Databases

Lecture 3 - Entity-Relationship-Model

Emily Lucia Antosch

HAW Hamburg

23.02.2025

Contents

1. Introduction

1.1 Where are we right now?

- Last time, we looked at SQL as the language in which we define our database.
- We learnt about different database objects and how they can help us achieve our business requirements.
- Today, we'll look at
  - what an ERM (Entity-Relationship-Model) is,
  - how we can use it to effectively conceptually design databases and
  - why conceptually designing a database prior to implementation can save us a lot of headache.

1.1 Where are we right now?

1. Introduction
2. Basics
3. SQL
4. **Entity-Relationship-Model**
5. Relationships
6. Constraints
7. More SQL
8. Subqueries & Views
9. Transactions
10. Database Applications
11. Integrity, Trigger & Security

1.2 What is the goal of this chapter?

- At the end of this lesson, you should be able to
  - design a database using the ER-model,
  - decide about which attributes, constraints and relations will help you achieve your requirements.

2. Entity-Relationship-Model

What is an ERM

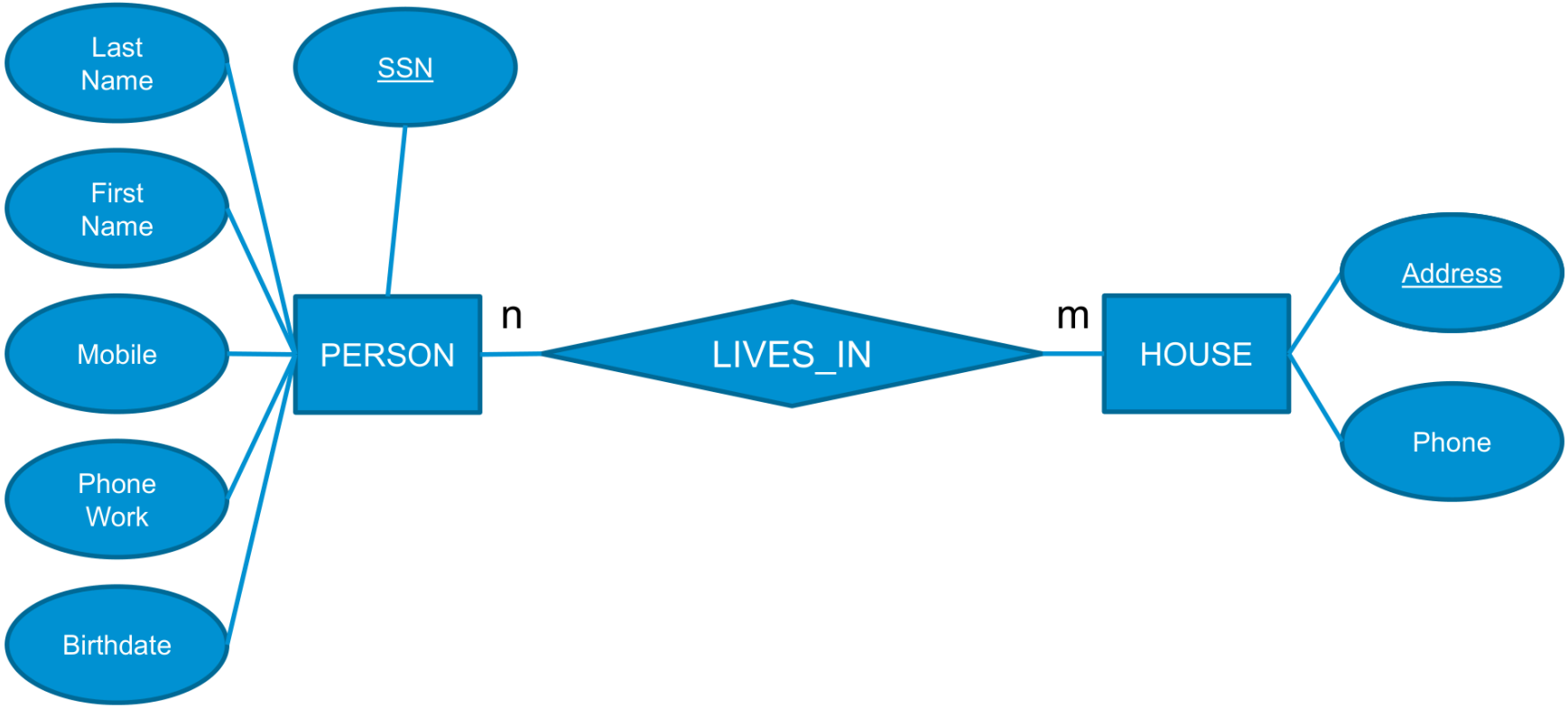
- Entity-Relationship-Model is model/diagram for the logical draft of the database
- The focus is on the business requirements
- This language is not implemented in any DBMS

A quick history of the ERM

- Introduced by Peter Chen in 1976.
- An ERM describes interrelated things of interest in a specific domain of knowledge.
- A basic ERM is composed of entity types (which classify the things of interest) and specifies relationships that can exist between entities (instances of those entity types).
- Elements:
  - Entity: A distinguishable thing existing in the real world.
  - Relationship: Between entities.
  - Attribute: Property of an entity or relationship.

