

CS422 Principles of Database Systems Entity-Relationship Model

Chengyu Sun
California State University, Los Angeles

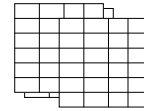
Designing Tables Is Not Easy

Problem in Real World

#42 Some Restaurant	
Date: Jul 09, 2008	Time: 03:07PM
Server: John	# of Guest: 2
Bill: 0060	Table: 42
1 Boiled Pork Wanton	4.95
1 Dumpling w/Crabmeat	8.00
1 Beef Noodle Soup	6.80
Subtotal:	19.75
GST:	0.99
Total:	20.74
Open Time: Jul 09, 2008 02:57PM	
Printed by: Cashier	



Tables in RDBM



Entity-Relationship (ER) Model

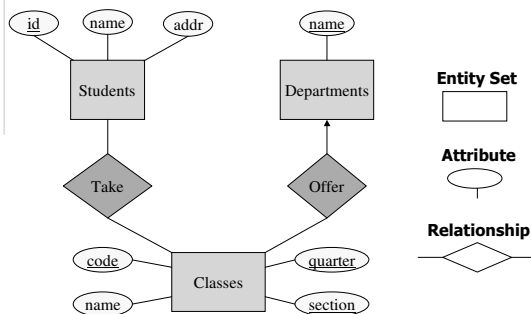
Problem → ER Model → Tables

- ◆ An *object-oriented* approach
- ◆ A visual representation of the design – ER Diagram
- ◆ Easily converted to relational model

Example: Problem Description

- ◆ Student
 - id, name, address
- ◆ Department
 - name
- ◆ Classes
 - code, name, quarter, section number
- ◆ Class offerings and enrollment

Example: ER Diagram



Entity Set and Attributes

- ◆ Entity Set is similar to *class* in an OO language
- ◆ Attributes are the properties of an entity set
 - Similar to the *class fields* in an OO language
 - Must have simple values like numbers or strings, i.e. *cannot be collection or composite type*

Keys

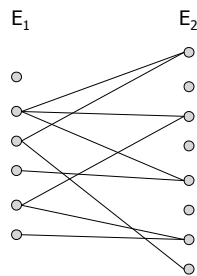
- ◆ A key is an attribute or a set of attributes that *uniquely* identify an entity in an entity set.
- ◆ Each entity set must have a key
- ◆ If there are multiple keys, choose one of them as the *primary key*

Types of Relationships

- ◆ Many-to-Many
- ◆ Many-to-One / One-to-Many
- ◆ One-to-One

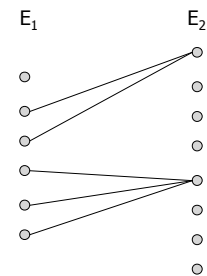
Many-to-Many Relationship

- ◆ Each entity in E_1 can be related to many entities in E_2
- ◆ Each entity in E_2 can be related to many entities in E_1



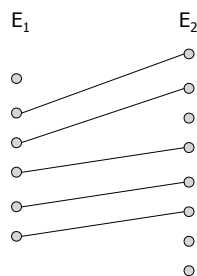
Many-to-One Relationship

- ◆ Each entity in E_1 can be related to one entity in E_2
- ◆ Each entity in E_2 can be related to many entities in E_1



One-to-One Relationship

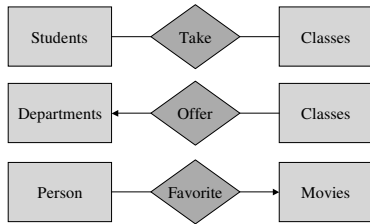
- ◆ Each entity in E_1 can be related to one entity in E_2
- ◆ Each entity in E_2 can be related to one entity in E_1



Relationship Type Examples

- ◆ Students and classes??
- ◆ Departments and classes??
- ◆ Person and Favorite movie??

