CS 337
Databases and DBMSs

Benjamin Dicken

Announcements

- Welcome Back
- PA 8
- Final Project
- UG Teaching Assistant Applications

Database != DBMS

Databases and DBMSs

- A ***Database (DB)*** is an organized collection of data, typically organized to model aspects of reality in a way that supports external processing
- A ***Database Management System (DBMS)*** is a computer software application that interacts with the user, other applications, and the *database* itself to capture and analyze data

Thanks, Wikipedia!

Databases and DBMSs

- A database is not a **program**
- It is a collection of information (typically one or more files on a computer) that represent something meaningful
- We will discuss several ways in which databases can **model** reality

Databases and DBMSs

- A database could be:
  - A single CSV file, where each line represents an entity, and each column a bit of information
  - A collection of files in a directory that have information related to each-other
  - An entire hard-drive with organized files and information
  - An building full of hard drives with petabytes of information

Databases and DBMSs

- A **DBMS** *is* a computer program
- A user should interact with the DBMS, not the database
  - A DBMS is a “middle man” between the user and the database
  - A robust DBMS provides features to add, remove, retrieve, and process data in a database
  - Users sends ***queries*** to a DBMS to interact with the data held within it

Activity

Databases and DBMSs

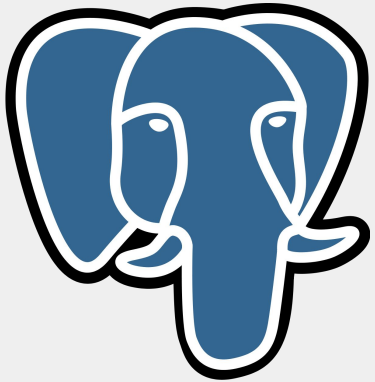
- Why is a DBMS necessary? Why Can’t we just manually view and edit files using programs like Excel?

Databases and DBMSs

- Why is a DBMS necessary? Why Can’t we just manually view and edit files using programs like Excel?
  - **Scale** is one issue
    - What if you have a billion pieces of information?
  - **Complexity** is another issue
    - What if you have very complex information and relationships to represent?

Databases and DBMSs

- Well-known DBMSs include
  - MySQL, SQLite, PostgreSQL, MongoDB, MariaDB, Microsoft SQL Server, Oracle, IBM DB2 ...



Databases and DBMSs

- A DBMS should provide functionality that allows for management of a database and its data
- These functionalities can be classified into four main functional groups

Databases and DBMSs

1) **Data definition**

2) **Update**

3) **Retrieval**

4) **Administration**

1) **Data definition**
2) **Update**
3) **Retrieval**
4) **Administration**

What do each of these DBMs requirements refer to? Write a sentence for each.

