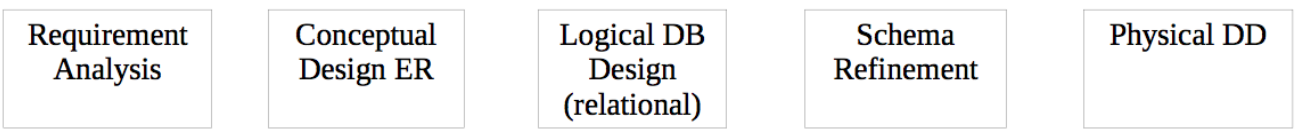
CS 166: Lab2 notes

March, 11 2018

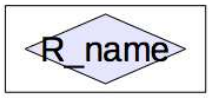
1. Database Design steps:
   - Requirement Analysis
     - What data should be stored in Database? Analysts + customer
     - What applications can be build on top of it?
     - Which operations are subject to performance requirements?
   - Conceptual Database Design
     - High level description of Analysis + constrains over these data are modeled with the ER model (semantic model used in database design)
     - Goal: Generate description of data which is understandable both by developers and users
     - The design must be precise in order to allow straightforward translation into the relational model (tables, attributes,..), which is used by the Database
   - Logical Database Design (ER schema → relational database schema)
     - Generally this step involves the conversion of the conceptual schema to Database-specific schema.
   - Schema Refinement
     - Analyze Relational Database Schema and identify problems
     - Ex: Student(ssn, name, numgrade, lettergrade) - functionally-dependent attributes
     - Normalizing relations
   - Physical Database Design
     - Make sure that the database meets the performance needs / workloads that are expected by the Analysis. (Indexes, Denormalize Relations)
   - Application & Security Design



2. Entities, Attributes, Entity Sets, Relationships, Relationship Sets

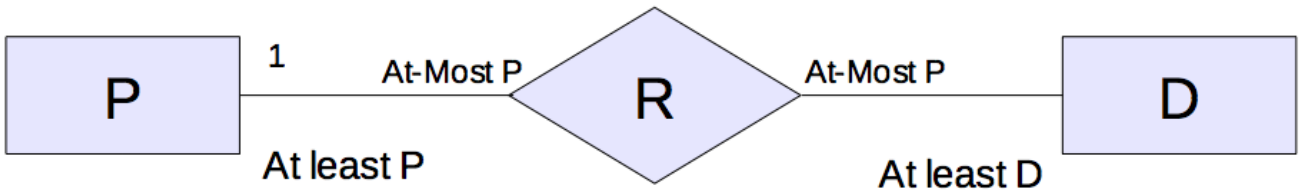
- Entity (Relation)
  - An entity is a real world object that can be distinguished from another object given some attributes
  - e.g. Employee, Manager are not different
  - but Employee, Projects are different
- Attribute
  - Several attribute characterize an entity. If an attribute is multi-value (address zip, address, aptno) create an entity
  - Domain → Possible values
  - Key → Set of attributes that uniquely identifies an entity (primary, secondary, candidate)
- Entity-Set
  - Toy, Appliance Department Employees under same set
- Relationship
  - Relates 2 or more entities
  - Descriptive Attributes
  - Ternary Relationship → Involves 3 relations
- Relationship Set
  - A set of similar relationships

3. Entity-Relationship (ER) Diagram
   - The ER model allows us to describe the data involved in a real world enterprise in terms of objects (entities) and their relationships
   - Provides the initial framework for developing an initial DB design
   - There are other variations of the ER model, mainly different on the way entities and their relationships are graphically represented
   - You should follow book notation
   - ER Data Structure Diagram
4. ER Notation Explanation

	Entity
	1:1, 1:M or M:1 Relationship
	M:N Relationship (Entity-Relationship)
	Relationship connector
	At-least-One Relationship connector
	At-most-One Relationship connector
	At-least-One and At-most-One Relationship connector
	Weak Entities
	Attribute (key if it is underlined)

