Lecture 1

# Database Design and Data Modelling - an Overview

Week 1

This series of lectures introduces:

- Enhanced Entity Relationship (EER) Data Model
- Methods to describe a universe of discourse (a business area) using the EER Data Model
- The Relational Data Model
- Mapping of an EER design onto a Relational Database Implementation
- How to describe and manipulate a relational database using the Structured Query Language (SQL).
- Principles of good database design

# Questions you should be able to answer after you studied this module

- How can information (e.g. about an enterprise) be represented ?
- What languages can be used for this task?
- How to manipulate this information (store and retrieve it) ?
- What standards exist?

# Possible answers

- Use the (Enhanced) Entity Relationship Model to represent data (or IDEF1x, UML, EXPRESS,...)
- Use the Relational Data Model to represent data (or OO, XML,…)
- Use SQL (Structured Query Language) for describing and manipulating data (or OSQL,…)

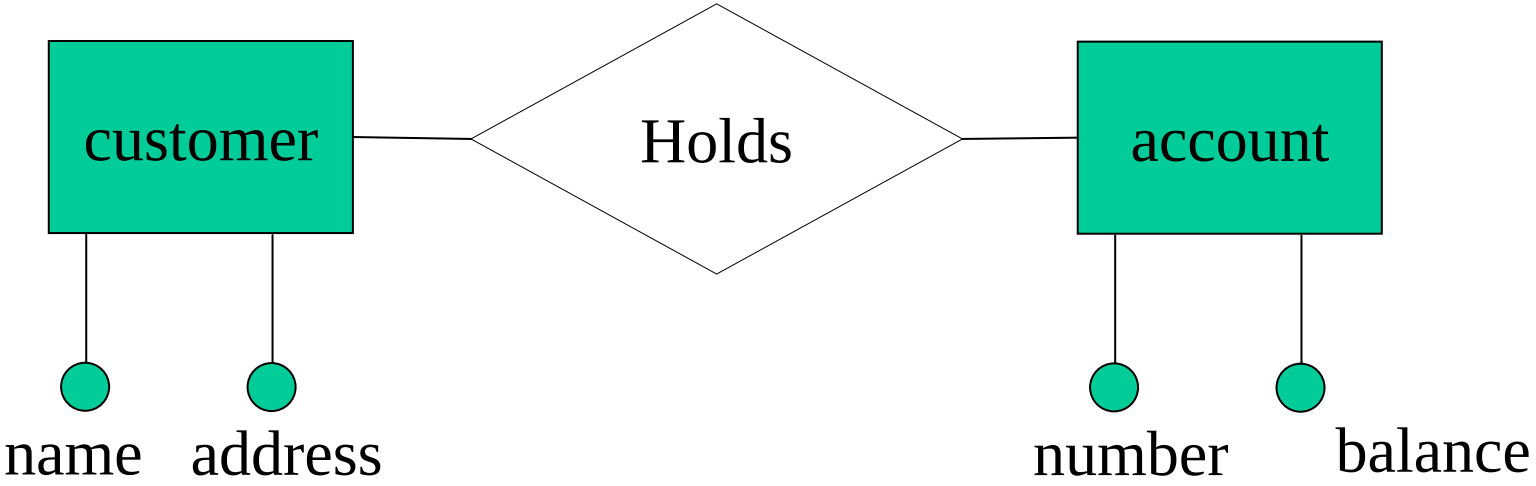
# The Universe of Discourse (UoD)

- The UoD is that domain of the enterprise about which information is to be stored in the database (accounting, personnel, materials management, scheduling, product catalog, customers...etc)

