

Discussion #1

Elena, Diwanshu

About Us

Elena Milkai

Grad Student in Computer Science Department,
UW-Madison.

Research Area: Database Systems

Email: milkai@wisc.edu

Diwanshu Jain

Grad Student in Computer Science Department,
UW-Madison.

Research Area: Optimization, Machine Learning,
Computer Vision, Systems

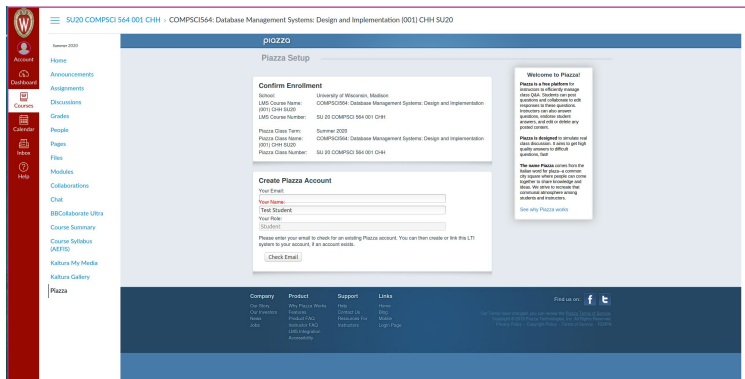
Email: djain23@wisc.edu

Outline

- Piazza
- Basic concepts needed for HW1
- Example
- Q&A

Piazza

1. Sign Up from Canvas



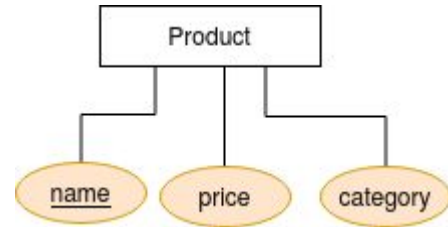
2. Search for “SU 20 COMPSCI 564 001 CHH” on Piazza

ER Model

- visual language used to specify:
 - the information that the DB must hold
 - the relationships between the components of that information
- consists of:
 - Entities
 - Attributes
 - Relationships
 - Constraints

Entity & Attributes

- **Entity:**
 - an object distinguishable from other object
- **Entity Set:**
 - a collection of similar entities
 - represented by rectangles
 - described using a set of attributes
 - must have a key attribute
- **Attribute:**
 - represented by ovals attached to an entity set
 - has atomic domain: string, integer, real, etc.



Relationships

- Association among two or more entities.
- Represented by diamonds.
- Multiplicity of relationships

- One-to-one



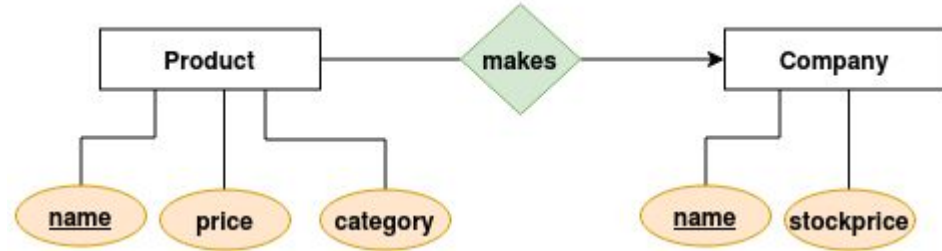
- Many-to-one



- Many-to-many



- Can have attributes.