2.1 Basics

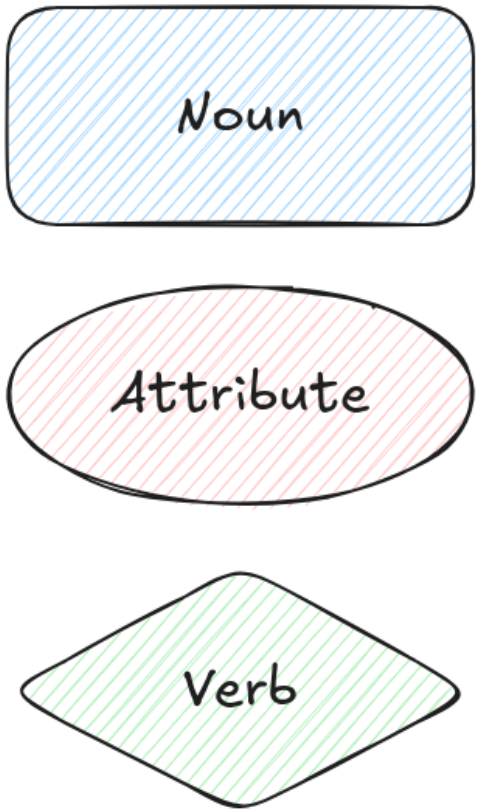
Conceptual Design with ERM



2.1 Basics

Conceptual Design with ERM

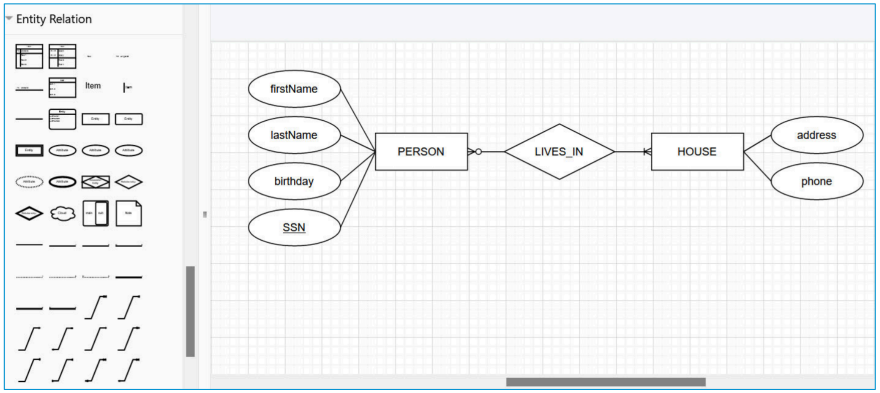
- Entity Type
  - Represented as a rectangle
  - Singular Noun
- Attribute Type
  - Represented as ovals
  - Noun
- Relationship Type
  - Represented as diamond
  - Always between entities
  - Verb & has cardinalities



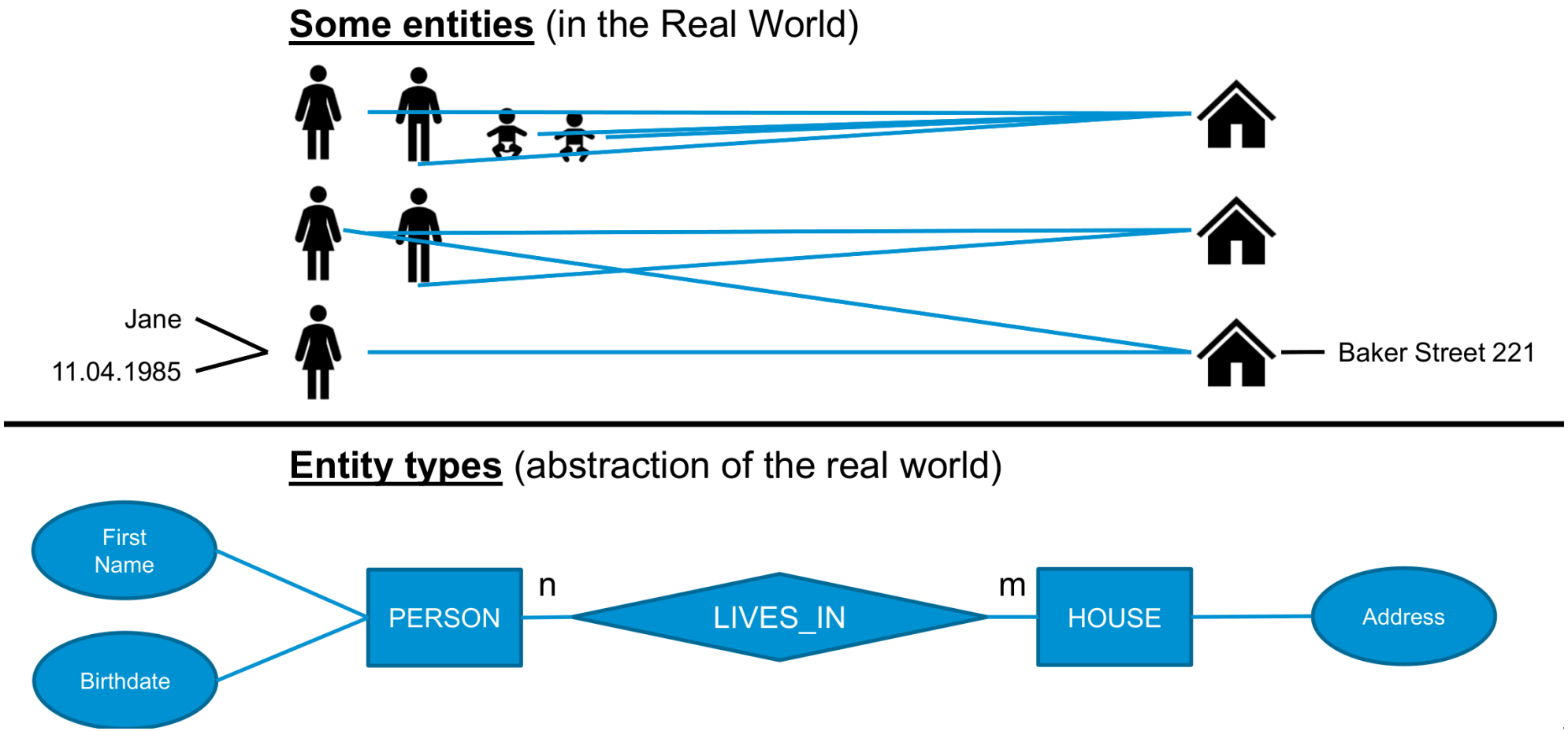
2.1 Basics

Online-Tools for ERM

- Creating ERMs can be done by using any drawing tool or just a piece of paper and a pen.
- Examples of drawing tools:
  - Excalidraw (Recommended)
  - Draw.io
  - Lucidchart
  - Creately



Entity Abstraction



Entity Abstraction

! Memorize

- An **entity** is a distinguishable thing that exists in the real world.

