



# INTRO TO BACKEND DEVELOPMENT

DAY 8

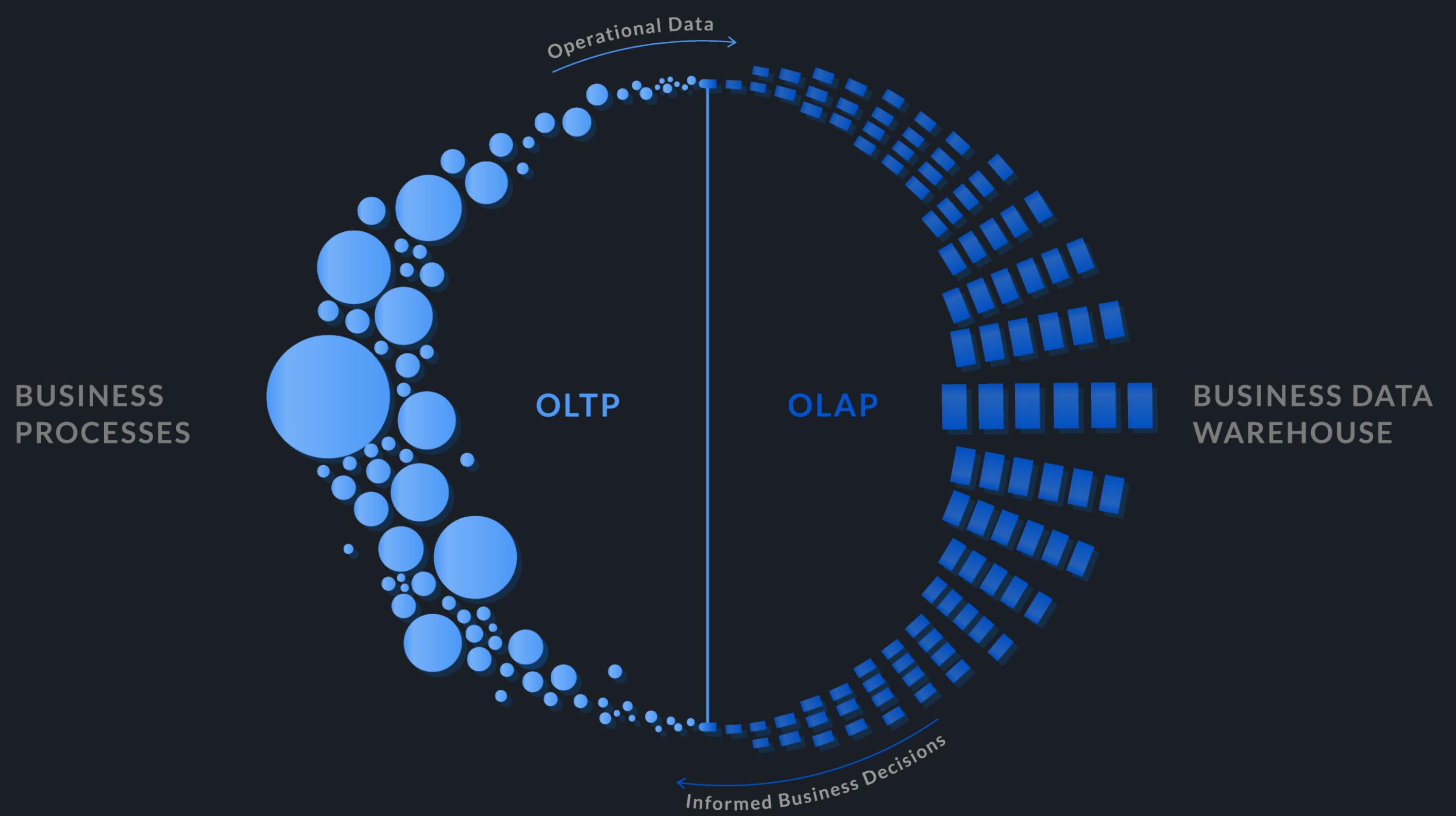
04/07/2021  
*Instructor - Casey Wilson*  
*TA - Kevin Dublin*

*Check In Time*



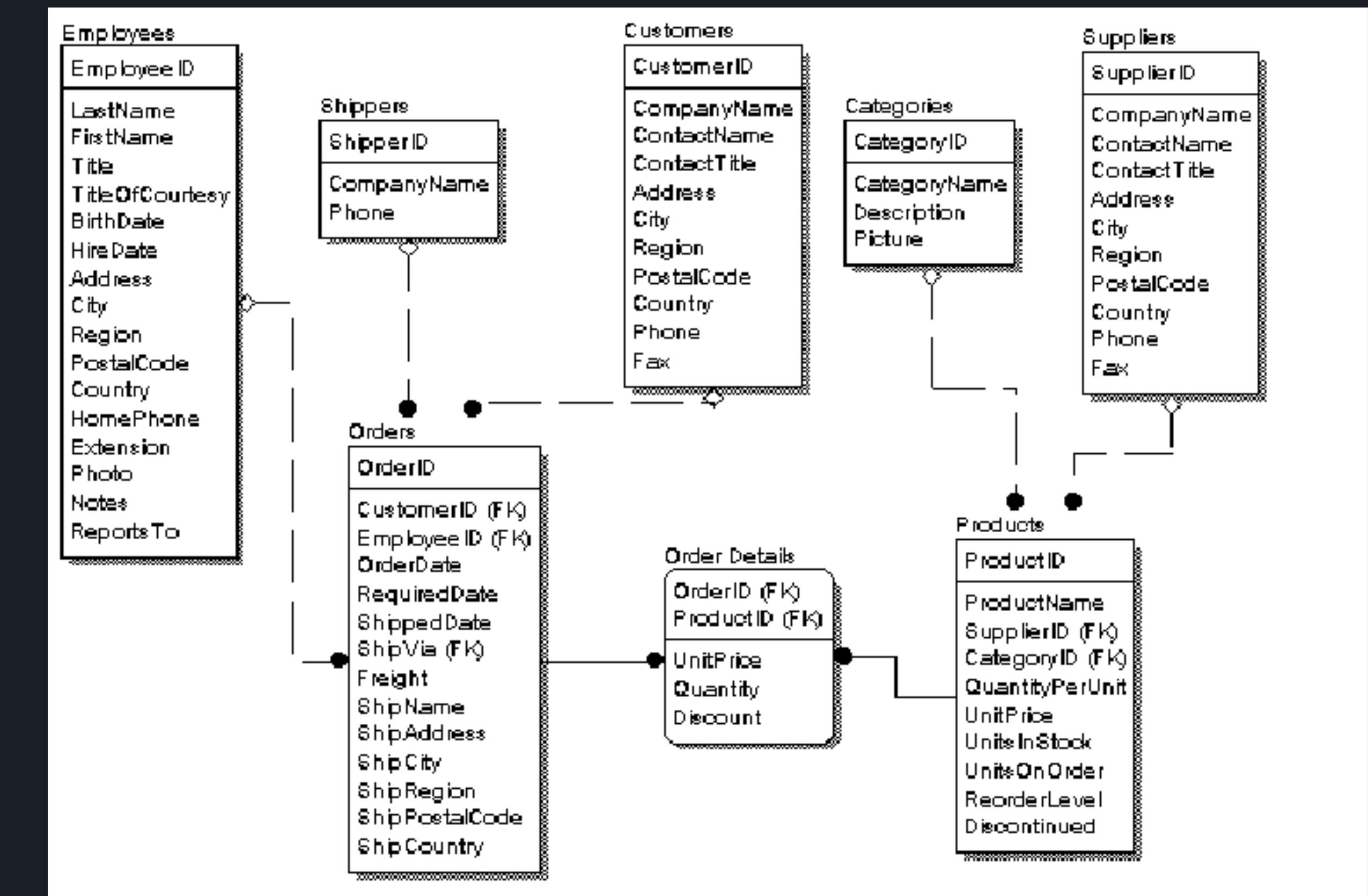
# SQL DBs → 2 Types

- OLTP
  - Online transaction processing
- OLAP
  - Online analytical processing



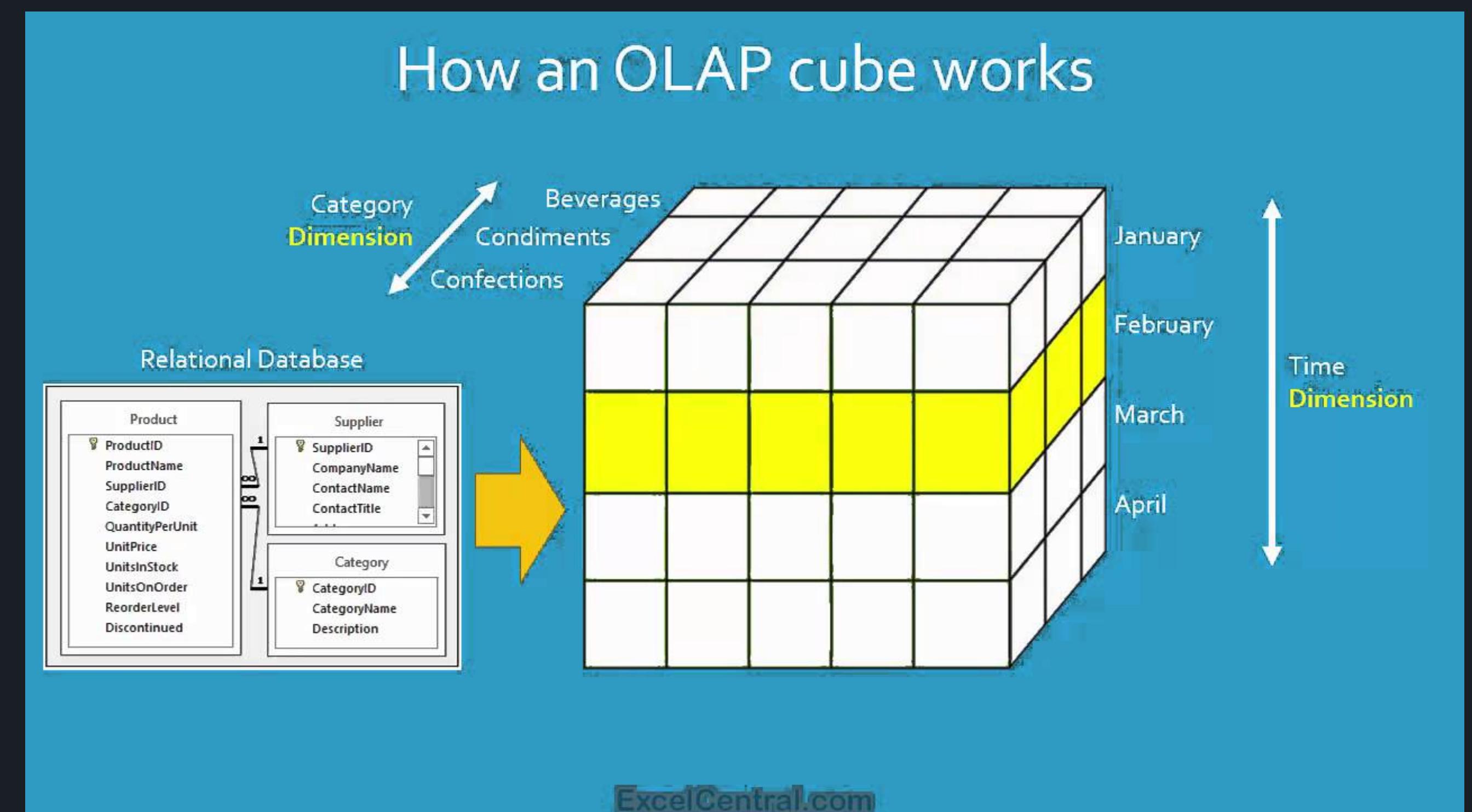
# *OLTP - Transactions*

- Online transaction processing
- Real-Time Data
- Normalized
- Application Based
- CRUD



