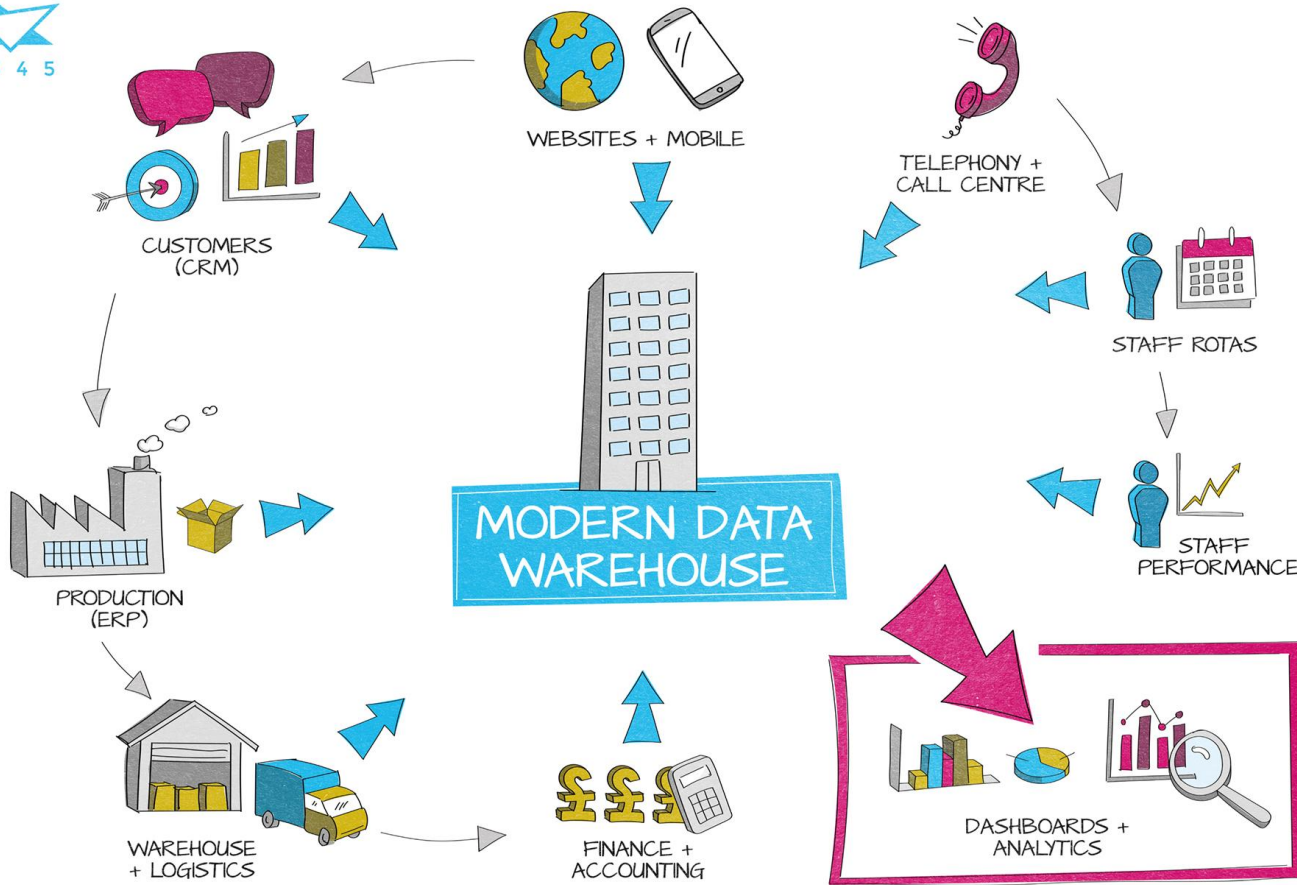
## **Lack of data strategy**

Lack of strategic data governance and planning create isolated systems

# Analogy

- We are trying to force the chaotic, concrete, and "opinion-filled" real world into an orderly, abstract, and "truth-seeking" model.





- Core Concept: Separate data from its operational environment and build a subject-oriented, integrated, non-volatile, and time-varying data collection to support management decision-making.