- An abstraction of entities would be an **entity type** (comparable to classes in OOP)
- Several entities make up an **entity set**
- An abstraction of relationships is called a **relationship type**

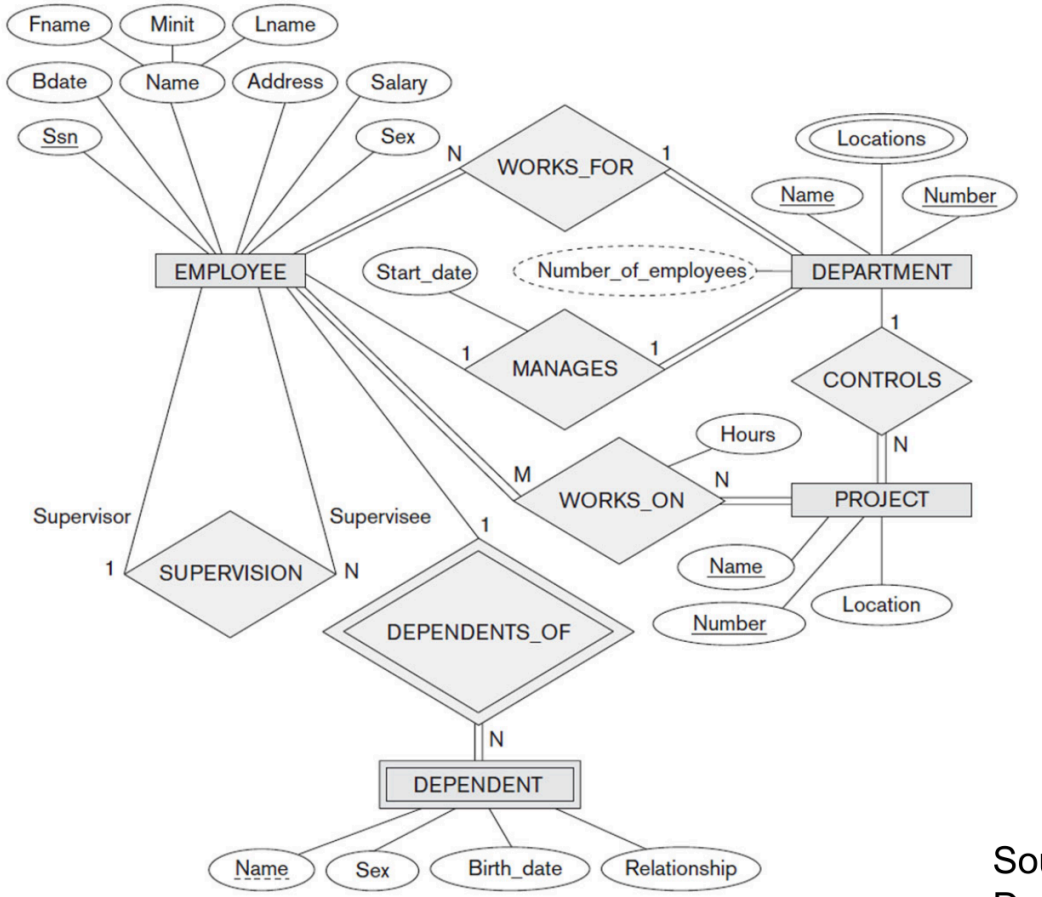
Entity Abstraction

Example

Imagine a company:

- A company is made-up of departments and each department has a unique name, a number and a manager.
- Each employee's name, social security number, address, salary and birth date is stored within our database.
- We also want to keep track of the hours per week per project, keep track of the supervisor.

ERM: Company Example



Source: Elmasri, Fundamentals of Database Systems, Page 204 ff

Entity Type

- An **entity type** is a basic object in an ERM.
- Represents a thing in the real world, like a car, a job or a person.
- An entity type has attributes, such as a name, an address or an age.
- A particular entity of that type will have values for each of these attributes.

Entity Type

! Memorize

- An entity type therefore defines a collection of entities, that have the same attributes.
- Each entity type can be defined by its name and its attributes.
- The collection of all entities of a particular entity type, so all the instances of this entity type, is called an **entity set**.

ERM: Entity Example

- Categories for entities could be
  - actual physical objects, people, roles, organizations,
  - actions, interfaces or general information
- An element is not an entity type
  - if it has neither attributes nor relationships,
  - only contains attributes that another entity type already has

? Question

What is a good name for an entity type?

ERM: Entity Example

Task 1

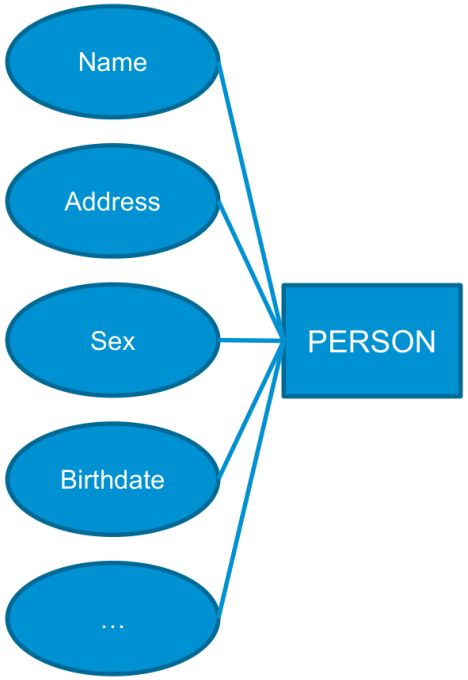
What are the entity types in the following examples?

- A company is organized in departments.
- Departments have a unique name, a unique number, a manager.
- A department oversees a number of projects, each with a name and a number.
- The company may store information about each employee like their name, their social security number and their salary.

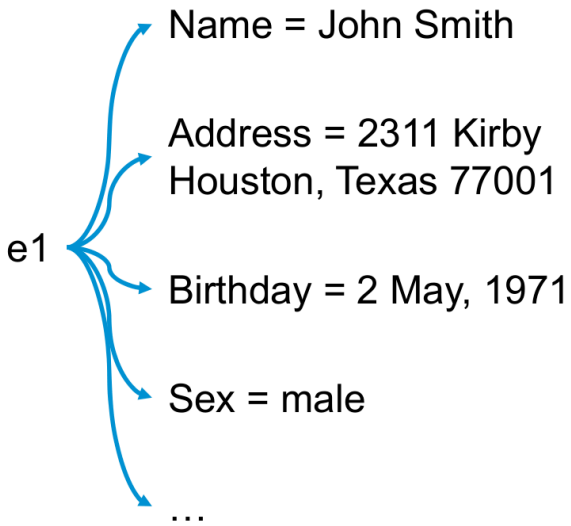
2.1 Basics

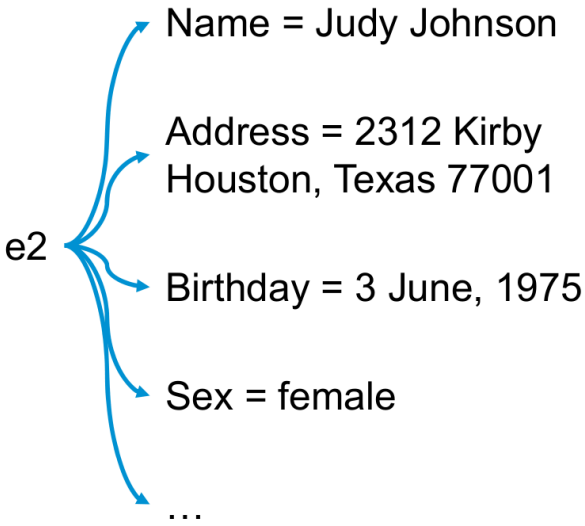
Entity Type

Entity Type



Entities





Source: Elmasri, Fundamentals of Database Systems, Page 204 ff

Attributes

- Is the attribute relevant to the problem you are trying to solve?
- An attribute must belong to an entity type (or a relationship type).
- Some of the attributes of an entity are important in identifying the entity. These are called **key attributes**.
- A good name for an attribute is unique within the entity type, but not necessarily across the entire model.