Databases and DBMSs

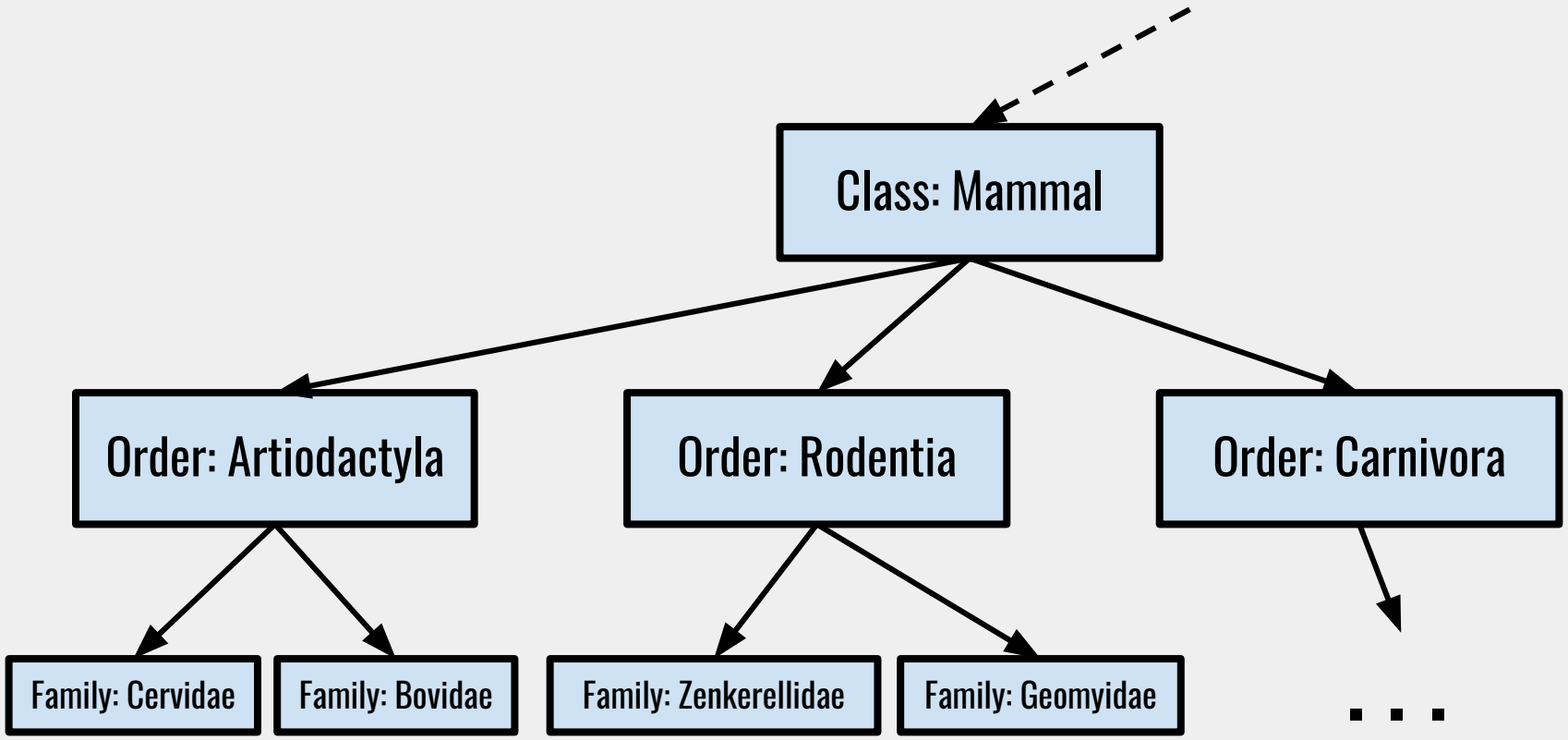
1) **Data definition** – Creation, modification and removal of definitions that define the organization of the data
2) **Update** – Insertion, modification, and deletion of the actual data

3) **Retrieval** – Providing information in a form directly usable or for further processing by other applications.
4) **Administration** – Registering and monitoring users, enforcing data security, monitoring performance, recovery, etc...

