BIRZEIT UNIVERSITY

Faculty of Engineering and Tecnology

Computer Science Department

Database Design

Chapter 2

Entity-Relationship Model

Databases Model the Real World

- "Data Model" translates real world things into structures computers can store
- Many models:
  - Relational, E-R, O-O, Network, Hierarchical, etc.
- Relational (more next time)
  - Rows & Columns
  - Keys & Foreign Keys to link Relations

Enrolled

sid	cid	grade
53666	Carnatic101	C
53666	Reggae203	B
53650	Topology112	A
53666	History105	B

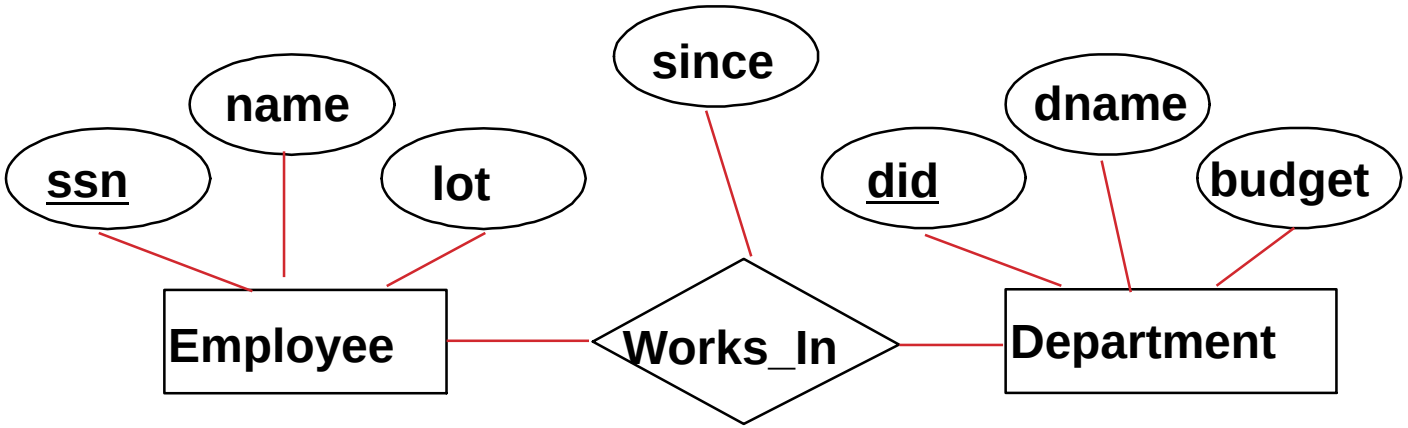
Students

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

One Solution: The E-R Model

- Instead of relations, it has:
  - Entities and Relationships

- These are described with diagrams
  - both structure, notation more obvious to humans



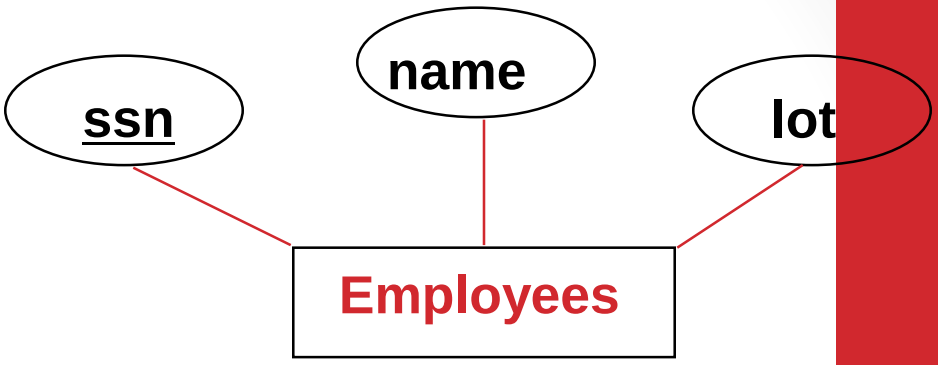
Steps in Database Design

- **Requirements Analysis**
  - user needs; what must database do and have?
- **Conceptual Design**
  - high level descr (often done w/ER model)
- **Logical Design**
  - translate ER into DBMS data model
- **Schema Refinement**
  - consistency, normalization
- **Physical Design**
  - indexes, disk layout
- **Security Design**
  - who accesses what, and how

Example: DBA for Bank of America

- Requirements Specification
  - Determine the requirements of clients (Database to store information about customers, accounts, loans, branches, transactions, …)
- Conceptual Design
  - Express client requirements in terms of E/R model.
  - Confirm with clients that requirements are correct.
  - Specify required data operations
- Logical Design
  - Convert E/R model to relational, object-based, XML-based,…
- Physical Design
  - Specify file organizations, build indexes

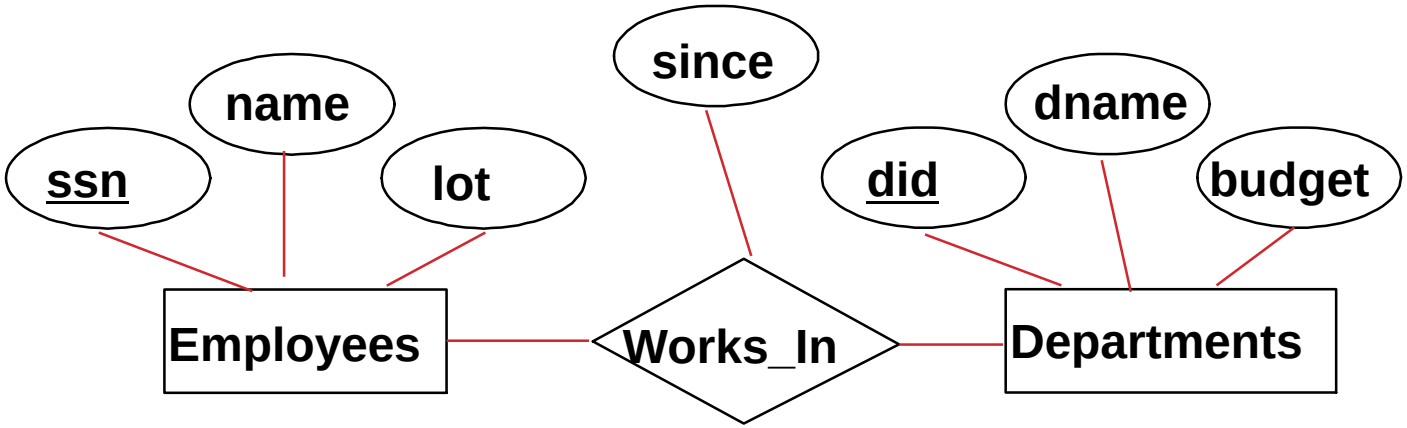
ER Model Basics



- ***Entity:***
  - Real-world thing, distinguishable from other objects.
  - Noun phrase (e.g., Bob Smith, Comm Ave Branch, Account 1234, etc)
  - Entity described by 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 consider hierarchies, anyway!)
  - Each attribute has a *domain*.