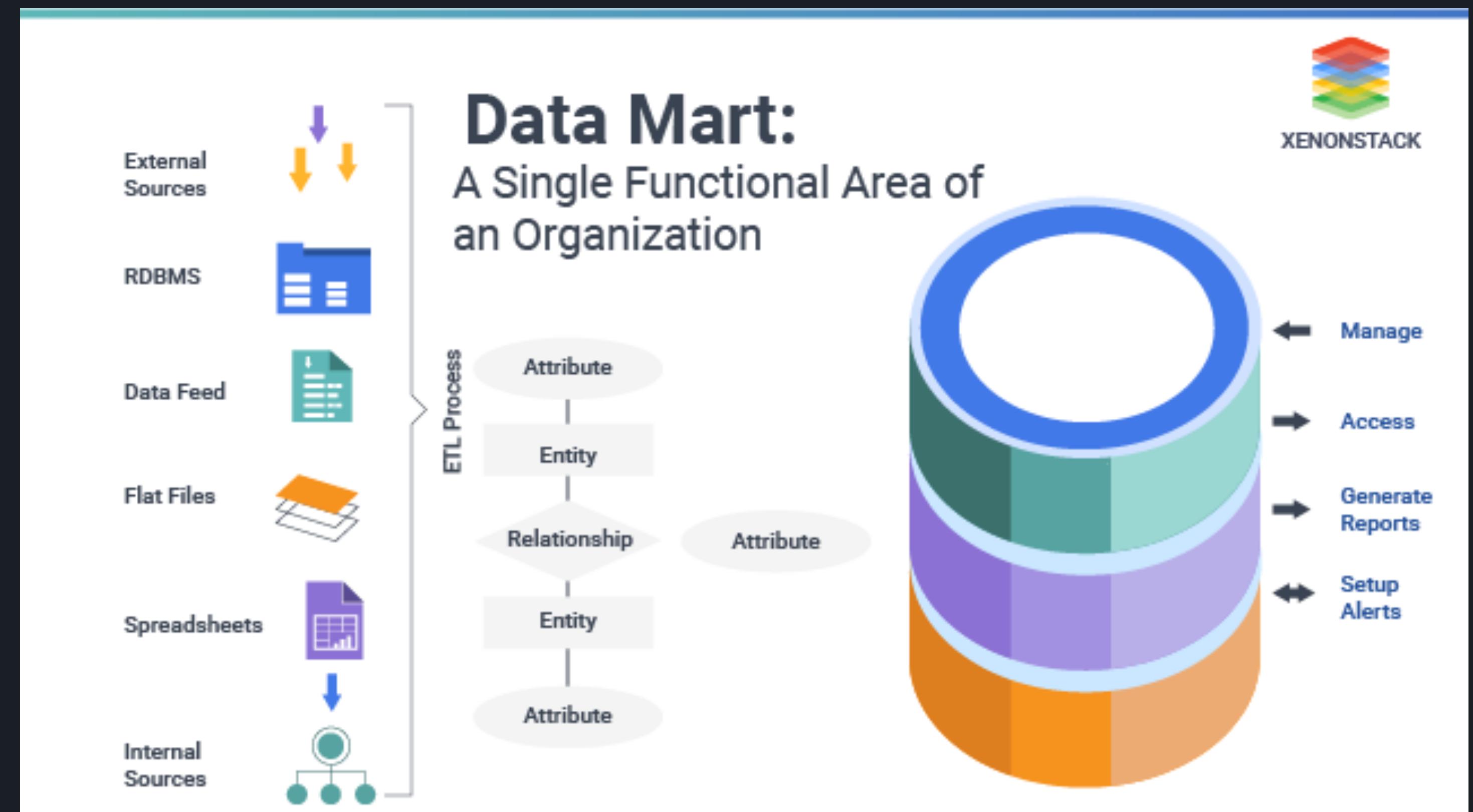
# OLAP - Analytics

- Online analytical processing
- Multiple Sources
- Multidimensional
- Not Normalized
- Business Intelligence



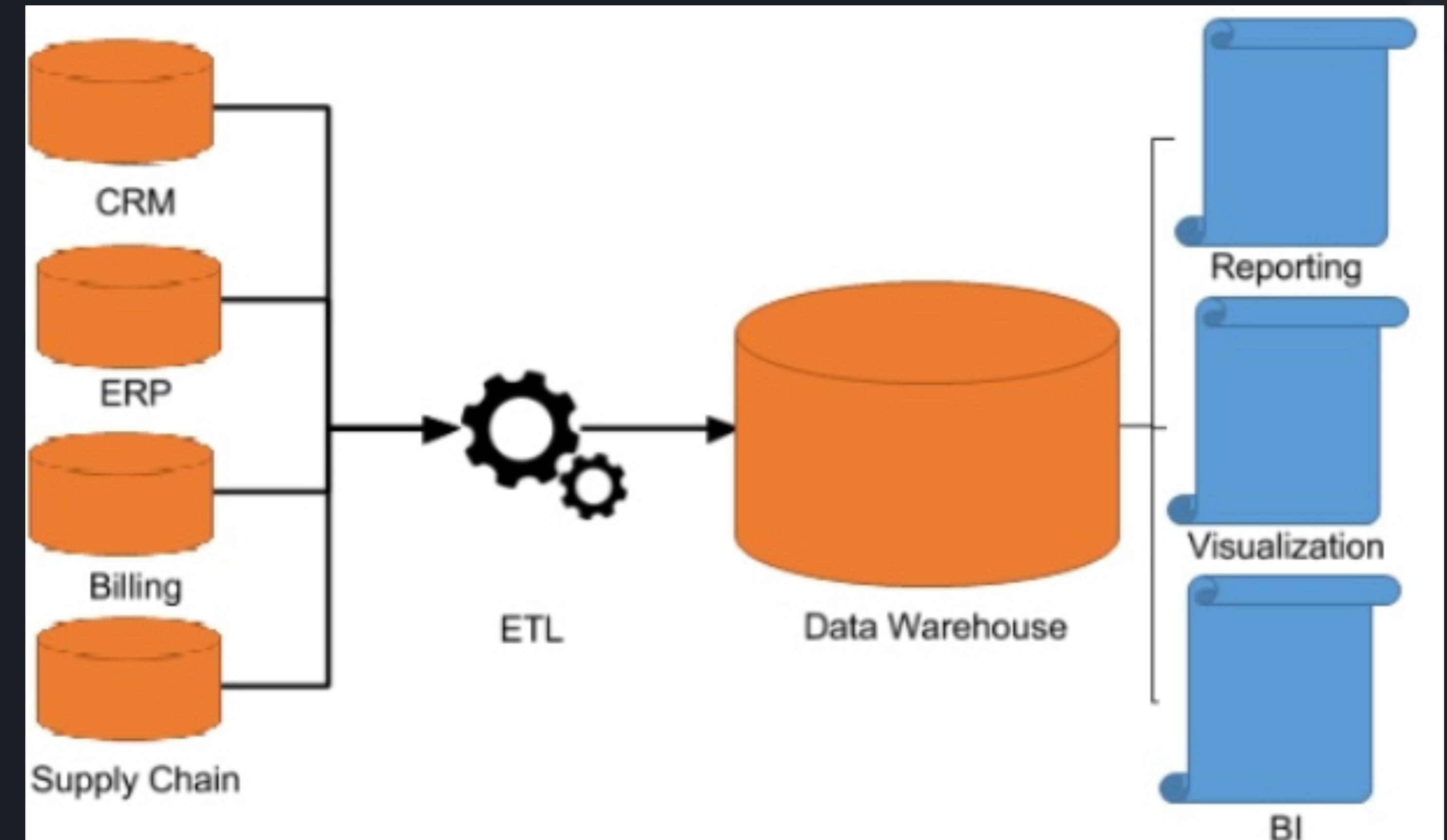
# Database Terms - Data Mart

- Data Mart
  - OLAP
  - Transforms data sources
  - Subject-Oriented
  - Logical View



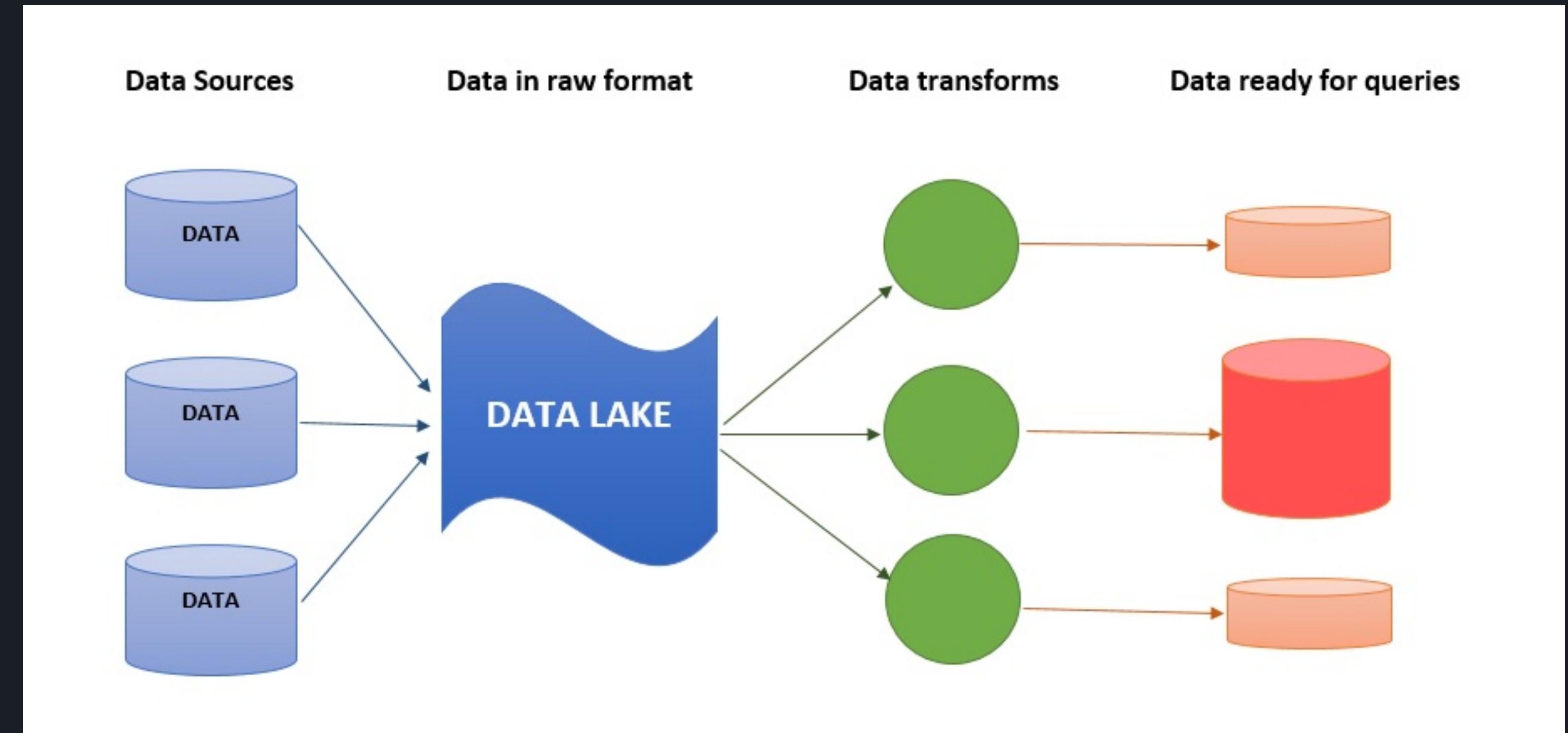
# *Database Terms - Data Warehouse*

- Data Warehouse
  - OLAP
  - Multiple Data Marts
  - Multi-Dimensional
  - Multi-Origin



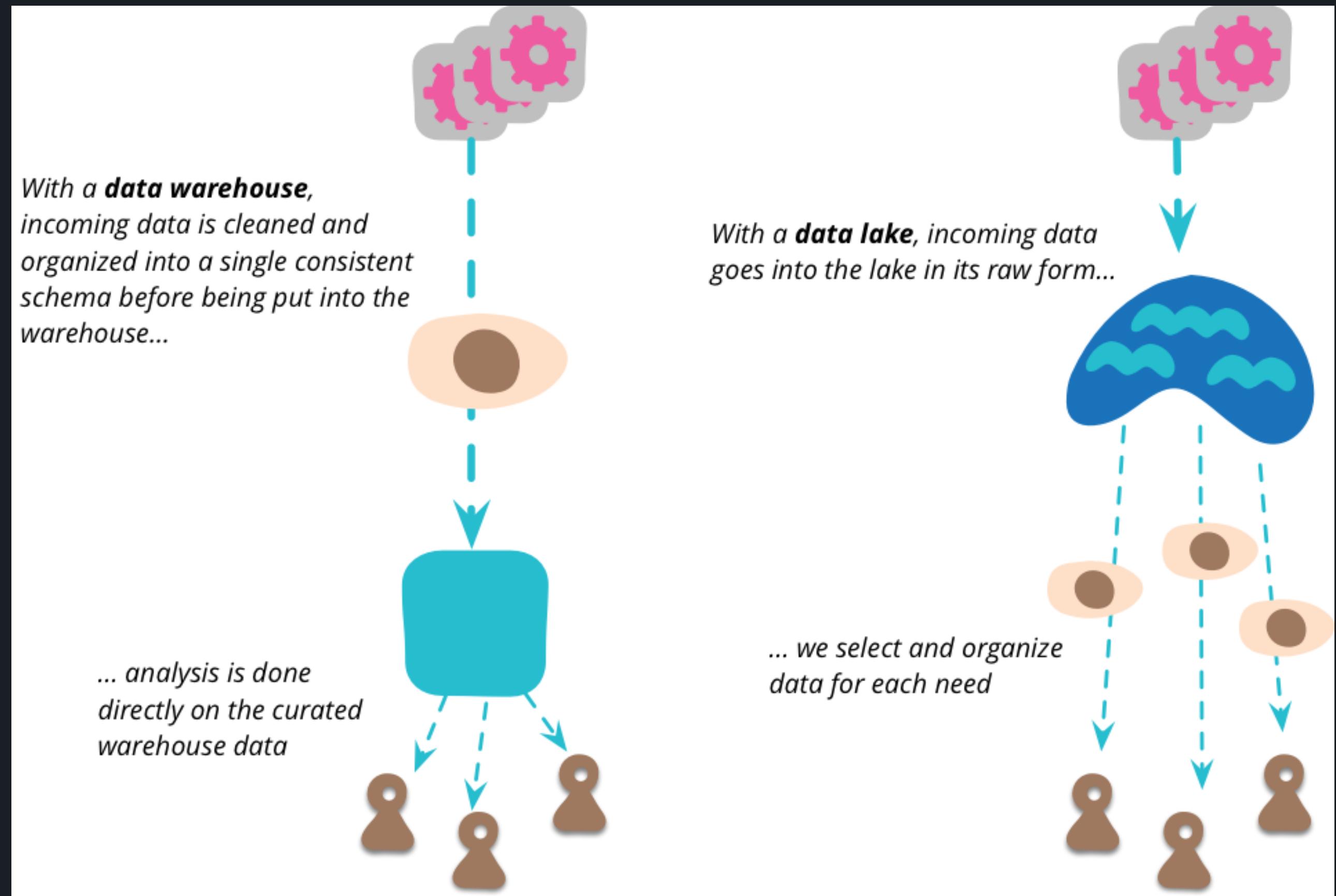
# *Database Terms - Data Lake*

- ❖ Data Warehouse
  - ❖ OLAP
  - ❖ Multiple Data Sources
  - ❖ Multi-Dimensional
  - ❖ Logical Layered

