Introduction to Database Systems

CSE 444

Lecture #7
Jan 24 2001

Announcements

- **Programming Assignment due tomorrow [Thu (1/25)]**
- **Mid Term Syllabus**
  - **Material in lectures**
  - **Textbook**
    - **Chapter 1, Chapter 2 (except 2.1 and ODL)**
    - **Chapter 3 (except 3.2), Chapter 4 (except 4.2, 4.3)**
    - **Chapter 5 (except 5.10)**
    - **Chapter 6, Chapter 7 (except 7.2)**
  - **Mid Term will be <u>in class closed book</u> exam**

Database Schema Design

Today's Reading:

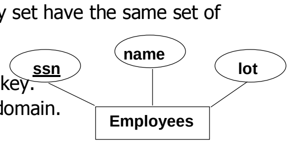
Sec 2 (except 2.1 and ODL discussions) and
Sec 3.1- 3.4 (except 3.1)

Overview of Database Schema Design

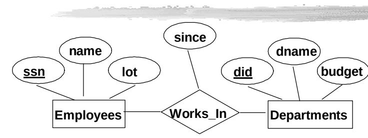
- <u>Conceptual design</u> (ER Model)
  - ER Diagram
    - What are the entities and relationships in the enterprise?
    - What are the integrity constraints or business rules that hold?
  - Map an ER diagram into a relational schema
- <u>Schema Refinement</u> (Normalization):
    - Check relational schema for redundancies and related anomalies
- <u>Physical Design</u>:
    - Determine physical structures

ER Model Basics

- <u>Entity</u>: Real-world object distinguishable from other objects. An entity is described (in DB) using a set of <u>attributes</u>.
- <u>Entity Set</u>: A collection of similar entities. E.g., all employees.
  - All entities in an entity set have the same set of attributes.
  - Each entity set has a key.
  - Each attribute has a domain.



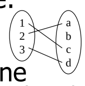
ER Model Basics

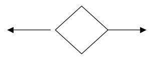


- Entity and Entity Set
  - Attributes (atomic but may be null)
- Relationship and Relationship Set
  - Attributes (atomic)

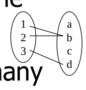
Multiplicity of E/R Relationships

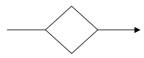
⌘one-one:



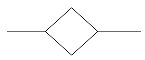


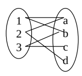
⌘many-one



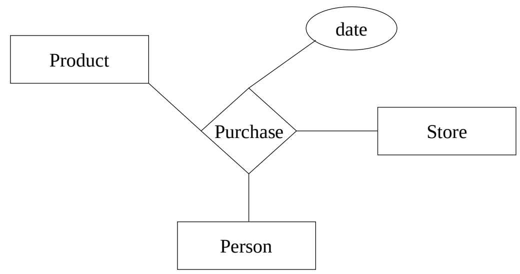


⌘many-many



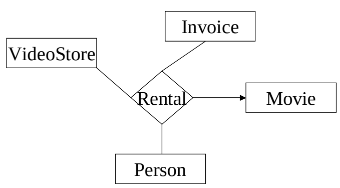


Multi-way Relationship



Arrows in Multiway Relationships (1)

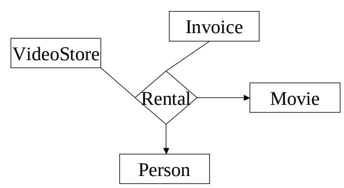
⌘Q: what does the arrow mean ?



⌘A: store, person, invoice determines the movie

Arrows in Multiway Relationships (2)

⌘Q: what do these arrow mean ?

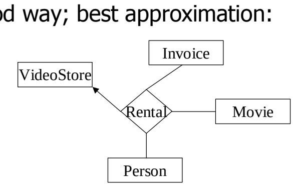


⌘A: store, person, invoice determines movie and store, invoice, movie determines person

Arrows in Multiway Relationships (3)

⌘Q: how do I say: “invoice determines store” ?

⌘A: no good way; best approximation:



⌘Why is this incomplete ?