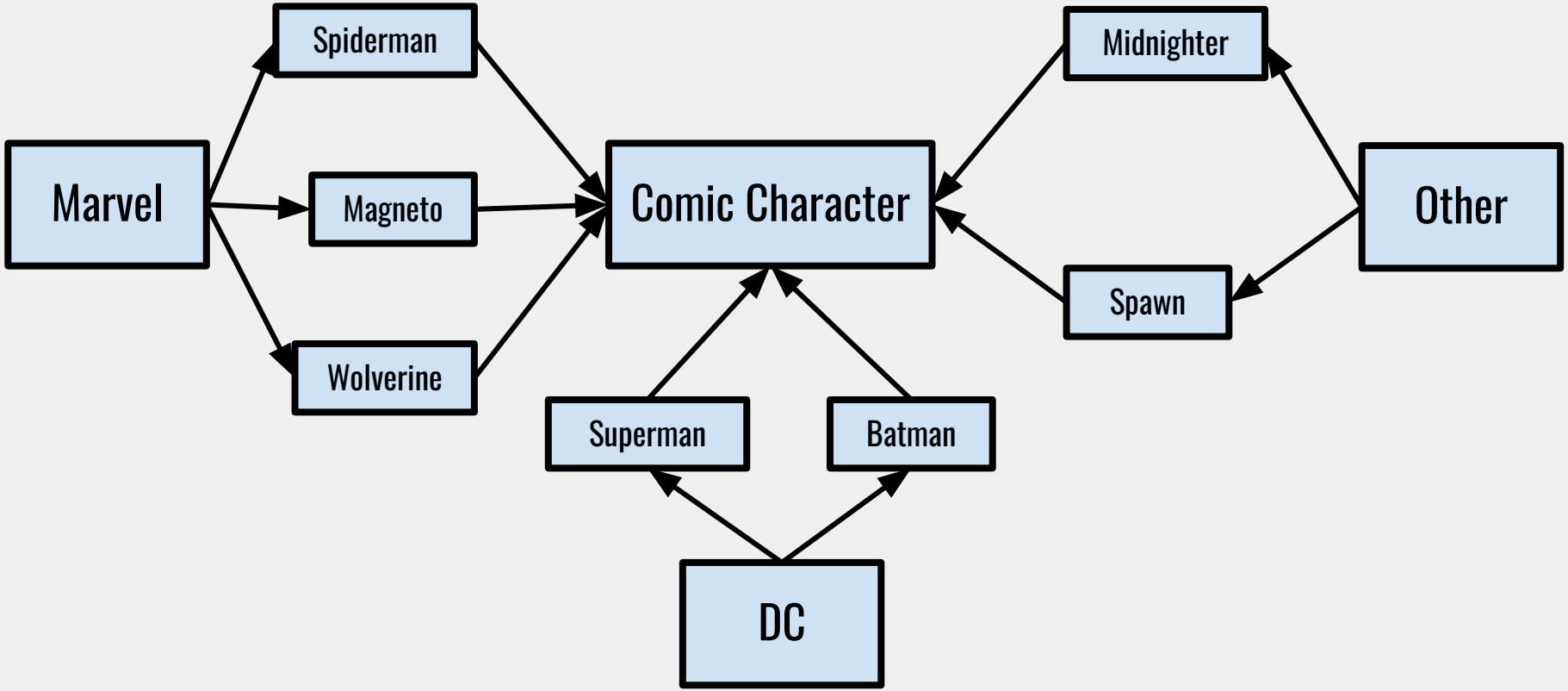
Databases and DBMSs

- Four main structure *types* of databases
  - Relational databases
  - Hierarchical databases
  - Network databases
  - Object-oriented databases

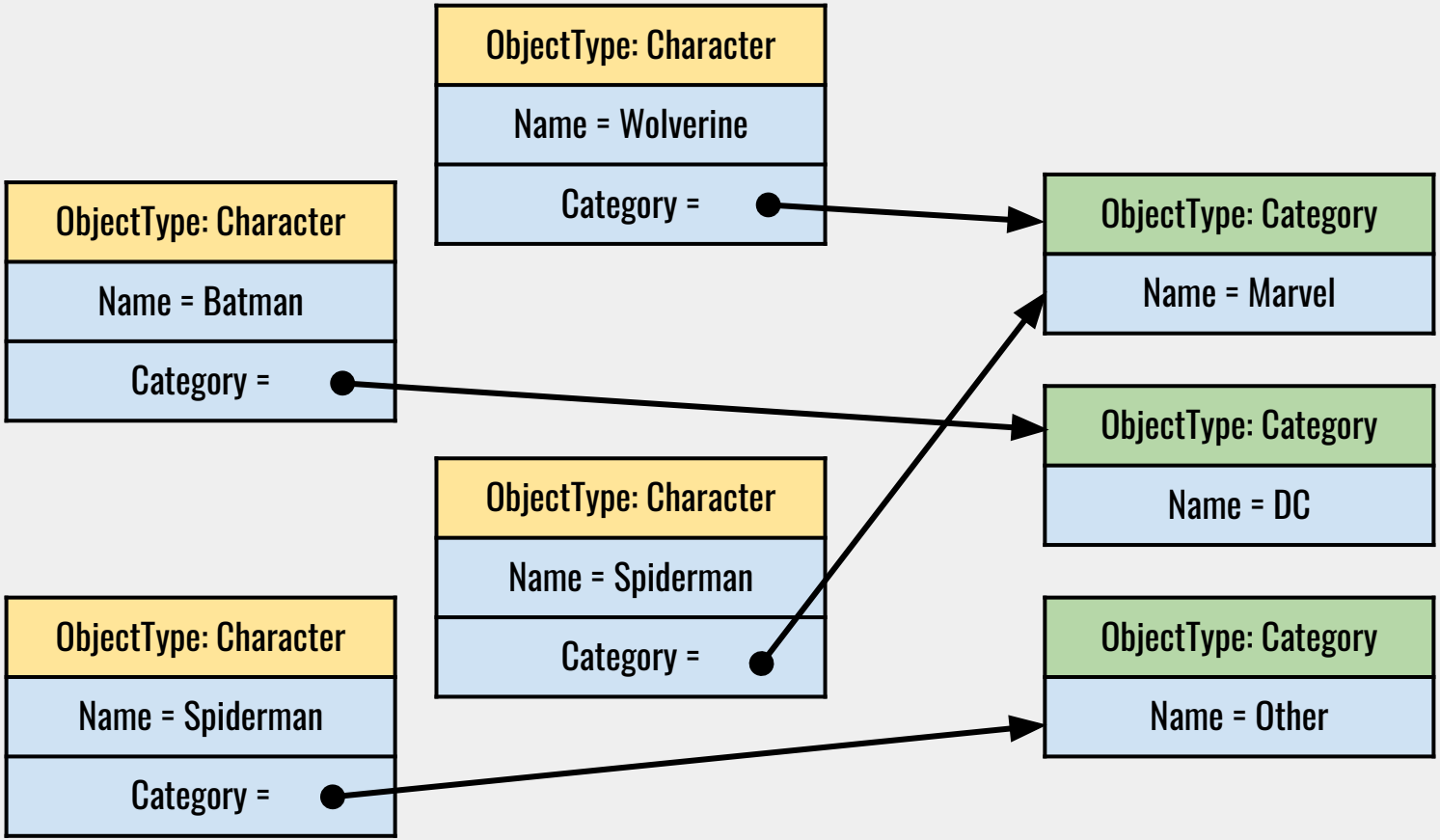
Hierarchical Model (Taxonomic Rank)