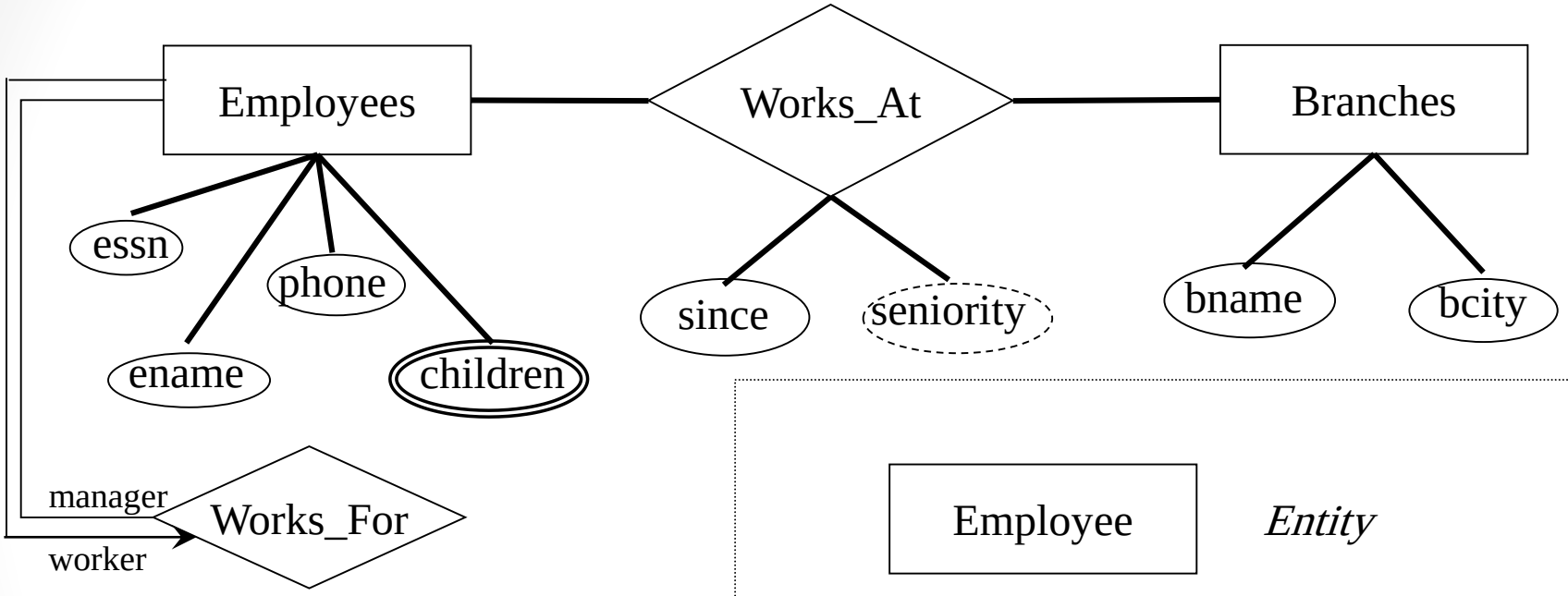
ER Model Basics (Contd.)



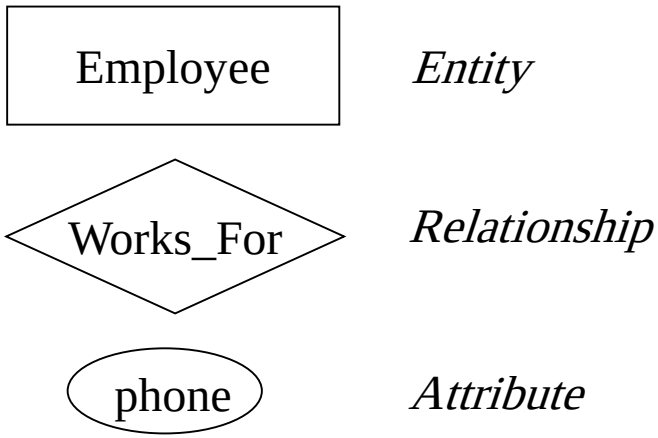
- ***Relationship***: Association among two or more entities. E.g., Bob Smith works in Pharmacy department.
  - relationships can have their own attributes.
  - Verb phrases (e.g., works_at, enrolled_in, etc)
- **Relationship Set:** Collection of similar relationships.
  - An *n*-ary relationship set *R* relates *n* entity sets $E_1 \ldots E_n$; each relationship in *R* involves entities $e_1 \in E_1, \ldots, e_n \in E_n$