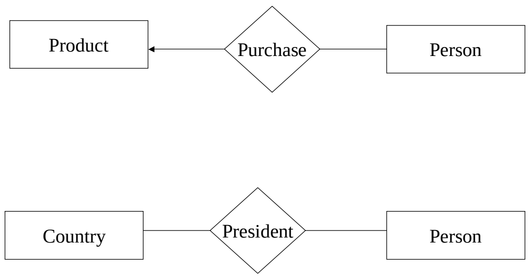
Design of ER Models

⌘Picking the right kind of element requires care
- Entity vs. attribute, entity vs. relationship, binary or n-ary relationship, roles,..
- Read Example 2.17 for illustration

⌘ Some design principles that help:
- Faithfulness
- Avoidance of redundancy
- Simplicity

Example1: Design Principles

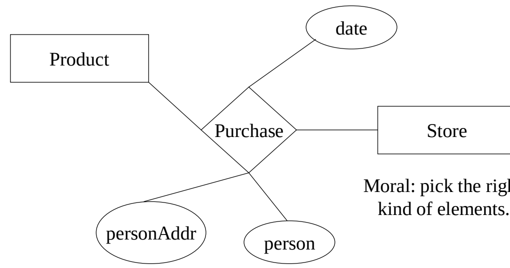
What's wrong?



Moral: be faithful!

13

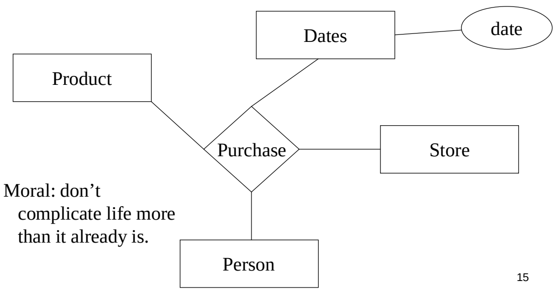
Example2: Design Principles



Moral: pick the right kind of elements.

14

Example3: Design Principles

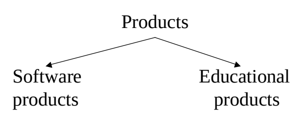


Moral: don't complicate life more than it already is.

15

Modeling Subclasses

Some objects in a class may be special
- •define a new class
- •better: define a *subclass*

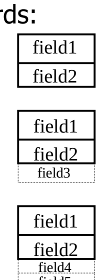


So --- we define subclasses (in ODL and in E/R).

16

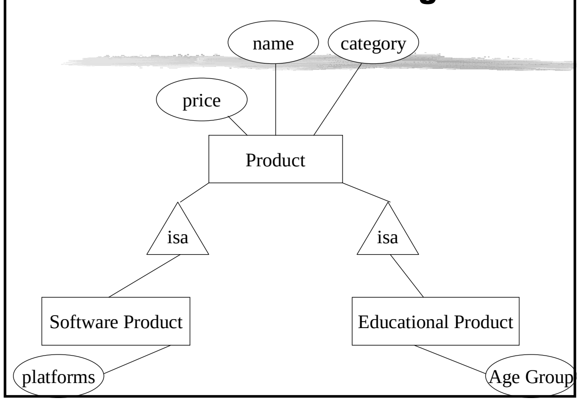
Understanding Subclasses

⌘Think in terms of records:
- ◰Product
- ◰SoftwareProduct
- ◰EducationalProduct



17

Subclasses in E/R Diagrams



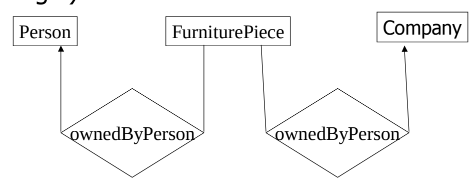
Modeling Union Types With Subclasses

Say: each piece of furniture is owned either by a person, or by a company

Modeling Union Types with Subclasses

Say: each piece of furniture is owned either by a person, or by a company

Solution 1. Acceptable, imperfect (What's wrong ?)



Constraints

- A constraint = an *assertion* about the database that must be true at all times
- Part of the database schema
- Correspond to *invariants* in programming languages

Modeling Constraints

Finding constraints is part of the modeling process.
Commonly used constraints:

Keys: social security number uniquely identifies a person.

Single-value constraints: a person can have only one father.

Referential integrity constraints: if you work for a company, it must exist in the database.

Domain constraints: peoples' ages are between 0 and 150.
General constraints: all others (at most 50 students enroll in a class)

Keys

A set of attributes that uniquely identify an object or entity:

Person: social security number
name
name + address
name + address + age

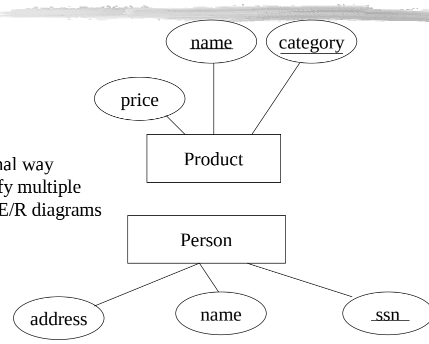
Perfect keys are often hard to find, so organizations usually invent something.