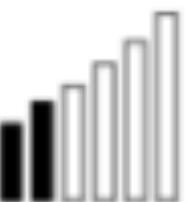


# Data Warehouse vs Data Lake

- Data Warehouse
  - Pre-Formatted
  - Designed for End User
- Data Lake
  - Raw Data
  - Designed for Data Scientist



# OLAP Summary

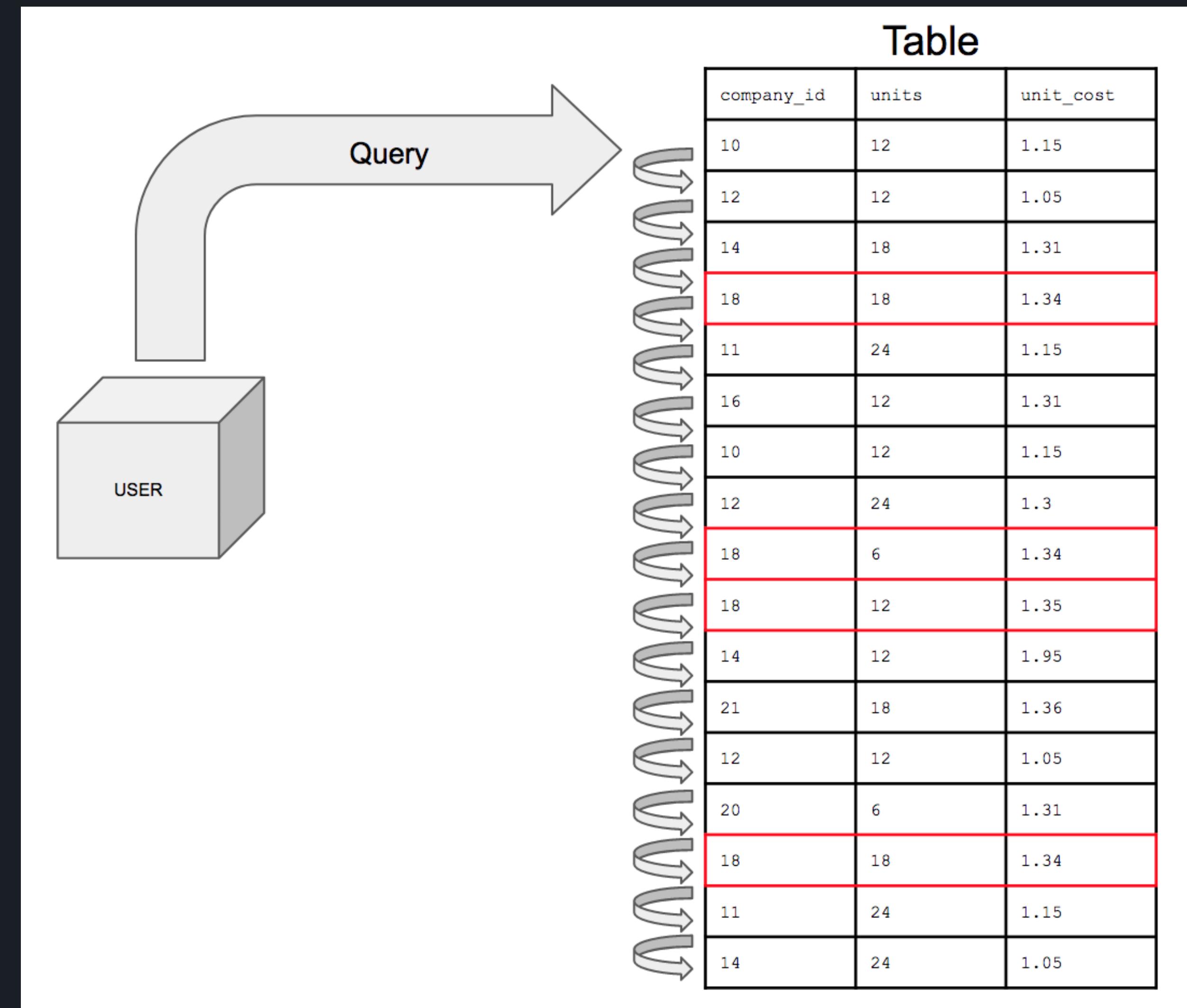
	Most Important Use Group & Use-Cases	Time-to-Market Questions & Solutions	Cost Implementation & Ownership	Users (# & Types)	Data Growth Volume & Variety
Data Lake	Predictive & Advanced Analytics	 Weeks - Months	\$\$\$\$\$		
Data Warehouse	Multi-Purpose Enabler of Operational & Performance Analytics	 Hours - Days	\$\$\$\$		
Data Mart	Line of Business Specific Reporting & Analytics	 Minutes - Hours	\$\$\$\$		

# *Database Terms - Indexing*

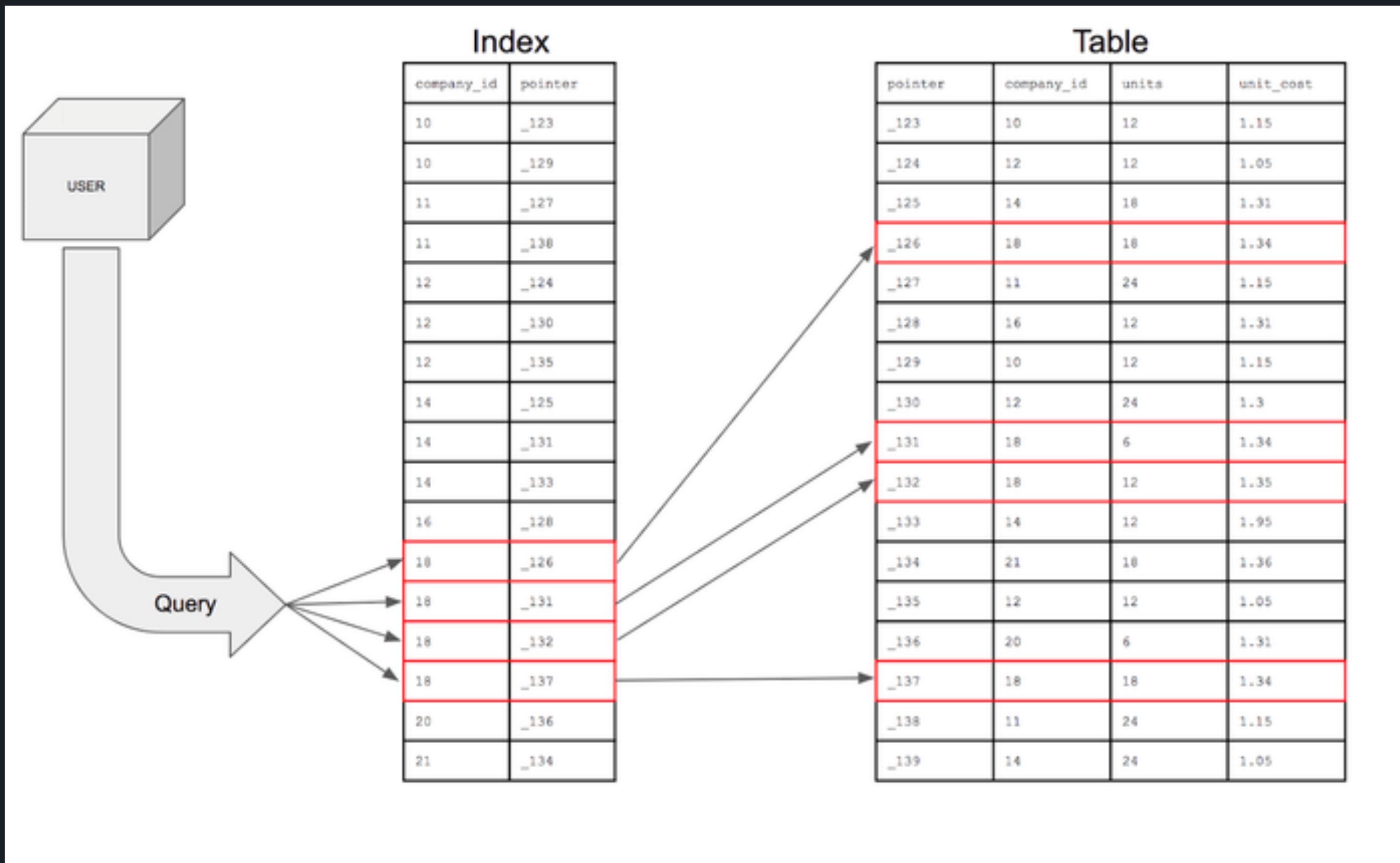
- ❖ Indexing
  - ❖ Similar to an Index of a Book
  - ❖ Helps “Find” Data in a Table
  - ❖ Indexed Data is Ordered
  - ❖ Indexed Data includes Pointer Reference

COMPANY_ID	POINTER
10	_123
10	_129
11	_127
11	_138
12	_124
12	_130
12	_135
14	_125
14	_131
14	_133
16	_128
18	_126
18	_131
18	_132
18	_137

# Database Terms - Without Indexing



# Database Terms - With Indexing



# Database Terms - Cardinality

## Cardinality Relationship

**C** Computer  
**S** Social  
**E** Education

**Cardinality** – The number of entities to which another entity can be associated through a relationship

The diagrams on the right show, in order:  
one-to-one  
one-to-many  
many-to-one  
many-to-many

The diagram illustrates four types of cardinality relationships using diamond symbols and lines:

- One-to-one: A diamond symbol with a line labeled "1" on each side.
- One-to-many: A diamond symbol with a line labeled "1" on the left and a line labeled "M" on the right.
- Many-to-one: A diamond symbol with a line labeled "M" on the left and a line labeled "1" on the right.
- Many-to-many: A diamond symbol with a line labeled "M" on both the left and right sides.

Relationship

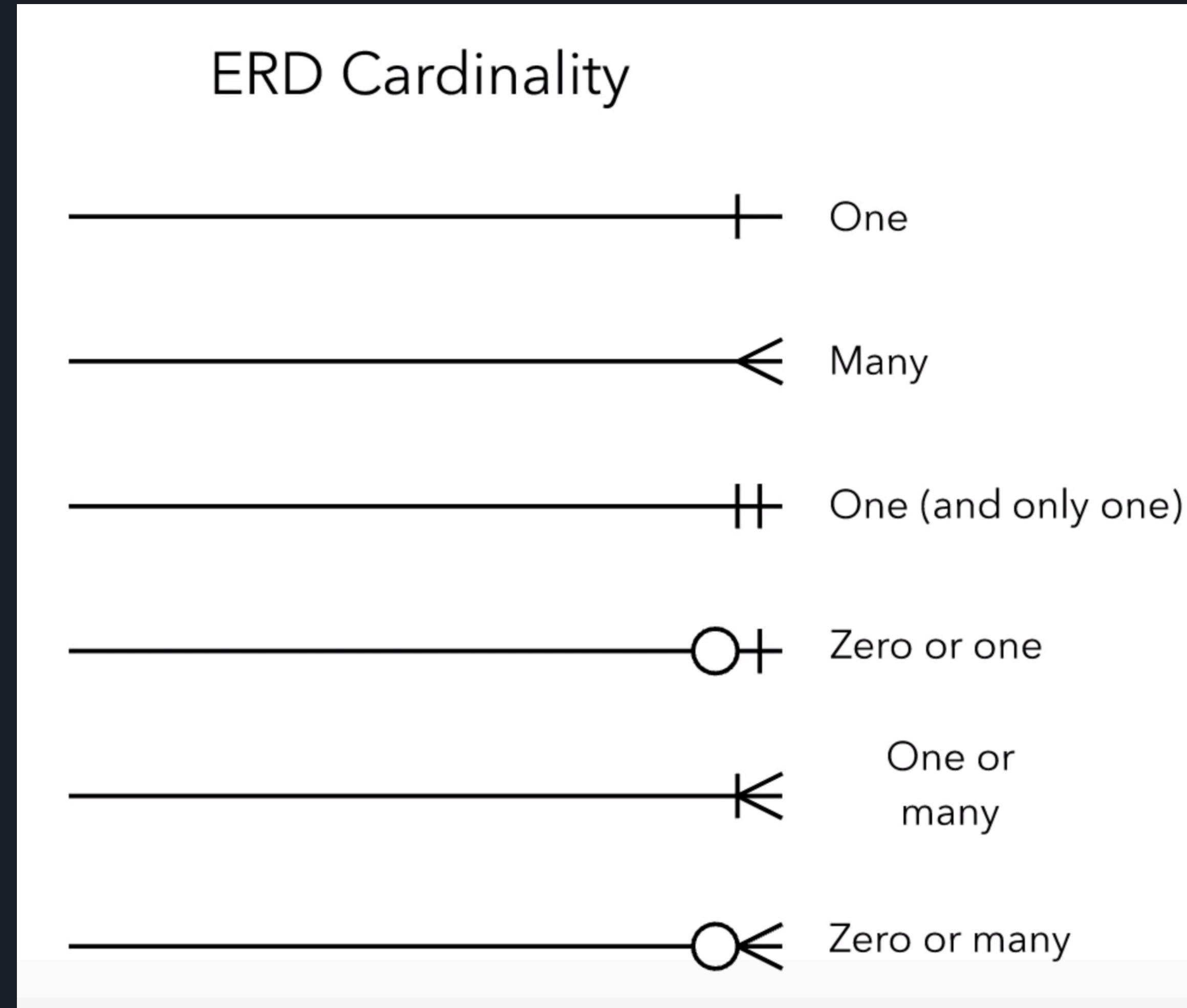
Relationship

Relationship

Relationship

Navigation icons: back, forward, search, etc.