# Enterprise Data Warehouse Benefits

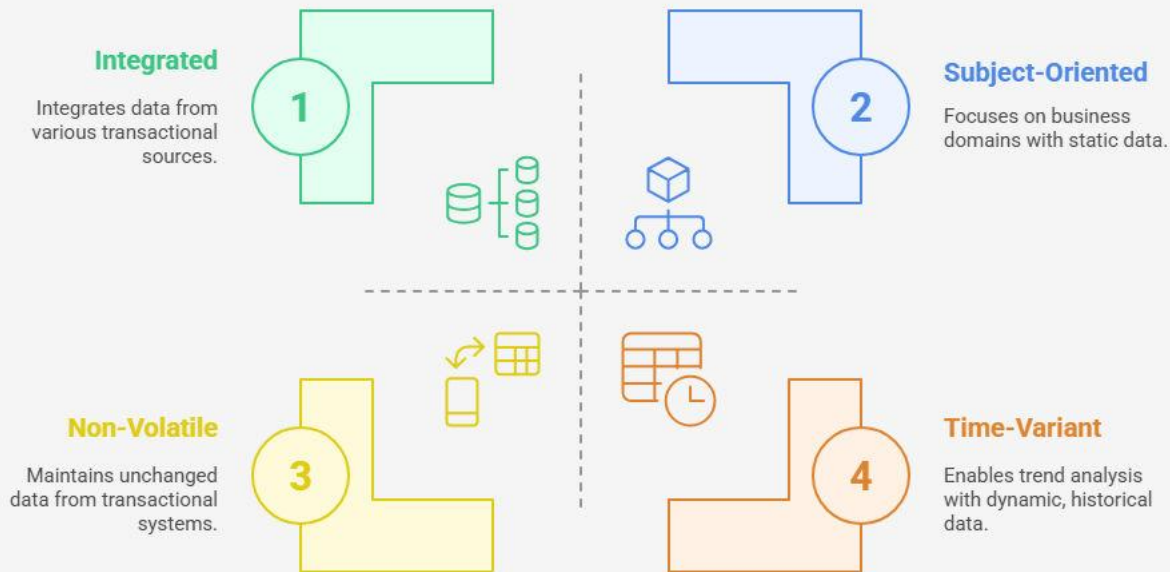


# Data Warehouse vs. Operational Database (OLAP vs. OLTP)

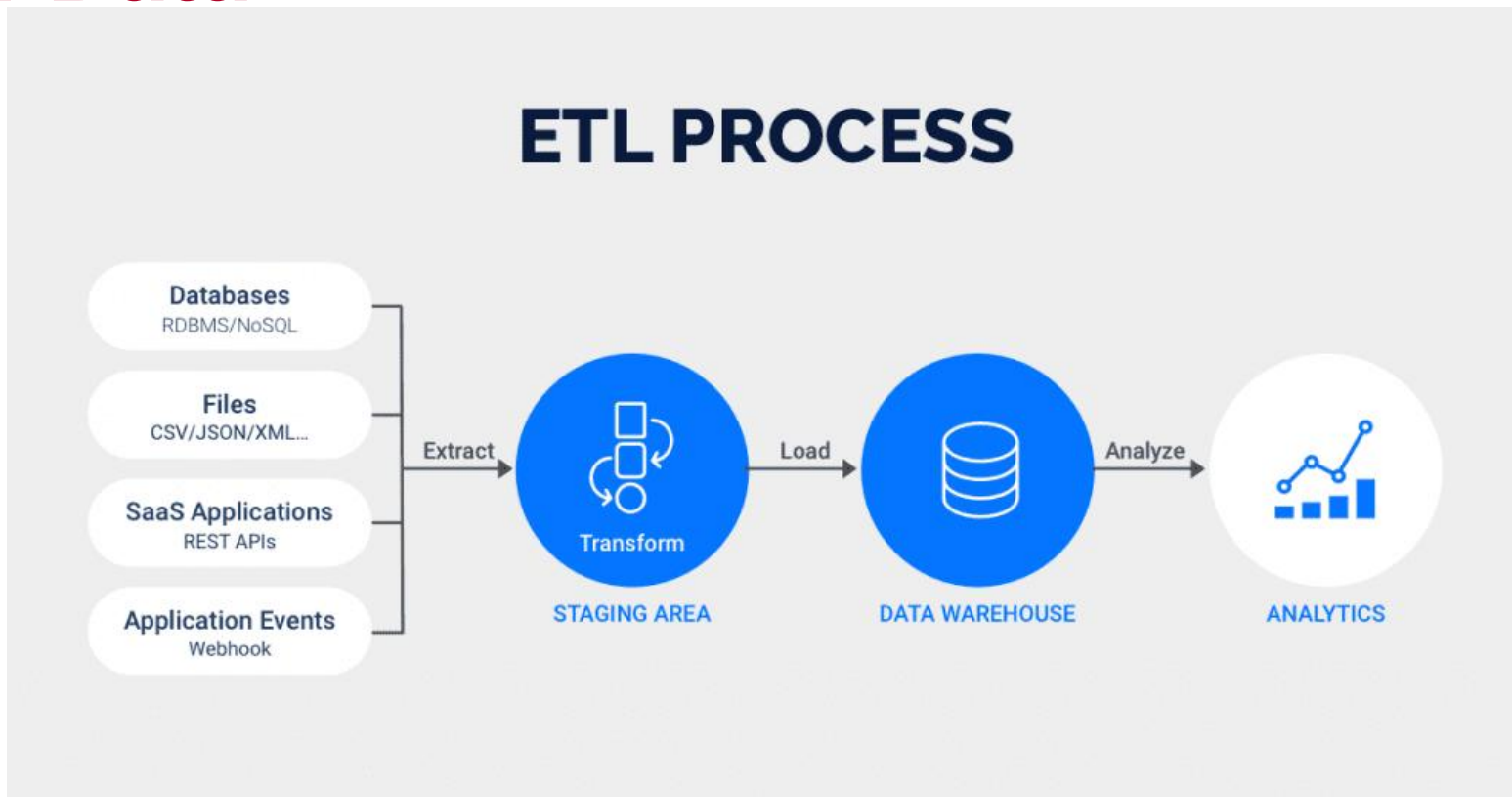
Characteristic	OLTP (Operational DB)	OLAP (Data Warehouse)
Purpose	Daily business operations	Support decision analysis
Users	Clerks, frontline staff	Analysts, managers
Data	Current, detailed	Historical, summarized
Operations	Frequent inserts, updates, deletes	Read-intensive (queries)
Model	Normalized (3NF)	Denormalized (Star/Snowflake)
Performance	Fast transaction processing	Fast complex query response