An object may have multiple keys:

employee number, social-security number

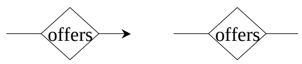
Keys in E/R Diagrams

No formal way to specify multiple keys in E/R diagrams



Single Value Constraints

- Each attribute can only have atomic (single) value
- Many to one relationship



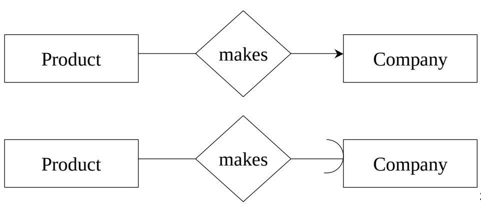
25

Referential Integrity Constraints

- The Referential Integrity Constraint explicitly requires a reference to exist
  - Specialization of single value constraint
  - Avoids situations where we refer to an object but get garbage instead
    - e.g. a dangling pointer in C/C++

26

Example: Referential Integrity



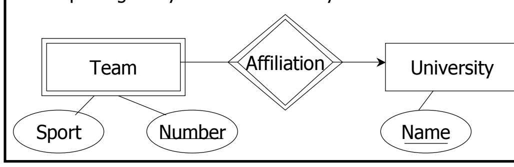
27

Weak Entity Sets

Entity sets are weak when their key attributes come from other classes to which they are related.

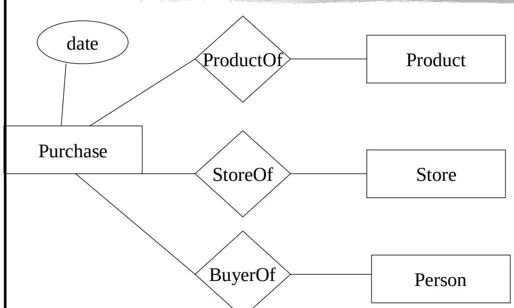
This happens if:

- part-of hierarchies
- splitting n-ary relations to binary.



28

Weak Entity Sets



29

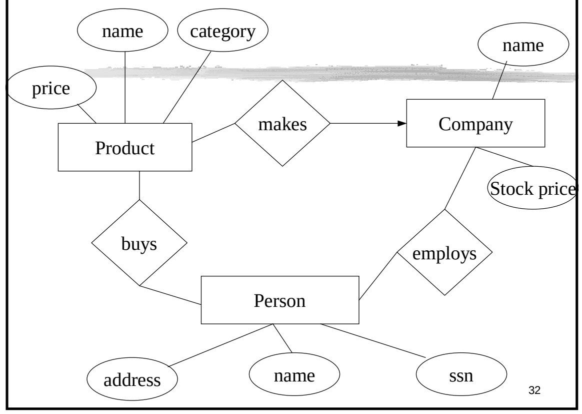
Modeling Constraints in ER

- Constraints play an important role in determining the best database design for an enterprise
- Several kinds of integrity constraints can be expressed in the ER model:
  - Keys
  - Referential constraints
- Some constraints cannot be expressed in the ER model:
  - Some functional dependencies
  - Domain constraints

30

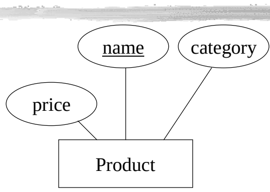
Translating E/R Diagrams into Relational Schemas

Reading: Chapters 3.1, 3,3, 3.4



32

Entity Sets to Relations

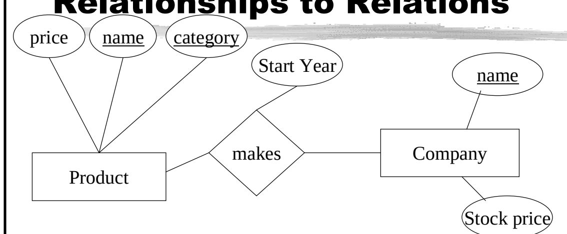


Product:

Name	Category	Price
gizmo	gadgets	$19.99

33

Relationships to Relations

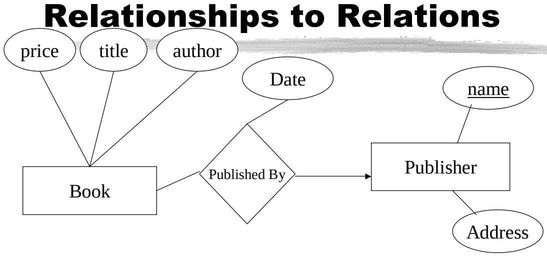


elation Makes (watch out for attribute name conflicts)