E/R Data Model

An Example

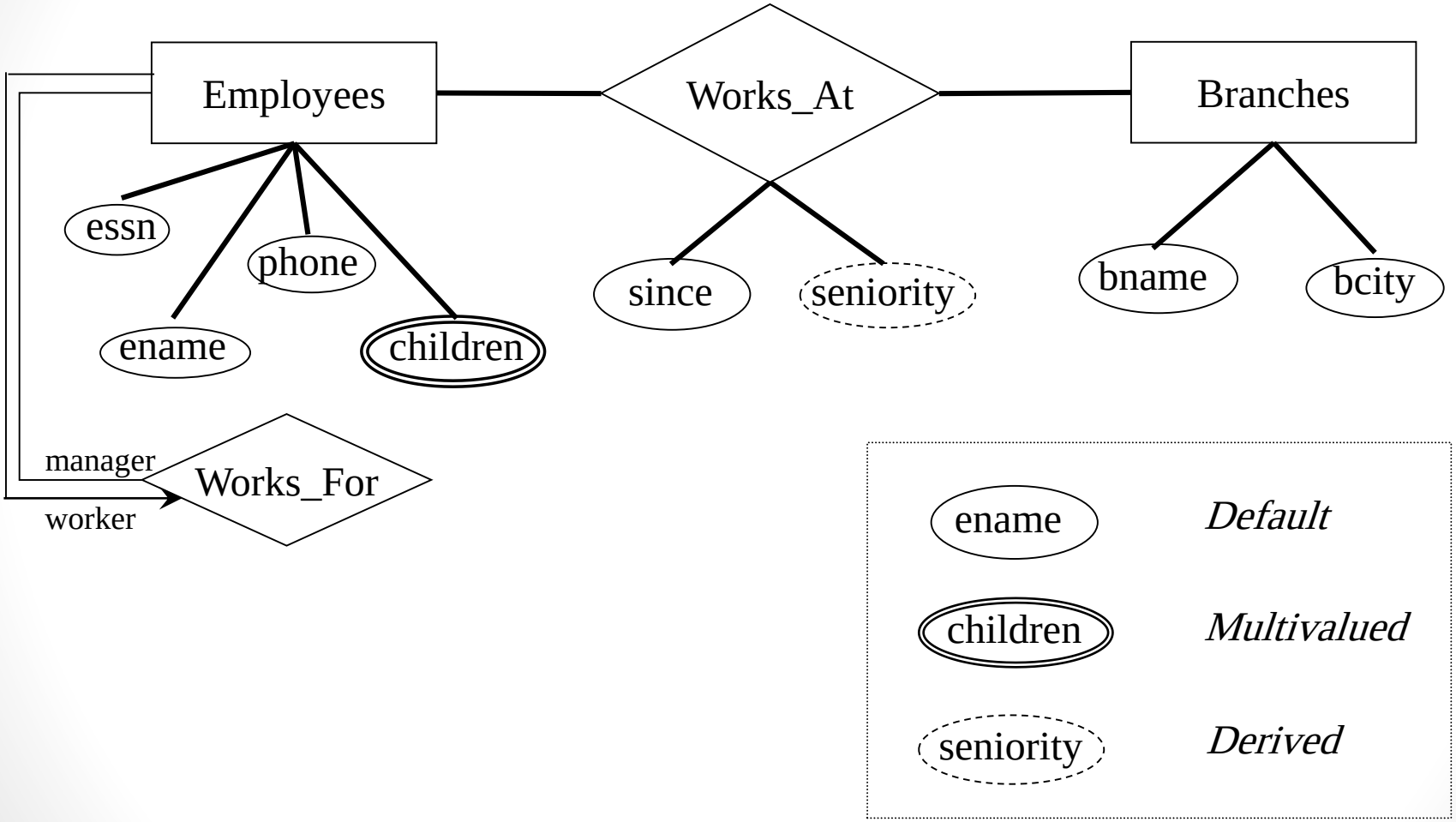


Lots of notation to come.



