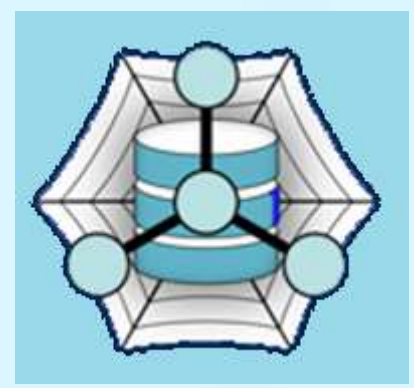
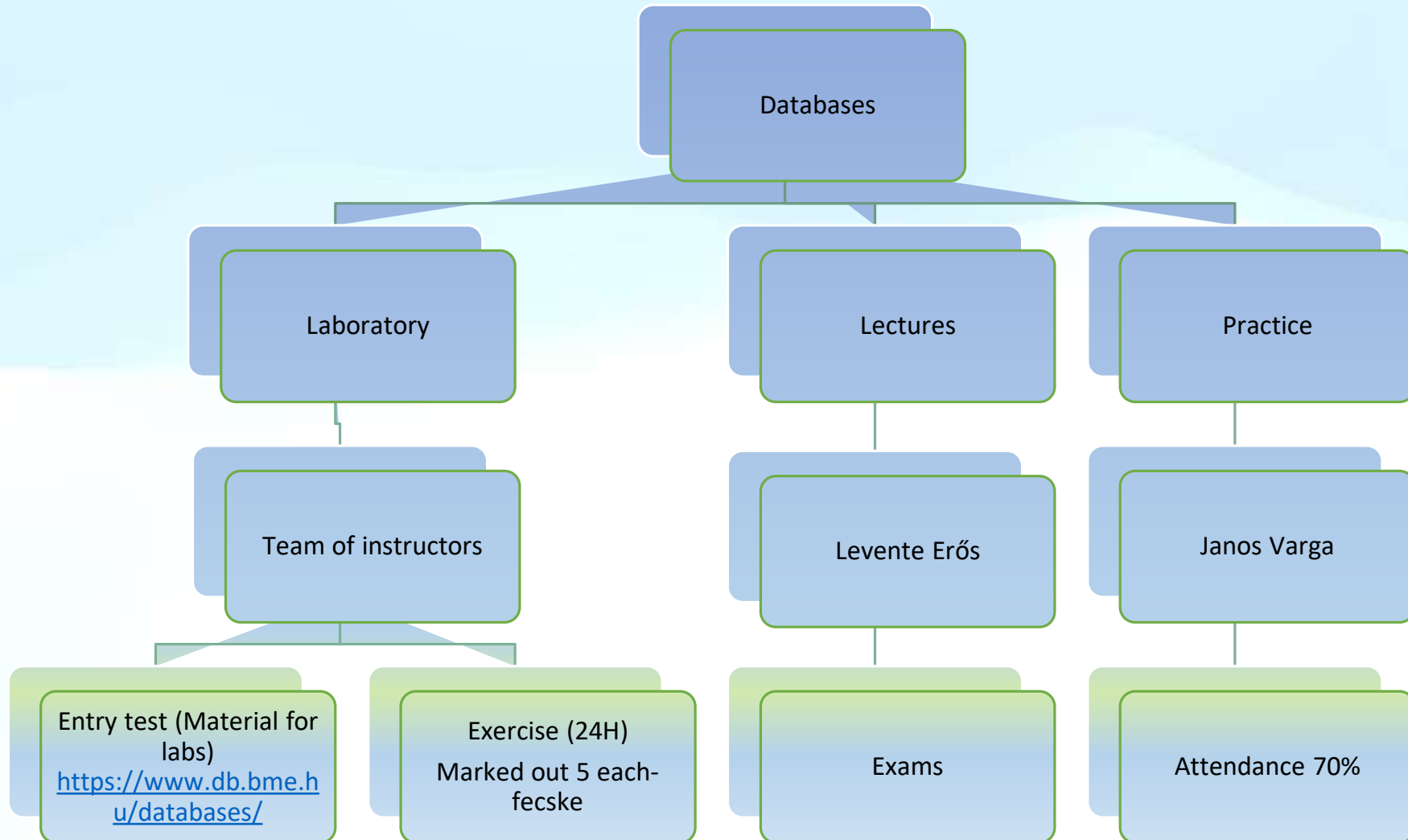


Databases



- Levente Erős – eros@db.bme.hu
- Ruba AlMahasneh - mahasnehr@tmit.bme.hu

Databases



- 1 Test
- Labs 5 Marks (averaged)
- If you participate the practices
- Then you have the **signature**
- Then you can sit for the **Final exam**
- <https://www.db.bme.hu/databases/>