# *Database Terms - Cardinality Cont.*

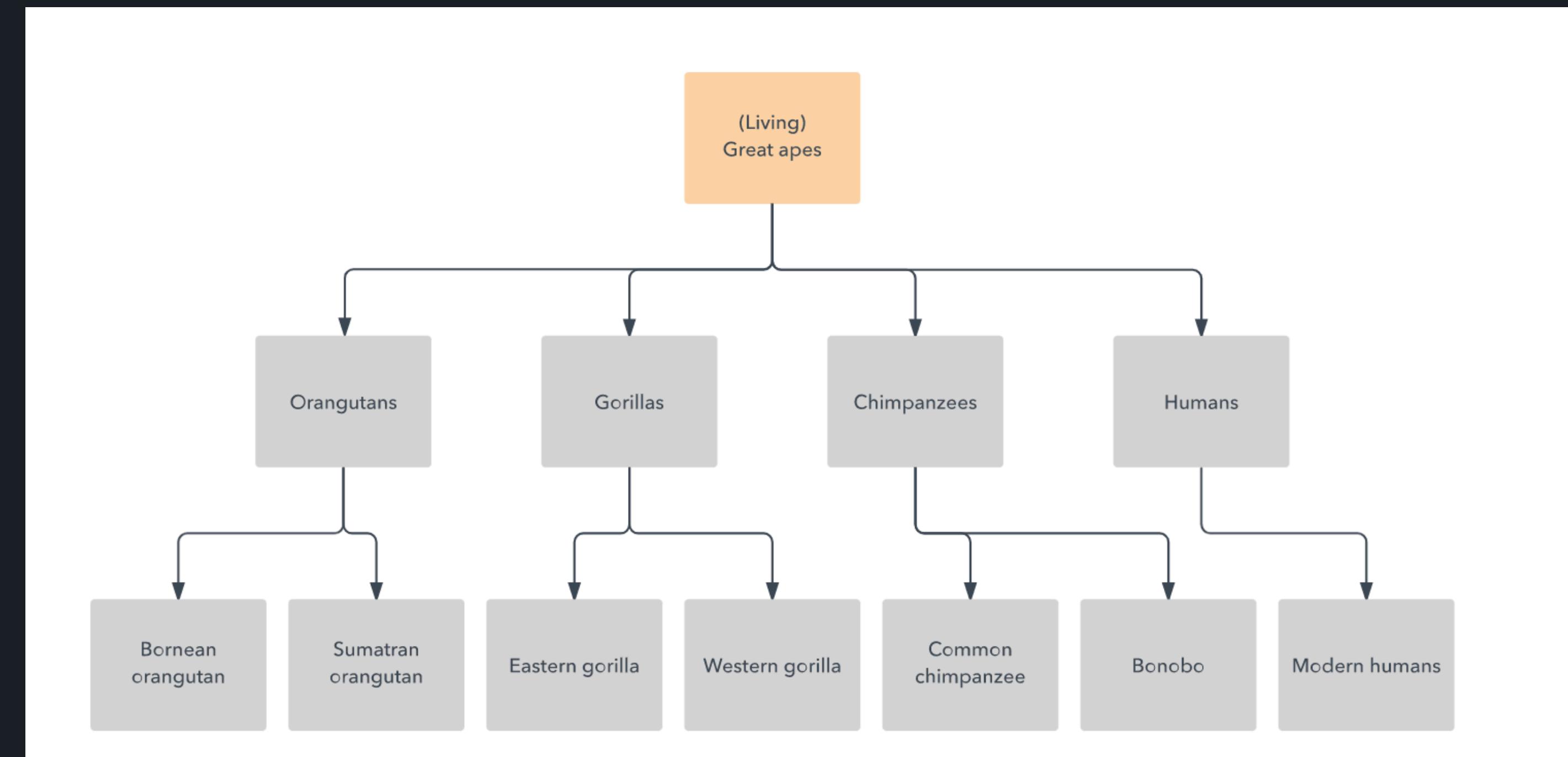


# *Database Design*

- ❖ Hierarchical database model
- ❖ Relational model (ER Model)
- ❖ Network model
- ❖ Object-oriented database model
- ❖ Document model
- ❖ Entity-attribute-value model
- ❖ Star / Snowflake schema
- ❖ The object-relational model

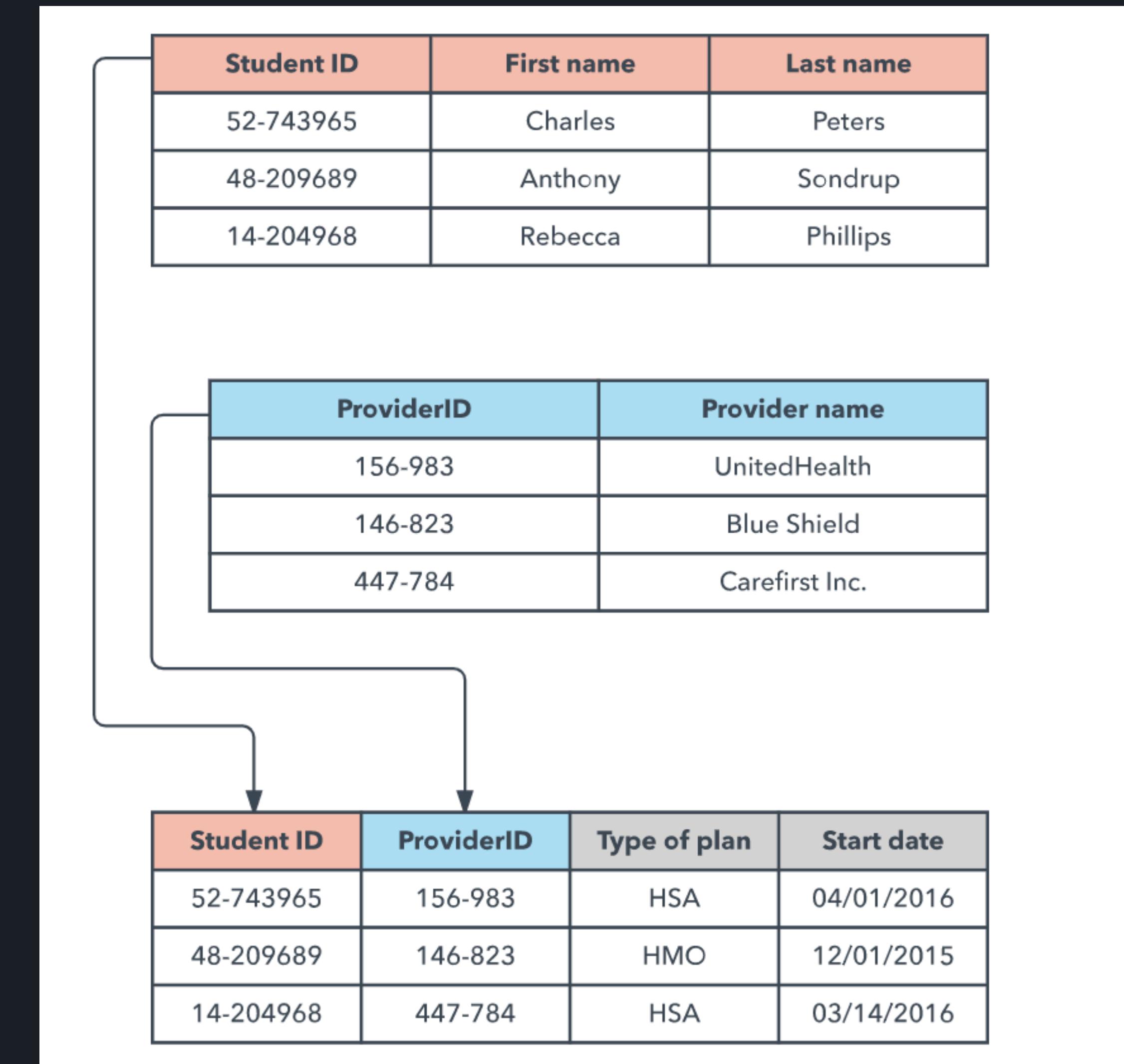
# *Database Design - Hierarchical*

- Hierarchical Database Model
  - Tree-Like Structure
  - Each Record has 1 Parent
  - Possibility for Siblings
  - Does not describe complex relationships



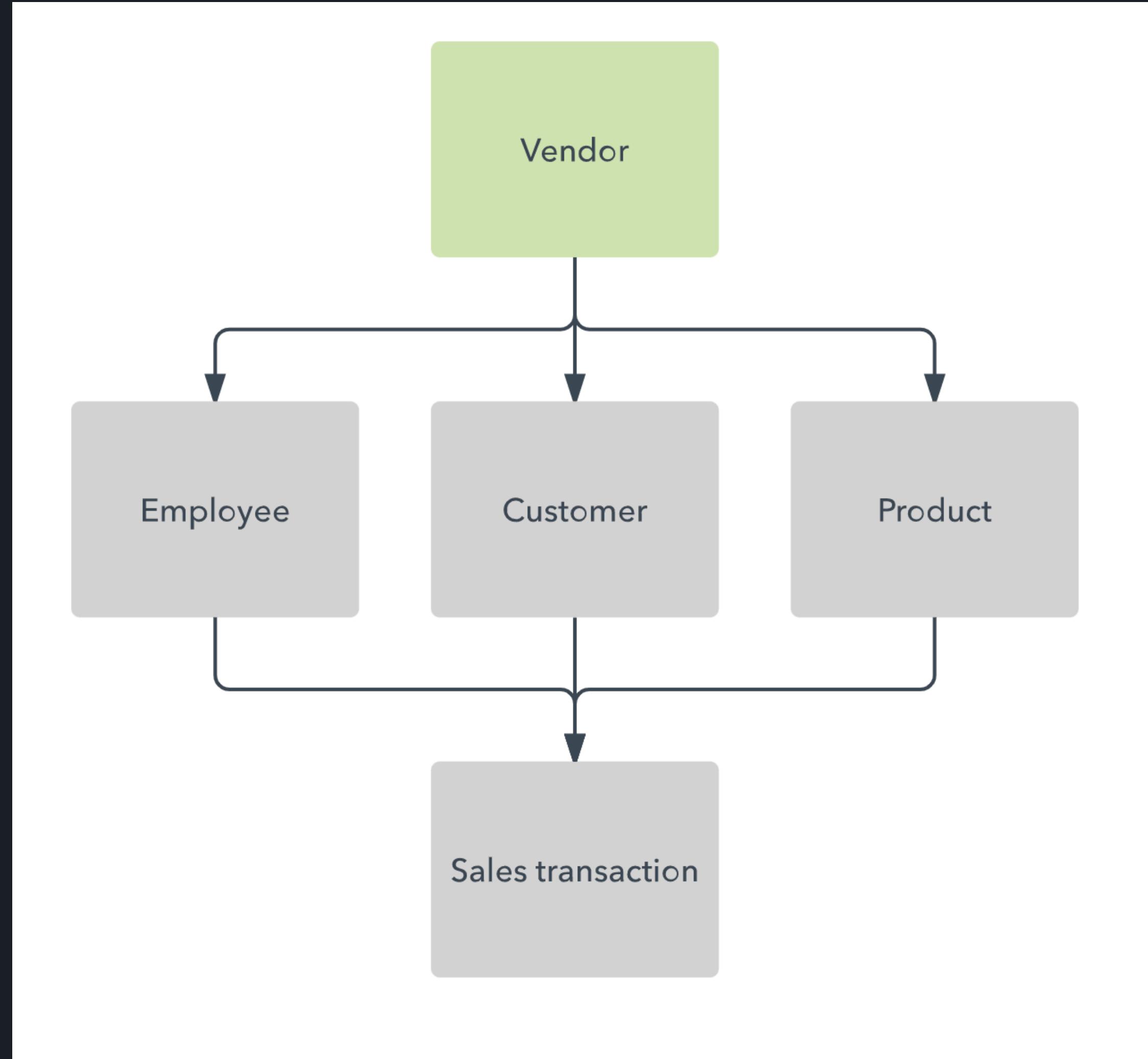
# *Database Design - Relational*

- Relational Database Model
  - Table Structure
  - Entity / Relationship
  - Foreign Key / Primary Key