Product-name	**Product-Category**	**Company-name**	Starting-year
gizmo	gadgets	gizmoWorks	1963

34

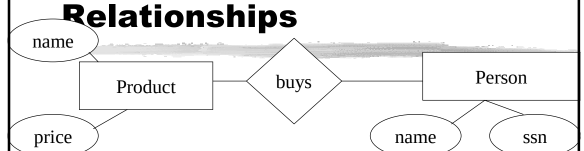
Relationships to Relations



Any other Options?

35

Rules for Binary Relationships



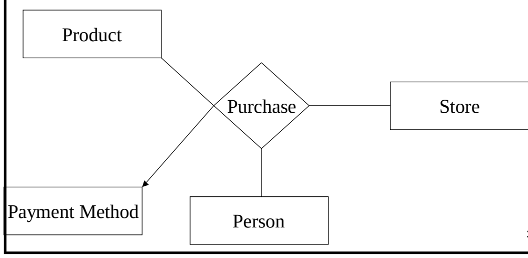
Several cases are possible for a binary relationship:

1. Many-many:
2. Many-one:
3. One-one:

36

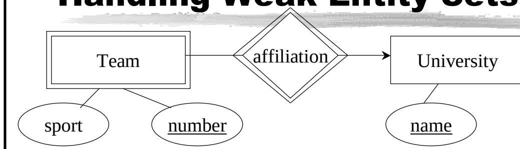
Example: Multiway Relationships

How will you map Purchase to a relation?



37

Handling Weak Entity Sets



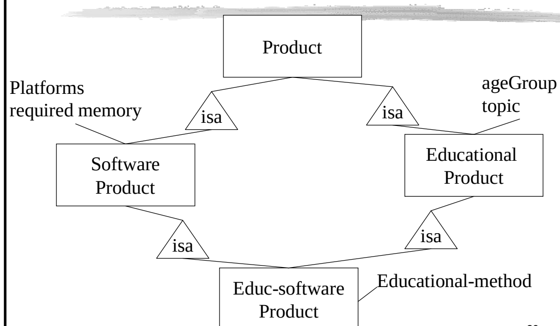
Relation Team:

Sport	Number	Affiliated University Name
mud wrestling	15	Montezuma State U.

- need all the attributes that contribute to the key of Team
- don't need a separate relation for Affiliation. (why ?)

38

Modeling Subclass Structure



39

Mapping Subclasses

Product(<u>name</u>, price, category, manufacturer)

EducationalProduct(<u>name</u>, ageGroup, topic)

SoftwareProduct(<u>name</u>, platforms, requiredMemory)

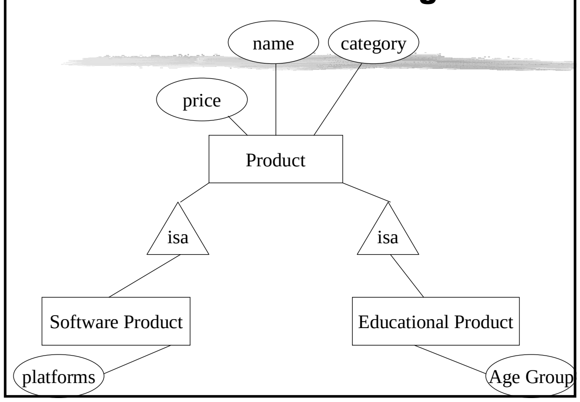
No need for a relation EducationalSoftwareProduct

Unless, it has a specialized attribute:

EducationalSoftwareProduct(<u>name</u>, educational-method)

40

Subclasses in E/R Diagrams



SQL for Keys and Reference Keys

```
CREATE TABLE  Books (
   isbn  CHAR(11),
   title  CHAR(20),
   pubname CHAR(25),
   pubdate  DATE,
   PRIMARY KEY  (isbn),
   FOREIGN KEY (pubname) REFERENCES Publishers (name))
```

42

E/R to Relations: Summary

- Entity set → Relation
- M-N Relationship → Relation (keys of related entities plus relationship attributes)
- Special Cases:
  - M-1 Relationship
  - 1-1 Relationship
- The resultant relational schema may have some nasty properties….