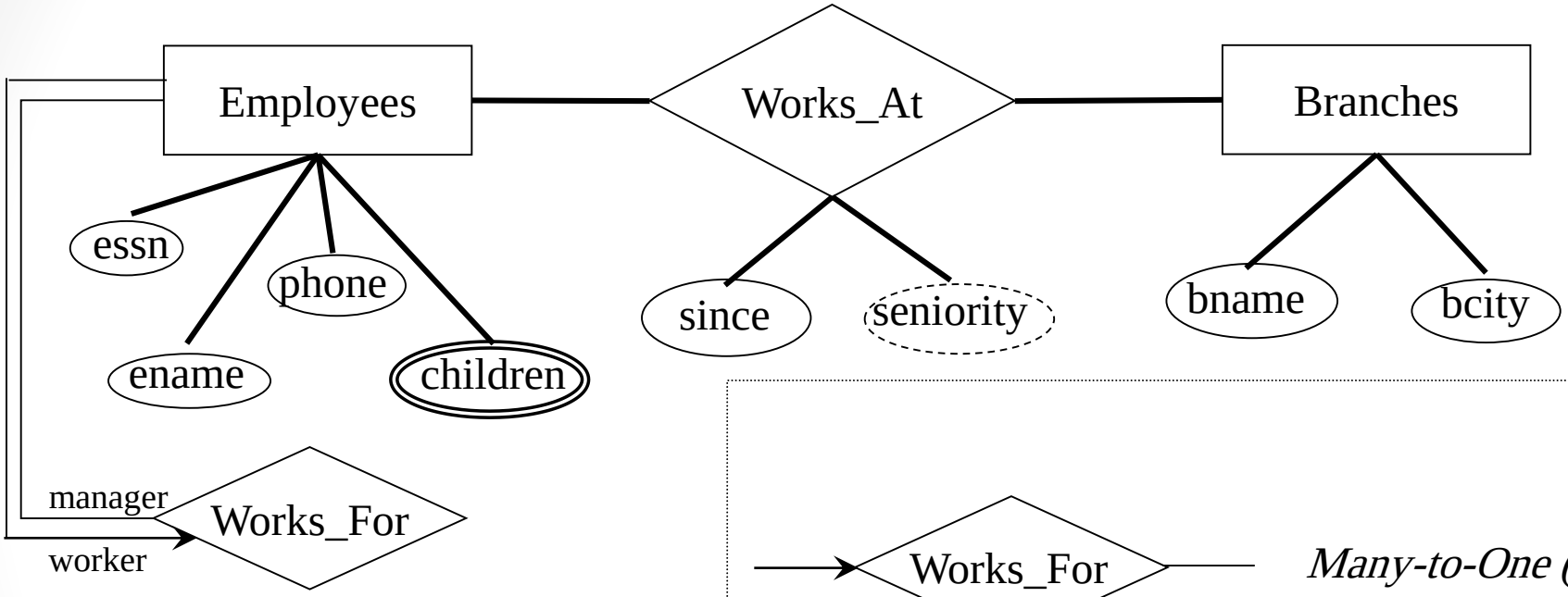
E/R Data Model

Types of Attributes



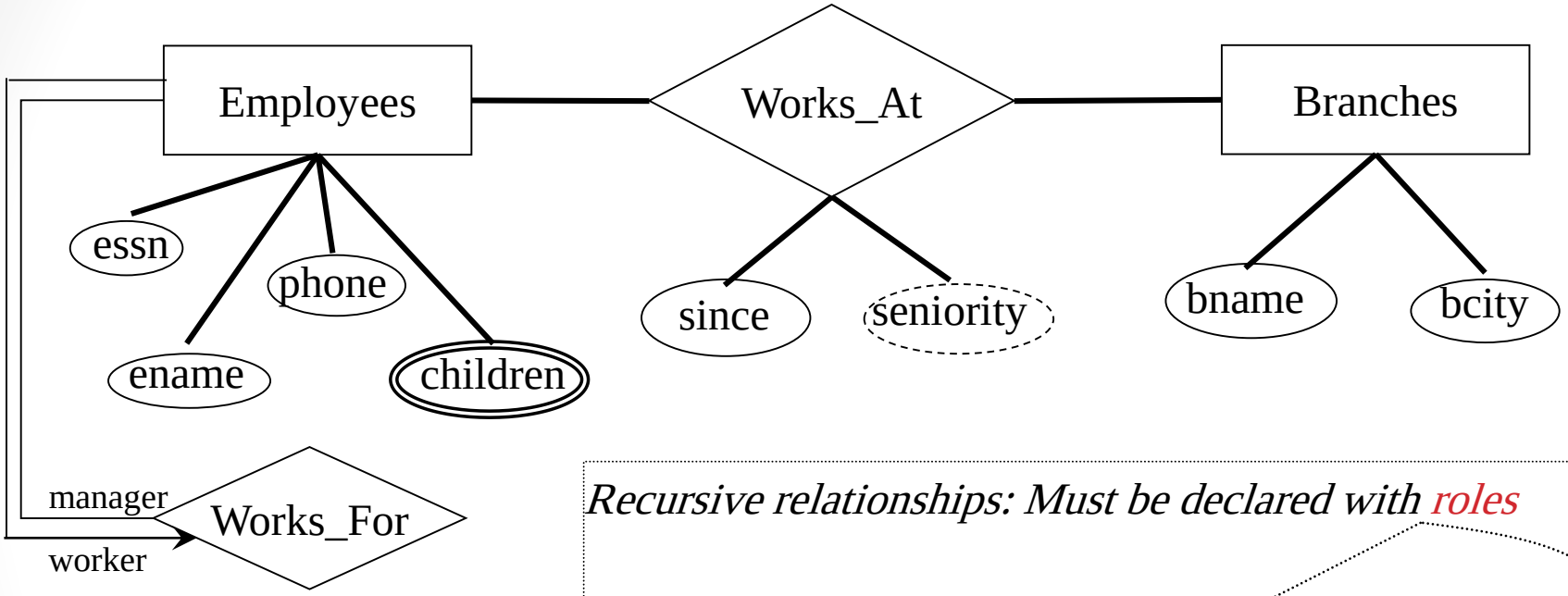
E/R Data Model

Types of relationships

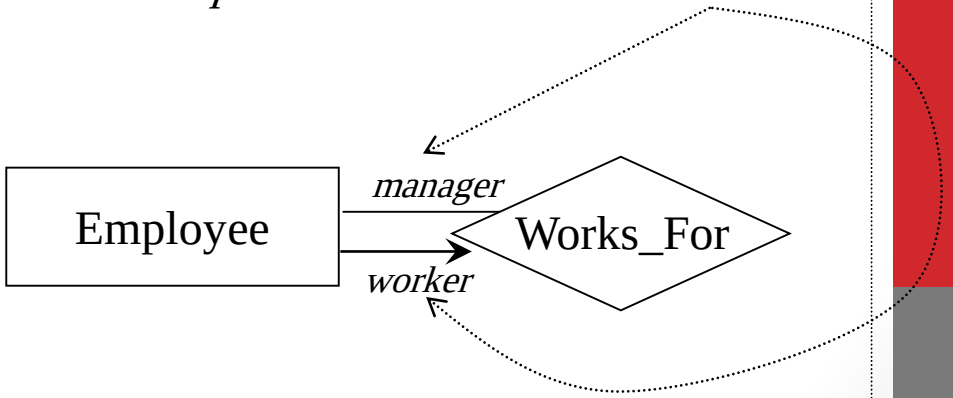


E/R Data Model

unary/Recursive relationships



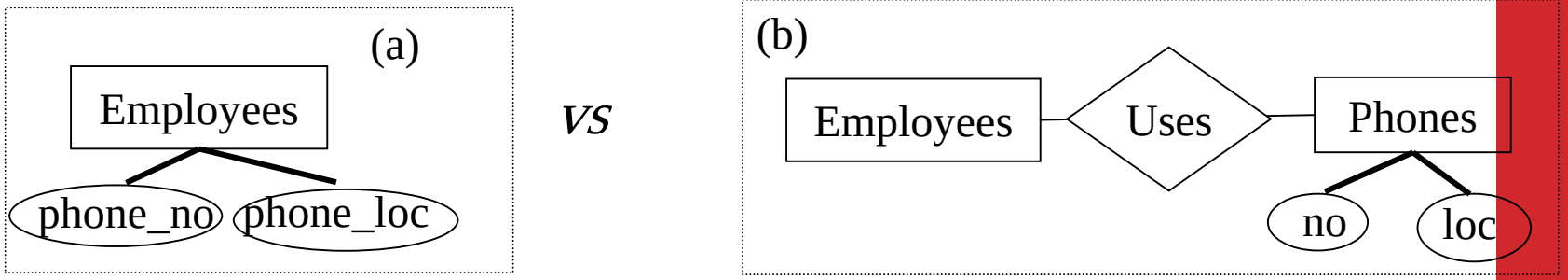
Recursive relationships: Must be declared with roles



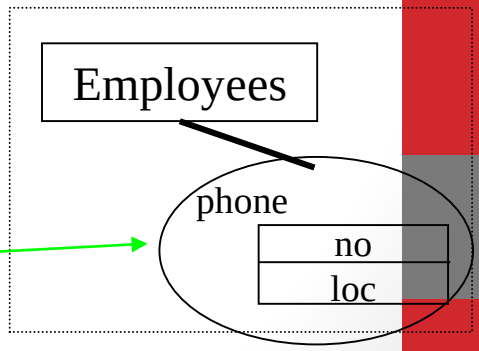
E/R Data Model

Design Issue #1: Entity Sets vs. Attributes

- An Example: Employees can have multiple phones



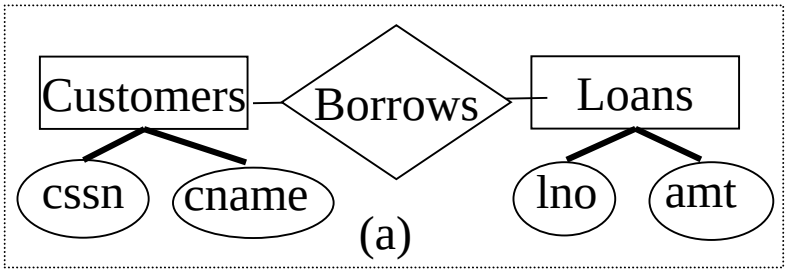
- To resolve, determine how phones are used
  - 1. Can many employees share a phone?
  - (If yes, then (b))
  - 2. Can employees have multiple phones?
  - (if yes, then (b), or (a) with multivalued attributes)
  - 3. Else
  - (a), perhaps with composite attributes