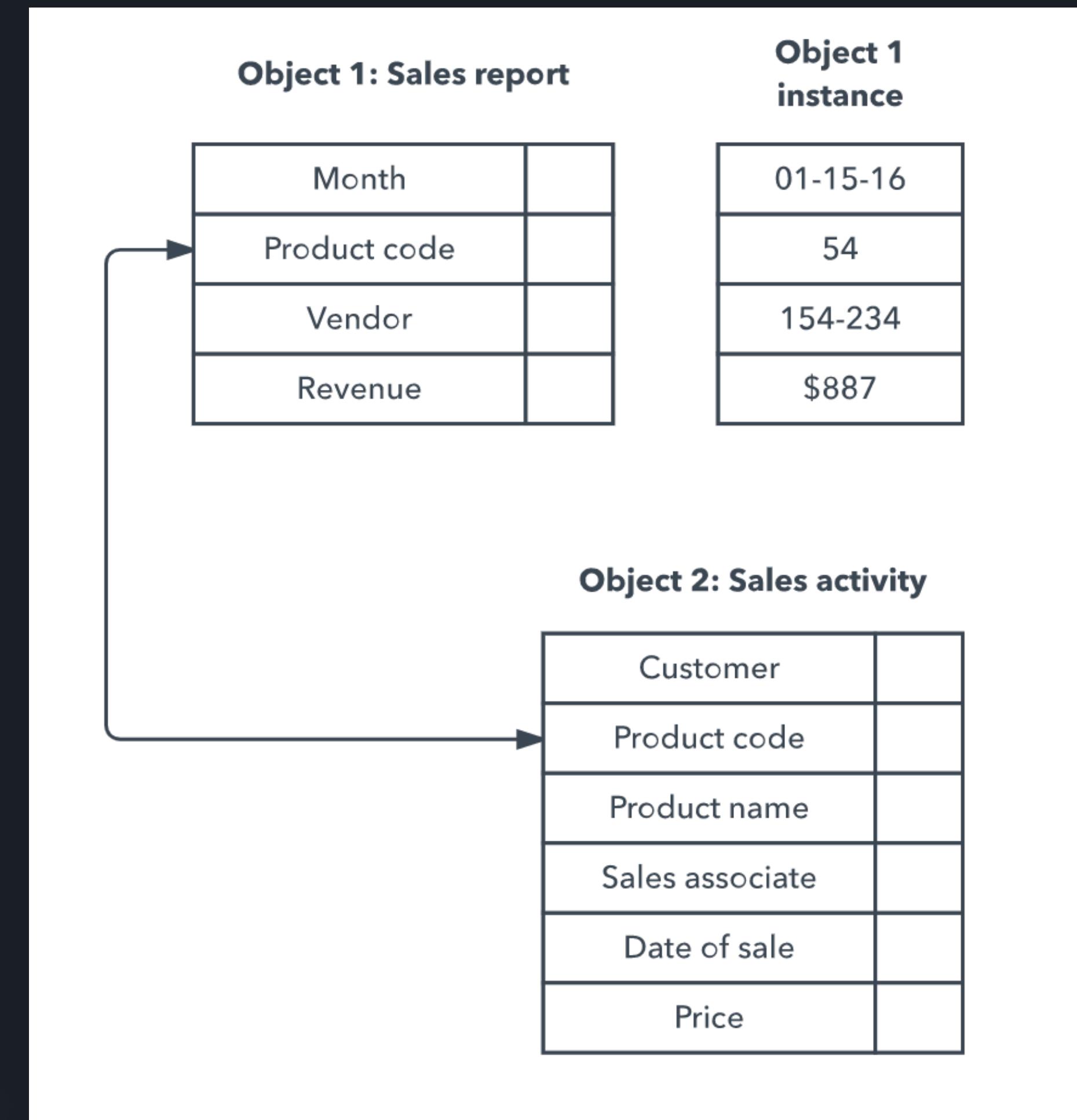
# *Database Design - Network*

- Network Database Model
  - Similar to Hierarchy
  - Allows for Many-To-Many Relationships
  - One Parent



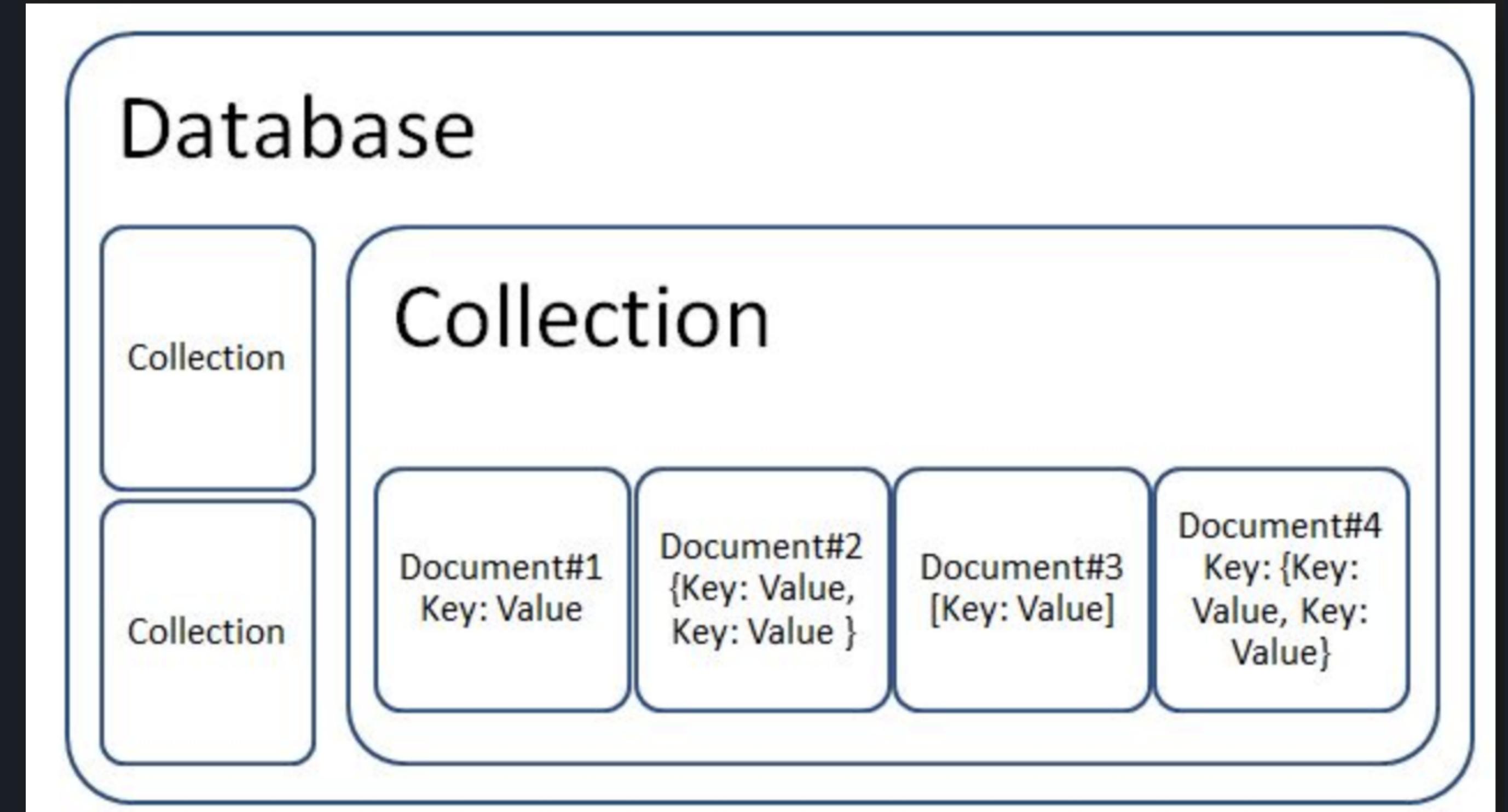
# *Database Design - Object Oriented*

- Object Oriented Database Model
  - Table Structure
  - Used for Multimedia and non-traditional data
  - Hyperlinking Capabilities



# *Database Design - Document*

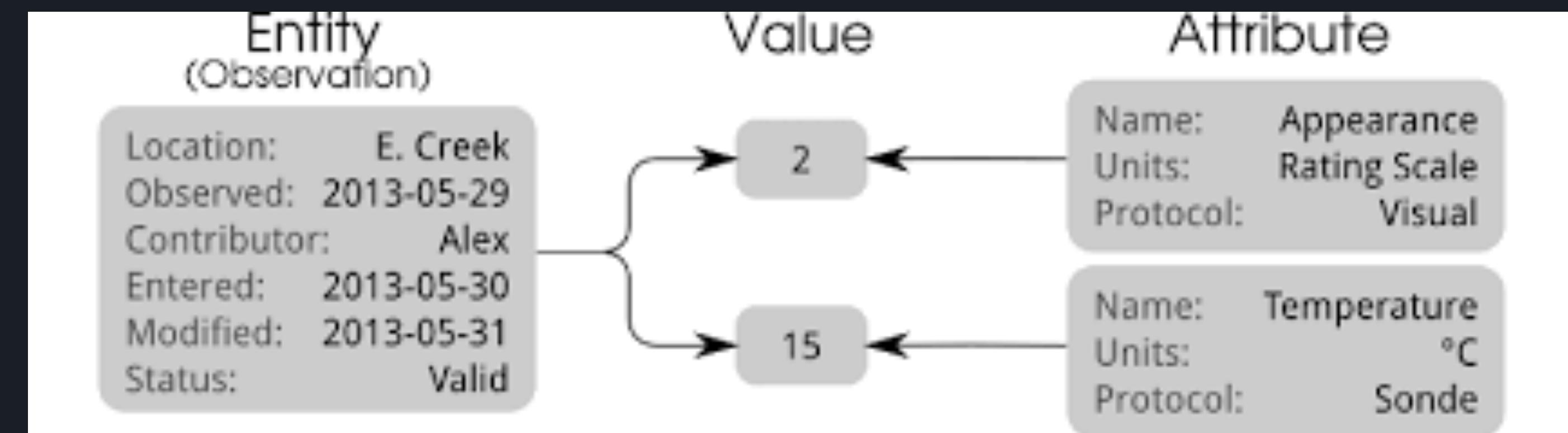
- Document Database Model
  - Document (semi-structured data)
  - Non-Atomic
  - NoSQL