What is a Database

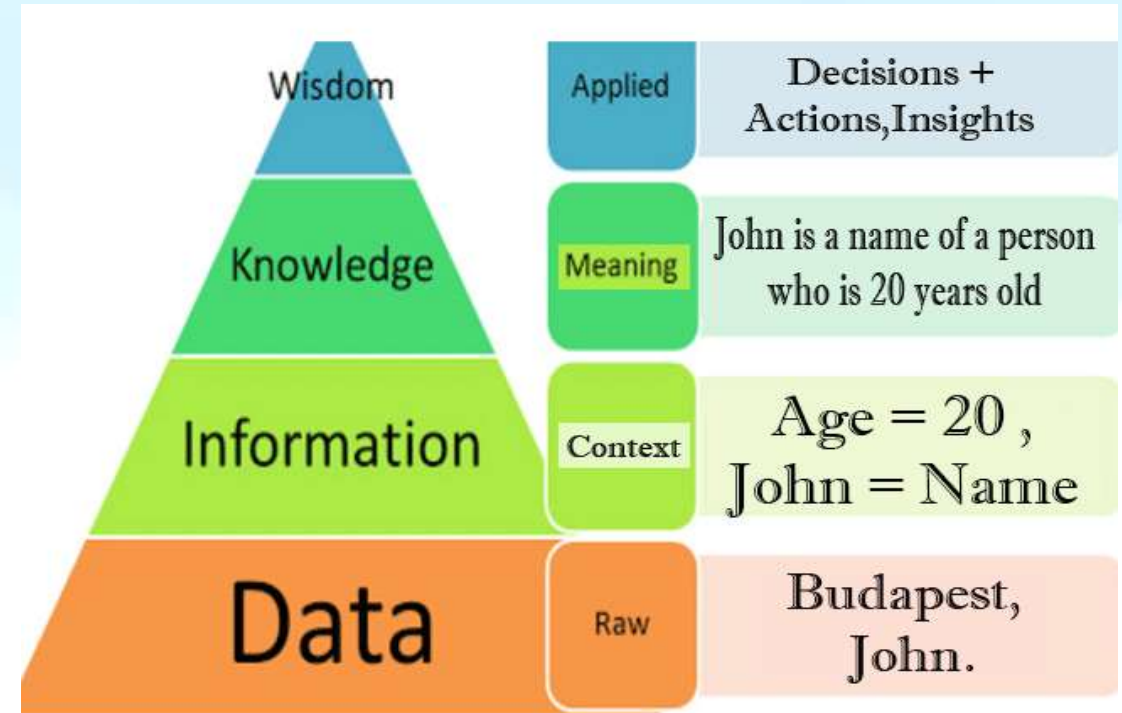
- **A Database:** is a *structured* collection of *knowledge* on a *subset* of the real world, typically stored electronically in a computer system. A database is usually *controlled* by a database management system (DBMS).
- Database management system (DBMS) is an important type of programming system, used today on the biggest and the smallest computers. DBMS allows us to *access* and *control* the data, find relationships, patterns ..etc
 1. Storing data in a structured way
 2. Providing means for accessing data

Which of these are considered Databases, explain?

- Data accessed through the internet
- Word documents
- Shopping list
- Phone book
- Excel sheet
- Can you think of other examples?
- How does the *structuredness* of the data affect accessing and extracting information from the database? What about the tools' complexity (sophistication level) ?

Data VS Information VS Knowledge

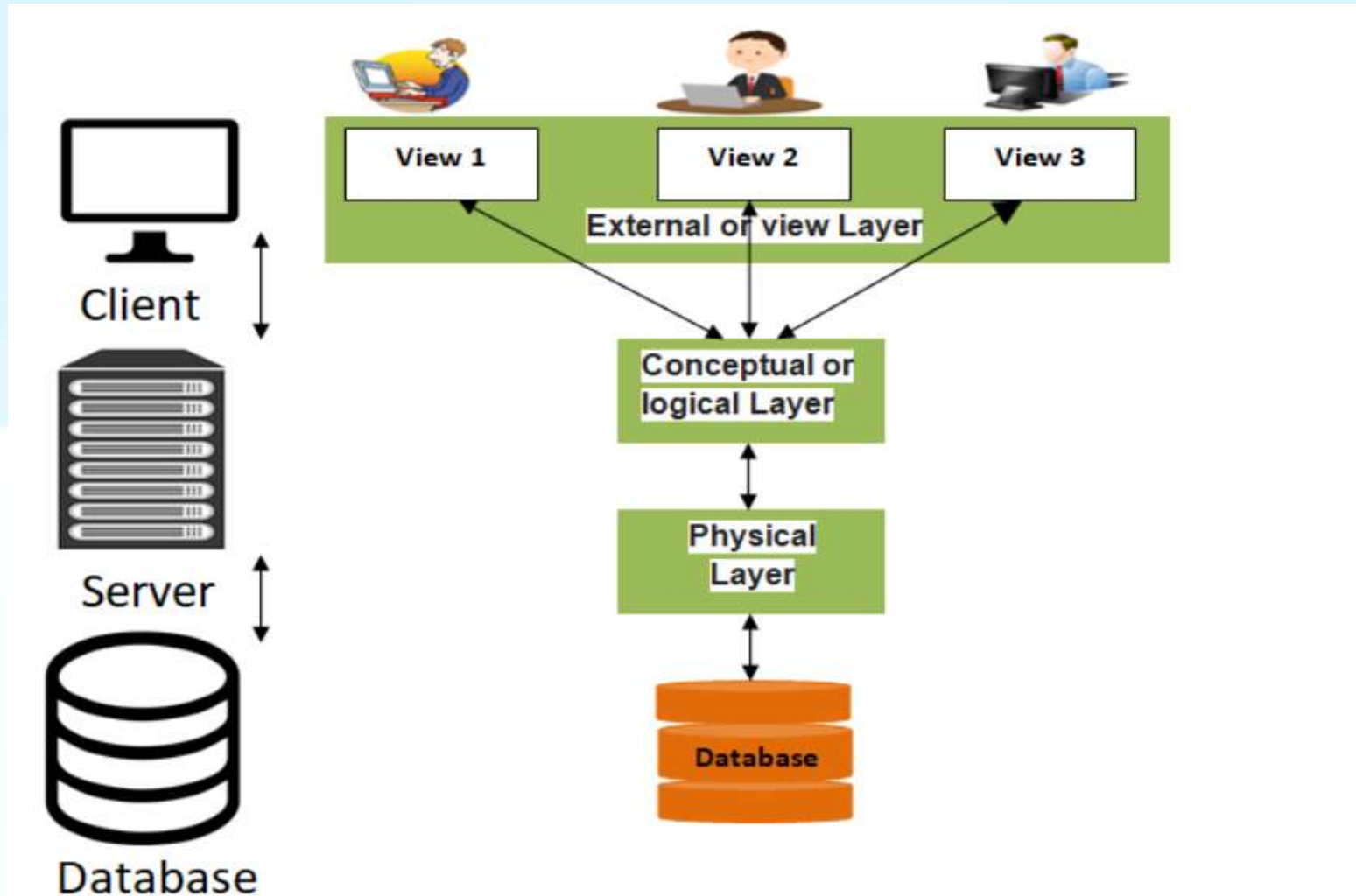
- *Data <> information*
- **Data** : does not have semantic or meaning yet [can be meaningless], 20, John, Budapest.
- **Information**: has idea behind it [processed data]. If we added context to it. Age = 20 , John = name, current location = Budapest
- **Knowledge**: John is a name of a person who is 20 years old. [*pieces of information related to each other => Row of a table*]
- We search for **knowledge** via DBMS. *Finding relations between pieces of information that represent the real world.*
- What other *knowledge* we can extract from this context?