Attributes

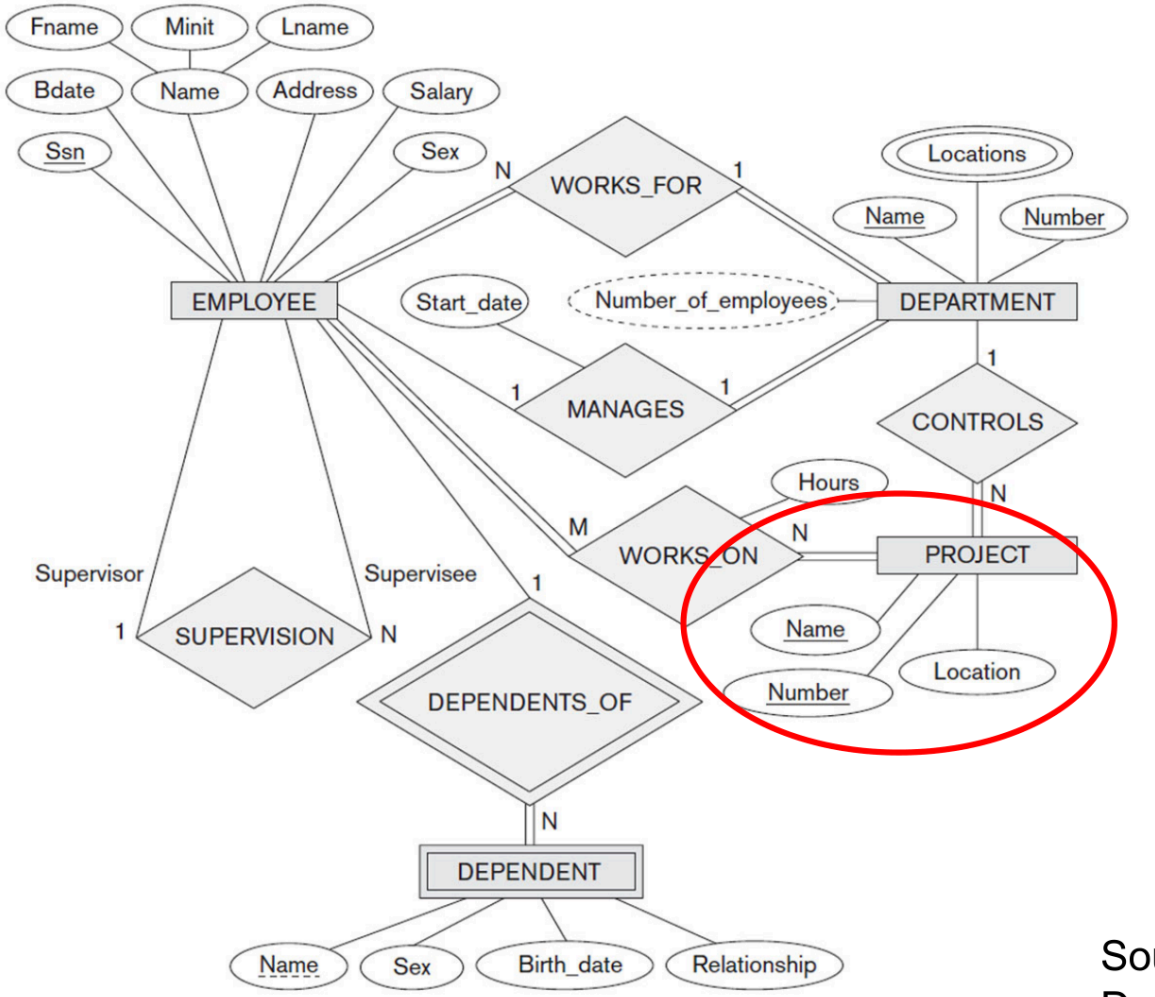
Task 2

When you look at the attributes of the entity project, what could be identifying or key attributes?

- A department controls a number of projects, each with a unique name, unique number and a single location.