# *Database Design - Entity-Attribute-Value*

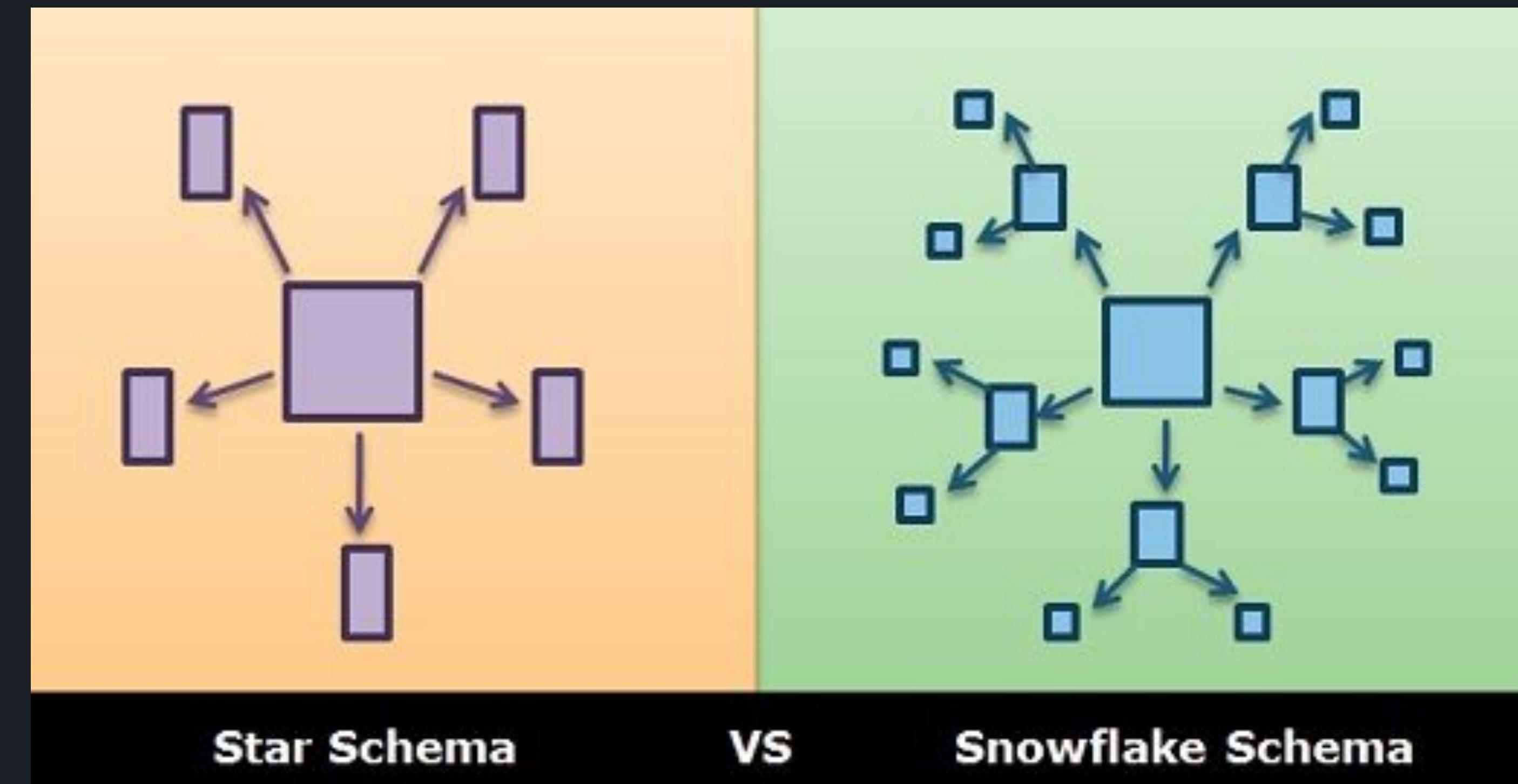
- EAV Database Model

- Quick Lookups
- GeoMapping
- Fact Finding



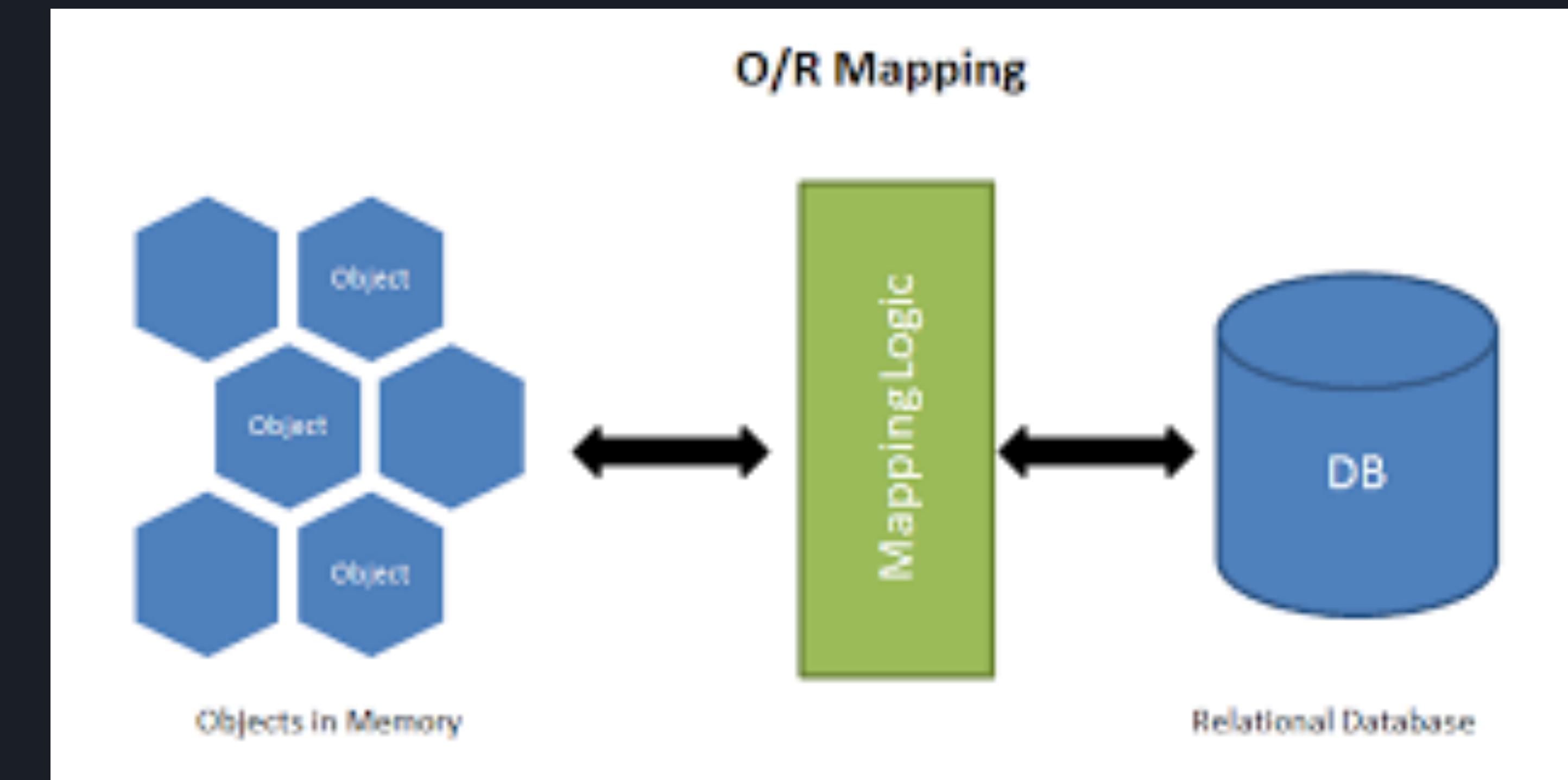
# *Database Design - Star / Snowflake*

- EAV Database Model
  - OLAP
  - Facts / Dimensions
  - Business Intelligence



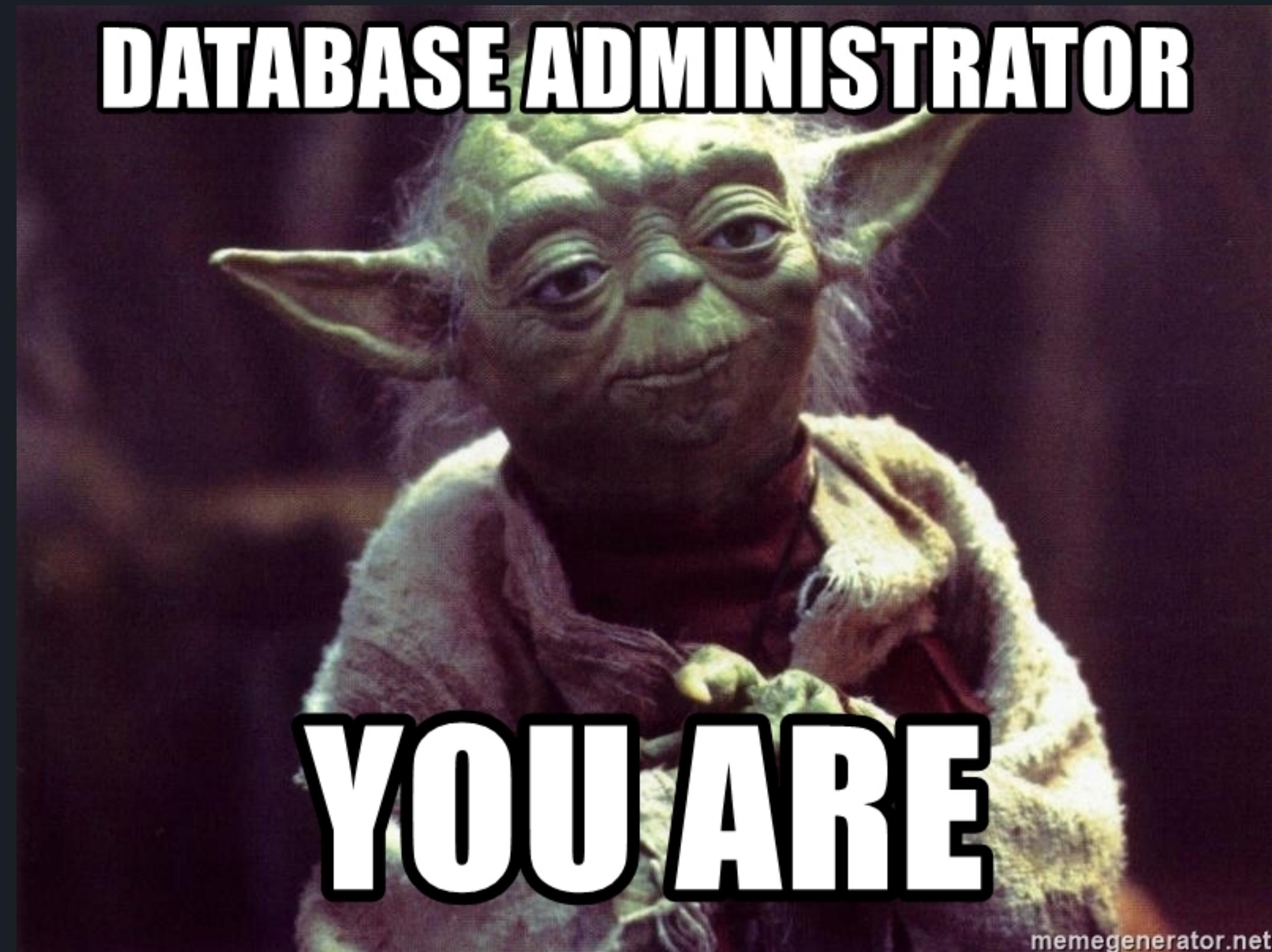
# *Database Design - Object Relational*

- Object Relational Model
  - Also called Hybrid Model
  - Abstracts ER Model
  - Basis for Django's ORM



# *So How Do Design a Database?*

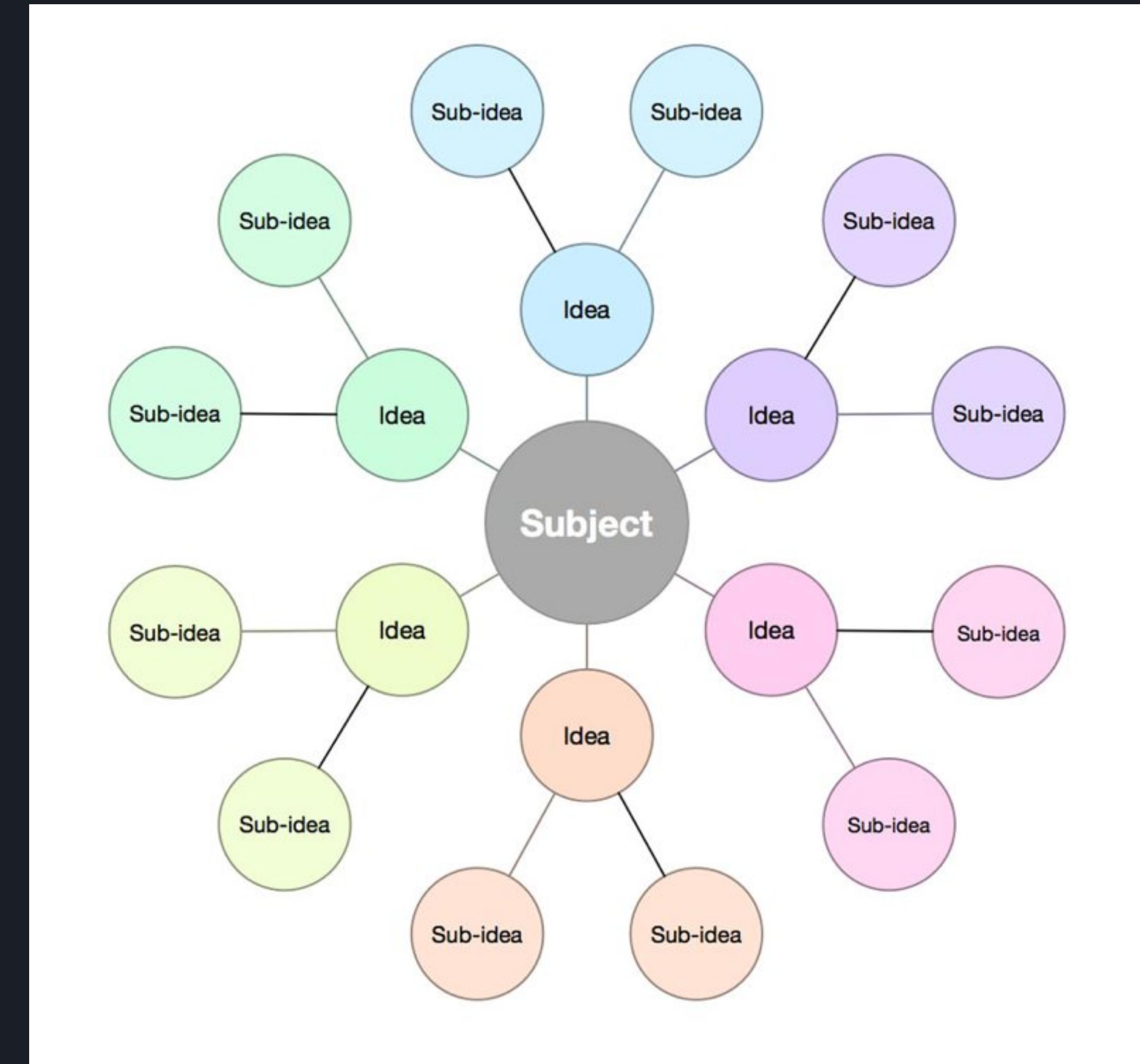
- 4 Steps
  - Conceptual Design
  - Logical Design
  - Normalization
  - Physical Implementation