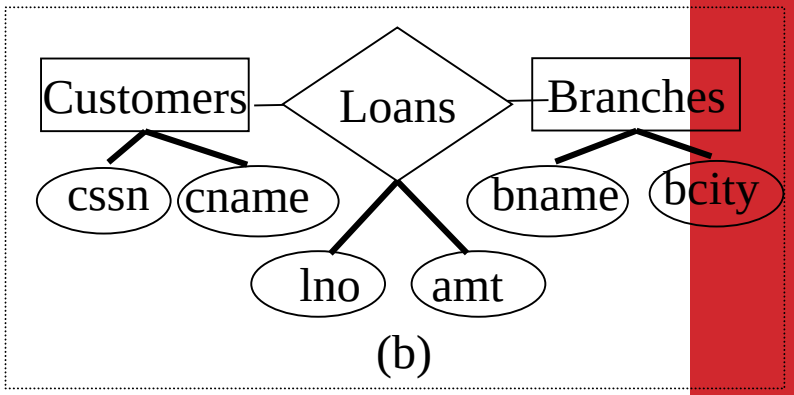
E/R Data Model

Design Issue #2: Entity Sets vs. Relationship Sets

- An Example: How to model bank loans



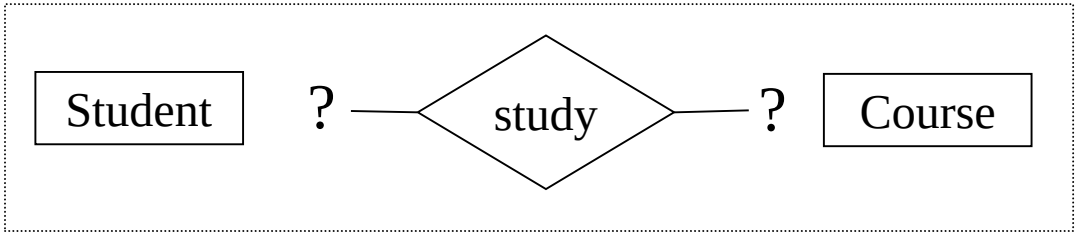
vs



- To resolve, determine how loans are issued
  - 1. Can there be more than one customer per loan?
    - *If yes, then (a). Otherwise, loan info must be replicated for each customer (wasteful, potential update anomalies)*
  - 2. Is loan a noun or a verb?
    - *Both, but more of a noun to a bank. (hence (a) probably more appropriate)*

E/R Data Model

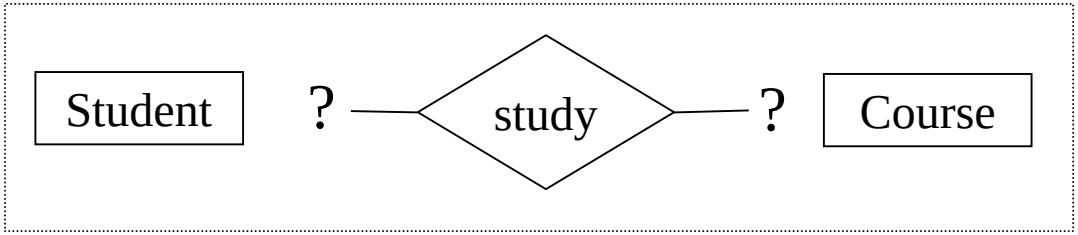
Design Issue #3: Relationship Cardinalities

- An Example:
  - 

- Variations on study:
  - 1. Can a student study multiple courses?
  - 2. Can a course be jointly held by more than 1 student?

E/R Data Model

Design Issue #3: Relationship Cardinalities



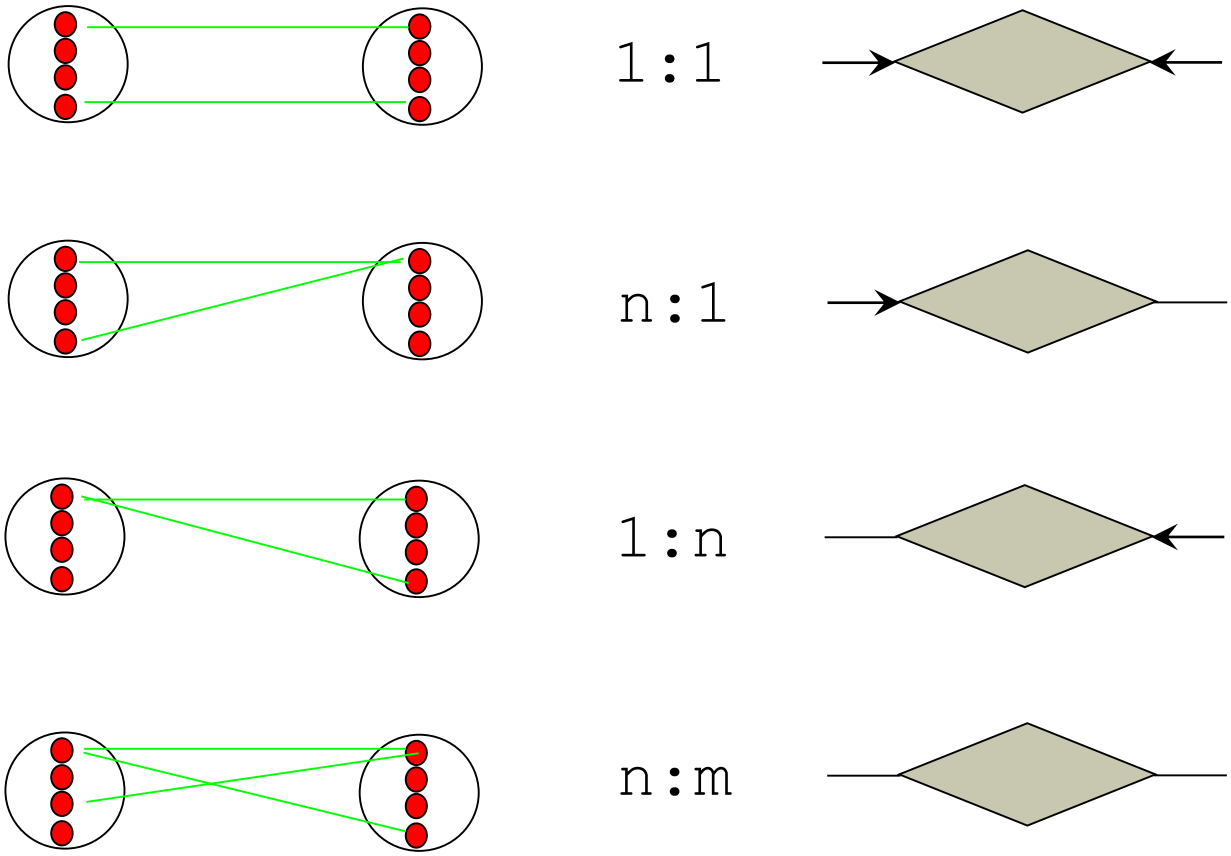
- Cardinalities of study:
  -

Type	Illustrated	Multiple courses?	Joint courses?
One-to-One (1:1)	study	No	No
Many-to-one (n:1)	study	No	Yes
One-to-many (1:n)	study	Yes	No
Many-to-many (n:m)	study	Yes	Yes