DBMS's capabilities

- a) Support for at least one data model, or mathematical abstraction through which the user can view the data.
- b) Support for certain high-level languages that allow the user to define the structure of data, access data, and manipulate data.
- c) Transaction management, the capability to provide correct, concurrent access to the database by many users at once.
- d) Access control, the ability to limit access to data by unauthorized users, and the ability to check the validity of data.
- e) Resiliency, the ability to recover from system failures without losing data.

Database Architecture



Database Architecture

- **External Data Model:** This is **highest level** architecture and closest to the user. The user doesn't need to know the database schema details such as data structure, table definition etc.
- **Logical Data Model:** The purpose is to develop technical map of rules and data structures. → ER modeling [Relational Data Models, Relations, Tables, Queries]
- **Physical Data Model:** Data blocks, data files, indices for searching. **responsible for allocating space of the data**
- **Why it is important to have those separate layers?**
- **What if some damage happened to the hard drive and I had to replace it do I lose all the information I have on the other layers?**

Why are Models Used?

What is modeling? To model/map a subset of our world in which we want to store data.

1. Subset of the world, why is that beneficial ?
2. Simplify (less detailed, manageable)

Two of the modeling techniques are the ER (entity–relationship) modeling and Relational Data Modeling.

Data model is mathematical formalism with two parts:

1. Describe the data and the connections /relationships between the data elements
2. Define *operations* on data elements: *Accessibility* of information through that model (query,add,update..etc)

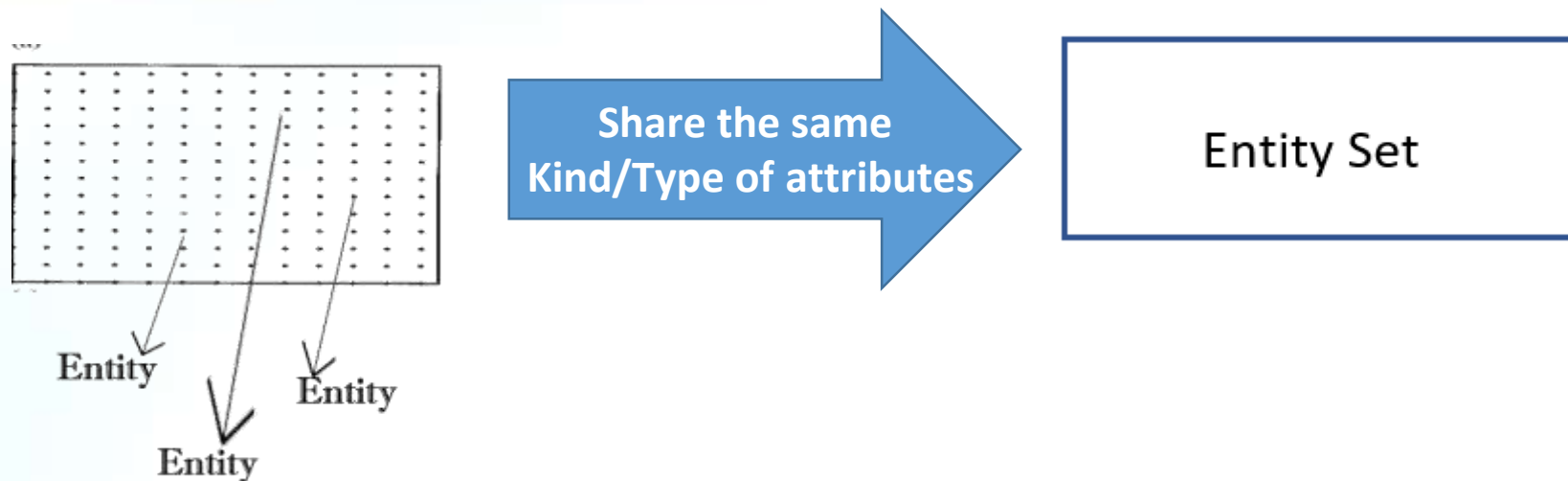
Note: MetaModel (Rules for creating a model)