# The Four Key Characteristics of a Data Warehouse

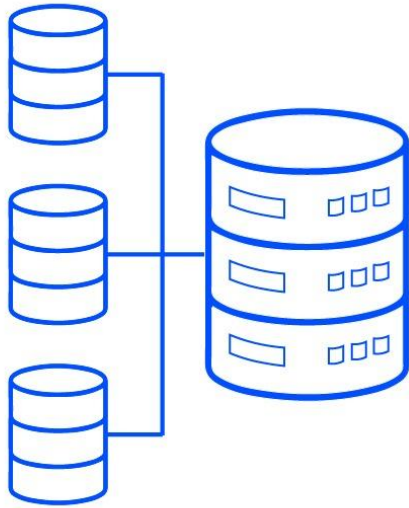
## Characteristics of a Data Warehouse



# ETL: The "Mover" and "Beautician" of Data

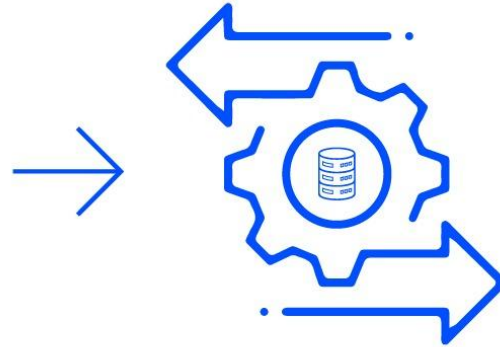


# The ETL Process Explained



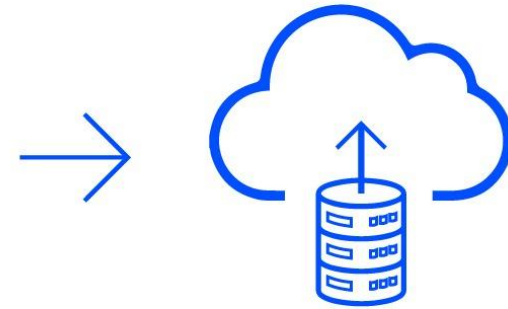
## Extract

Retrieves and verifies data  
from various sources



## Transform

Processes and organizes  
extracted data so  
it is usable



## Load

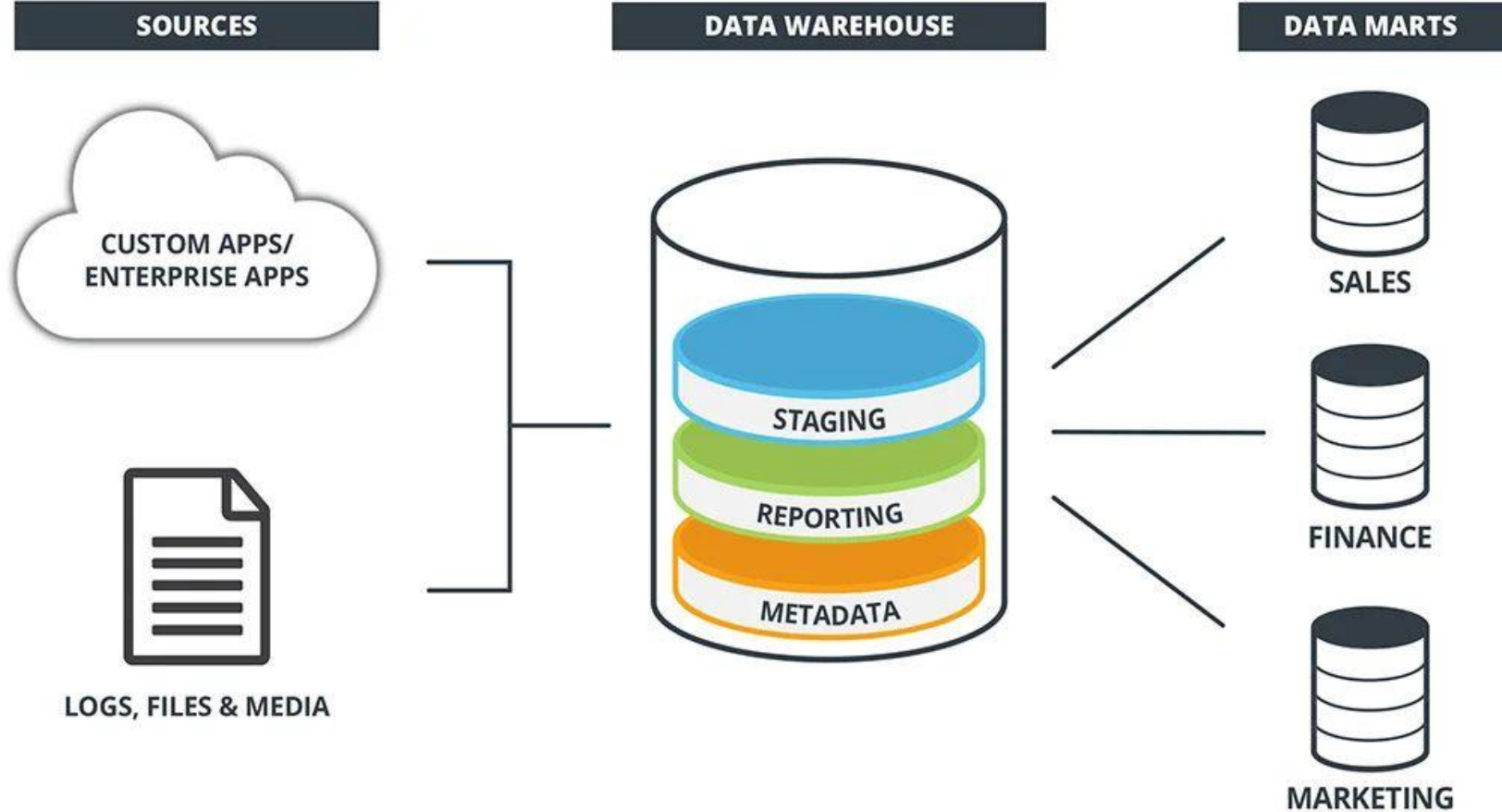
Moves transformed data  
to a data repository

# ETL Example

---



# The Layered Architecture of a Data Warehouse



# Staging Area

- Function: Temporarily stores raw data extracted from source systems; where the 'T' in ETL happens.
- Often physically separate from the core warehouse. Functionalities:
  - Temporarily store data extracted from different data sources before transforming it and loading it into the destination system;
  - Clean and normalise the data to eliminate duplicates, inconsistencies, missing or erroneous values, etc;
  - Apply validation and quality rules to ensure that the data is complete, accurate and consistent;
  - Apply transformations to change the format, structure and values of data to adapt them to the requirements of the destination system;
  - Enable consistency and conformity checks on data before it is finally loaded into the destination system.