Relationship Types in ER Diagram

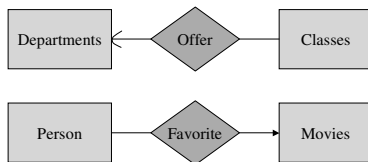


◆ An arrow is used to indicate the "one" side

A Closer Look at "One" and "Many"

- ◆ One
 - 0 or 1
 - Exactly 1 → Referential Integrity
- ◆ Many
 - 0..N
 - 1..N
 - N..M (*Example??*)

Referential Integrity in ER Diagram



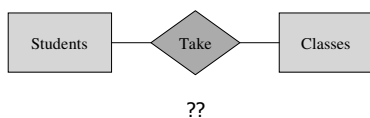
◆ An circular arrow is used to indicate "Exactly 1"

One vs. Exactly One

- ◆ Both lead to foreign key constraint in SQL
 - One: foreign key + NULL
 - Exactly one: foreign key + NOT NULL
- ◆ It's usually not too important to distinguish the two in ER design

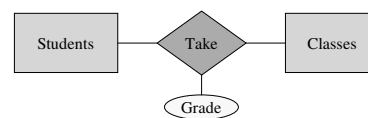
Example: Grades

- ◆ Store the grades the students received for their classes
- ◆ A grade is a single letter A, B, C, D, or F



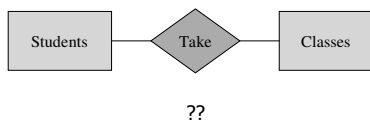
Relationship Attributes ...

- ◆ Sometimes it's useful to attach an attribute to a relationship.



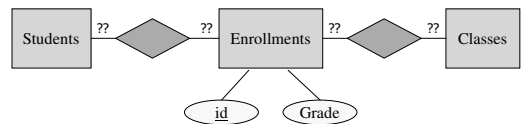
... Relationship Attributes ...

- ◆ Some variations of ER model does not allow relationships to have attributes



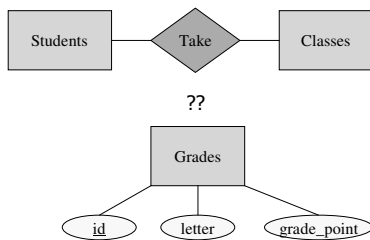
... Relationship Attributes

- ◆ If something needs an attribute, it probably should be an entity set

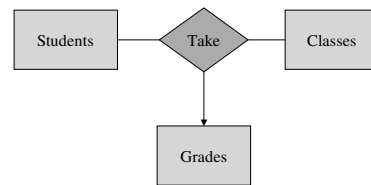


Example: More about Grades

- ◆ Make *Grades* an entity set



Multiway Relationship

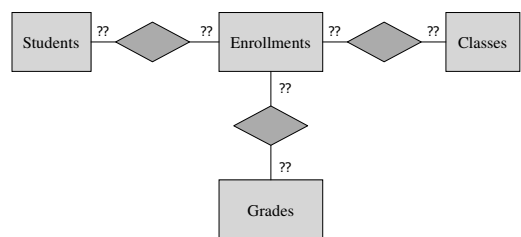


- ◆ Why there is an arrow pointing to *Grades*??

"Arrows" in Multiway Relationships

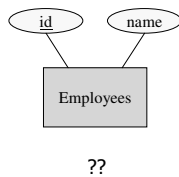
- ◆ In multiway relationships, an arrow points to an entity set **E** means that if we select one entity from each of the other entity sets in the relationship, those entities are related to at most one entity in **E**.

Convert Multiway Relationship to Binary Relationship



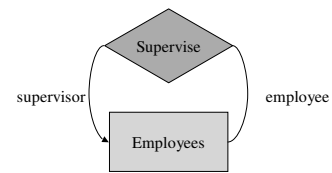
Example: Employees and Supervisors

- ◆ Each employee has a supervisor
- ◆ A supervisor is an employee

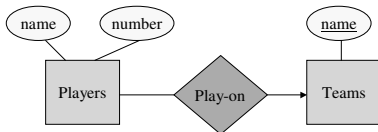


Roles

- ◆ An entity set may appear in the same relationship more than once.
- ◆ Label the edges with names called Roles



Example: Players and Teams

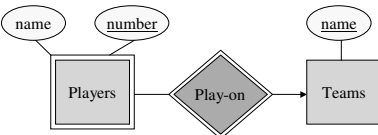


- ◆ What's the key for *Players*??

Weak Entity Set

- ◆ Entity set E is said to be weak if in order to identify entities of E uniquely, we need to follow one or more many-one relationships from E and include the key of the related entities from the connected entity sets.