E/R Data Model

Design Issue #3: Relationship Cardinalities (cont)

- In general...
  - ù

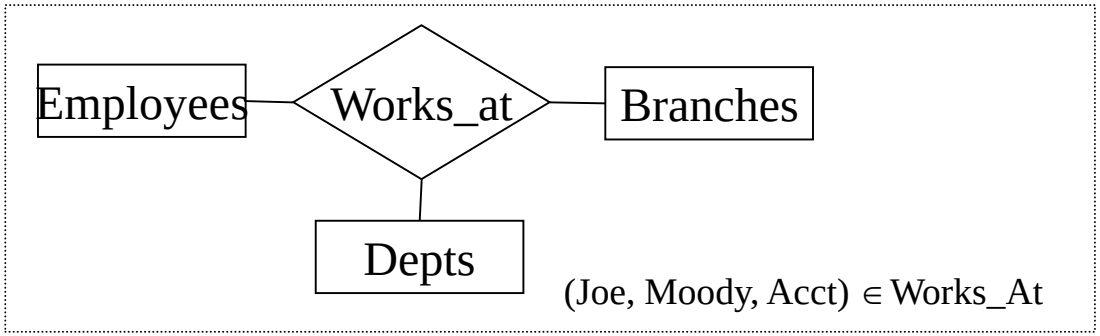


E/R Data Model

Design Issue #4: N-ary vs Binary Relationship Sets

- An Example: Works_At

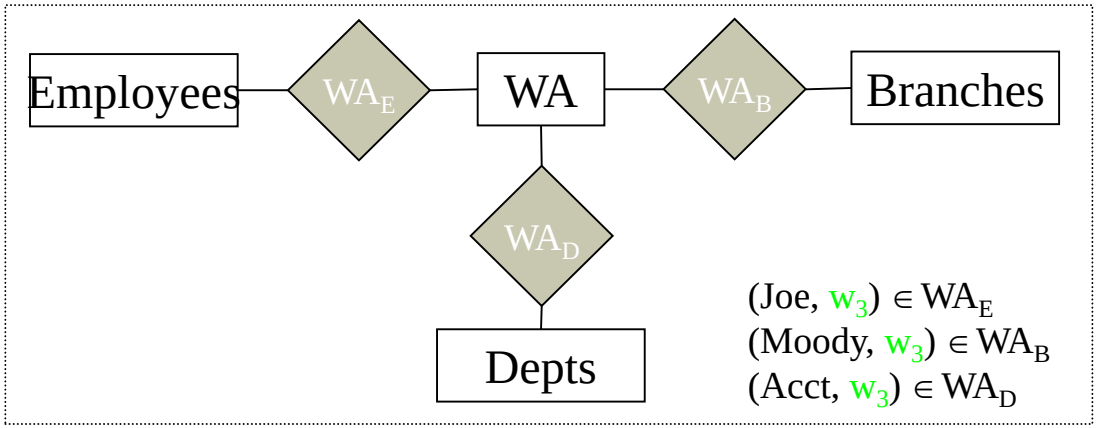
Ternary:



Choose n-ary when possible! (Avoids redundancy, update anomalies)

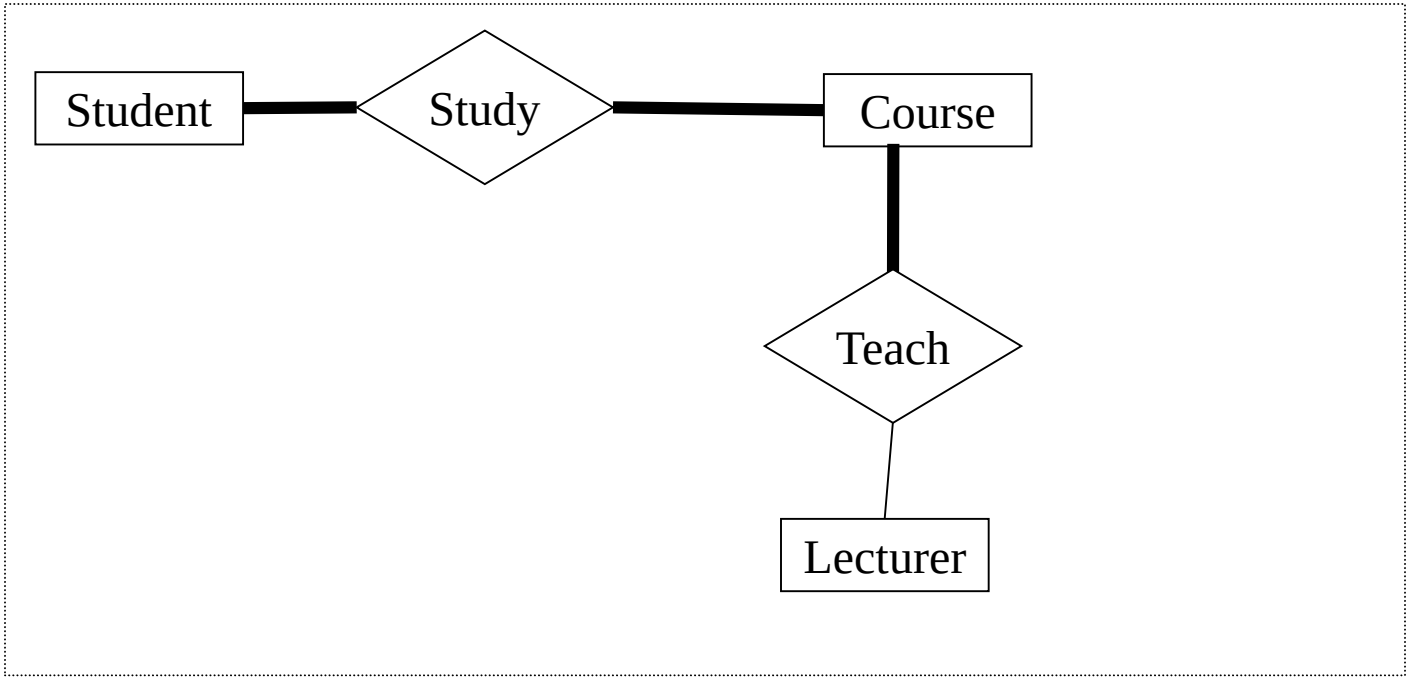
vs

Binary:



E/R Data Model

Design Issue #5: Total participation vs. Partial participation Relationship

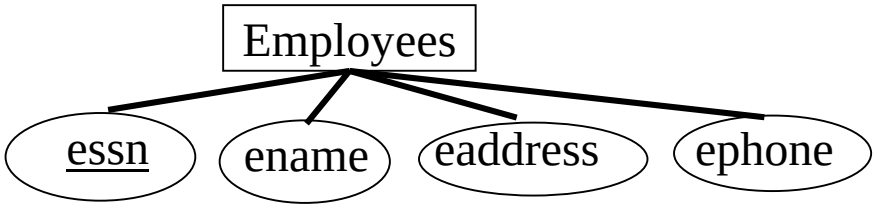


- Variations
- 1. The student must study at least one course.
  - **Total** participation: Thick line
- 2. There is No constraint that the lecturer must teach courses.
  - **Partial** participation: Normal line

E/R Data Model

Keys

- Key = set of attributes identifying individual entities or relationships



- A. Superkey:
  - *any attribute set that distinguishes identities*
  - *e.g., {essn}, {essn, ename, eaddress}*
- B. Candidate Key:
  - *"minimal superkey" (can't remove attributes and preserve "keyness")*
  - *e.g., {essn}, {ename, eaddress}*
- C. Primary Key:
  - *candidate key chosen as **the** key by a DBA*
  - *e.g., {essn} (denoted by underline)*

E/R Data Model

Existence Dependencies and Weak Entity Sets

- Idea:
  - Existence of one entity depends on another
- Example: The Employee insurance can cover all of his children's

Weak Entity Set

Identifying Relationship

Total Participation

E/R Data Model

Existence Dependencies and Weak Entity Sets

- Idea:
  - Existence of one entity depends on another
- Example: The Employee insurance can cover all of his children's

Weak Entity Set

Identifying Relationship

Total Participation

E/R Data Model

Weak Entity

Weak Entity Sets

- existence of Childs depends upon Employee
- have no superkeys: different childs records (for different Employees) can be identical
- instead of keys, discriminators: discriminate between childs for a given Employee

- We say:
  - Employee is owner in Insurance_coverage
  - Child is weak entity

E/R Data Model

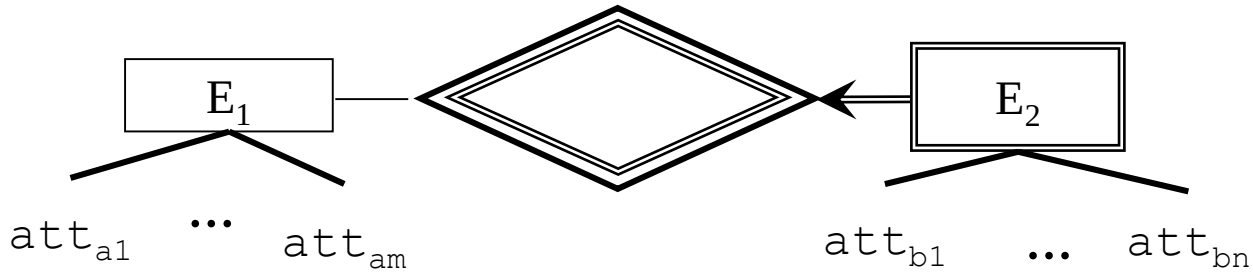
Weak Entity

═══ Total Participation

One-to-many

E/R Data Model

Existence Dependencies and Weak Entity Sets



- Q. Is $\{att_{b1}, \ldots, att_{bn}\}$ a superkey of E_2?