- Function: Stores integrated, cleaned, transformed enterprise-wide historical data.
- The model is relatively stable; it is the single source of truth.

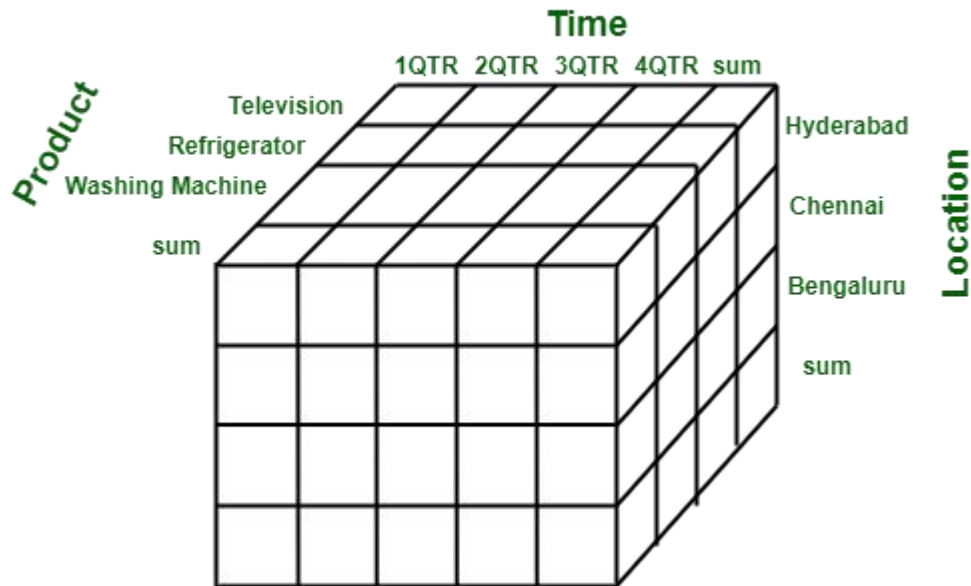
- Definition: A subset of the data warehouse, tailored for a specific department or business line (e.g., Sales Mart, Finance Mart).
- Two Construction Approaches:
  - Top-Down: Derived from the enterprise DW, ensuring consistency.
  - Bottom-Up: Departmental data marts are built first, then integrated into an DW.

- Why Do We Need a Data Model?
  - Improve query performance.
  - Enhance data understandability.
  - Provide a foundation for OLAP.

# The Multi-Dimensional Data Model

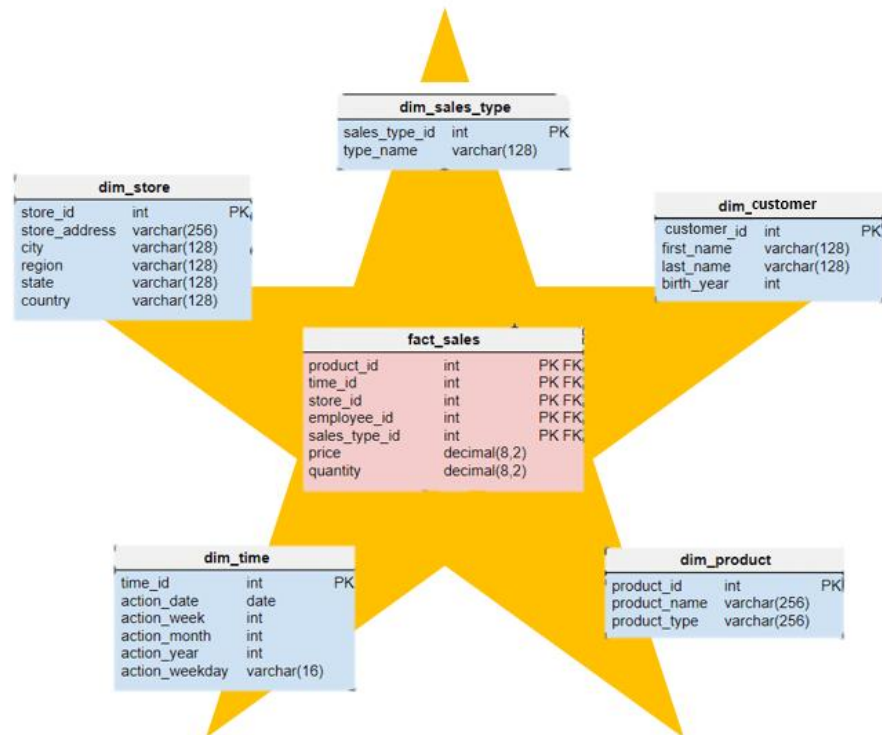
## Core Idea

- Facts: The measures to be analyzed (e.g., Sales Amount, Quantity Sold).
- Dimensions: The perspectives for analyzing facts (e.g., Time, Location, Product).
- Analogy: Facts are "verbs," dimensions are "nouns."