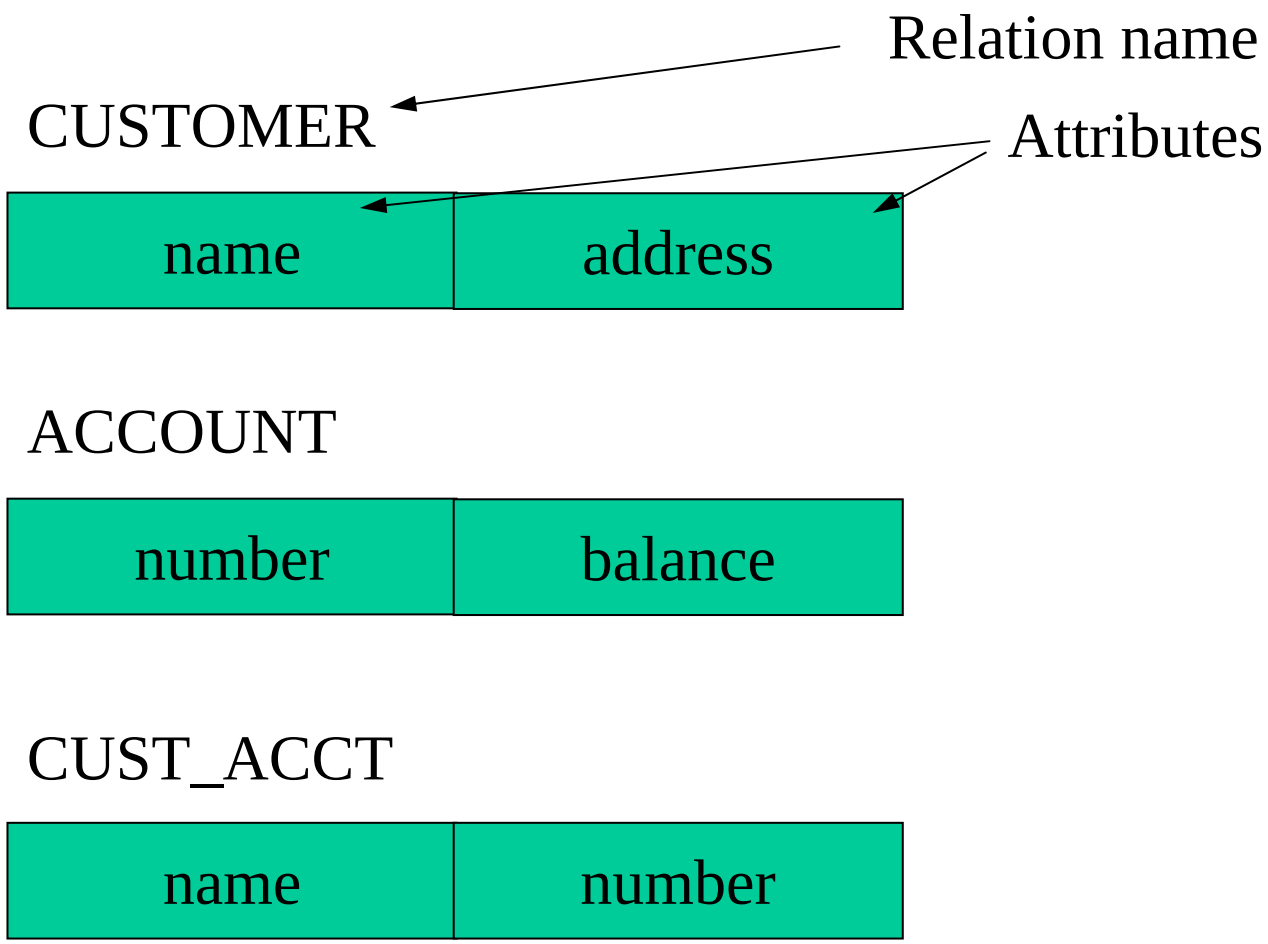
# Information or Data?

- Data: facts
- Information: interpreted data (what element of news is carried by the data)
- Databases store data but users *interpret* the retrieved data and it becomes information (for them)