Network Model



Object-oriented Model



Relational Model

Name
Batman
Superman
Spiderman
Spawn
Midnighter
Magneto
Wolverine

Name
Marvel
DC
Other

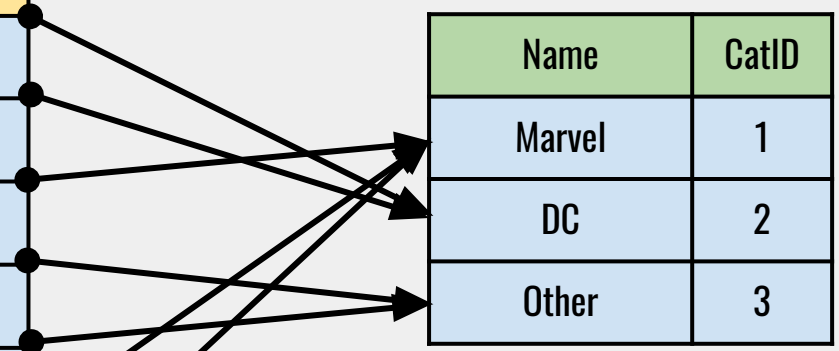
Relational Model

Name	Character ID	CatID
Batman	1	2
Superman	2	2
Spiderman	3	1
Spawn	4	3
Midnighter	5	3
Magneto	6	1
Wolverine	7	1

Name	CatID
Marvel	1
DC	2
Other	3

Relational Model

Name	Character ID	CatID
Batman	1	2
Superman	2	2
Spiderman	3	1
Spawn	4	3
Midnighter	5	3
Magneto	6	1
Wolverine	7	1



Name	CatID
Marvel	1
DC	2
Other	3

Databases and DBMSs

- Four main structure *types* of DBMSs
  - Relational databases
  - Hierarchical databases
  - Network databases
  - **Object-oriented databases**

Object - oriented model

- It is the task of a database administrator, or backend dev, to model some real-world information and interactions with the relational model
- Typically, each real-world ***entity*** has its own document