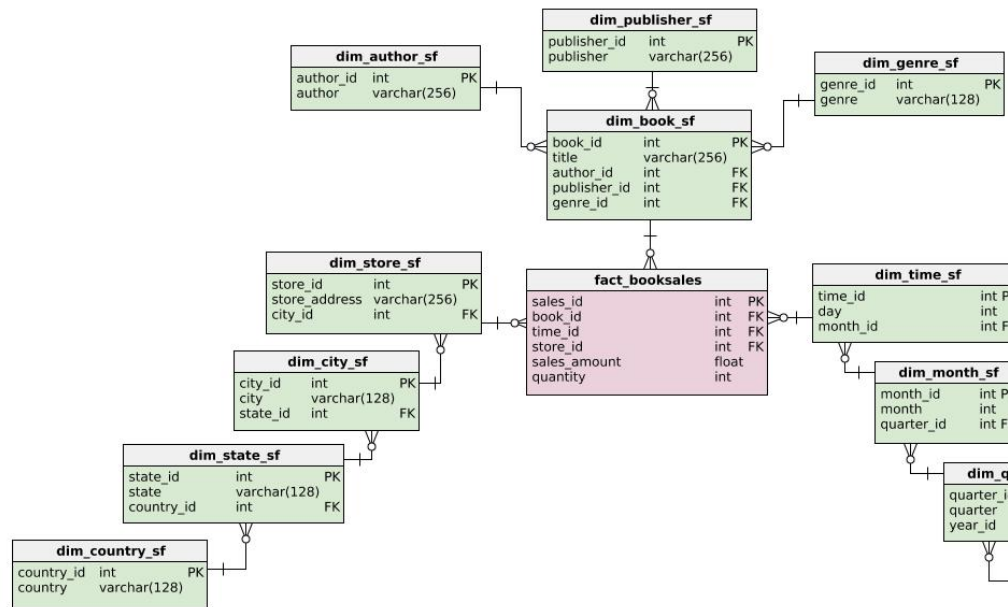
# Star Schema

- Diagram: One central fact table connected to multiple dimension tables.
- Characteristics:
  - Simple structure, easy to understand.
  - High query performance (denormalized).
  - The most common modeling approach.



# Snowflake Schema

- Diagram: The dimension tables in a star schema are normalized forming a hierarchy.
- Characteristics:
  - Saves storage space.
  - Structure is more normalized.
  - Queries can be slower due to more joins.



- Composition: Dimension Keys (Foreign Keys) + Measures (Facts).
- Types:
  - Transaction Fact Table: Records a specific event (e.g., a single sale).
  - Periodic Snapshot Fact Table: Records a regular status (e.g., end-of-month account balance).
  - Accumulating Snapshot Fact Table: Tracks a workflow process (e.g., order status flow).

## Transactional Vs Snapshot tables

**Transactional fact table: Sales**

Date	Product	Customer	Store	Amount
2020-01-18	XYZ	XYZ	XYZ	\$100
2020-01-20	XYZ	XYZ	XYZ	\$200
2020-01-25	XYZ	XYZ	XYZ	\$150

**Snapshot fact table: Stock On Hand**

Date	Product	Warehouse	Qty	Value
2020-01-18	XYZ	XYZ	1	\$50
2020-01-19	XYZ	XYZ	1	\$50
2020-01-20	XYZ	XYZ	2	\$100
2020-01-21	XYZ	XYZ	2	\$50
2020-01-22	XYZ	XYZ	2	\$50
2020-01-23	XYZ	XYZ	3	\$150
2020-01-24	XYZ	XYZ	3	\$150
2020-01-25	XYZ	XYZ	3	\$150

# Dimension Tables

- Composition: Surrogate Key (Primary Key) + Dimension Attributes (textual descriptive fields).
- Characteristics: Contain rich descriptive information for filtering and grouping.