2.1 Basics

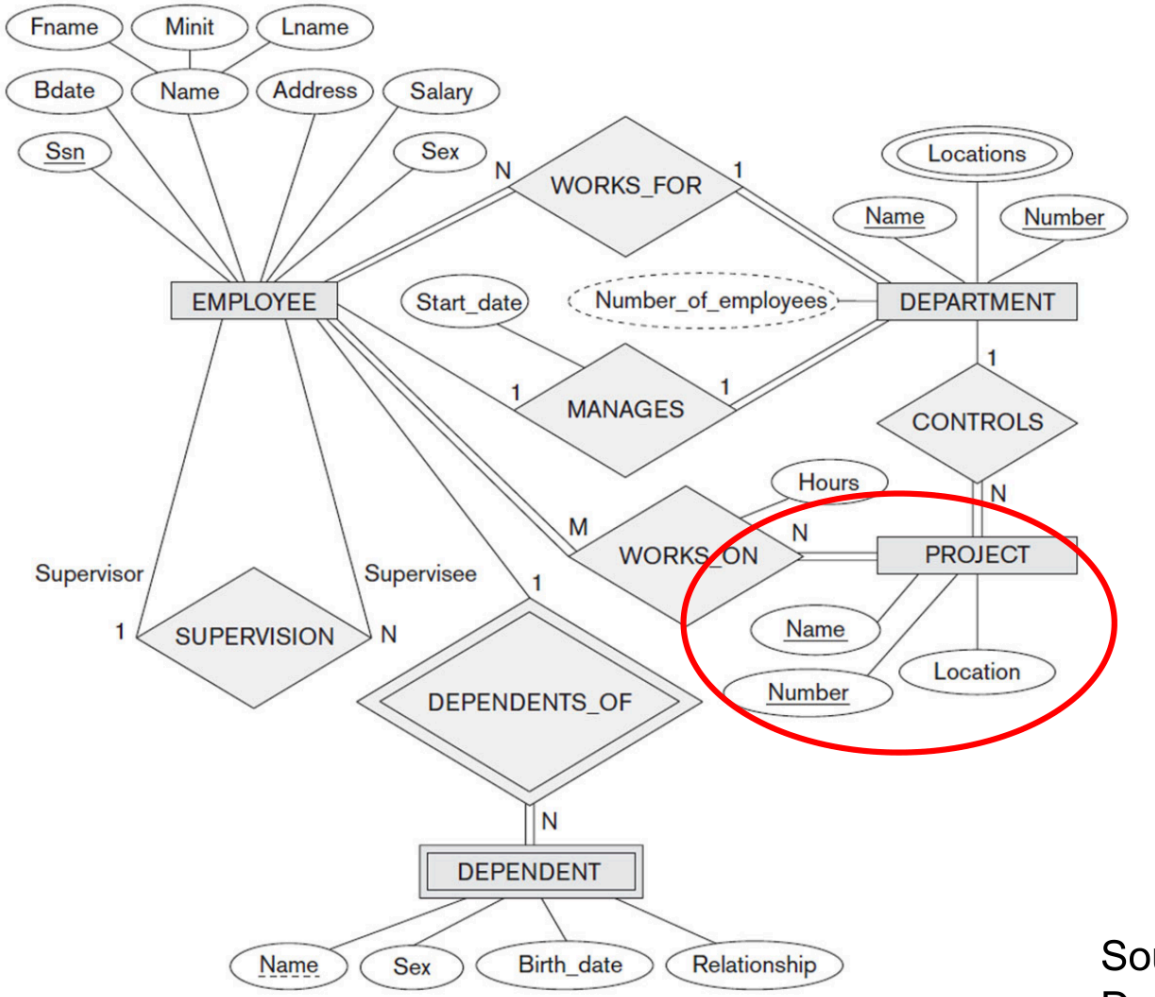
Attributes



Source: Elmasri, Fundamentals of Database Systems, Page 204 ff

2.1 Basics

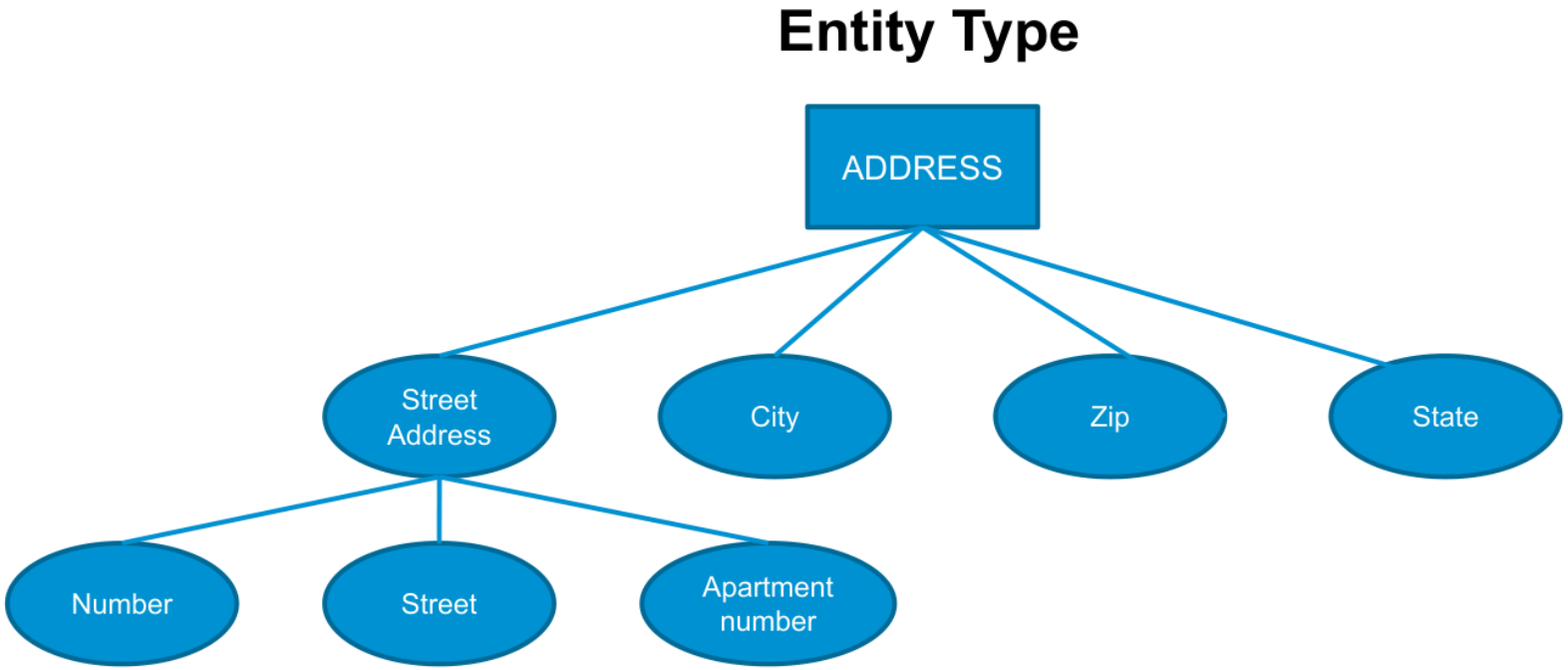
Attributes



Source: Elmasri, Fundamentals of Database Systems, Page 204 ff

- Composite vs. Simple (atomic) attributes
  - Attributes which are not divisible are called simple or atomic attributes
  - Composite attributes can form a hierarchy
  - Composite attributes are useful to model situations in which a user sometimes refers to the composite attribute as a unit but at other times refers specifically to its components
  - If the composite attribute is referenced only as a whole, there is no need to subdivide it into component attributes
  - Composite attributes are attached to their component attributes by straight lines

Attributes in Entity Types



Source: Elmasri, Fundamentals of Database Systems, Page 204 ff

Key Attributes in entity sets

- How can we identify an actual entity within an entity set?
- Attributes must be used → Key Attributes (also called identifying attributes)
- Sometimes several attributes together form a key attribute (identifying attribute), meaning that the combination of the attribute values must be distinct for each entity
  - If a set of attributes possesses this property, the proper way to represent this in the ER model that is to define a composite attribute and designate it as a key attribute of the entity type