Databases and DBMSs

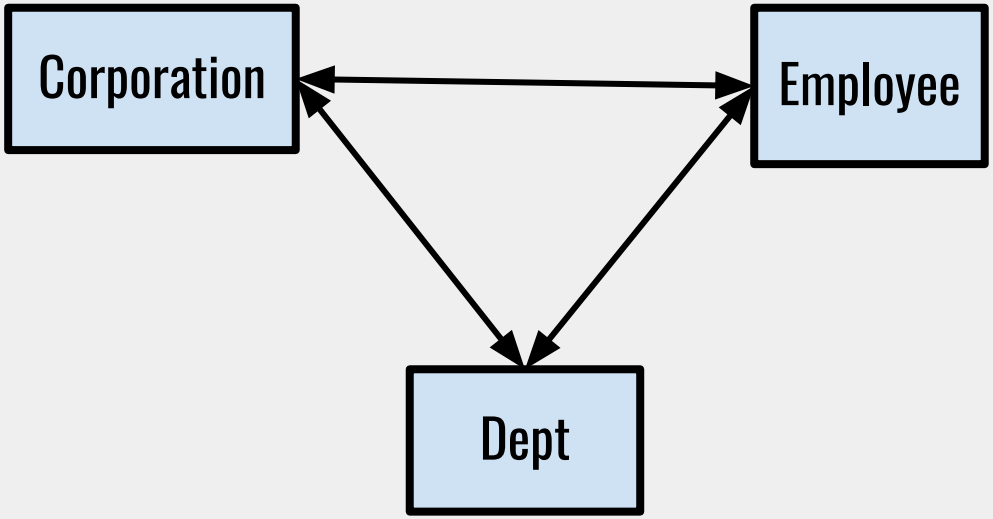
How to model the following real-world information with a relational model?

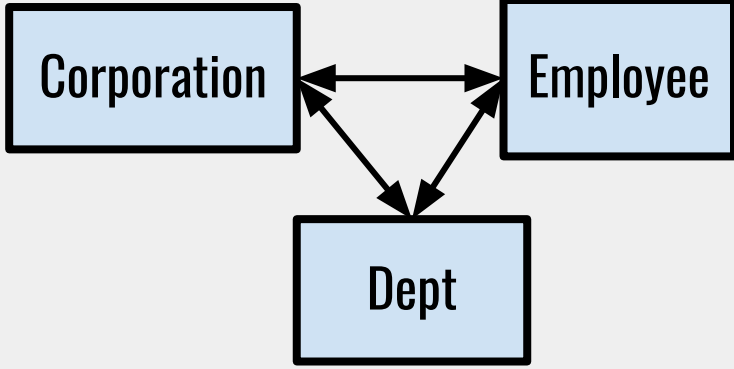
John Smith makes $***80,000*** /yr in the ***Engineering*** dept for **IBM**
Janet Carrie makes ***$120,000*** /yr in the ***Marketing*** dept for ***IBM***
Michael Johnson makes ***$150,000*** /yr in the ***Marketing*** dept for ***Raytheon***
Samantha Jones makes ***$100,000*** /yr in the ***Finance*** dept for ***Raytheon***
Luigi deSantis *makes* ***$85,000*** */yr in the* ***Finance*** *dept for* ***IBM***
Yuri Bezmenov makes ***$115,000*** /yr in the ***Research*** dept for **Raytheon**
Jocob Robinson makes $***70,000*** /yr in the ***Engineering*** dept for **IBM**
. . . .

Databases and DBMSs

- In this set of info, there are three types of entities
  - **Employee**
  - **Departments**
  - **Corporations**
- What can we use to model this?

Databases and DBMSs




Corporation
Employee
Dept

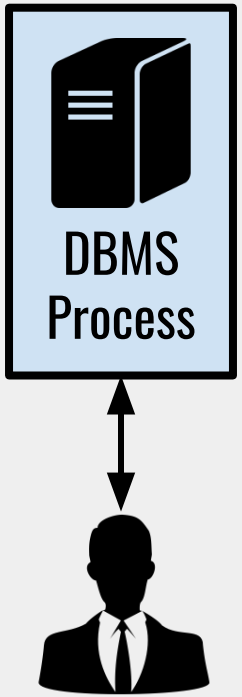
Types of Relationships

- Relationships
  - **1-to-1:** Do not need extra table for relationship
  - **1-to-many:** Sometimes want extra table for relationship (but not necessary)
    - If only need to go one-way, no need
    - If want to go both ways easily, then need!
  - **many-to-many:** Always have extra relation for relationship