The Entity-Relationship (ER) Model

- Description of the conceptual without the attention to efficiency or physical database design (segregation of duties).
- Real database systems are built depending on relational model (server as blueprints).
- Obtaining the conceptual scheme that offers the most efficiency can be quite difficult but ER models can facilitate this process. There are numerous good enough solutions but it's the engineers/architecture preference.

Entities

- **Entity**: a thing (physical or logical) that exists and is distinguishable from others (by distinctive features, properties = attributes)
- For example : each person is an *entity*. Each automobile is an *entity*. An ant is an entity if we had a way to distinguish one from another (say paint little numbers on their backs). We are interested in the data about entities → We store it in the database.

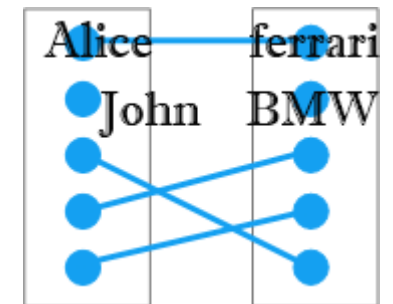
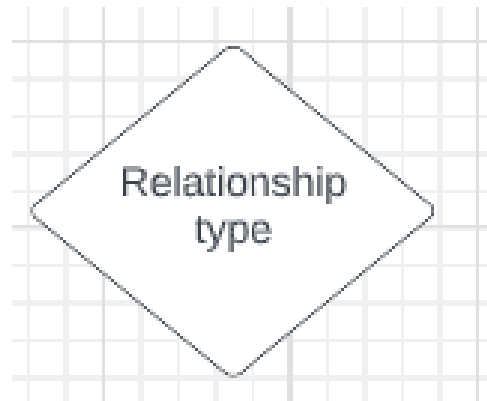


Entity Sets/Types

- A group consisting of all "similar" **entities** forms an **entity set**.
- Examples of entity sets are (all persons, all red-haired persons, all automobiles)
- Do all attributes have the same *values*?
- Example: The Hungarian's Department of Motor Vehicles may design its database scheme to have an entity set AUTOMOBILES. This includes all the automobiles presently registered in Hungary, not all automobiles in the world.
- Other examples you can think of ?

Relationship Sets/Types

- Diamonds represent the relationship sets defined among two or more entity sets. There are two Types of relationship sets (identifying/determining relationship and relationship).
- A relationship set among entity sets is an ordered list of entity sets.



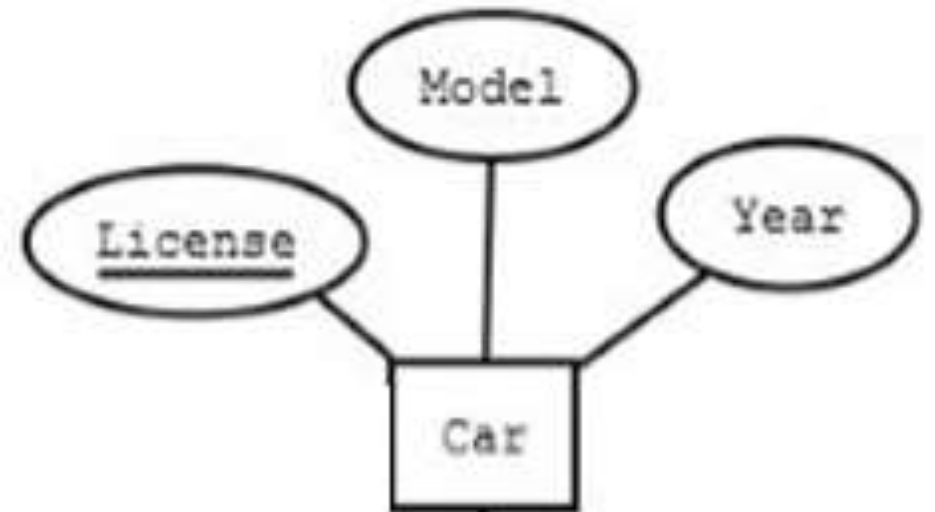
Attributes and Keys

- Each entity set has a set of attribute types associated with it.
- Each entity set must have a distinguishable key. **Why?**
- Entity sets have properties called attribute types. Each entity in the set is associated with a *value* from a domain of values for that attribute.
- Domain for an attribute usually a set of integers, real numbers, or character strings etc.
- An attribute set (or set of attributes) whose values uniquely identify each entity in an entity set is called a **key** for that entity set.



Attributes and Keys

- In the figure. How many entity sets do we have ? What are their attribute types?
- Do all cars have the same license plate (values)?
- **Question:** What are possible attribute types of the entity set *PERSONS* ? What is each attribute set's possible *domain*?