memegenerator.net

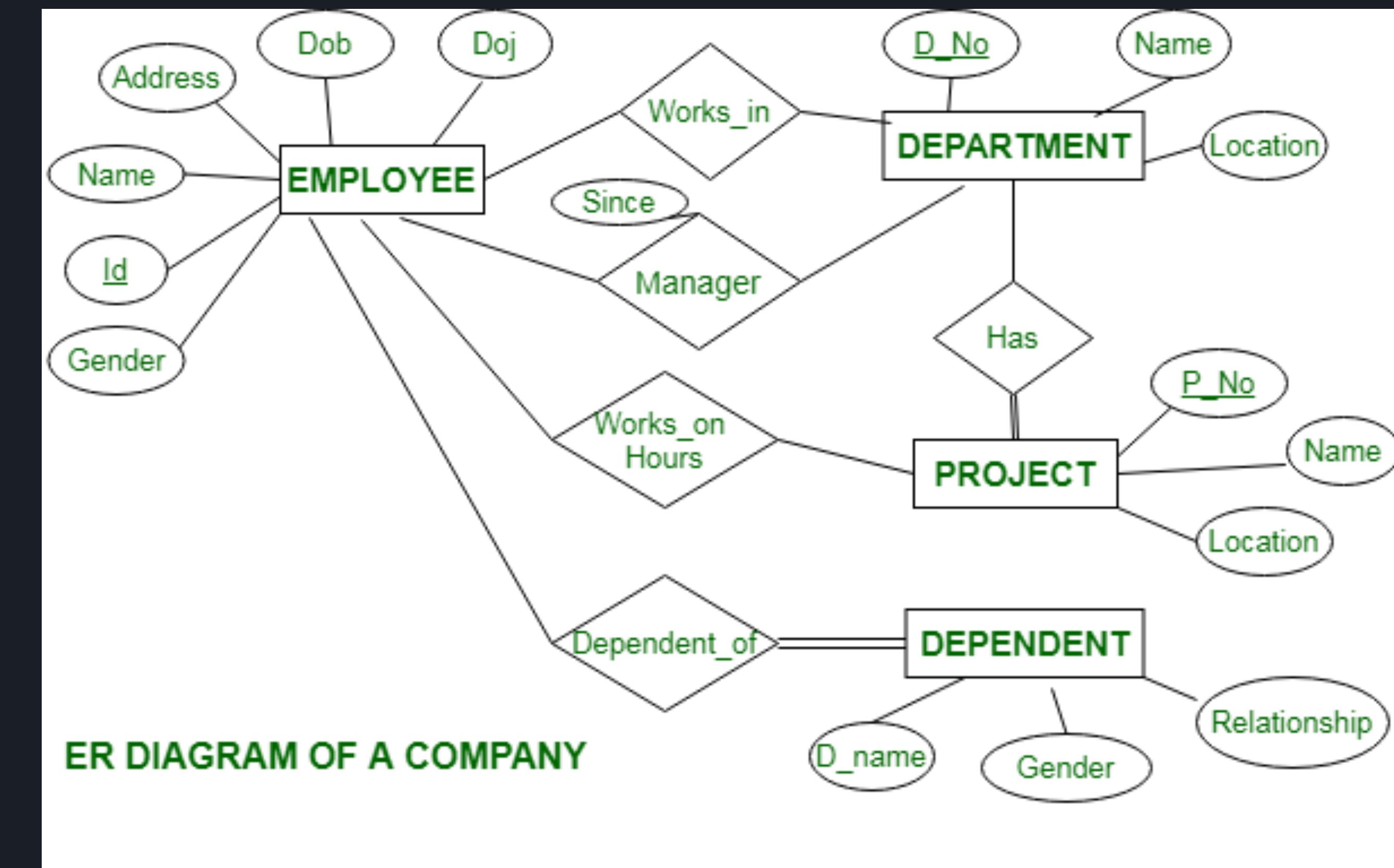
# *Design* → Conceptual

- ❖ Conceptual
  - ❖ High Level
  - ❖ Basic Needs
  - ❖ Brainstorming Phase



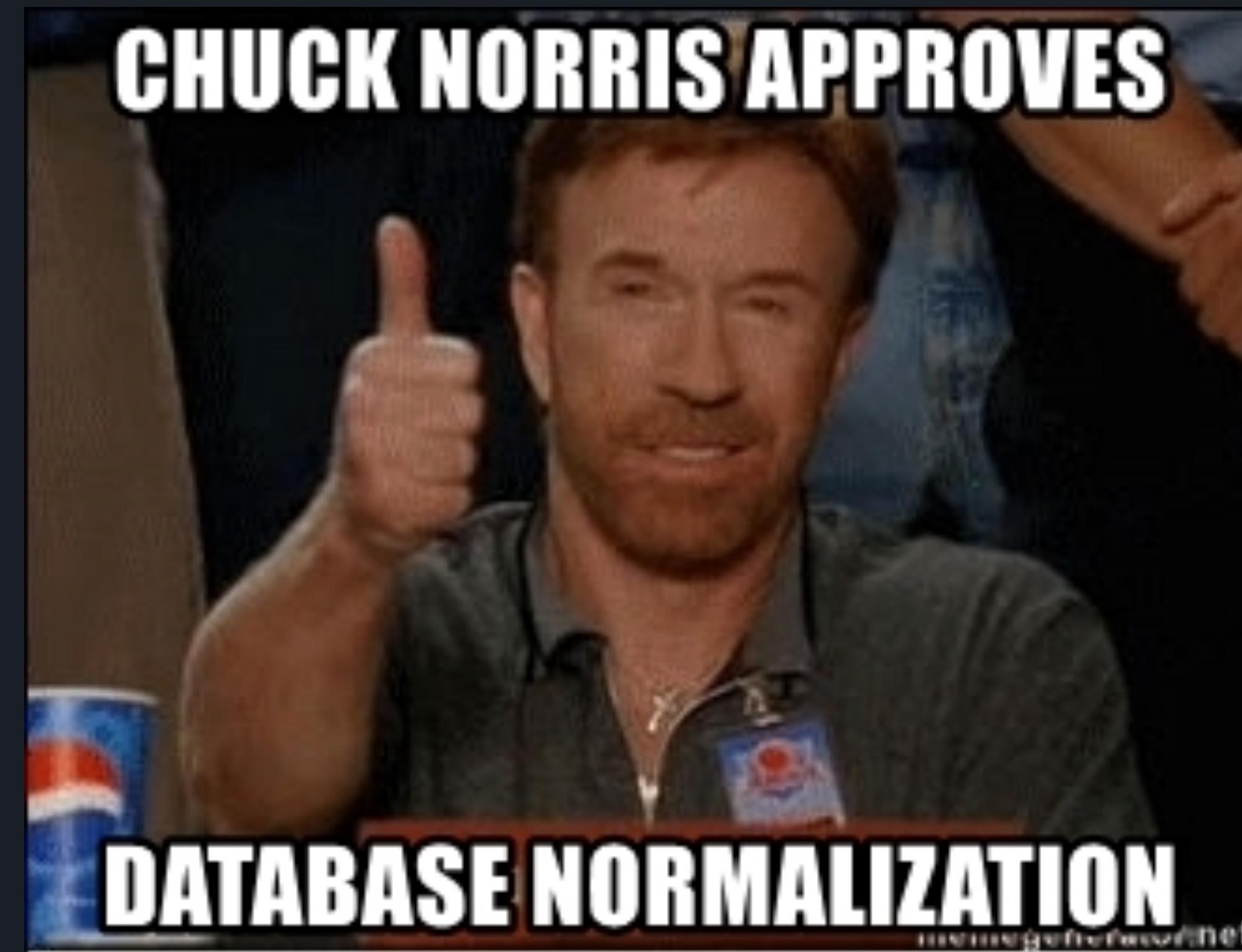
# *Design → Logical*

- Logical
  - Data Modeling
  - Map Tables
  - ER diagram



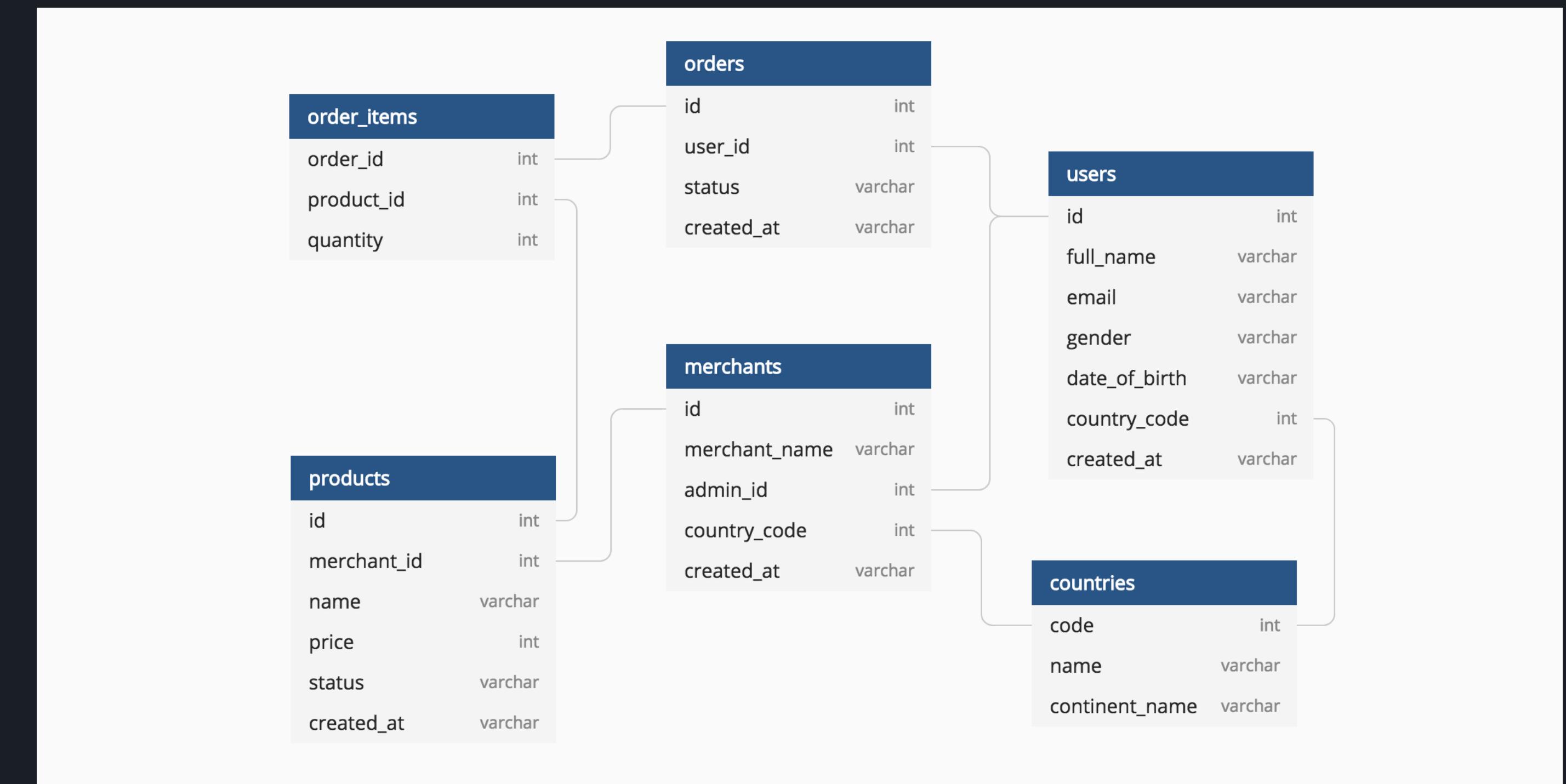
# *Design → Normalization*

- Normalization
  - Remove Duplicates
  - Check Logic
  - Maintain ACID



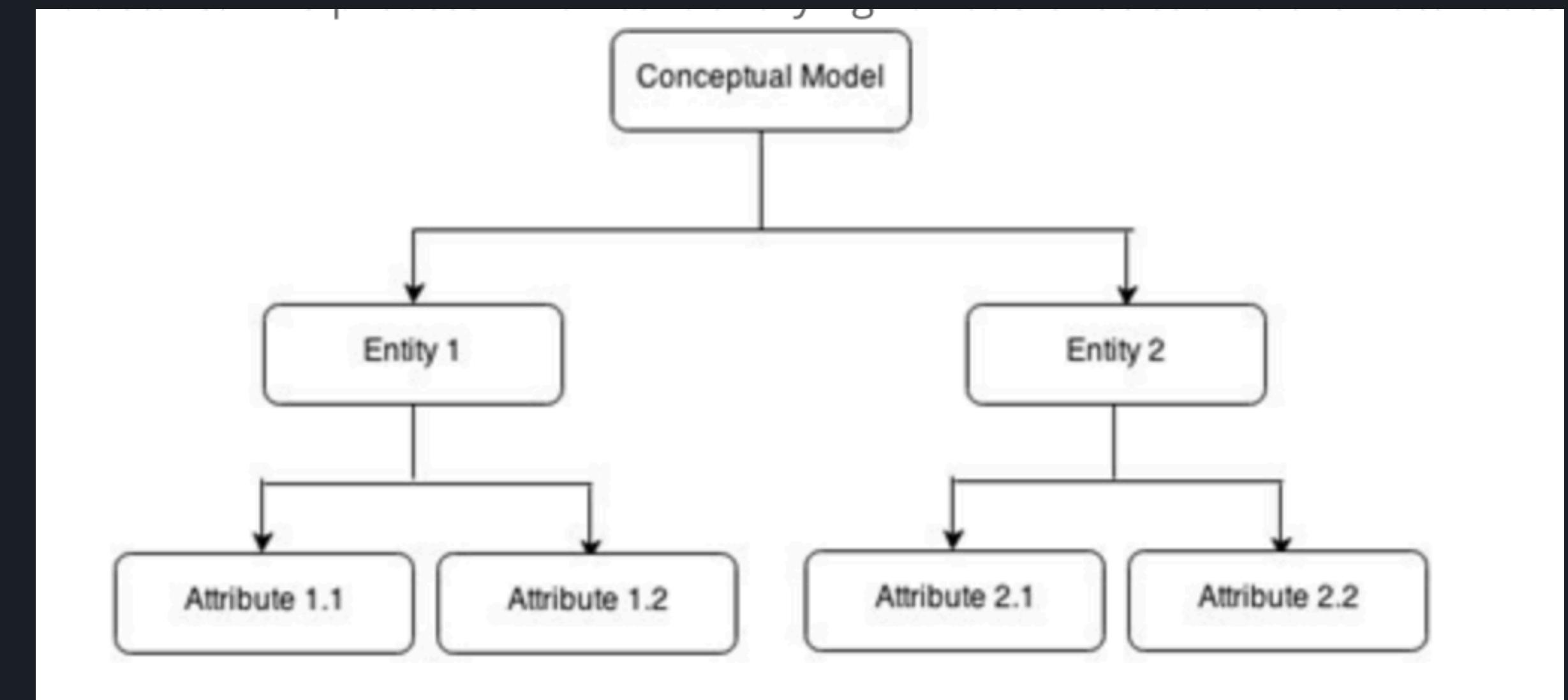
# *Design → Physical*

- Physical
  - DDL
  - Schema Diagrams
- Create Database



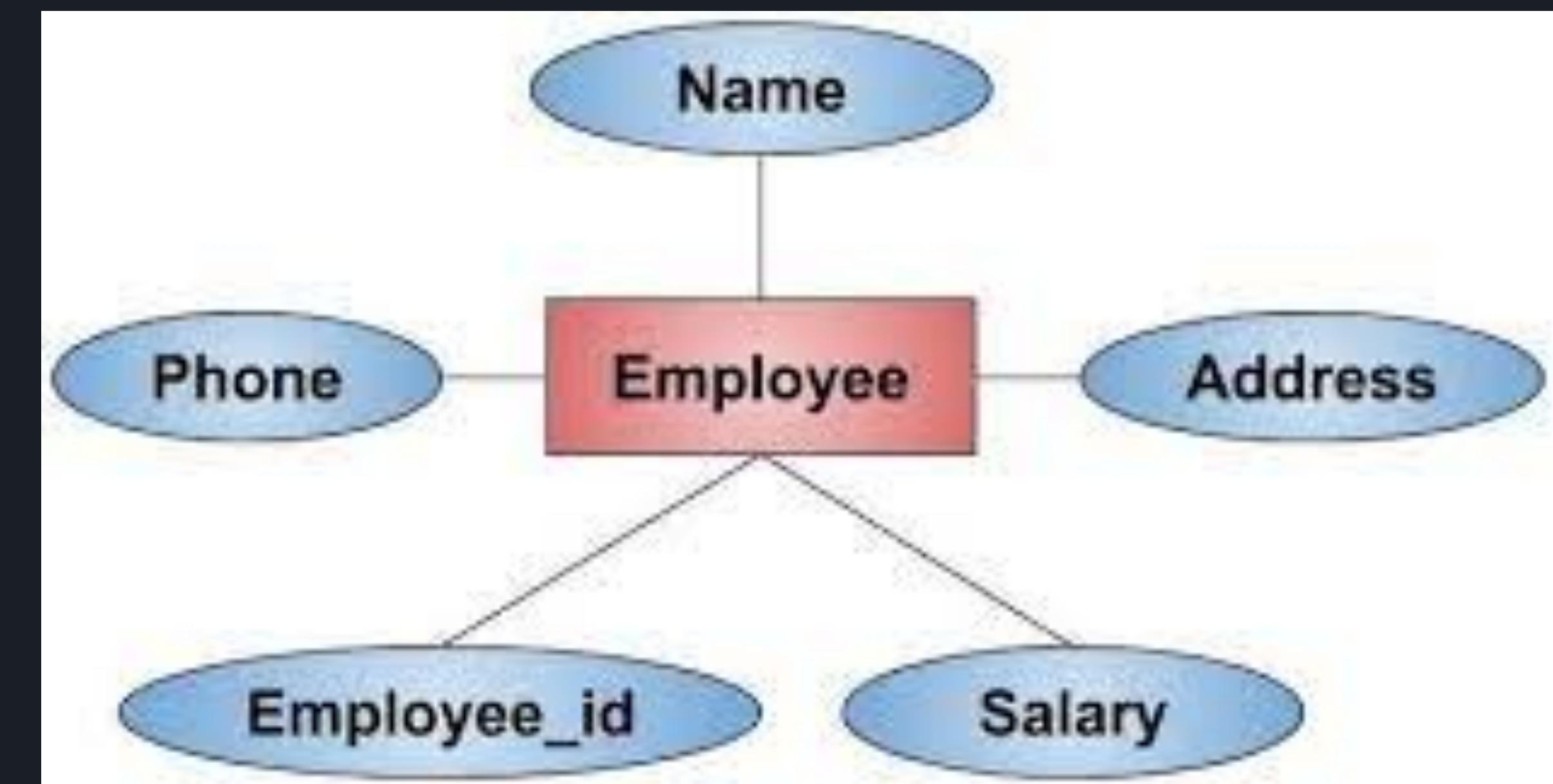
# *Other Design Ideas - Top Down*

- Top Down Approach
  - Start with Abstract
