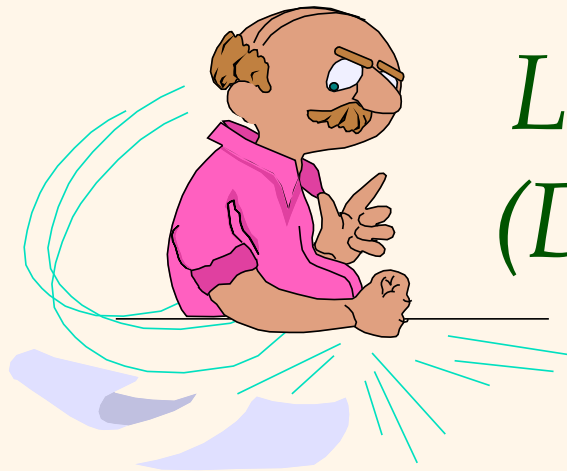


Introduction to Data Management

**** The “Online” Edition ****

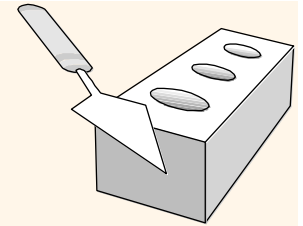
Lecture #2 – Part 2 (Database Design...)



Instructor: Mike Carey
mjcarey@ics.uci.edu

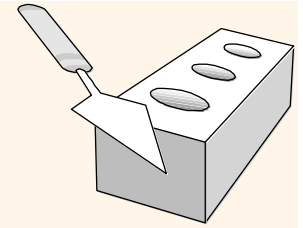


Notices (Addendum)



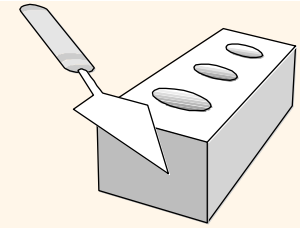
- ❖ The problem with pre-recorded lectures is that one can lose track of which day a particular lecture will be played on!
 - Lecture 2 – Part 1 was recorded yesterday, for example
 - Up-to-the-minute announcements can therefore be missed
 - *Ex:* A McGraw-Hill eBook is coming! (Watch Piazza.)
- ❖ After discussion with the TAs/Readers about grading and checking with the Dean of ICS...
 - Future lectures will be optional this quarter
 - As a result, we will also be eliminating the exams
 - For simplicity, all HW assignments will be cancelled
 - All grades in this class will simply be “A”s

...Not!



❖ Never mind..!

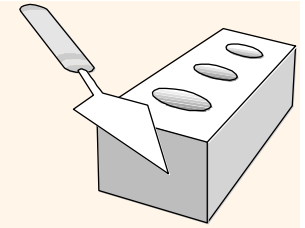
- The information on the wiki still stands. 😊



The Database Design Process

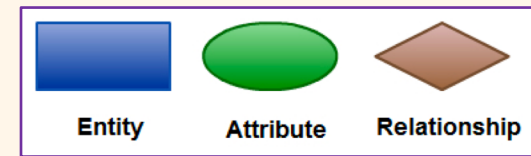
- ❖ Essentially a top-down process.
 1. Requirements gathering (interviews)
 2. Conceptual design (using E-R model)
 3. Platform selection (which DBMS?)
 4. Logical design (for target data model)
 5. Physical design (for target DBMS & workload)
 6. Implement (and test, of course 😊)
- ❖ *Notes:*
 - Expect backtracking, iteration, and then incremental changes over time
 - Our targets: Relational model & RDBMSs

Steps 1 & 2 in Database Design



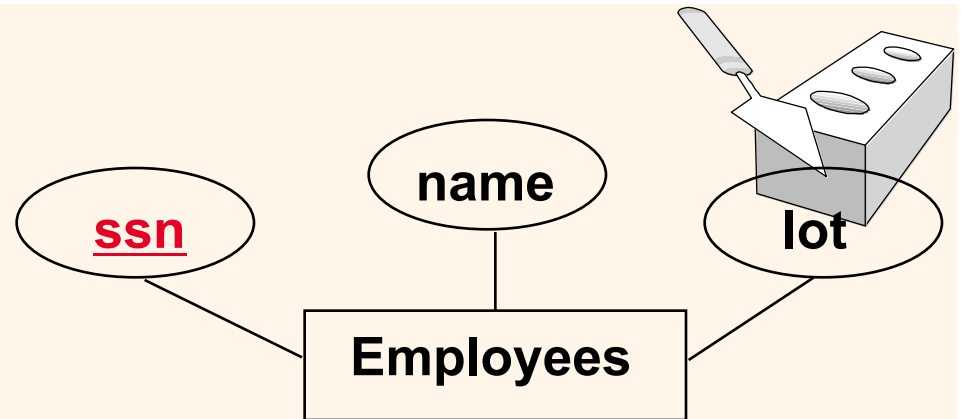
❖ Conceptual design (ER-based):