Weak Entity Sets in ER Diagram

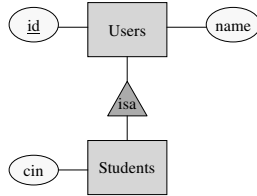


- ◆ The key of a weak entity set consists of its own key attributes *and* the key attributes of the supporting set

From Weak to Strong

- ◆ We can usually create unique IDs for entity sets

Subclass



- ◆ Super class must have all the key attributes

Summary of ER Diagram

- ◆ Entity Set
 - Attributes, key
 - Weak entity set
- ◆ Relationship
 - Many-to-Many, Many-to-One, One-to-One
 - Attributes
 - Multiway relationship
 - Subclass

Relational Model

- ◆ Proposed by Edgar F. Codd in early 1970's
- ◆ Data is stored in tables (a.k.a. relations)
- ◆ All major database systems these day are relational

student_id	first_name	last_name	birthday
2000001	John	Doe	1970-1-1
2000002	Jane	Doe	1971-1-1
2000003	Tom	Smith	1962-2-2

About Relational Model

- ◆ Attributes must be of simple type
- ◆ No order among attributes
- ◆ No order among records

Table (Relation)

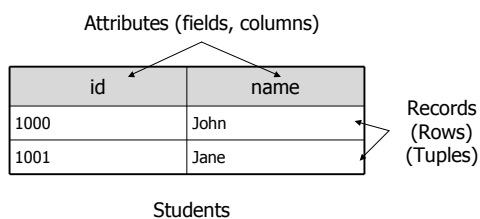


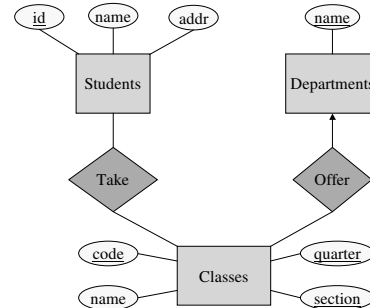
Table and Database Schema

- ◆ Table schema
 - Name of the table, and the names and types of the attributes
 - E.g.
 - students(id:integer, name:string)
 - or just students(id, name)
- ◆ Database schema
 - Schemas of all the tables in the database

Basic Rules of ER to Relational Conversion

- ◆ A entity set is converted to a table
- ◆ A many-to-many relationship is also converted to a table, including
 - Its own attributes
 - Key attributes from the associated entity sets
- ◆ A many-to-one relationship is merged into the "many" side with a foreign key to the "one" side

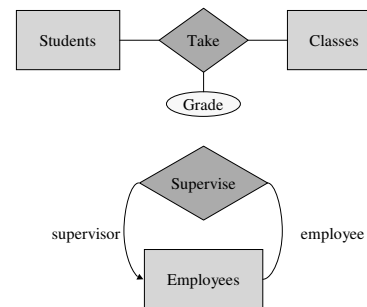
Conversion Example: ER Diagram



Conversion Example: Relational Schema

Students(id, name, address)
 Departments (name)
 Classes (code, name, quarter, section, department_name)
 Takes(student_id, code, quarter, section)

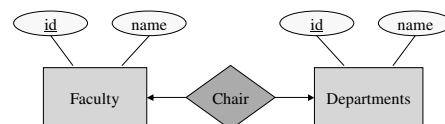
More Conversion Examples



Special Cases of Conversion

- ◆ One-to-One relationship
- ◆ Multiway relationship
- ◆ Weak entity set
- ◆ Subclass

Converting One-to-One Relationship ...



... Converting One-to-One Relationship

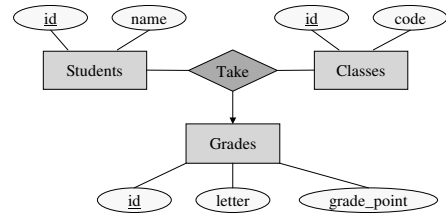
◆ Which one of the following makes more sense??

Faculty(id, name, chair_of_department)
 Departments(id, name)

or

Faculty(id, name)
 Departments(id, name, department_chair)

Converting Multiway Relationship

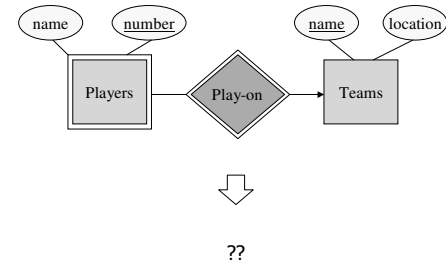


◆ Should this relationship be treated as many-to-many or many-to-one??

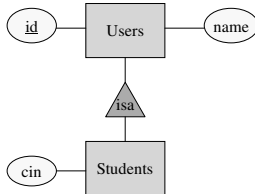
Converting Weak Entity Set ...