- Notice that such a composite key attributes must be minimal; that is, all component attributes must be included in the composite attribute to have the uniqueness property

- Key attributes are underlined
- If two attributes are underlined separately, then each is an identifying attribute on its own

Key Attributes in entity sets

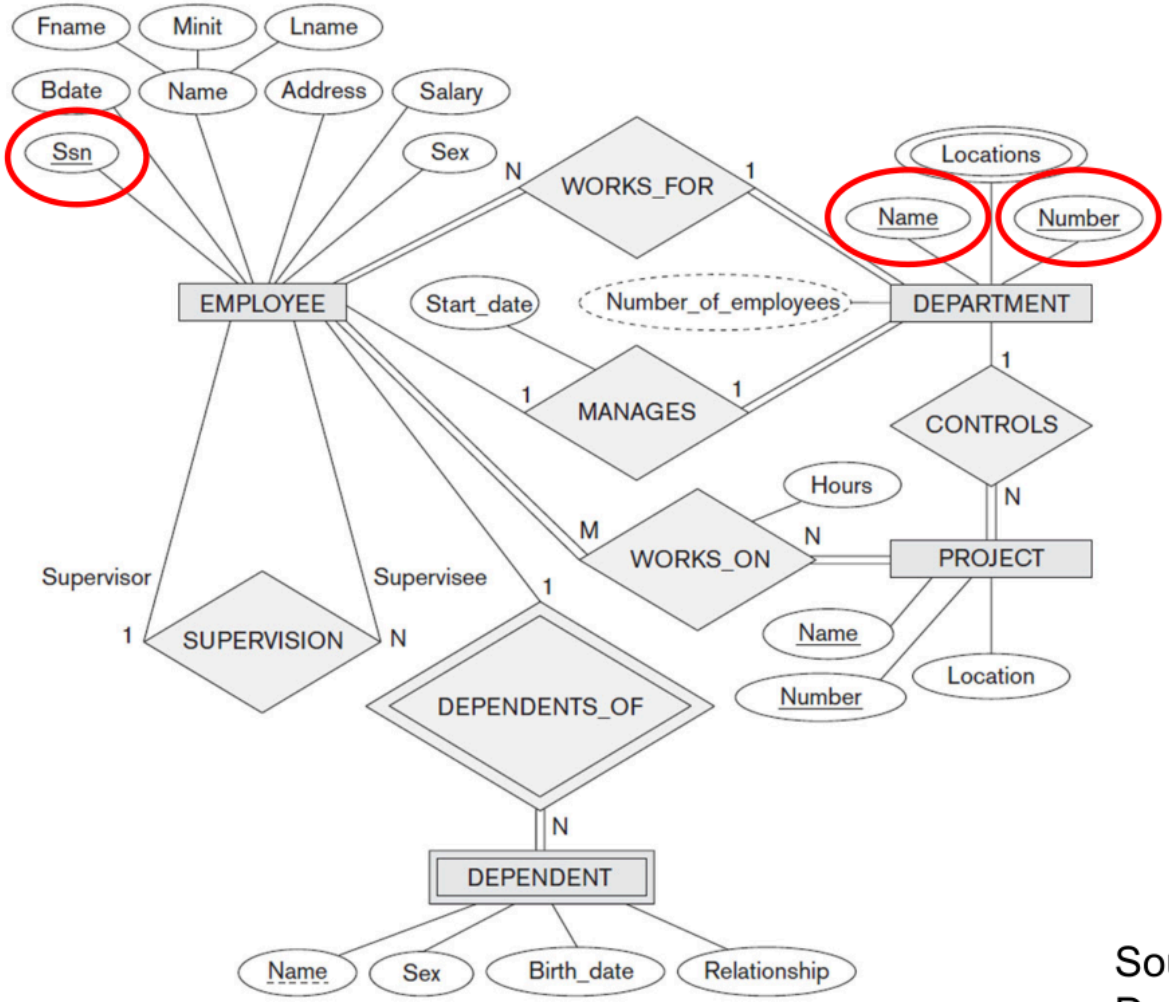
Task 3

What are key attributes for entity type `EMPLOYEE` and `DEPARTMENT`?

- A company is organized in departments.
- Departments have a unique name, a unique number, a manager.
- A department oversees a number of projects, each with a name and a number.

- The company may store information about each employee like their name, their social security number and their salary.