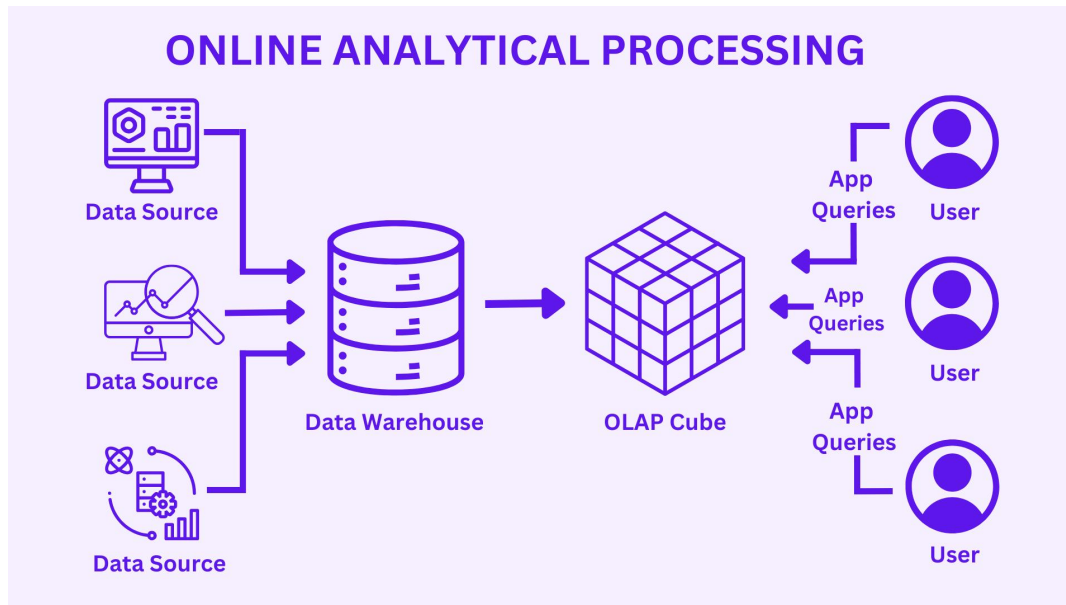
100 %

Results Messages

	DateKey	DateInt	YearKey	QuarterOfYear	MPH_MonthOfYear	MonthOfYear	DayOfMonth	MonthName	MonthInCalendar	QuarterInCalendar	DayC
1	2022-01-02 00:00:00.000	20210467	2022	1	1	1	2	January	2022-01-02 00:00:00.000	Q1 2022	Sund
2	2022-01-01 00:00:00.000	20210466	2022	1	1	1	1	January	2022-01-01 00:00:00.000	Q1 2022	Satur
3	2021-12-31 00:00:00.000	20210465	2021	4	12	12	31	December	2021-12-31 00:00:00.000	Q4 2021	Frida
4	2021-12-30 00:00:00.000	20210464	2021	4	12	12	30	December	2021-12-30 00:00:00.000	Q4 2021	Thur
5	2021-12-29 00:00:00.000	20210463	2021	4	12	12	29	December	2021-12-29 00:00:00.000	Q4 2021	Wed
6	2021-12-28 00:00:00.000	20210462	2021	4	12	12	28	December	2021-12-28 00:00:00.000	Q4 2021	Tues
7	2021-12-27 00:00:00.000	20210461	2021	4	12	12	27	December	2021-12-27 00:00:00.000	Q4 2021	Monc
8	2021-12-26 00:00:00.000	20210460	2021	4	12	12	26	December	2021-12-26 00:00:00.000	Q4 2021	Sund
9	2021-12-25 00:00:00.000	20210459	2021	4	12	12	25	December	2021-12-25 00:00:00.000	Q4 2021	Satur
10	2021-12-24 00:00:00.000	20210458	2021	4	12	12	24	December	2021-12-24 00:00:00.000	Q4 2021	Frida
11	2021-12-23 00:00:00.000	20210457	2021	4	12	12	23	December	2021-12-23 00:00:00.000	Q4 2021	Thur

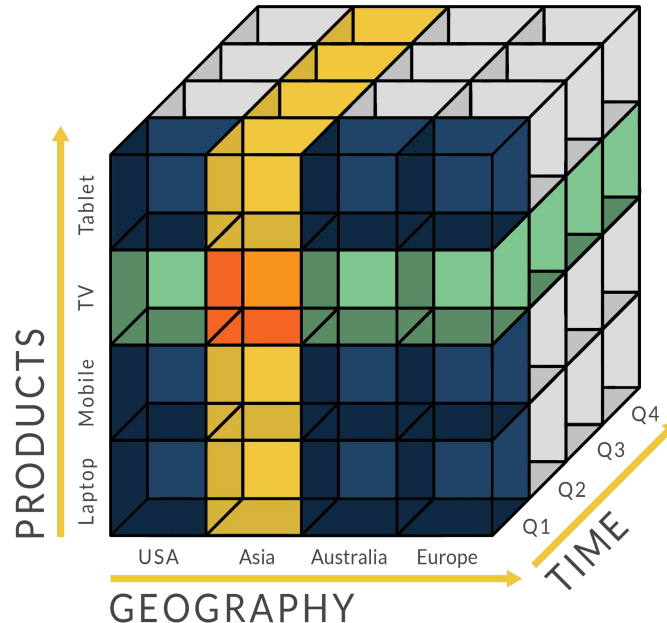
# What is OLAP?

- Definition: Online Analytical Processing. A technology that enables users to quickly, consistently, and interactively access data from multiple perspectives.
- Goal: Support complex analysis and decision-making.



# Core Concept: The Cube

- Diagram: A 3D data cube (Product, Time, Region -> Sales Amount).
- Explanation: The physical or logical implementation of the multi-dimensional model; the foundation for OLAP operations.

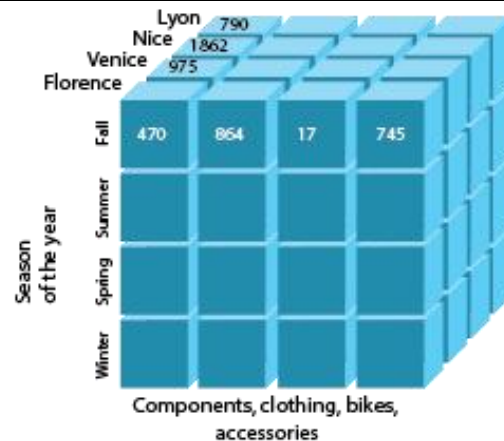


# Core OLAP Operation: Slice

- Definition: Selecting a value on one dimension of the cube.
- Example: Selecting data for "Time = January 2023".



1

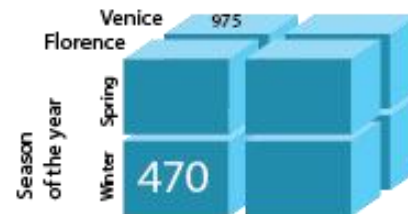


Slice  
for time  
="winter"

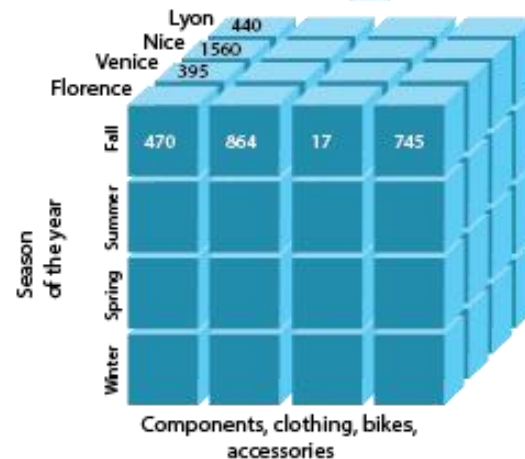


# Core OLAP Operation: Dice

- Definition: Selecting a range of values on one dimension.
- Example: Selecting data for "Time in Q1 2023".
- <https://www.budgetbytes.com/how-to-slice-dice-and-mince/>