- What are the *entities* and *relationships* in the enterprise?
- What information about these entities and relationships should we store in the database?
- What are the *integrity constraints* or *business rules* that hold?
- A database schema in the ER Model can be represented pictorially (using an *ER diagram*).
- Can map an ER diagram into a relational schema (manually or using a design tool's automation).



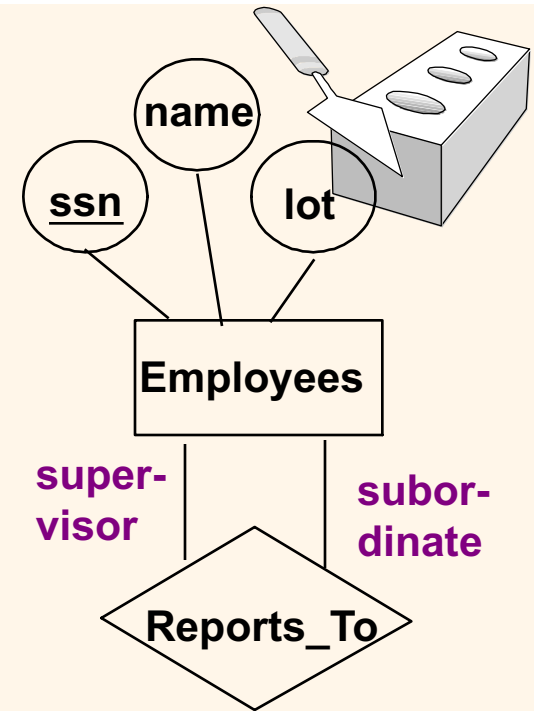
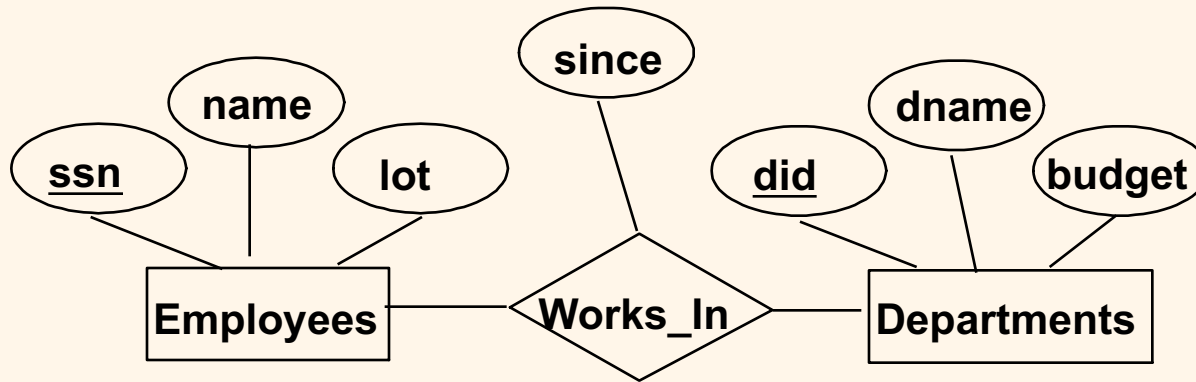
ER Model Basics

(Entities)



- ❖ Entity: Real-world object, distinguishable from all other objects. An entity is described (in DB-land) using a set of attributes.
- ❖ Entity Set: A collection of similar entities. E.g., all employees.
 - All entities in an entity set have the same set of attributes. (Until we get to ISA hierarchies...)
 - Each entity set has a key (a unique identifier); this can be either one attribute (an “atomic” key) or several attributes (called a “composite” key)
 - Each attribute has a domain (similar to a data type).

