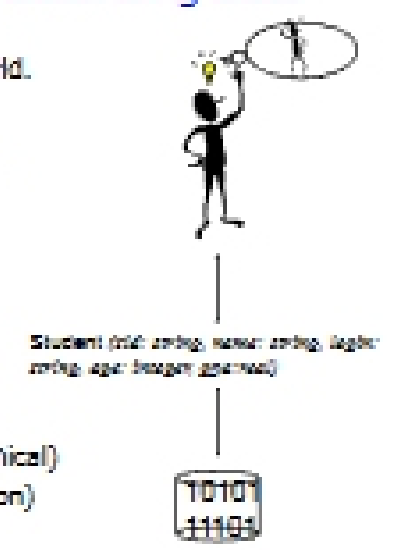
Data and Queries in the Relational Model

CS 162 Guest Lecture
Mike Franklin
April 6, 2011

> A relationship, I think, is like a shark, you know? It has to constantly move forward or it dies. And I think what we got on our hands is a dead shark.
>
> Woody Allen (from Annie Hall, 1979)

Data Models – Describing Data

- A *Database design* encodes some portion of the real world.
- A *Data Model* is a set of concepts for thinking about this encoding.
- Many models have been proposed.
- We will look at two related models:
  - i) Entity-Relationship (graphical)
  - ii) Relational (implementation)



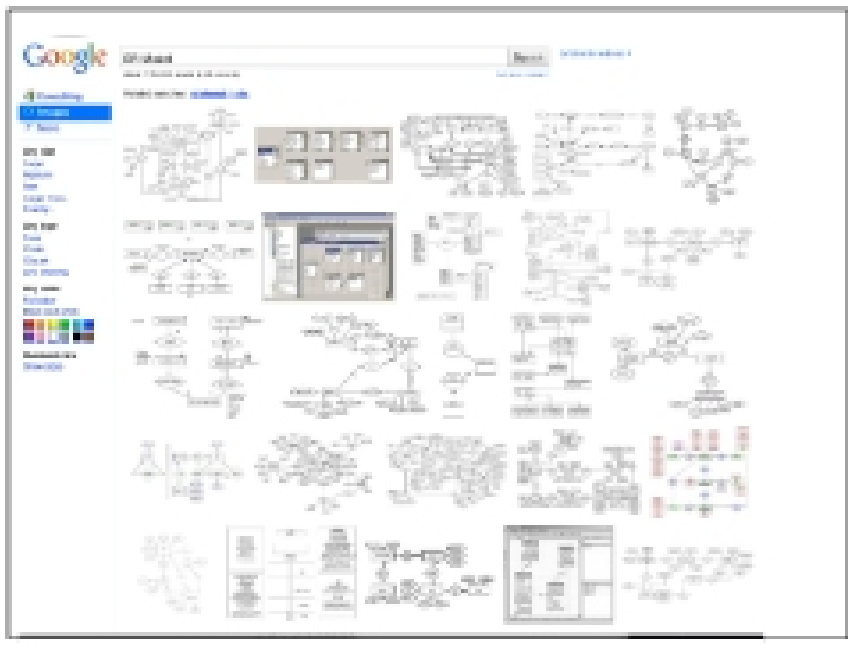
Steps in Database Design

- Requirements Analysis
  - user needs; what must the database capture?
- Conceptual Design
  - high level description (often done w/ER model)
- Logical Design
  - translate ER into DBMS data model
    - Typically: "relational" model as implemented by SQL
- Schema Refinement - consistency, normalization
- Physical Design - indexes, disk layout
- Security Design - who accesses what, and how

Conceptual Design using ER

- What are the entities and relationships?
- What info about E's & R's should be in DB?
- What *integrity constraints* (*business rules*) hold?
- *ER diagram* is a representation of the 'schema'
- Can map an ER diagram into a relational schema.