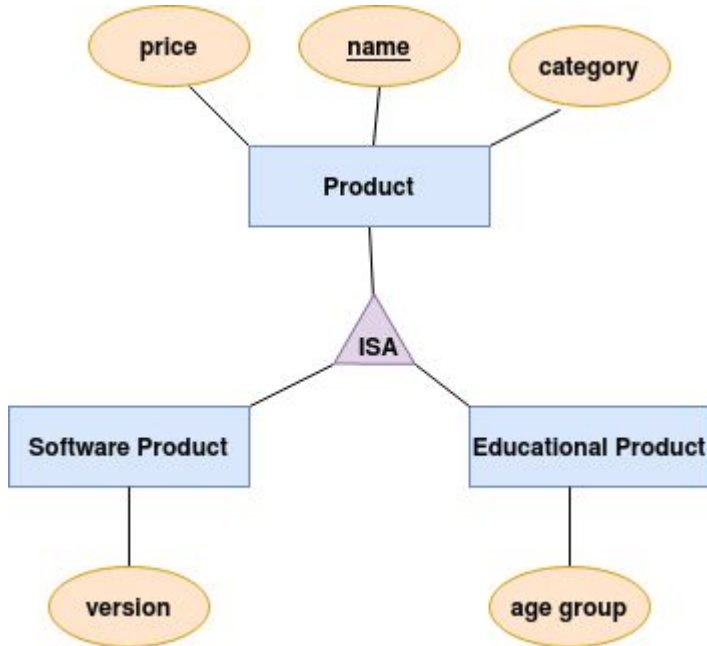


Constraints

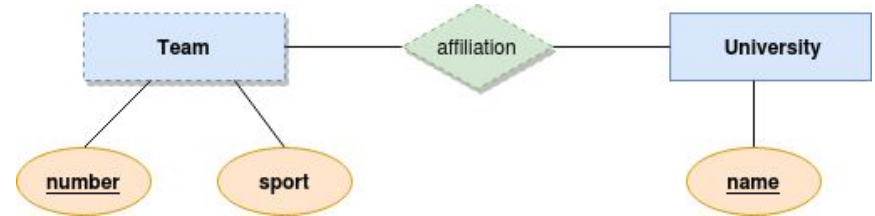
- constraint: an assertion about the database that must be true at all times.
- types of constraints
 - keys: SSN uniquely identifies a person
 - single-values: a person can have only one father
 - participation: each student must have a major
 - referential integrity: if you work for a company it must exist in the database
 - domain: age is between 0 and 150
 - other: at most 80 students in a class
- very important for the design of the database
 - more semantics to data
 - prevent wrong data entry
 - allow us to refer to entities
 - efficient storage, data lookup, etc.

ER Additional Features

SUBCLASSES



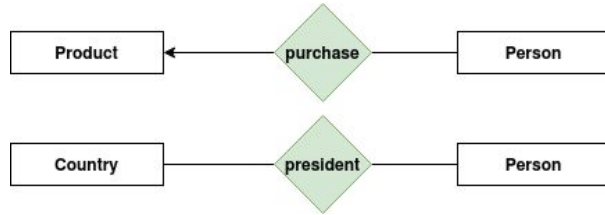
WEAK ENTITY SETS: entity sets are weak when their key attributes come from other classes to which they are related



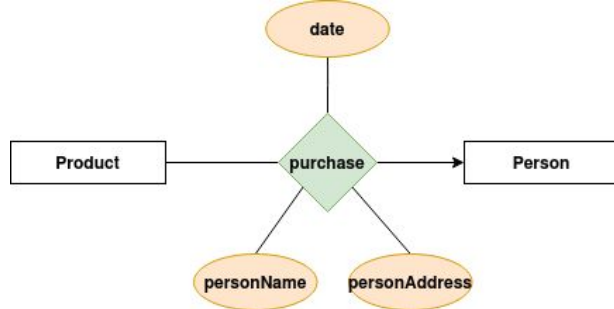
- entities of an entity set need “help” to identify them uniquely!

ER Design Principles

1. Be faithful to the APP



2. Avoid redundancy



3. Don't use an entity set when an attribute will do.

ER vs Relational Model

ER model

- many concepts: entities, relations, attributes, etc.
- well-suited for capturing the app requirements
- not well-suited for computer implementation

Relational model

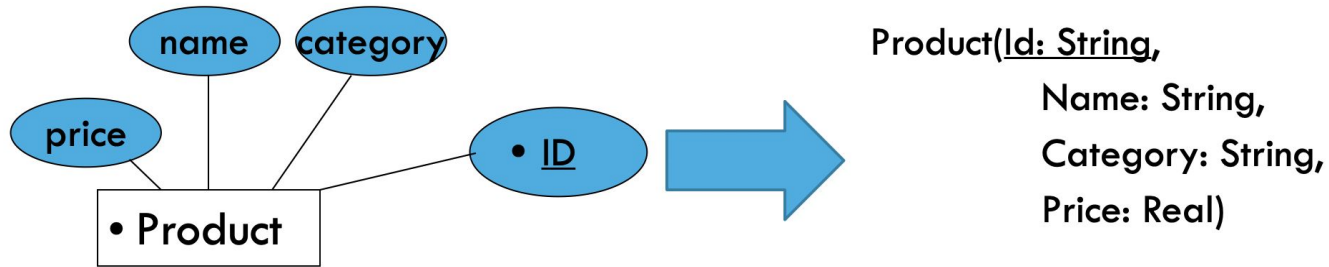
- has just a single concept: relation
- world is represented with a collection of tables
- well-suited for efficient manipulations on computers