  A: No

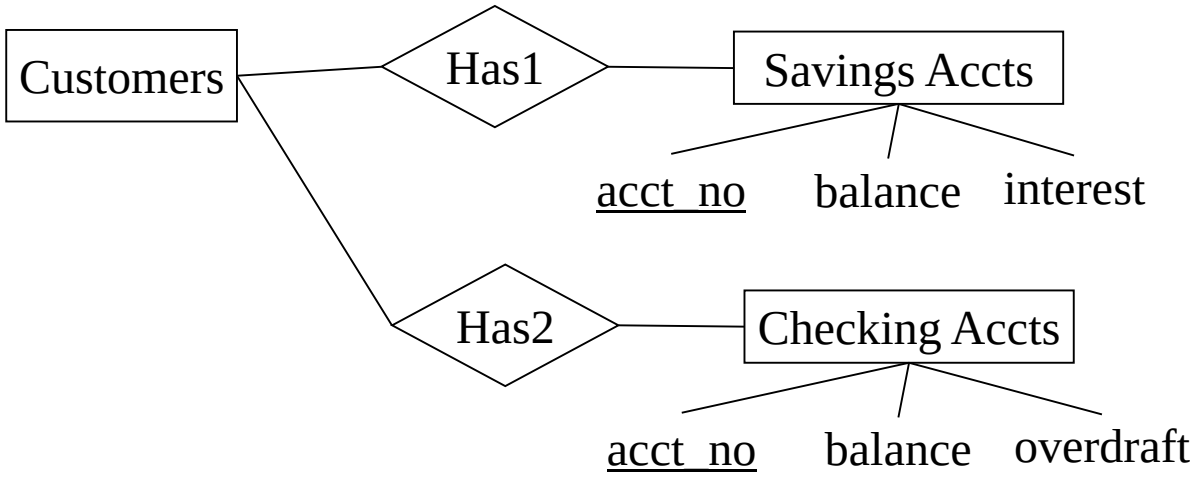
- Q. Name a candidate key of E_2

  A: $\{att_{a1}, att_{b1}\}$

E/R Data Model

Extensions to the Model: Specialization and Generalization

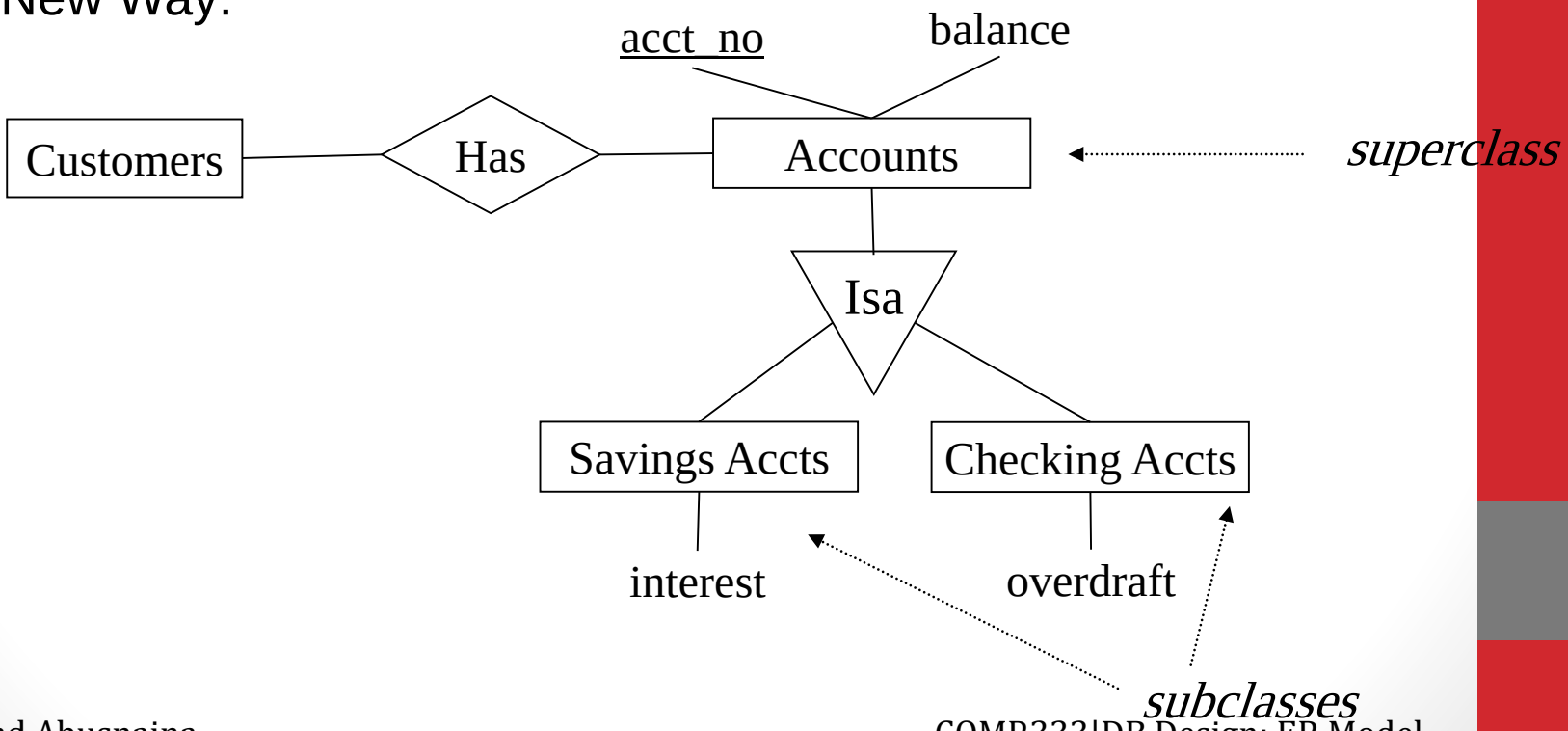
- An Example:
  - Customers can have checking and savings accts
  - Checking ~ Savings (many of the same attributes)
- Old Way:



E/R Data Model

ISA : Specialization and Generalization

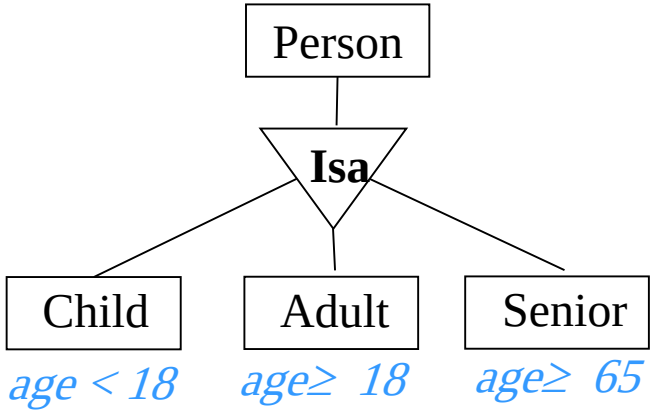
- An Example:
  - Customers can have checking and savings accts
  - Checking ~ Savings (many of the same attributes)
- New Way:



E/R Data Model

Extensions to the Model: Specialization and Generalization

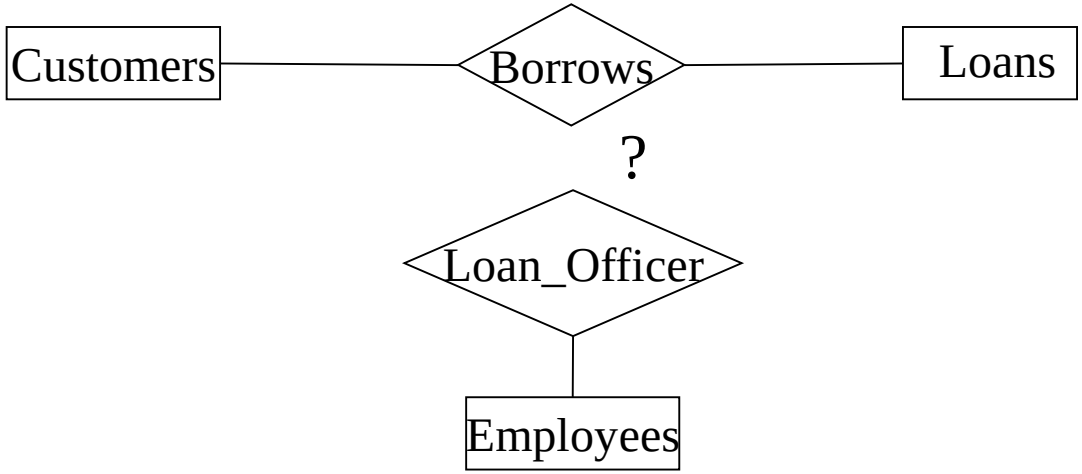
- Subclass Distinctions:
  - 2. Overlapping vs. Disjoint
    - *Overlapping: Entities can belong to >1 entity set (e.g., Adult, Senior)*
    - *Disjoint: Entities belong to exactly 1 entity set (e.g., Child)*



E/R Data Model

Extensions to the Model: Aggregation

- E/R: No relationships between relationships
  - E.g.: Associate loan officers with Borrows relationship set



- Associate Loan Officer with Loan?

E/R Data Model

Summary

- *Entities, Relationships (sets)*
- *Both can have attributes (simple, multivalued, derived, composite)*
- *Cardinality or relationship sets (1:1, n:1, n:m)*
- *Keys: superkeys, candidate keys, primary key*
  - *DBA chooses primary key for entity sets*
  - *Automatically determined for relationship sets*
- *Weak Entity Sets, Existence Dependence, Total/Partial Participation*
- *Specialization and Generalization (E/R + inheritance)*

These things get pretty!

- Many E-R diagrams will be covered!