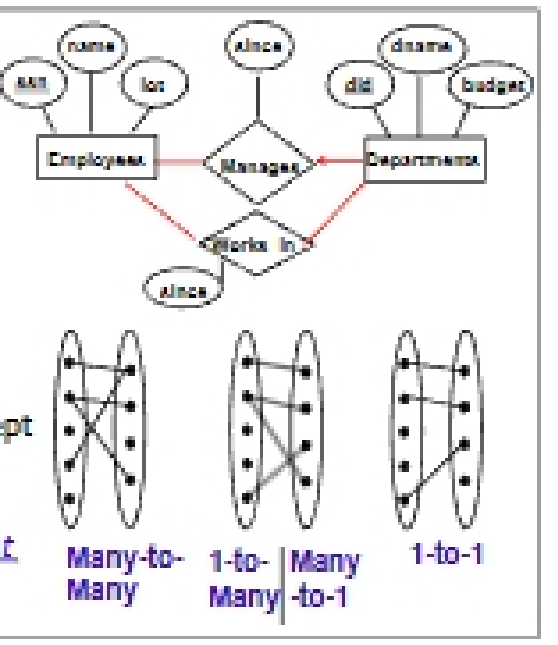
- Conceptual design is where the SW/data engineering *begins*
  - Rails "models"



ER Example

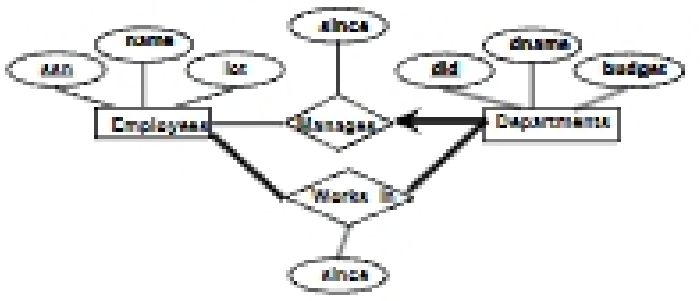
An employee can work in many departments; a dept can have many employees.

In contrast, each dept has at most one manager, according to the *key constraint* on Manages.



Participation Constraints

- Does every employee work in a department?
- If so: a *participation constraint*
  - participation of Employees in Works_In is *total* (vs. *partial*)
  - What if every department has an employee working in it?
- Basically means "at least one"



Implementation: The Relational Model

- The E-R model is not directly implemented by most DBMSs.
- Fairly easy to map an E-R design to a Relational Scheme
- The Relational Model is Ubiquitous
  - MySQL, PostgreSQL, Oracle, DB2, SQLServer, ...
  - Note: some "Legacy systems" use older models
    - e.g., IBM's IMS
- Object-oriented concepts have been merged in
  - Early work: POSTGRES research project at Berkeley
  - Informix, IBM DB2, Oracle 8i
- As has support for XML (semi-structured data)

Relational Database: Definitions

- *Relational database:* a set of *relations*
- *Relation:* made up of 2 parts:
  Schema : specifies name of relation, plus name and type of each column

  Students(*sid*: string, *name*: string, *login*: string, *age*: integer, *gpa*: real)

  Instance : the actual data at a given time
  - #rows = *cardinality*
  - #fields = *degree / arity*

Some Synonyms

Formal	Not-so-formal 1	Not-so-formal 2
Relation	Table	
Tuple	Row	Record
Attribute	Column	Field
Domain	Type	

Ex: Instance of Students Relation

sid	name	login	age	gpa
53666	Jones	jones@cs	18	3.4
53688	Smith	smith@eecs	18	3.2
53650	Smith	smith@math	19	3.8

- Cardinality = 3, arity = 5 , all rows distinct
- Do all values in each column of a relation instance have to be distinct?

SQL - A language for Relational DBs

- Say: "ess-cue-ell" or "sequel"
  - But spelled "SQL"
- Data Definition Language (DDL)
  - create, modify, delete relations
  - specify constraints
  - administer users, security, etc.
- Data Manipulation Language (DML)
  - Specify queries to find tuples that satisfy criteria
  - add, modify, remove tuples