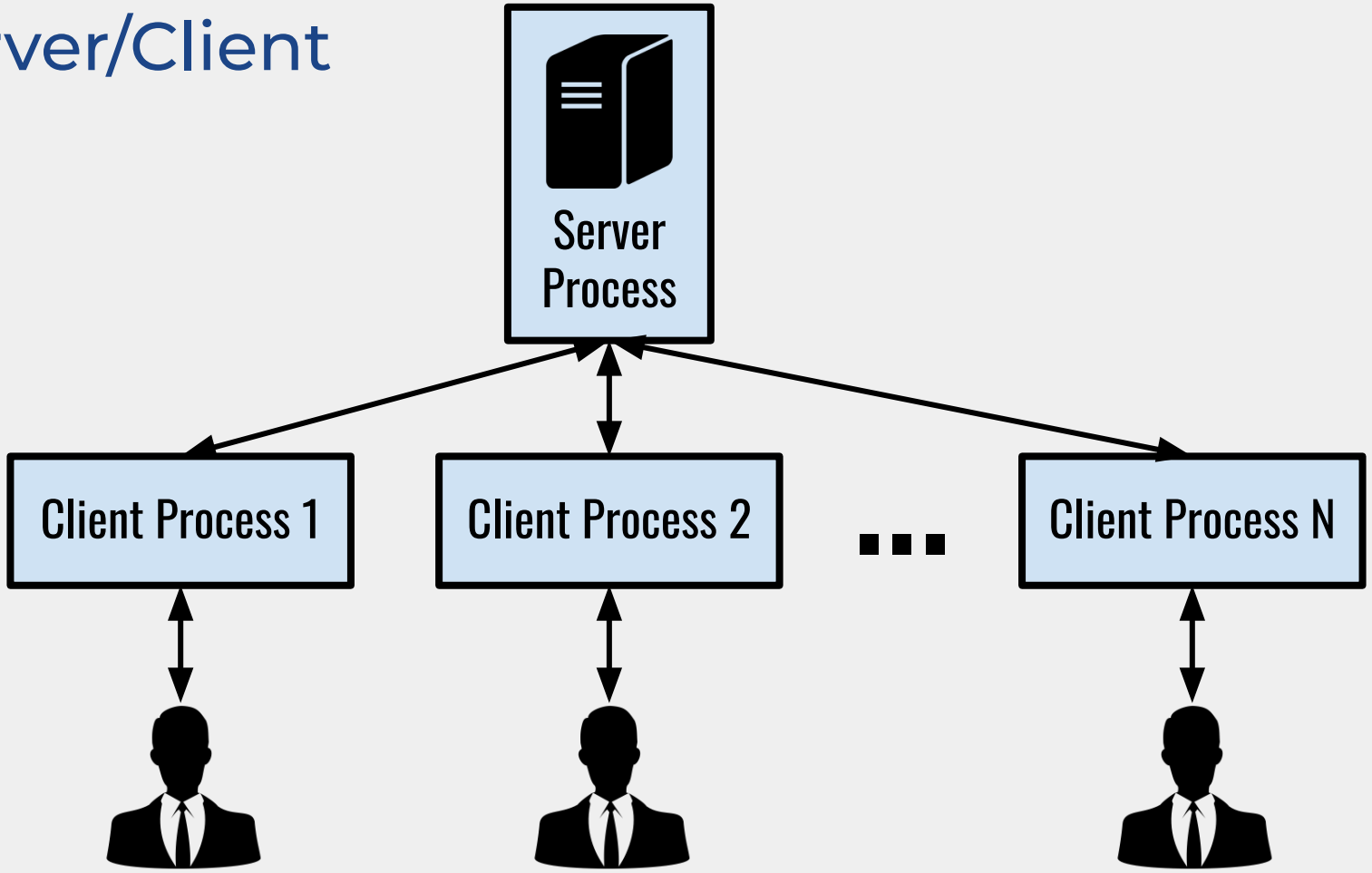
Databases and DBMSs

- In this set of info, there are three types of entities
  - **Employee**
  - **Departments**
  - **Corporations**
- What can we use to model this?