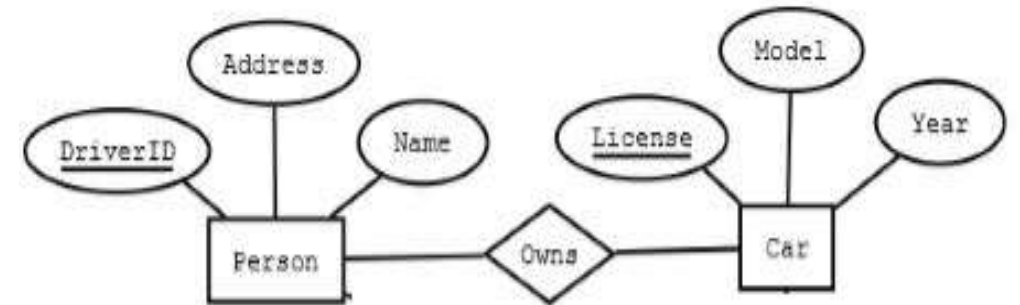
Entity-Relationship Diagram-ERD Elements- Chen Notation

Rectangles represent *entity sets*.

Circles represent *attributes*. They are linked to their entity sets by (undirected) edges.

Sometimes, attributes that are part of the key for their entity set will be underlined.

Diamonds represent *relationship sets*. They are linked to their constituent entity sets by edges, which can be undirected edges or directed edges.



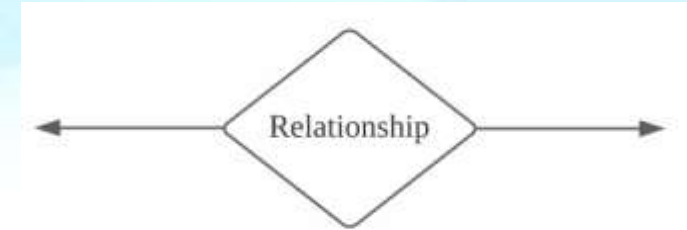
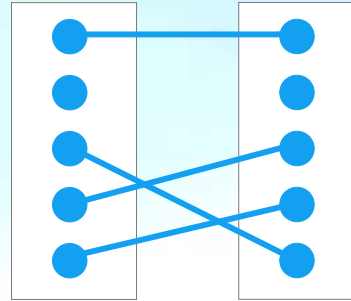
Relationship set's Cardinality-Chen Notation

- A relationship set where two entity sets are participating is called a *binary relationship type*
- Each relationship instance connects exactly two entities by a label (semantic).
- Cardinalities of **binary** relationship sets represents the maximum number of entities of an entity set A that can be assigned by the relationship set (type) to a single entity of the other entity set B.
- Why we used Binary? Is there other relationship types you can think?

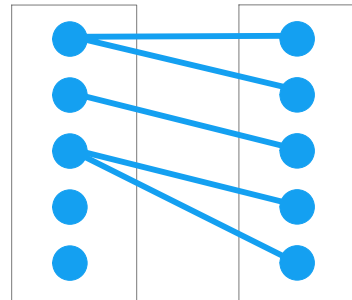
Relationship Set Cardinalities -Chen

Notation

- One-to-One (1:1) relationship set: Each entity of both sets can be connected to at most one entity of each other.



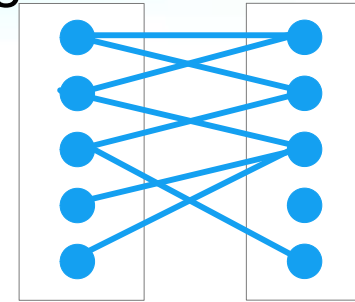
- One-to-Many (1: N) relationship set: Each entity of the second entity set can be connected to *at most* one entity of the first entity set. Each entity of the first entity set can be connected to an arbitrary number of entities of the second entity set can be connected to at most one entity of each other.



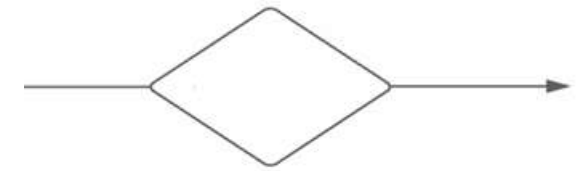
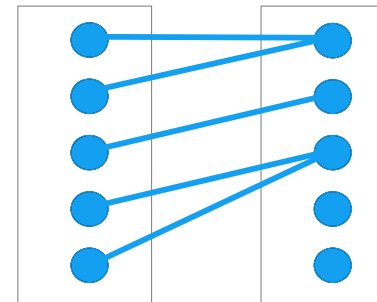
Relationship Set Cardinalities-Chen Notation

- Many-to-Many (M: N) relationship set: Each entity of both entity sets can be connected to an arbitrary number of entities of each the other entity set.

Ex: There are two parents for each child, and a given individual *may* have any number of children.



- Many-to-One (M: 1) relationship set: one entity in entity set $E2$ is associated with zero or more entities in entity set $E1$, but each entity in entity set $E1$ is associated with *at most* one entity in entity set $E2$.