  - Develop Each Entity through Relations
  - Think Hierarchical



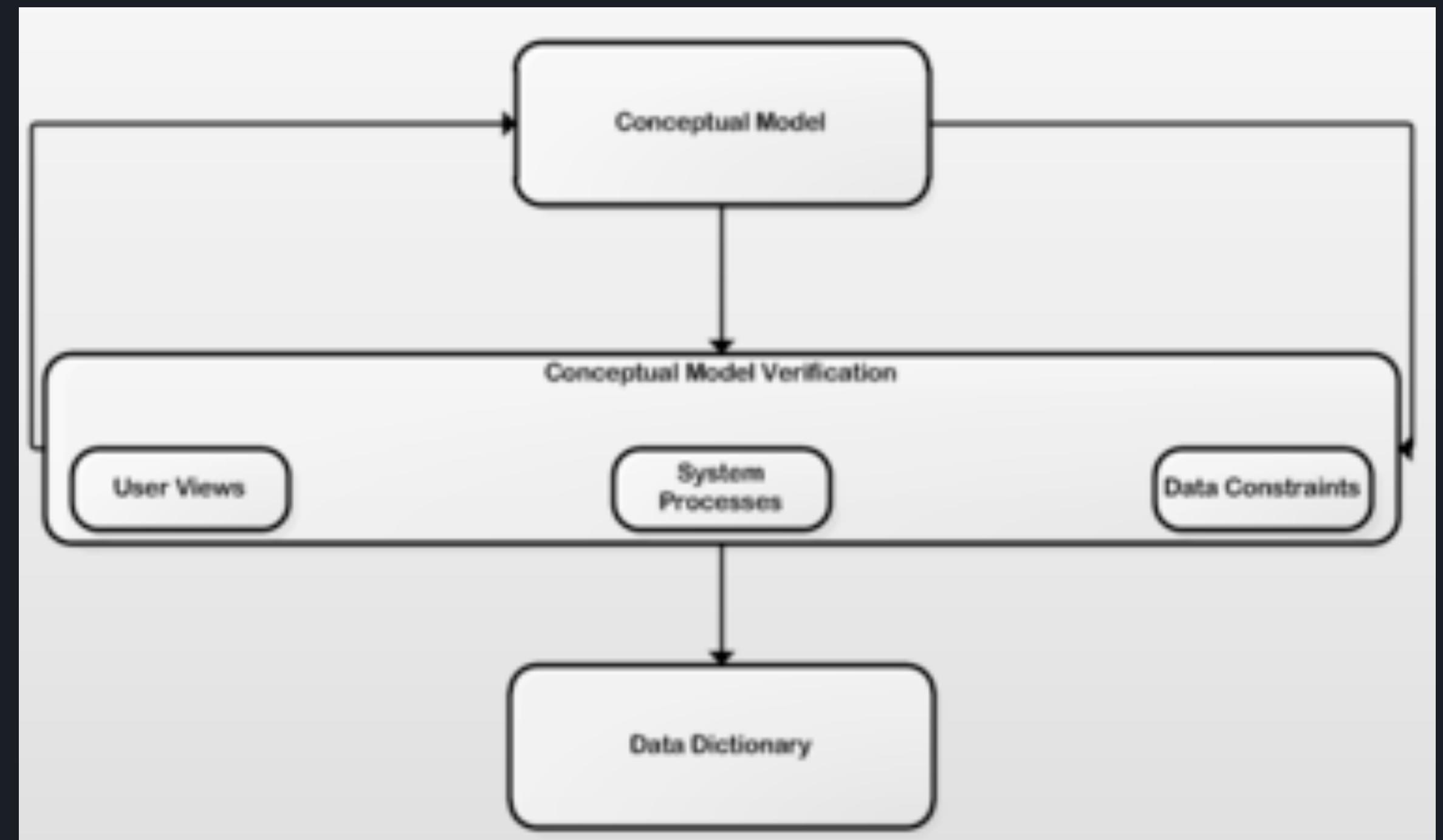
# *Other Design Ideas - Bottom Up*

- Bottom Up Approach
  - Start with Details
  - Develop Each Entity In Isolation
  - Think About Individual Parts



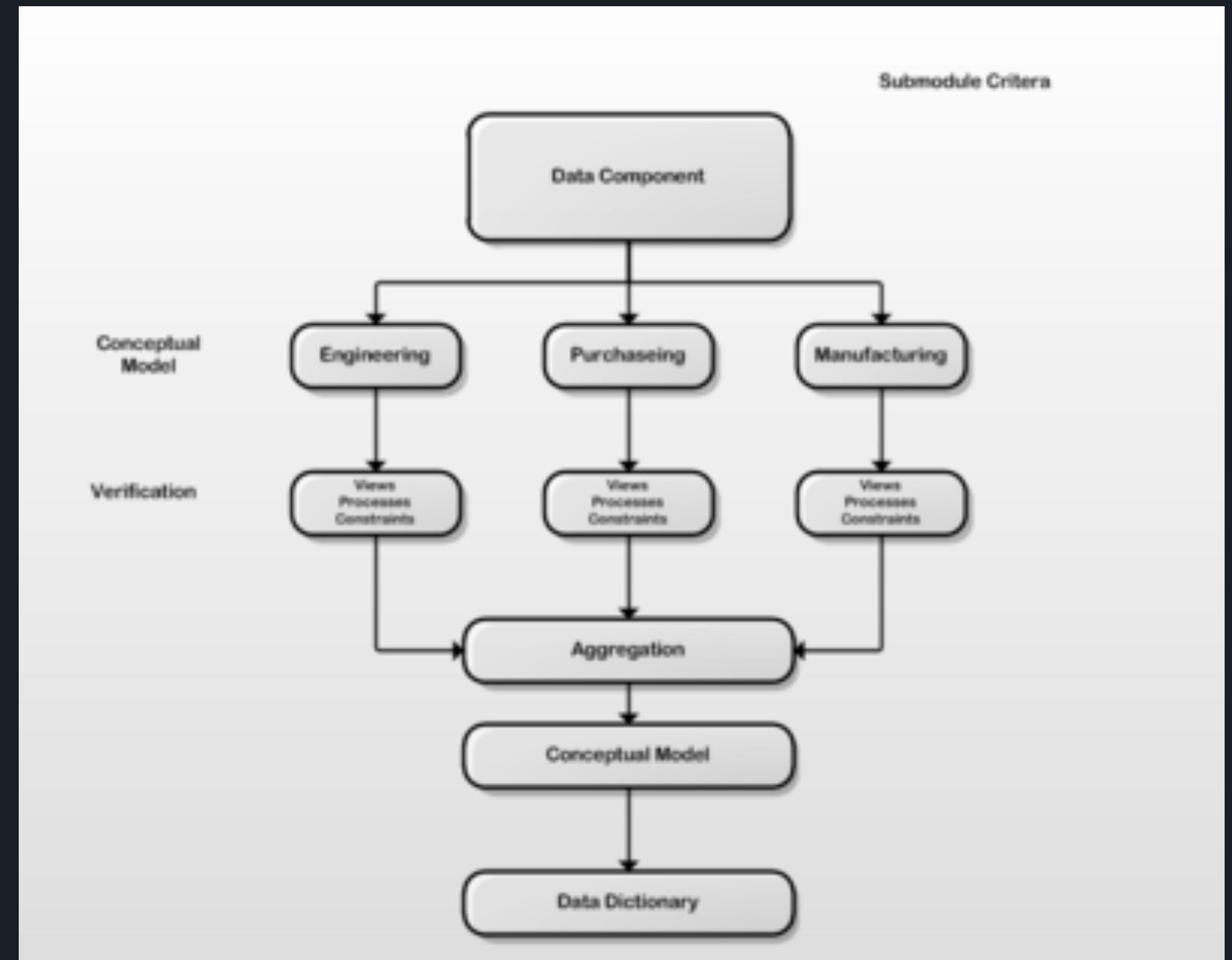
# *Other Design Ideas - Centralized*

- Centralized Approach
  - Small Scale
  - Core Design that branches outward
  - “Start-Up” Mentality



# *Other Design Ideas - Decentralized*

- Decentralized Approach
  - Large Scale
  - Column Based Approach
  - “FAANG” Mentality



# Questions?

when your lecturer asks if you have any questions



DAILY CHALLENGE -> BYODB



# Take Home Challenge