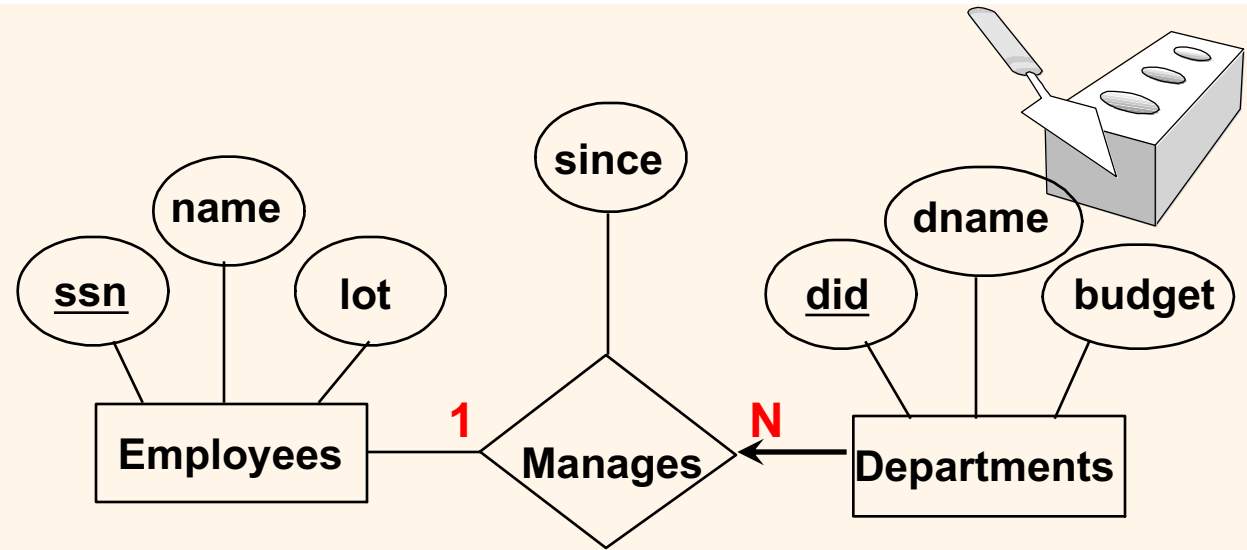
ER Model Basics (Relationships)



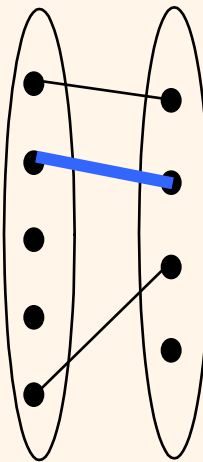
- ❖ **Relationship**: Association among two or more entities. E.g., Santa Claus works in the Toy department.
- ❖ **Relationship Set**: Collection of similar relationships.
 - An n-ary relationship set R relates n entity sets E1 ... En; each relationship in R involves entities e1:E1, ..., en:En
 - One entity set can participate in different relationship sets – or in different “**roles**” in the same set.

Cardinality Constraints

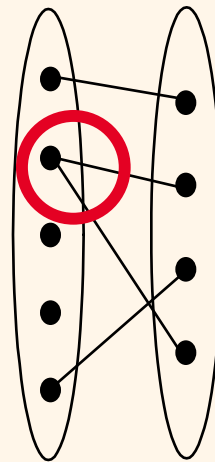
- ❖ Consider Works In:
An employee can work in many departments; a dept can have many employees.
- ❖ In contrast, each dept has at most one manager, according to the cardinality constraint on Manages above.



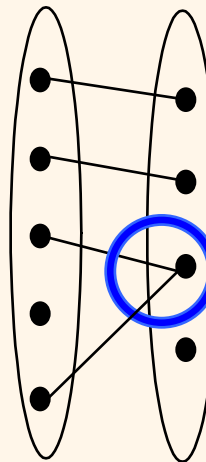
(Note: A given employee can manage several departments)



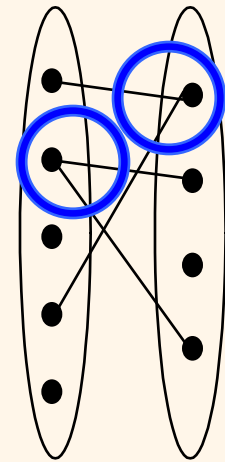
1-to-1
(1:1)



1-to Many
(1:N)



Many-to-1
(N:1)



Many-to-Many
(M:N)

To Be Continued...