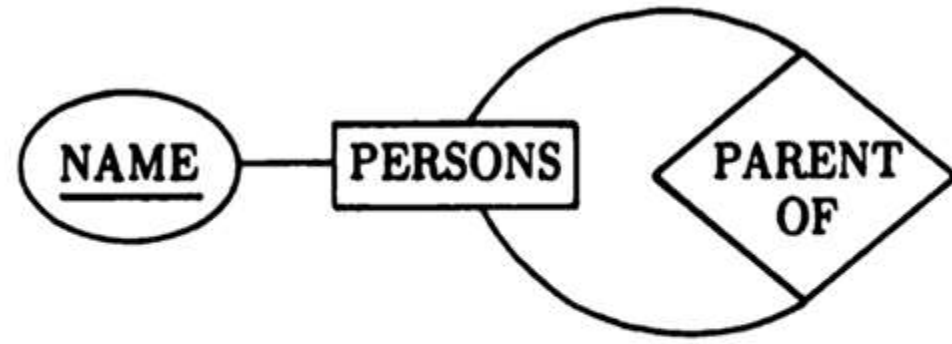


Quiz

Suppose we have an entity set PERSONS and PARENT OF relationship set

What is the possible relationship?

What are the entity sets?



M-M Relationship

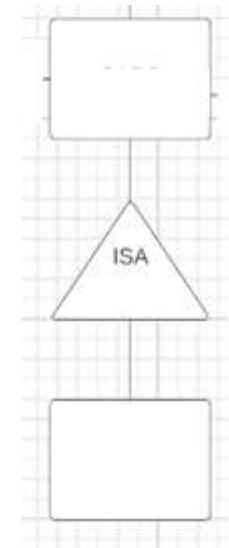
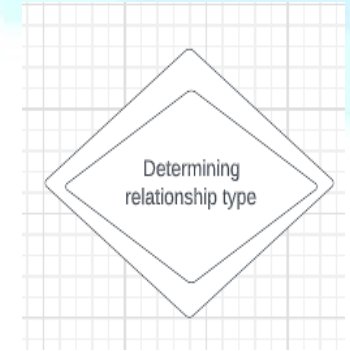
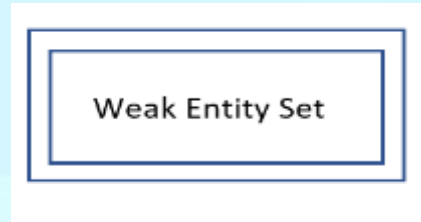
Question: Apply this concept on relationship of *enrolment* between *courses* and *students*.

Example

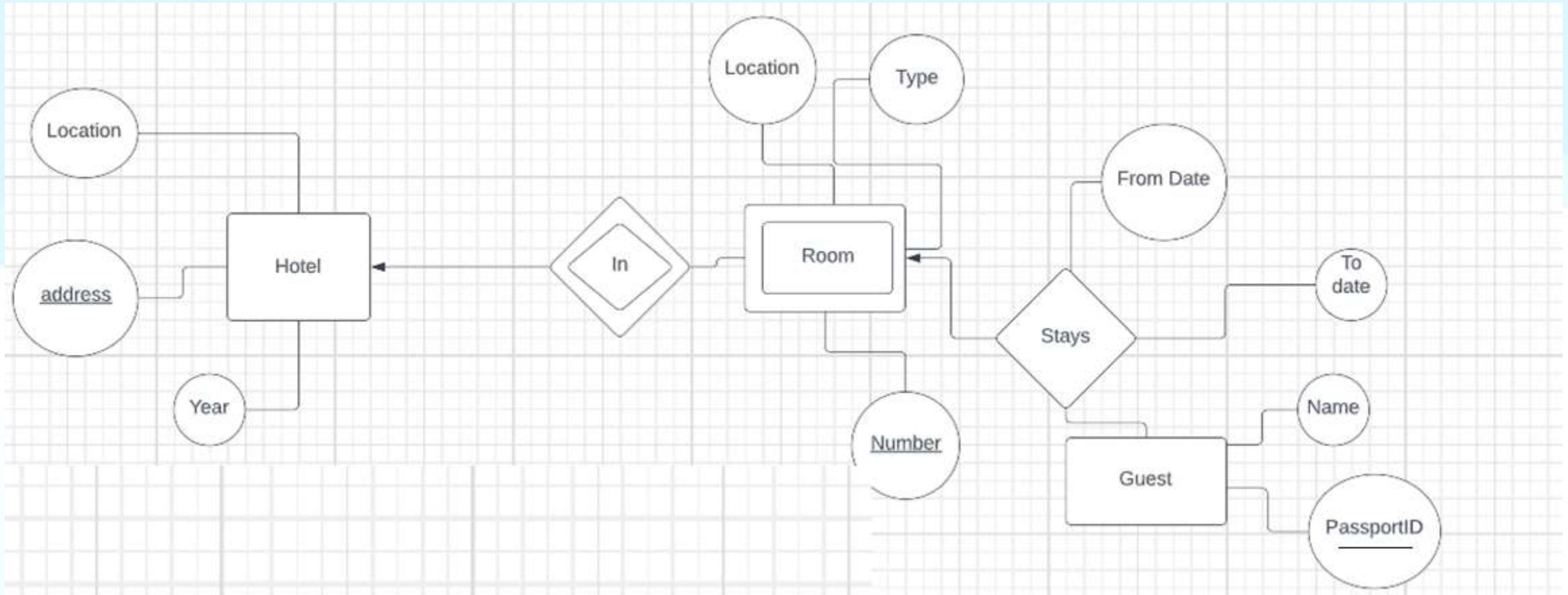
- Suppose we want to represent the relationship between Person(s) and his Car(s)
- Draw the ER diagram ?
- Define the attributes for Persons ?
- Define the attributes for Cars?
- What other entity sets you might think of that are related to this Diagram?