Key attributes in entity sets

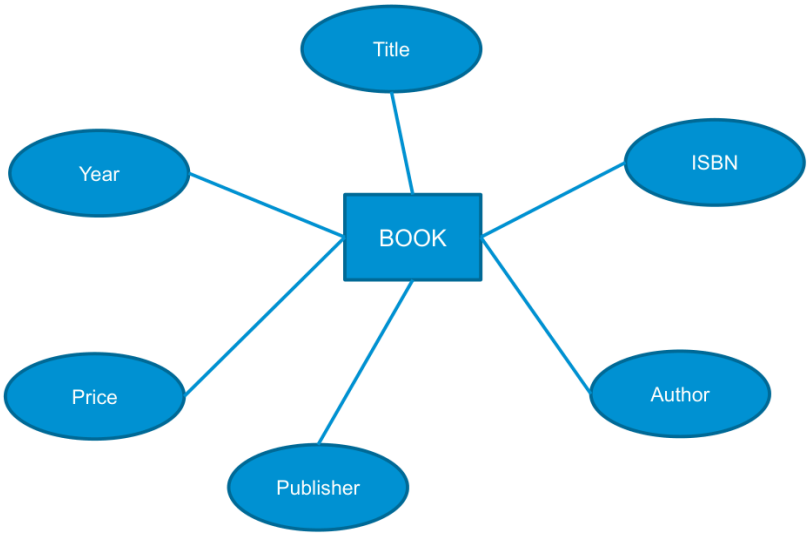


Source: Elmasri, Fundamentals of Database Systems, Page 204 ff

Key attributes in entity sets

Task 4

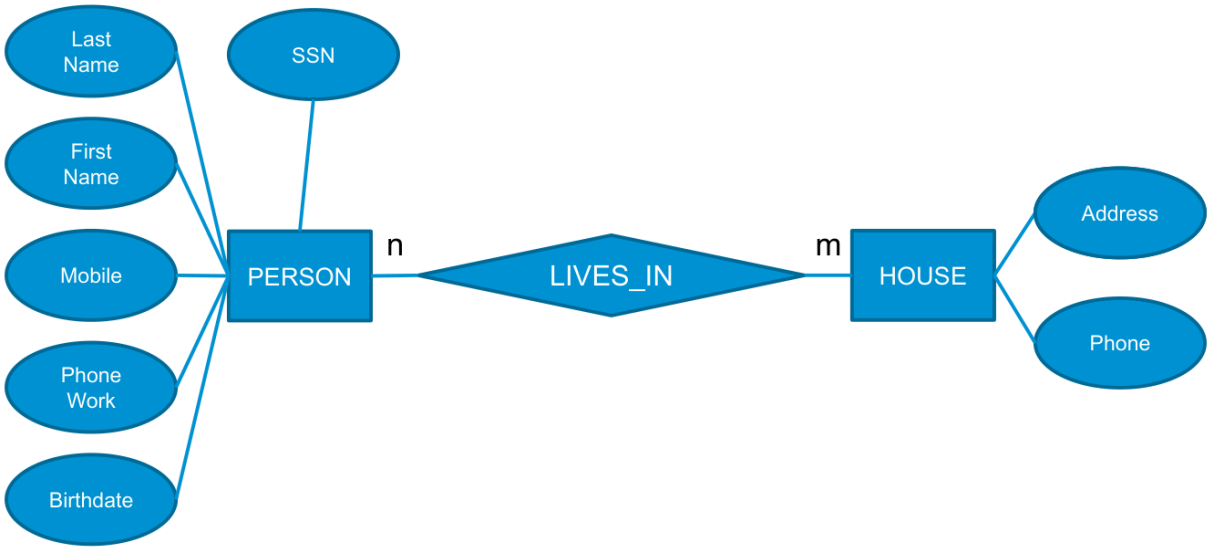
What are key attributes for BOOK?



Key attributes in entity sets

Task 5

What are key attributes for PERSON and HOUSE?



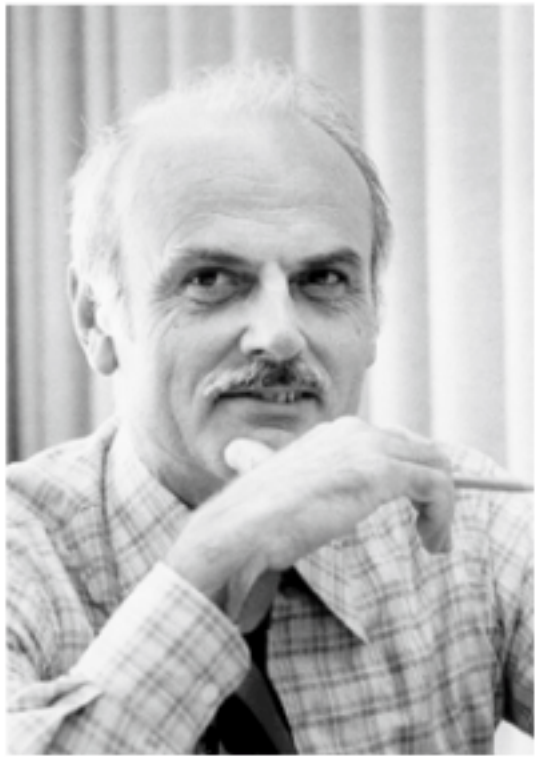
3. The relational model

3.1 What is the relational model

RM: A quick history

3.1 What is the relational model

- Edgar F. Codd invented the relational model in 1970 and won the Turing price for it.
- The model has become widely accepted.
- The model is based on relations, that are subset of the Cartesian product.
- Everything is modelled in tables.



Source: www.wikipedia.org

3.1 What is the relational model

Name	Matr_no	Term
John Meyer	123456	2
Judy Fisher	234567	4
William Smith	345678	3

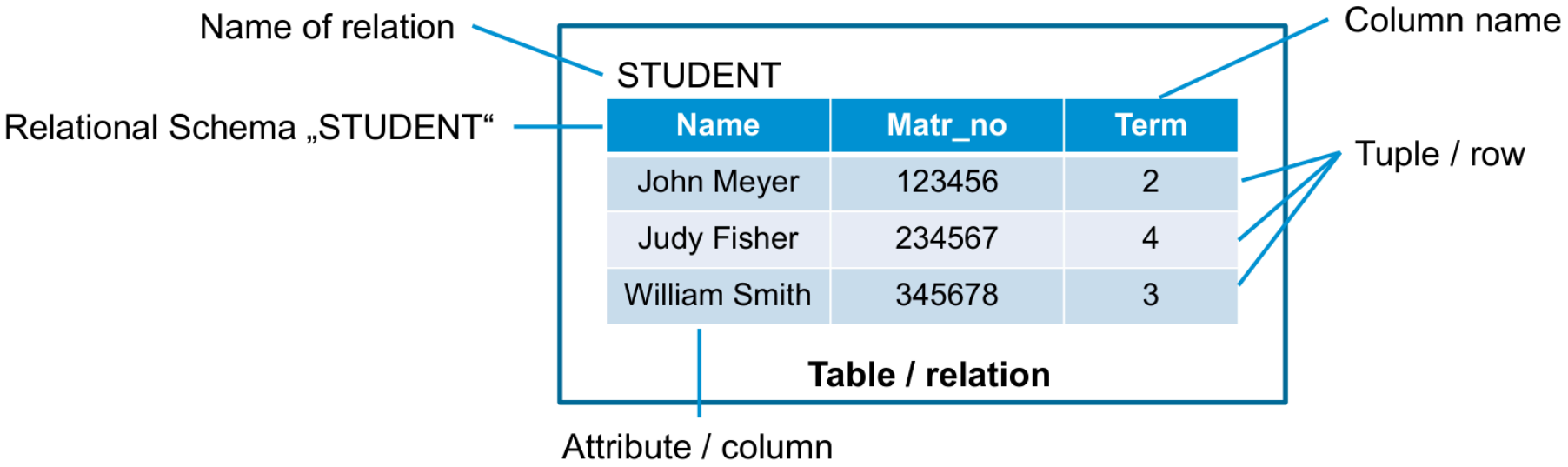
3.1 What is the relational model

RM: The model

- The relational schema describes objects and relationships as a relational schema.
- A relational schema consists of a set of attributes
- Each attribute belongs to a value range/type
- A database schema consists of a set of relational schemas
- A relation displays the current data for the relational schema
- The set of relations is called the database (or the state of the DB)
- An element of a relation is called a tuple, which is simply a row

3.1 What is the relational model

RM: The model



Attribute	Type
Name	String
Matr_no	Integer
Term	Integer

3.1 What is the relational model

RM: The model

Name of relation

Relational Schema „STUDENT“

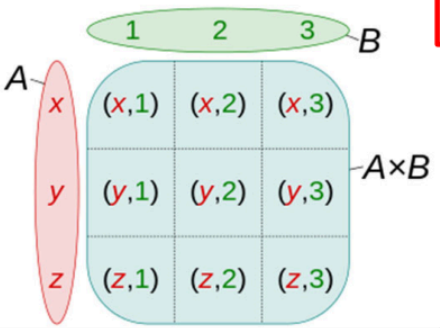
Column name

Tuple / row

STUDENT

Name	Matr_no	Term
John Meyer	123456	2
Judy Fisher	234567	4

A table can be created by taking the Cartesian product of a set of rows and a set of columns.
If the Cartesian product rows × columns is taken, the cells of the table contain ordered pairs of the form (row value, column value).