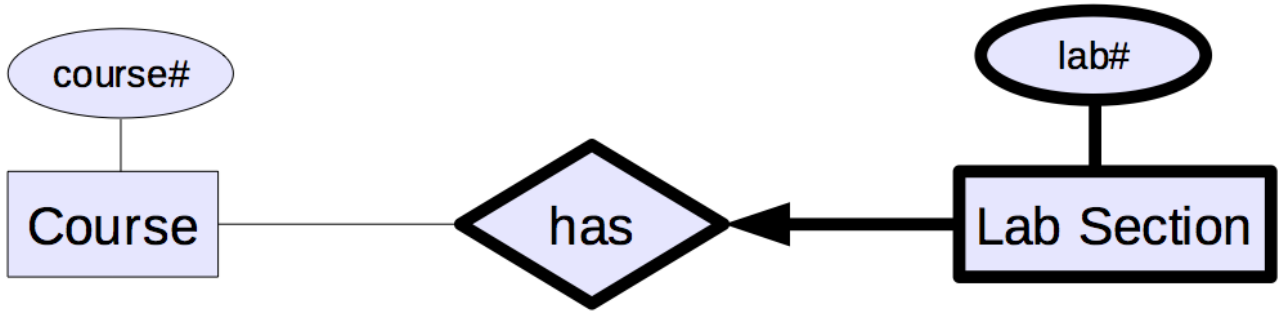
5. Additional Feature of ER Model

   - Key Constrains
     - The "At least – At most" question
     - 1:1 : Each professor works in at most 1 department. At most 1 professor work works in each department.