- ◆ The table for a weak entity set includes its *complete key* as well as its own non-key attributes
- ◆ A supporting relationship is redundant and yields no relation

... Converting Weak Entity Set



Converting Subclass ...



... Converting Subclass

- ◆ Object-oriented approach
 - One table per class
 - Each entity belongs to exact one table
- ◆ ER approach
 - One table per class
 - Each entity may appear in multiple tables
- ◆ NULL approach
 - One table per class hierarchy

Object-Oriented Approach

id	name
1000	John

Users

id	name	cin
1001	Jane	212345678

Students

ER Approach

id	name
1000	John
1001	Jane

Users

user_id	cin
1001	212345678

Students

NULL Approach ...

id	name	cin
1000	John	NULL
1001	Jane	212345678

Users

... NULL Approach

Discriminator field

id	user_type	name	cin
1000	staff	John	NULL
1001	student	Jane	212345677

Users

Comparison of Subclass Conversion Approaches

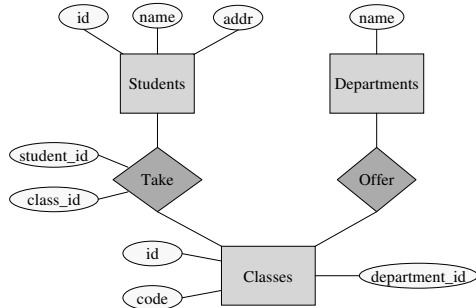
- ◆ Constraints and data integrity
- ◆ Query performance

Q1: find the number of users
Q2: find the number of students

Summary of ER to Relational Conversion

- ◆ Basic rules
 - Entity set
 - Many-to-many relationship
 - Many-to-one relationship
- ◆ Special cases
 - One-to-one relationship
 - Multiway relationship
 - Weak entity set
 - Subclass

Common Problems in ER Diagram



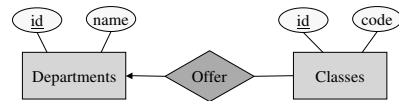
Common Mistakes of ER Design

- ◆ ER diagram
 - Missing arrows
 - Missing keys
 - Redundant foreign keys
- ◆ ER to relational conversion
 - Missing tables for many-to-many relationships

Variations of ER Model

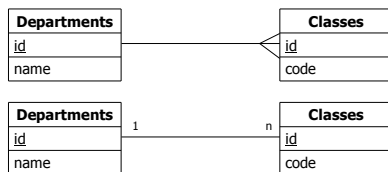
- ◆ Rule differences
 - Relationships cannot have attributes
 - Anything that have attributes should be an entity set
 - Simplifies ER diagram
 - No multiway relationship
 - Better mapping to OO languages
- ◆ Notational differences
 - Classic ER diagram
 - OO style ER diagram

Classic ER Diagram



- ◆ Highlight the three distinct components of the ER model
- ◆ Allow relationship attributes and multiway relationships
- ◆ *Difficult to draw*
- ◆ *Can look messy even for a relatively simple schema*

OO Style ER Diagram

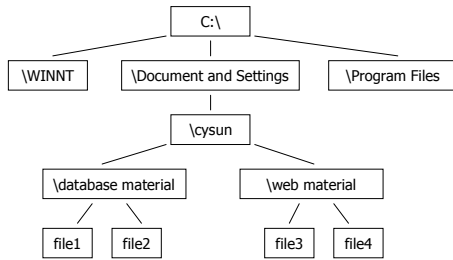


- ◆ Easier to draw, especially using class/table design tools
- ◆ Looks cleaner
- ◆ *No relationship attributes*
- ◆ *No multiway relationships*

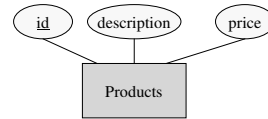
Design Example 1: Restaurant

#42 Some Restaurant	
Date: Jul 09, 2008	Time: 03:07PM
Server: John	# of Guest: 2
Bill: 0060	Table: 42
1 Boiled Pork Wanton	4.95
1 Dumpling w/Crabmeat	8.00
1 Beef Noodle Soup	6.80
	Subtotal: 19.75
	GST: 0.99
Total:	20.74
Open Time: Jul 09, 2008 02:57PM	
Printed by: Cashier	

Design Example 2: Folders and Files



Design Example 3: Price That Changes



What if we want to model price that changes??