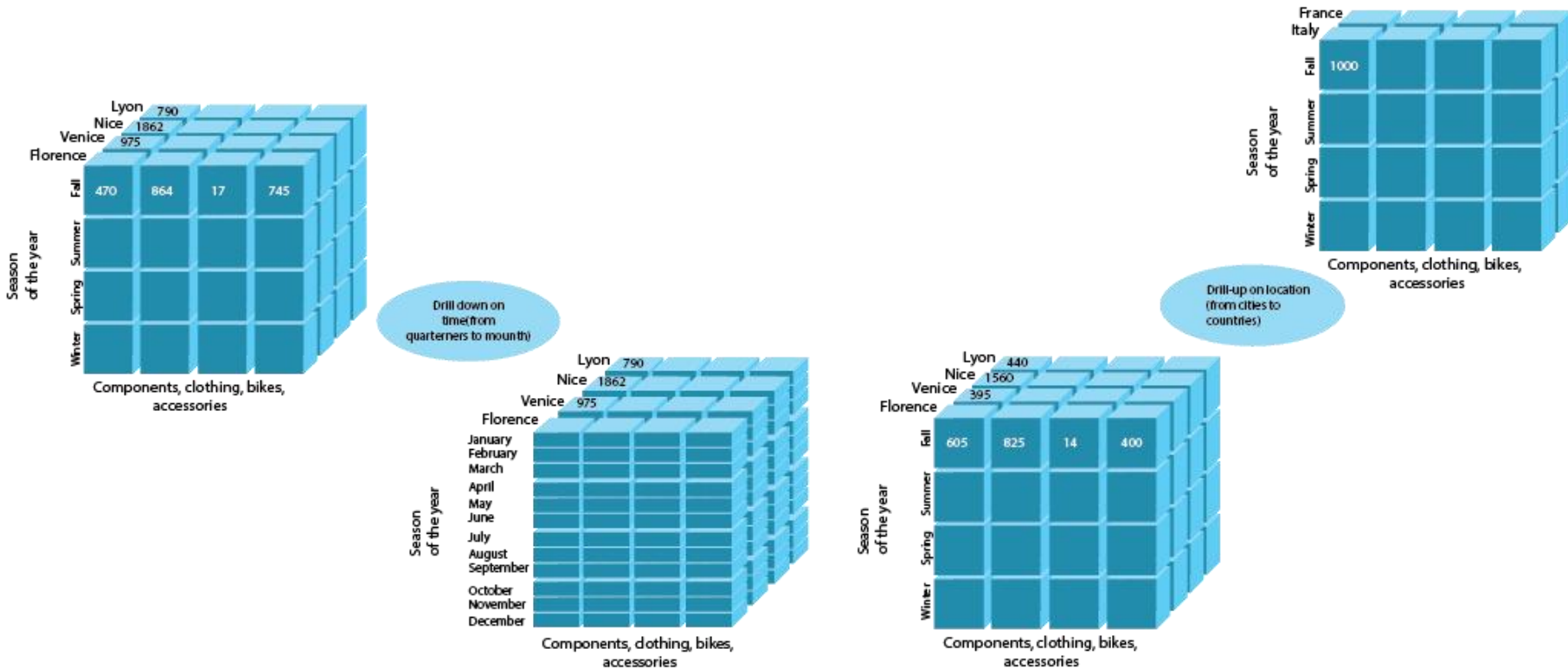


Dice for (location = "Venice" or "Florence")  
and (season = "Winter" or "Spring") and  
(item = "components" or "clothing")



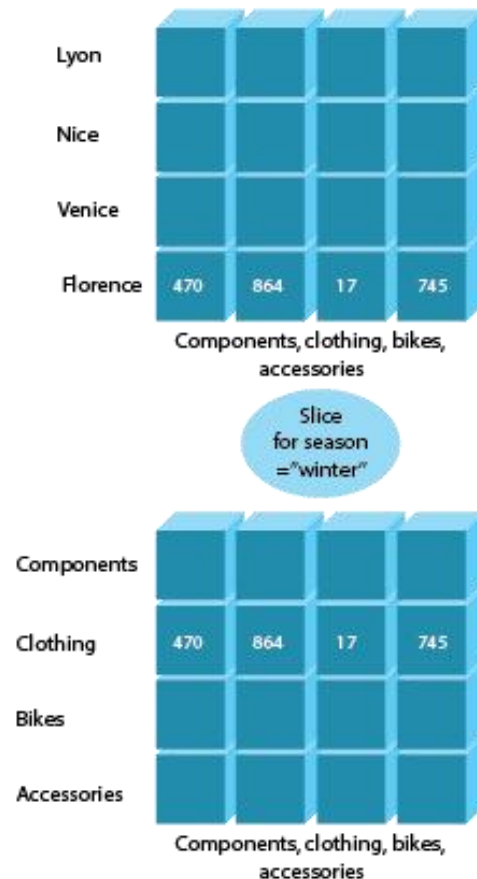
# Core OLAP Operation: Drill

- Drill-Down: Moving from summarized data to more detailed data (e.g., from "Year" to "Quarter" to "Month").
- Roll-Up: The inverse of drill-down, aggregating detail data to a higher level.



# Core OLAP Operation: Pivot (Rotate)

- Definition: Swapping the row and column dimensions to provide a different view of the data.
- Example: Swapping "Product" on rows with "Region" on columns.



# Cross-Tab: 2D Representation of Multi-Dimensional Data

- Diagram: A standard pivot table / crosstab.
  - Rows: Product Category
  - Columns: Year
  - Intersection: Sales Amount
- Explanation: This is the most intuitive representation of OLAP operation results.