Extended ER Modeling

- Weak entity set
- “Strong” Entity set
- Determining relationship type
- Specialization (**Isa** Hierarchies) - specialized entity sets



Example-Weak Entity Set

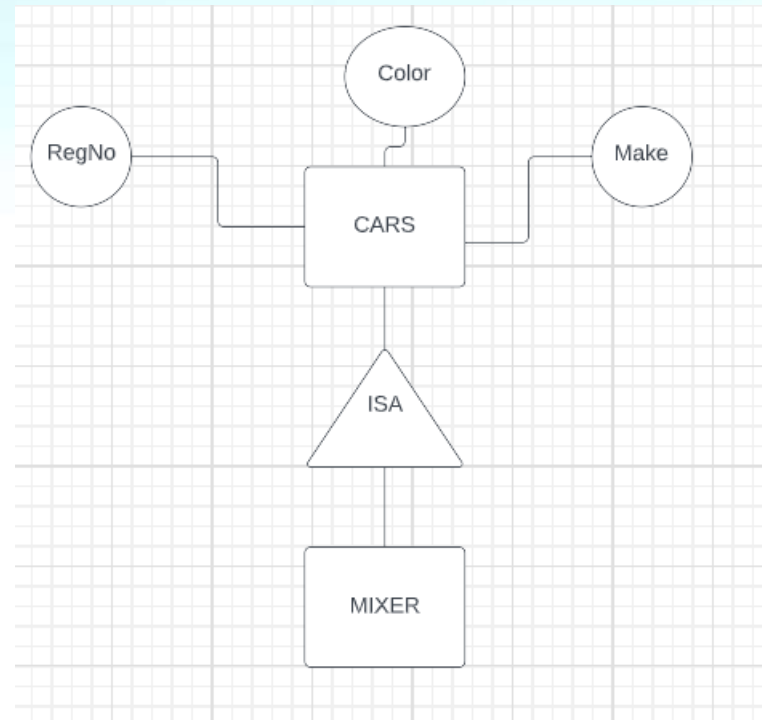


Example-Weak Entity Set

- In the previous example what are the key attribute sets for the weak entity ? (Rooms)
- Can the Room# by itself serve as key attribute set? Why?
- What do we call those key attribute sets that the weak entity set borrow from the strong entity set?
- What do we call the relationship set (that has double border line diamond shape) that connects the weak entity set with the strong entity set?

Isa Hierarchies (Specialization)

- A **isa** B , read " A is a B " if entity set B is a generalization of entity set A , or equivalently, A is a special kind of B .