43

Functional Dependencies

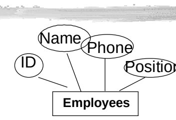
Reading: Chapter 3.5, 3.6

Motivation

- Subjective nature of E-R diagram may not capture all relationships
- ER → Relational translation may not satisfactory :
  - Some relationships not easily captured by translation
- Recognizing Functional Dependencies helps refine schema

45

Example



EmpID	Name	Phone	Position
E0045	Smith	1234	Clerk
E1847	John	9876	Salesrep
E1111	Smith	9876	Salesrep
E9999	Mary	1234	lawyer

46

Example (Contd)

- The specific ER Diagram did not capture that position has a unique telephone number
- What if:
  - All current salespersons resign
  - Can I update Smith's phone?
  - Can I add a salesperson *Roy* with phone *6923*?

47

The Evils of Redundancy

- Redundancy is at the root of these problems:
  - redundant storage, insert/delete/update anomalies
- Integrity constraints, in particular functional dependencies, can be used to identify problem schema
- Use decomposition judiciously to overcome these issues
  - Replacing ABCD with, say, AB and BCD, or ACD and ABD

Functional Dependencies

Definition:

If two tuples agree on the attributes

$A_1, A_2, \dots A_n$

then they must also agree on the attributes

$B_1, B_2, \dots B_m$

Formally: $A_1, A_2, \dots A_n \longrightarrow B_1, B_2, \dots B_m$

Motivating example for the study of functional dependencies:

Name	Social Security Number	Phone Number

49

In General

⌘To check $A \longrightarrow B$, erase all other columns

…	A	…	B	
	X1		Y1	
	X2		Y2	
	…		…	

⌘check if the remaining relation is many-one (called ***functional*** in mathematics)

50

Example

EmpID	Name	Phone	Position
E0045	Smith	1234 ←	Clerk
E1847	John	9876 ←	Salesrep
E1111	Smith	9876 ←	Salesrep
E9999	Mary	1234 ←	lawyer

51

Keys and SuperKeys

Product: name → price, manufacturer
Person: ssn → name, age
Company: name → stock price, president

Key of a relation is a set of attributes that:

- functionally determines all the attributes of the relation
- none of its subsets determines all the attributes.

Superkey: a set of attributes that contains a key.

52

A Property of Functional Dependency

Splitting/Combining Lemma

$A_1, A_2, \dots A_n \longrightarrow B_1, B_2, \dots B_m$ Is equivalent to

$A_1, A_2, \dots A_n \longrightarrow B_1$
$A_1, A_2, \dots A_n \longrightarrow B_2$
…
$A_1, A_2, \dots A_n \longrightarrow B_m$

53

Inferring Implied Functional Dependency

Reflexivity

$A_1, A_2, \dots A_n \longrightarrow A_i$ Always holds

Why ?

54

Inferring Implied Functional Dependency (contd.)

Augmentation Rule:

If $A_1, A_2, \ldots A_n \longrightarrow C_1, C_2 \ldots, C_p$

then $A_1, A_2, \ldots A_n$ B1,..BK $\longrightarrow C_1, C_2 \ldots, C_p$ B1,..BK

Why ?

55

Inferring Implied Functional Dependency (contd.)

Transitive Closure Rule:

If $A_1, A_2, \ldots A_n \longrightarrow B_1, B_2 \ldots, B_m$

and $B_1, B_2, \ldots B_m \longrightarrow C_1, C_2 \ldots, C_p$

then $A_1, A_2, \ldots A_n \longrightarrow C_1, C_2 \ldots, C_p$

Why ?

56

Inference of Implied FD (contd.)

- Armstrong's axioms
  - Reflexivity
  - Augmentation
  - Transitivity
- A sound and complete inference rule to obtain all implied functional dependencies

57

Closure of a set of Attributes

Given a set of attributes {**A1, ..., An}** and a set of dependencies S.

Problem: find all attributes *B* such that:
any relation which satisfies S also satisfies:
A1, ..., An → B

The **closure** of *{A1, ..., An}*, denoted $\{A1, \ldots, An\}^+$, is the set of all such attributes *B*

What is the relationship between closure and keys?

58

Closure Algorithm

Start with X={A1, ..., An}.

Repeat until X doesn't change **do**:

if $B_1, B_2, \ldots B_n \longrightarrow C$ is in S, **and**
$B_1, B_2, \ldots B_n$ are all in X, **and**
C is not in X

then

add C to X.

59

Example

A B → C
A D → E
B → D
A F → B

Closure of {A,B}: X = {A, B, }

Closure of {A, F}: X = {A, F, }

60