Mathematically:
Relation STUDENT ⊆ String × Integer × Integer

Cartesian Product

Source: www.wikipedia.org

3.1 What is the relational model

RM: The model

- Objects are described using relations
  - Relations can be viewed as tables
  - But: Not like a spreadsheet table!
- There can be links between relations
- Attributes describe properties
- Possible attribute values are defined by the domain

3.1 What is the relational model

RM: The model

Informally:

- A relational model represents the database as a collection of relations
- Each relation resembles a table of values or, to some extent, a flat file of records
- When a relation is thought of as a table of values, each row in the table represents a collection of related data values
- A row represents a fact that typically corresponds to a real-world entity or relationship

3.1 What is the relational model

- The table name and column names are used to help to interpret the meaning of the values in each row
- All values in a column are of the same data type

3.1 What is the relational model

RM: The model

Formally:

- A row is called a tuple
- A column header is called an attribute
- The table is called a relation
- The data type describing the types of values that can appear in each column is represented by a domain of possible values

3.1 What is the relational model

RM: The math behind it

- Example:
  - `ROOM(room_num, function, seats)`
  - where $\text{function} = \{\text{auditorium}, \text{lab}, \text{office}, \text{administration}\}$

3.1 What is the relational model

RM: The math behind it

Idea

- A Relation Schema R is a set of attributes $(A_1, A_2, ..., A_n)$.
- Each attribute A_i is the name of a role played by a certain domain D in the relational schema R.
- A domain D of attribute A_i is denoted as $\mathrm{dom}(A_i)$.
- The degree (or arity) of a relation is the number of attributes n of its relational schema.

3.1 What is the relational model

RM: The math behind it

3.1 What is the relational model

RM: The math behind it

- Relational Schema:

```
1  BOOK                                  SQL
2  (ISBN,
3  title,
4  author,
5  publisher,
6  year,
7  price)
```

- Relational Schema with types:

```
1  BOOK                                  SQL
2  (ISBN: integer,
3  title: string,
4  author: string,
5  publisher: string,
6  year: integer,
7  price: real)
```

⟶ Relation `BOOK` is of degree six.

3.1 What is the relational model

RM: The math behind it

- A relation (or relational state) r of the relation schema $R(A_1, A_2, ..., A_n)$, also denoted by $r(R)$, is a set of m-tuples

$$r = (t_1, t_2, ..., t_m)$$

- Each n-tuple t is an ordered list of n values $t =< v_1, v_2, ..., v_n >$, where each value v_1, $1 \leq i \leq n$, is an element of $\mathrm{dom}(A_i)$ or is a special `NULL` value.
- The i^{th} value in tuple t, which corresponds to the attribute A_i, is referred to as $t[A_i]$ or $t.A_i$ (or $t[i]$ if we use the positional notation).

3.1 What is the relational model

RM: The math behind it

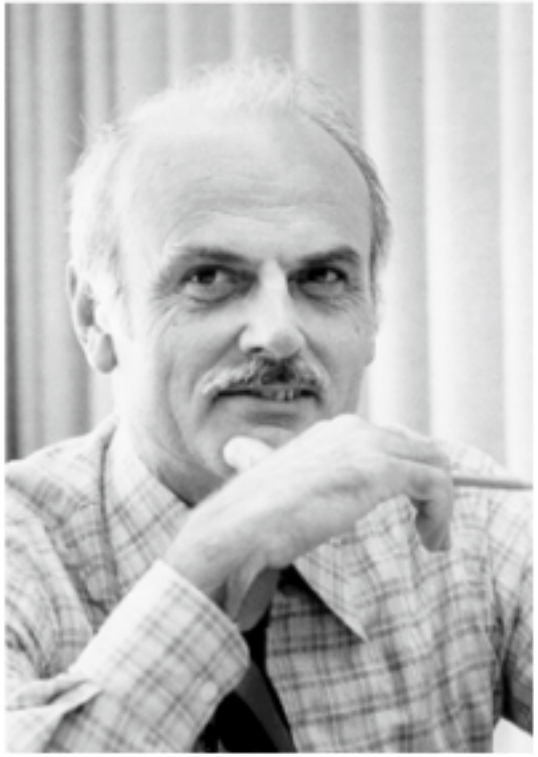
! Memorize

- A relation is a set of rows.
  - meaning: no order, no row number
  - no duplicates

3.1 What is the relational model

RM: The math behind it

3.1 What is the relational model



Source: www.wikipedia.org

3.1 What is the relational model

RM: The math behind it

- A relation (or relational state) $r(R)$ is a mathematical relation of degree n on the domains $\mathrm{dom}(A_1), \mathrm{dom}(A_2), ..., \mathrm{dom}(A_n)$, which is a subset of the Cartesian product (denoted by $\times$) of the domains that define R:

$$r(R) \subseteq (\mathrm{dom}(A_1) \times \mathrm{dom}(A_2) \times ... \times \mathrm{dom}(A_n))$$

- If $|D|$ is the total number of values in a domain D, the total number of tuples in the Cartesian product is

$$|\mathrm{dom}(A_1)| \times |\mathrm{dom}(A_2)| \times ... \times |\mathrm{dom}(A_n)|$$

4. License Notice

4.1 Attribution

- This work is shared under the CC BY-NC-SA 4.0 License and the respective Public License.
- https://creativecommons.org/licenses/by-nc-sa/4.0/
- This work is based off of the work by Prof. Dr. Ulrike Herster.
- Some of the images and texts, as well as the layout were changed.
- The base material was supplied in private, therefore the link to the source cannot be shared with the audience.