Translation

- entity set $E \rightarrow$ relation with attributes of E
- relationship $R \rightarrow$ relation with attributes being keys of related entity sets + attributes of R

Some Rules (Taken from Slide Set #2)

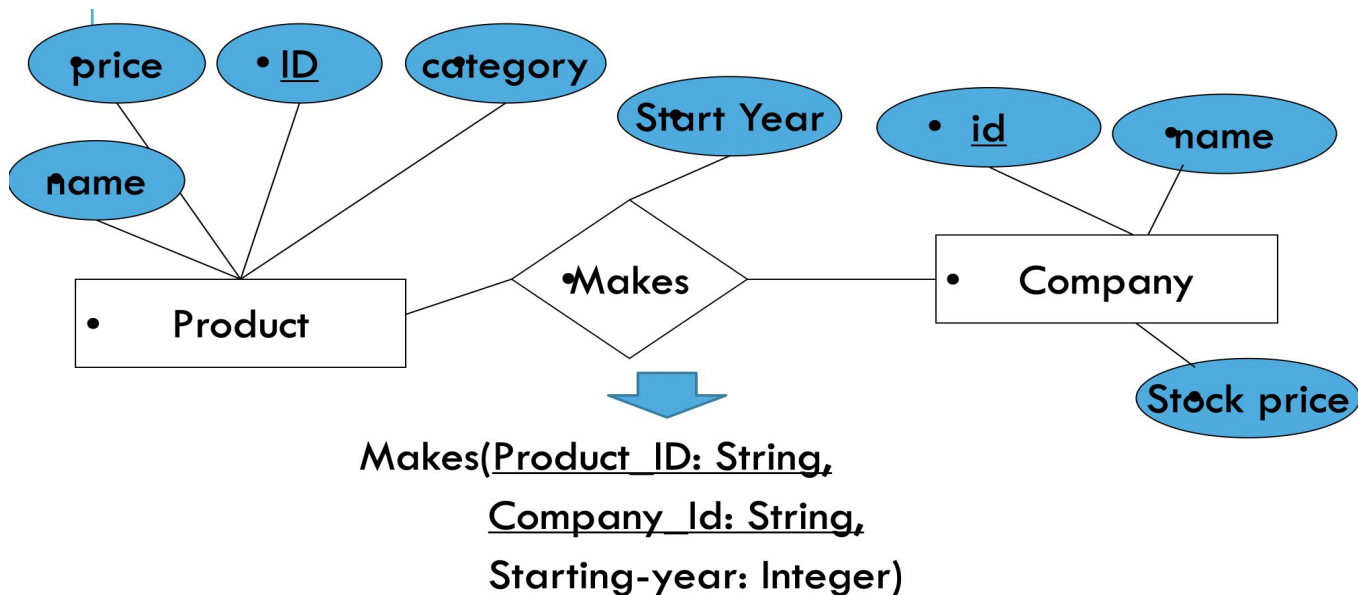
Rule #1: Each attribute of the entity set becomes an attribute of the table.



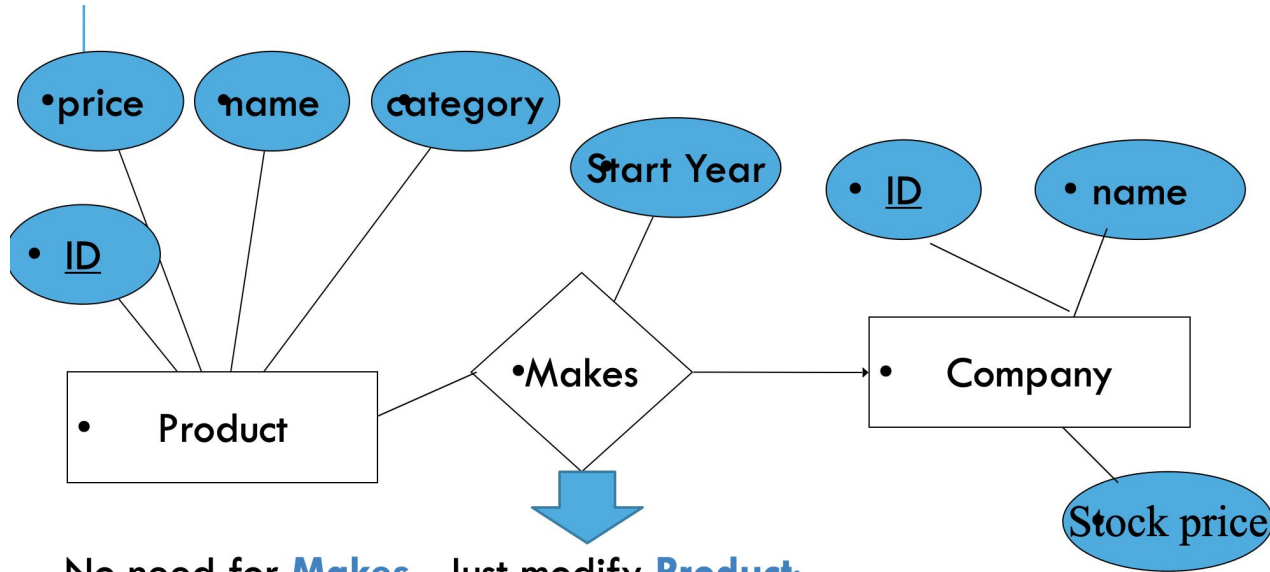
Relationship Sets (w/o constraints) to Relation