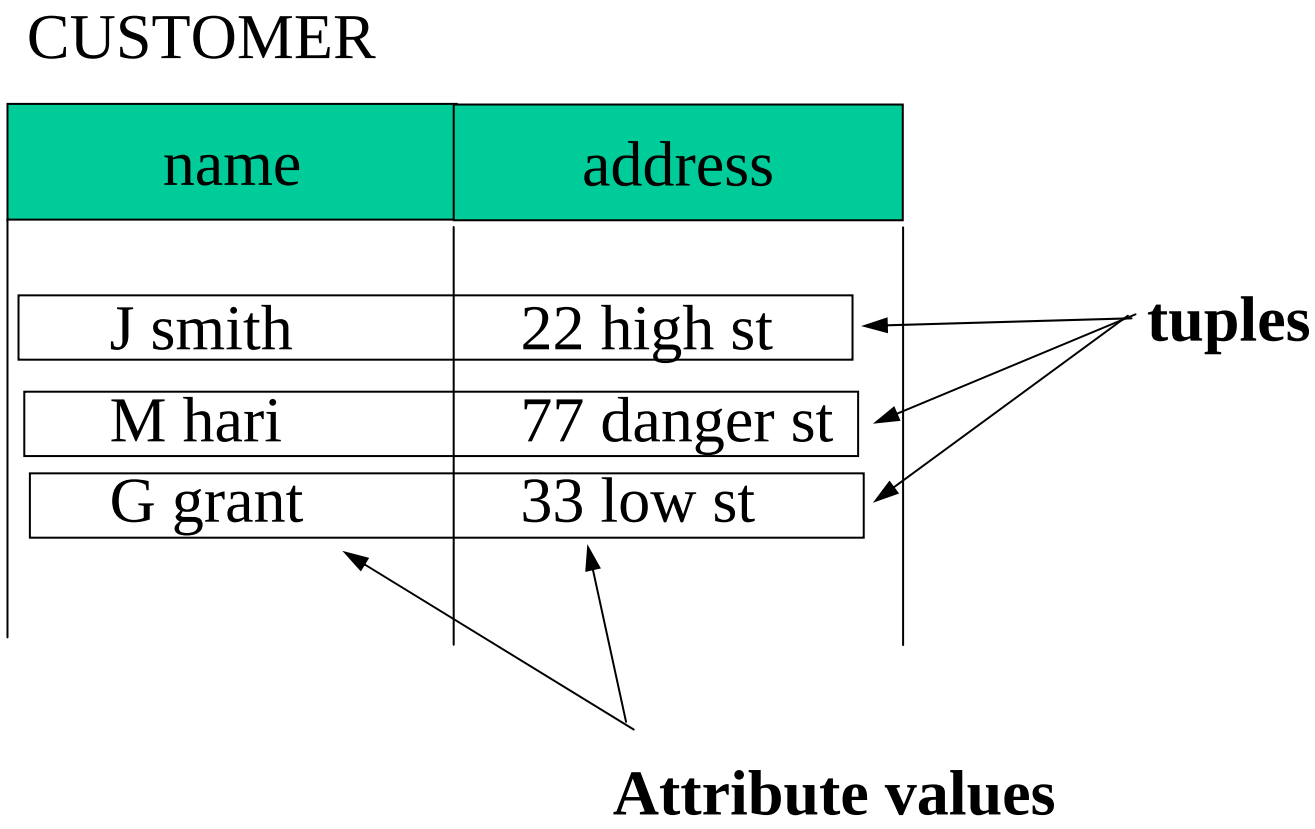
# Example Entity Relationship Diagram



# Relation schema



# Relation instance



# Databases can be queried using

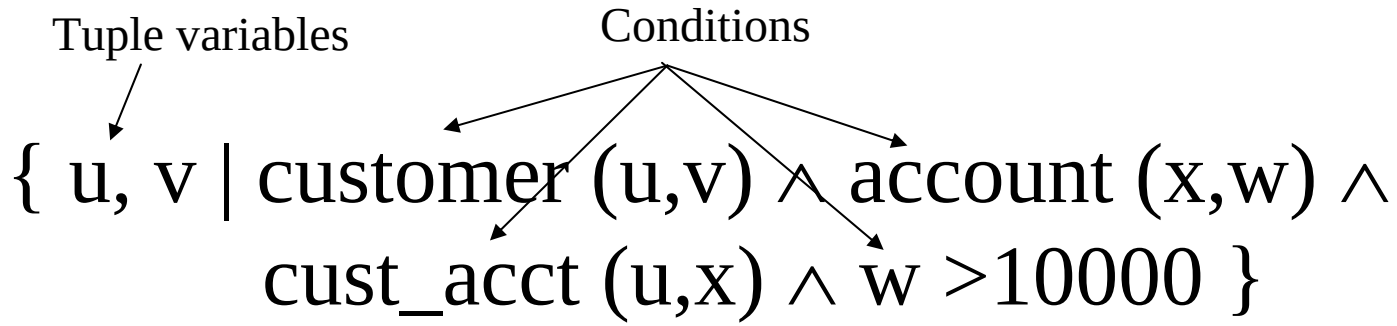
- Relational calculus
- Relational algebra
- SQL

# Relational calculus

The query

*"Show names and addresses of customers with balances > $10,000"*

can be expressed as:



# Relational algebra

The query

*"Show names and addresses of customers with balances > $10,000"*

can be expressed as:

$$\Pi_{\text{name,adress}} (\sigma_{\text{balance}>10000} (\text{customer} * \text{account} * \text{cust\_acct}) )$$

# SQL

The query

*"Show names and addresses of customers with balances > $10,000"*

can be expressed as:

```
SELECT name, address
FROM customer c , account a, cust_acct ca
WHERE
    c.name = ca.name AND
    ca.number = a number AND
    a.balance >10000;
```

# The End