Standalone

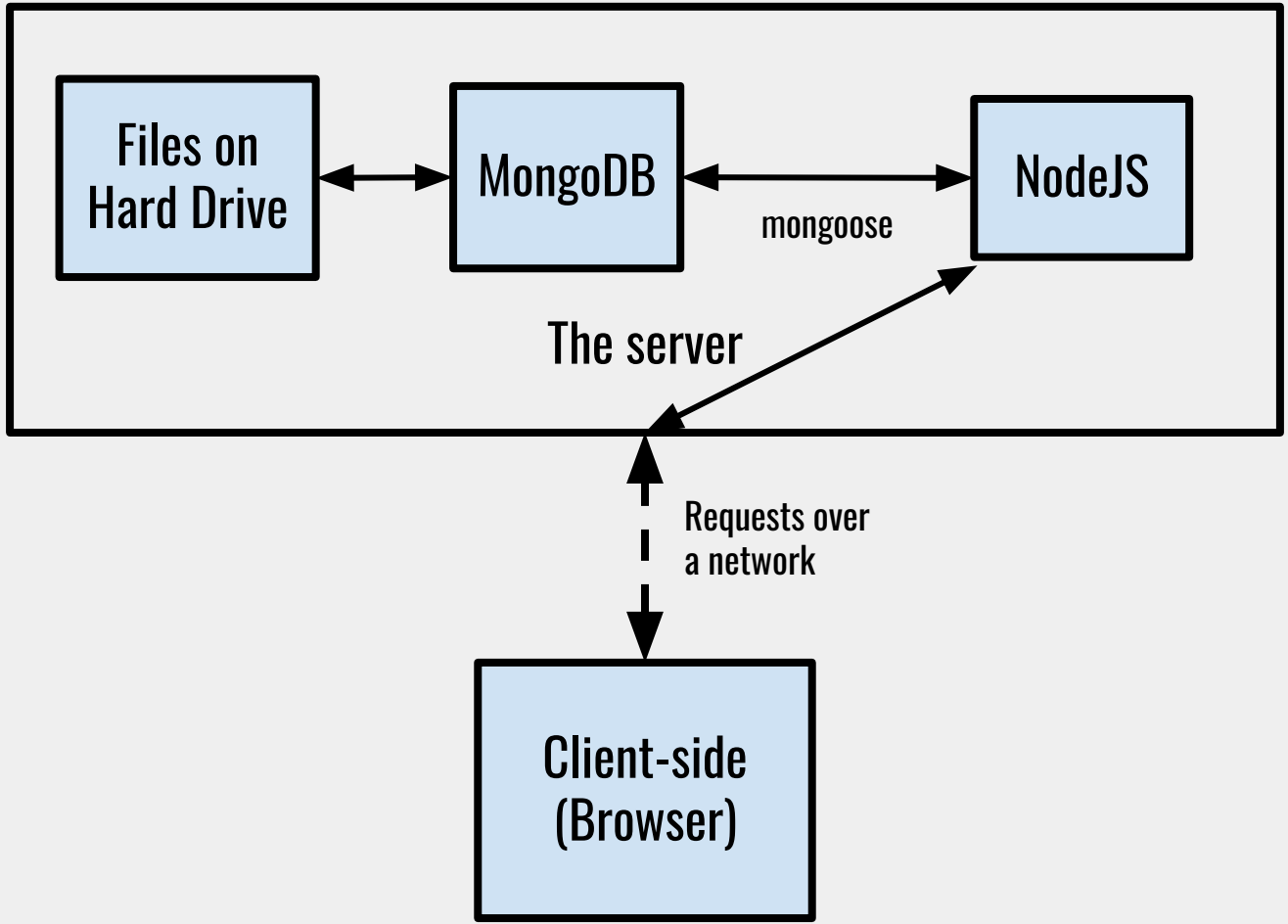


Server/Client



MongoDB

- In particular, we will be learning the **MongoDB** DBMS because:
  - Good for using with Js
  - Works well with nodeJs and express (mongoose)
  - Part of the popular MEAN stack


Files on
Hard Drive
MongoDB
NodeJS
mongoose
The server
Requests over
a network
Client-side
(Browser)