Cross tabulation		What is Your Favorite Baseball Team?			
		Toronto Blue Jays	Boston Red Socks	New York Yankees	Row Totals
In What City Do You Reside?	Boston, MA	11	33	7	51
	Row Percent	21.57%	64.71%	13.73%	34.93%
	Montreal, Canada	23	14	9	46
	Row Percent	50.00%	30.43%	19.57%	31.51%
	Montpellier, VT	22	13	14	49
	Row Percent	44.90%	26.53%	28.57%	33.56%
	Column totals	56	60	30	146
	Column Percent	38.36%	41.10%	20.55%	100.00%

# From Cube to Cross-Tab

- Visual Flow: 3D Data Cube -> Via Slice/Pivot Operations -> 2D Cross-Tab.
- Emphasis: The cross-tab is the output of OLAP analysis.

## Pivot

df

```
df.pivot(index='foo',  
          columns='bar',  
          values='baz')
```

	foo	bar	baz	zoo
0	one	A	1	x
1	one	B	2	y
2	one	C	3	z
3	two	A	4	q
4	two	B	5	w
5	two	C	6	t

Stacked

Record

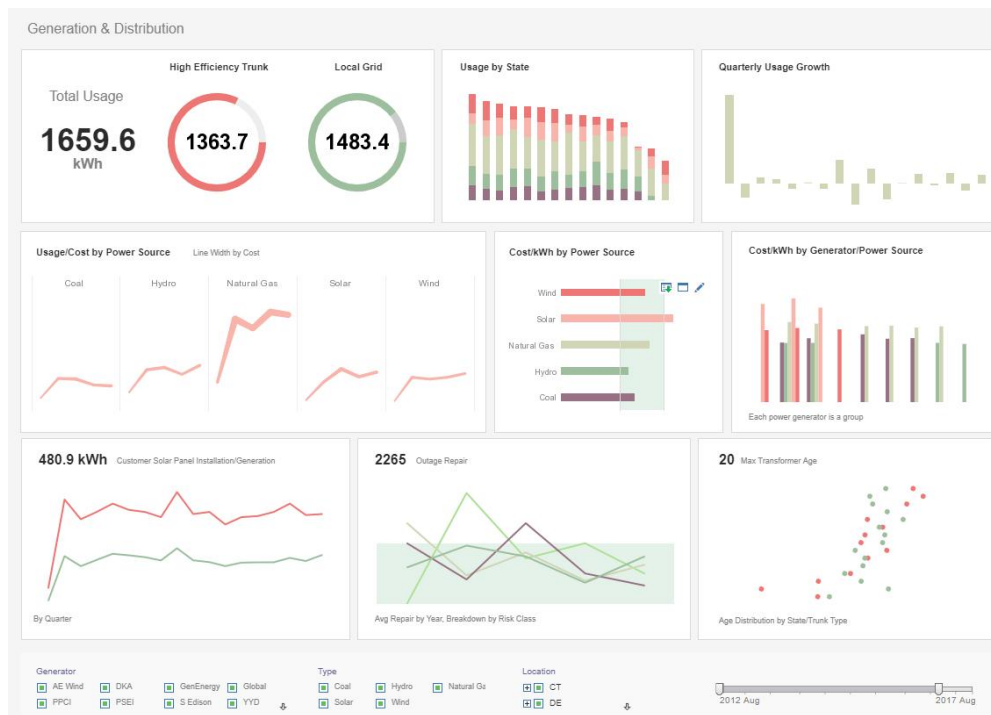
bar	A	B	C
foo			
one	1	2	3
two	4	5	6

# Reporting Systems: Solidified Insights



Xi'an Jiaotong-Liverpool University  
西交利物浦大學

- Definition: Generate fixed-format reports (daily, weekly, monthly) regularly based on the data warehouse.
- Characteristics: Standardized, automated.



# Common Chart Types

---

- Trend Analysis: Line Chart
  - Comparison Analysis: Column Chart, Bar Chart
  - Composition Analysis: Pie Chart, Stacked Column Chart
- Distribution Analysis: Scatter Plot, Histogram
  - Relationship Analysis: Heat Map
  - Geospatial Analysis: Map



# Overview: A three-coordinate system (you can define your own)

Algorithm	X: Model Complexity	Y: Regularization Strength	Z: Ensemble Method	Positioning Description
Linear Regression	Low (near origin)	None (near origin)	None (near origin)	The simplest baseline model, located in the (low, low, low) corner of the coordinate system.
Ridge Regression	Low	High (+Y)	None	Builds upon linear regression by adding L2 regularization, moving it in the +Y direction.
Lasso Regression	Low	High (+Y)	None	Similar to Ridge, but uses L1 regularization (which also performs feature selection). Also located in the +Y direction.
Decision Tree	Medium-High (+X)	Low (can be increased via pruning)	None	A single tree is a flexible model with higher complexity than linear models, placing it in the +X direction.
Support Vector Regression (SVR)	Medium (depends on kernel)	Medium (controlled by parameter C)	None	Its core idea of maximizing the margin is a form of regularization, placing it in the medium region of both X and Y.
K-Nearest Neighbors (KNN)	Medium (controlled by K)	Low	None	Complexity is determined by the number of neighbors K. Low K means high complexity (+X), high K means low complexity. Typically has no explicit regularization.
Random Forest	High (each tree is complex)	Medium (Bagging acts as regularization)	High (+Z)	Integrates multiple decision trees via Bagging. It possesses high complexity (+X) and strong ensemble characteristics (+Z). The Bagging process also reduces variance, providing a regularization effect (+Y).
Gradient Boosting Machine (GBM)	High	High (via learning rate/tree depth)	High (+Z)	Integrates multiple weak learners (typically trees) sequentially via Boosting. It employs strong regularization (+Y) by limiting tree depth, using a learning rate, etc., and is a powerful ensemble method (+Z).
Neural Network	Very High (Far +X)	Adjustable (Dropout, L2, etc.)	None (typically)	Possesses extremely high expressive power. Techniques like Dropout and weight decay can move it in the +Y direction to prevent overfitting given its complex structure.



*Cheatsheet of ML*

- We will review this in the lab next week (Data Warehouse for this Week)