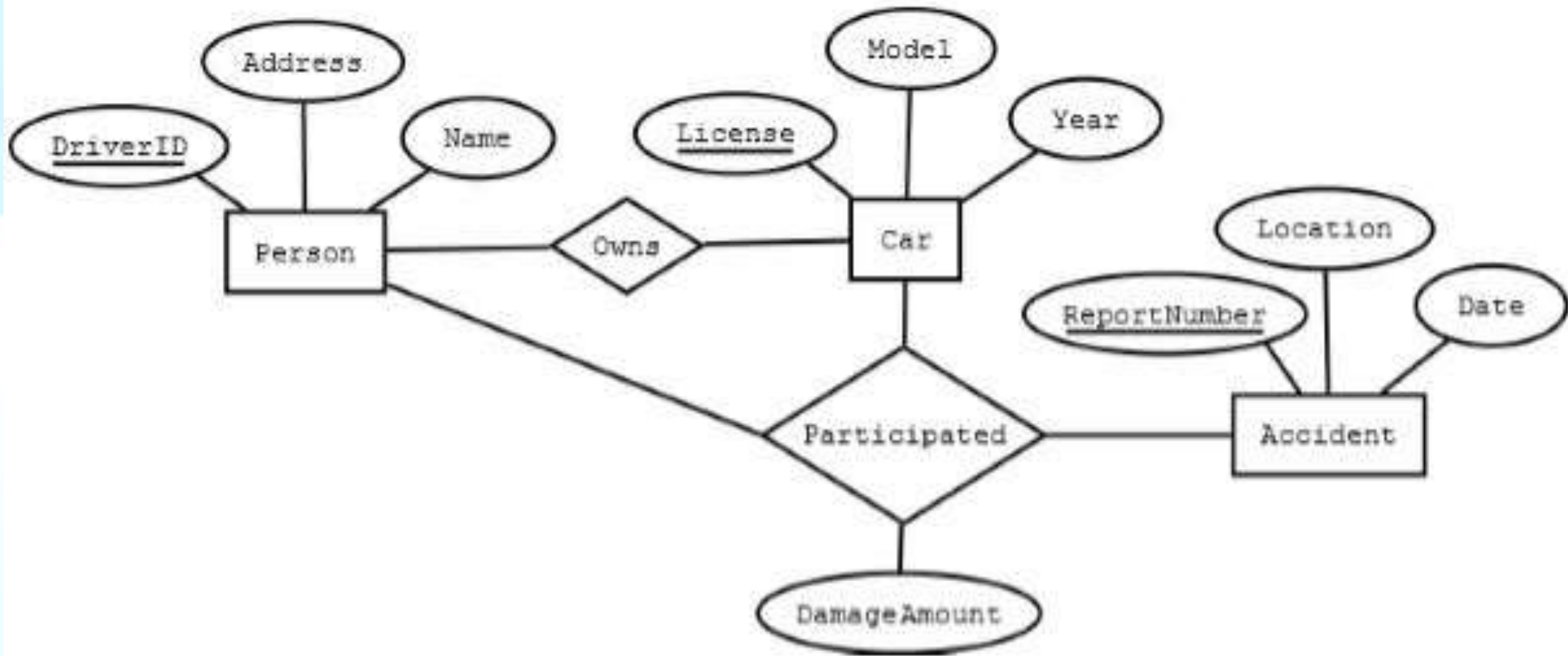


Borrowed Key Attribute Sets

Example: A corporation might well have an entity set EMPLOYEES, with attributes such as ID_NO, NAME, and SALARY. If this corporation was a baseball team, some employees are the players, would have other important attributes, like BATTING-AVG or HOME-RUNS, BUT the other employees would not have. How can we design the entity sets (for PLAYERS and EMPLOYEES)?

- what is the Key attribute (that will uniquely identify any player) of the Entity Set PLAYERS?
- Does this entity set have its own key or it borrows it from EMPLOYEES Entity set? Explain?

Arity- Relationship Sets Degree



Brain teaser

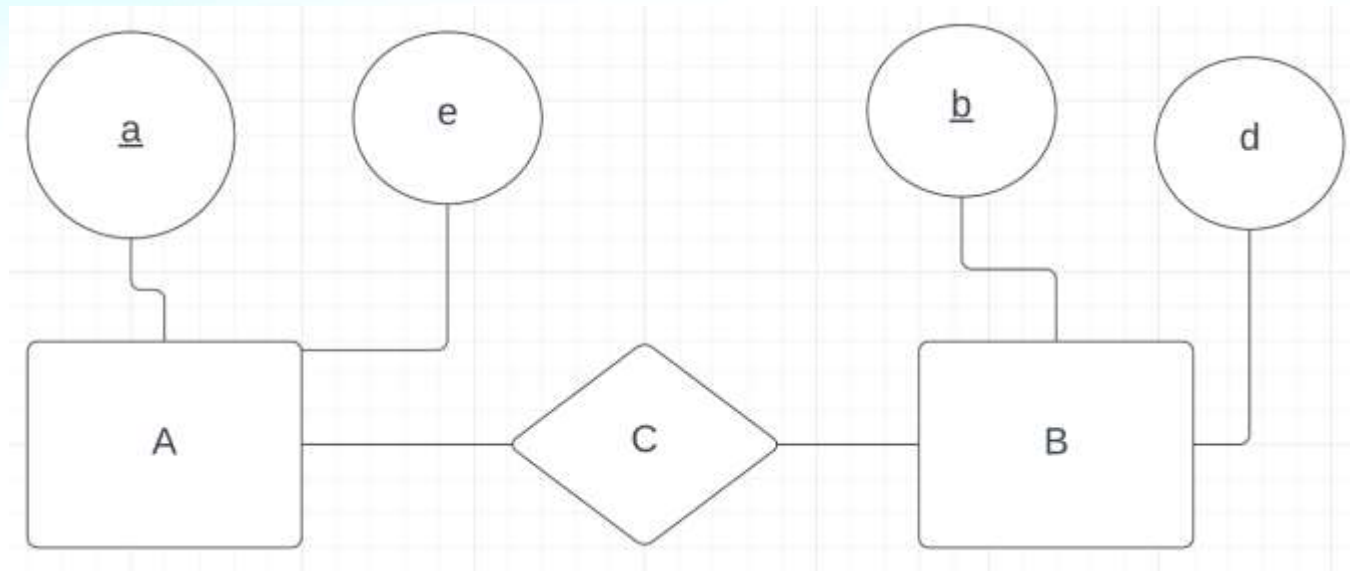
Why ER looked at as a half-data model?

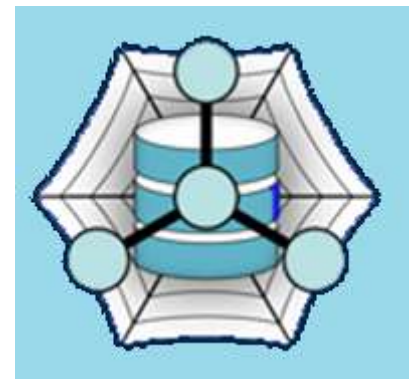
Example-Enhance this ER Model?

- What are the Entity sets in the diagram ?
- What is the relationship set between DEPTS and MANAGERS?
- What is the relationship set between DEPTS and EMPS?
- Name each Entity Set's Key attribute.

Transcribing Relationship types

- Relationship instances are identified by the connected entities
- A weak entity set is identified by its key (if any) and the key(s) of the determining connected entity sets.





Thank You

- Levente Erős – eros@db.bme.hu
- Ruba AlMahasneh - mahasnehr@tmit.bme.hu