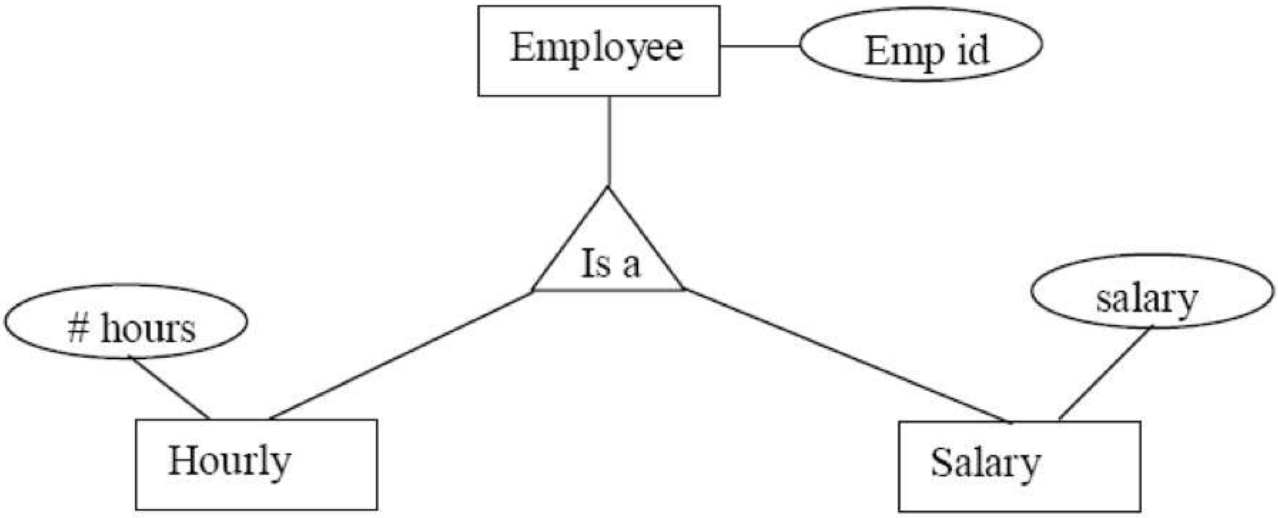


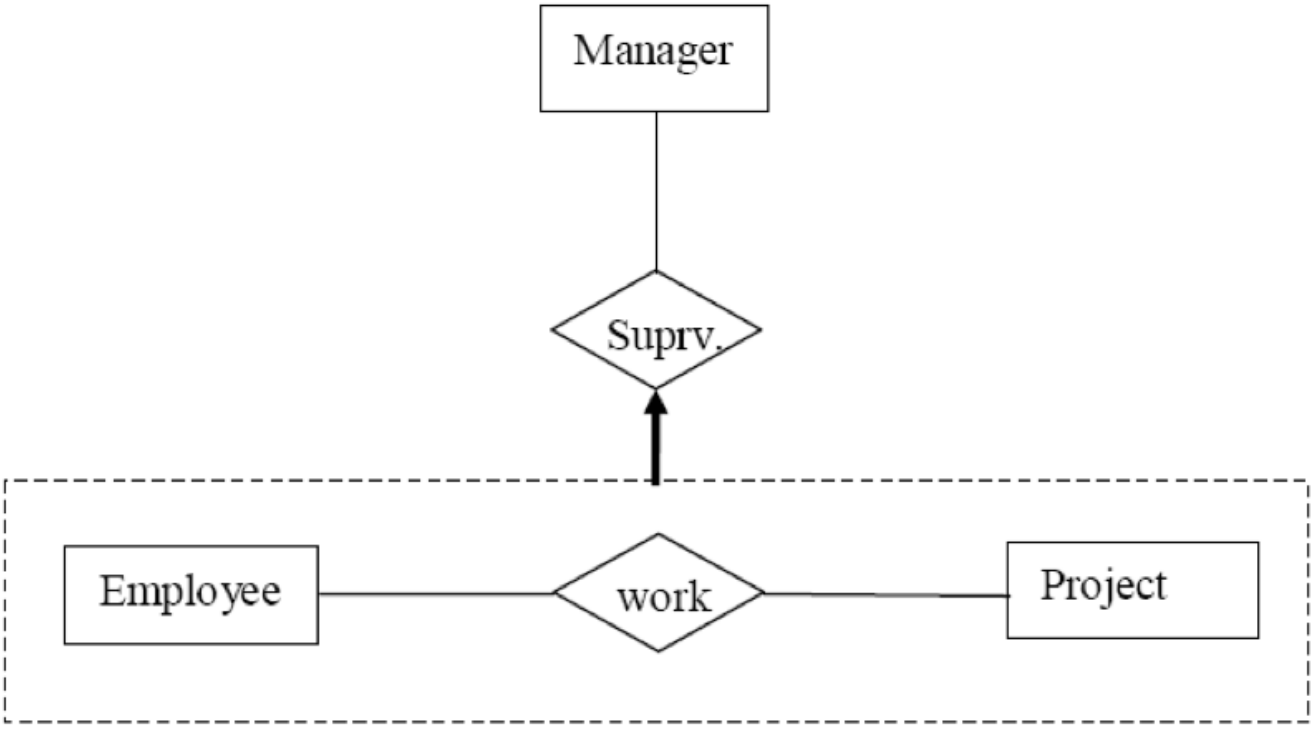
     - 1:N : Each professor works in at most N departments . In each department at most 1 professor work.
     - N:M : Each professor works in at most N departments . In each department at most M professor work.
   - Participation Constrains (partial VS total)
     - Employee works in at least 1, at most N Departments (TOTAL participation in relationship)
     - Employee works in at least 0, at most N Departments (PARTIAL participation in relationship)
   - Weak Entities

– A weak entity can be identified uniquely ONLY by considering the primary of another relation



- Class Hierarchies & Aggregation





6. Exercises

   (a) A university database contains information about professors (identified by social security number, or SSN) and courses (identified by courseid). Professors teach courses; each of the following situations concerns the Teaches relationship set. For each situation, draw an ER diagram that describes it (assuming that no further constraints hold).

   - Professors can teach the same course in several semesters, and each offering must be recorded.
   - Professors can teach the same course in several semesters, and only the most recent such offering needs to be recorded. (Assume this condition applies in all subsequent questions.)
   - Every professor must teach some course.
   - Every professor teaches exactly one course (no more, no less).
   - Every professor teaches exactly one course (no more, no less), and every course must be taught by some professor.
   - Now suppose that certain courses can be taught by a team of professors jointly, but it is possible that no one professor in a team can teach the course. Model this situation, introducing additional entity sets and relationship sets if necessary.

   Entity Vs Attribute (e.g. address) Depends on application. Rule of Thumb: if multi-value attribute split into Entity

   Entity Vs Relationship Rule of Thumb: if a relationship is M:N think of making it an entity

   - Easier for your design.
   - During conversion to relational model you will do it anyway

- Rename by concatenating: Student, Courses → StudentCourses

(b) Create an ER diagram for a car-insurance company with the following information:

- People own one or more cars each
- A car cannot have multiple owners
- Each person has the following attributes: driver-id, name, address
- Cars have license, model, year attributes
- Each car has a number of recorded accidents (zero to any) associated with it
- Accidents have report-number, location, date attributes
- The car owner might not be the driver of the car involved in the accident
- For each accident record the damage-amount
- Step one: Define Entity Set
  - People: driver-id, name, address
  - Car: license, model, year
  - Accident: report-number, location, date
- Step two: Relationship sets
  - owns: a person can own 0 or more cars. A car has only one owner
  - participates: a driver and a car participate in an accident. Damage-amount is the attribute of participates
- Step three: build ER diagram