Rule #2: The attributes of the relation include:

1. The primary key attributes of each participating entity set.
2. The descriptive attributes of the relationship set



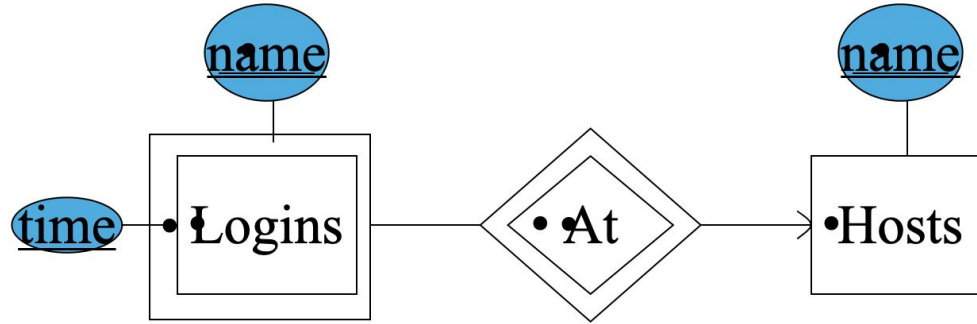
Combining 2 relations



No need for **Makes**. Just modify **Product**:

Product(Product_ID: String, Product_Name:String,
category: String, price: Real, StartYear: Integer, Company_ID)
Company(ID: String, name: String, StockPrice: Real)

Handling Weak Entity Sets

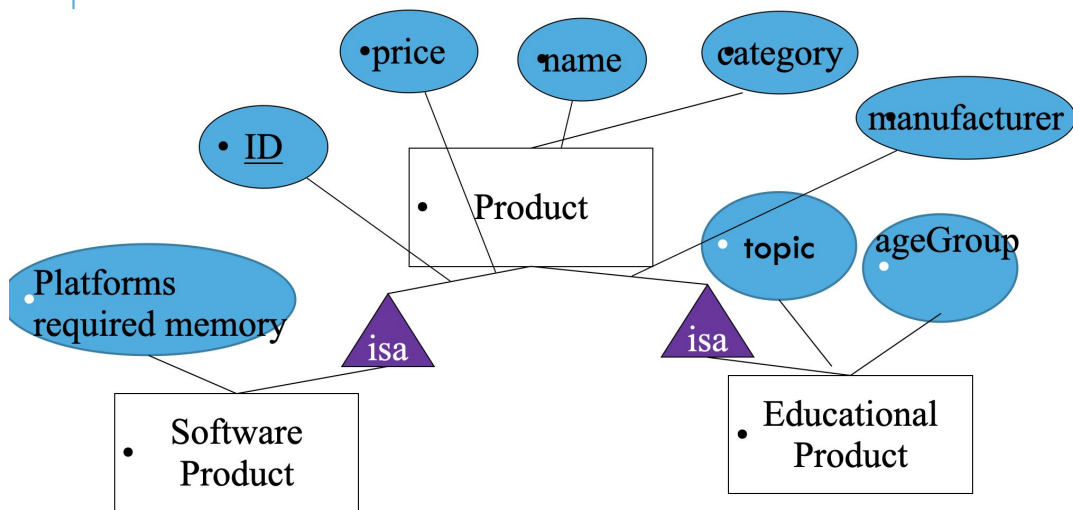


Hosts(hostName)

Logins(loginName, hostName, time)

•
At becomes part of
Logins

Translating Subclass Entities



ER Approach

Product(ID: String, name:String, price:Real,
category:String,
manufacturer:String)

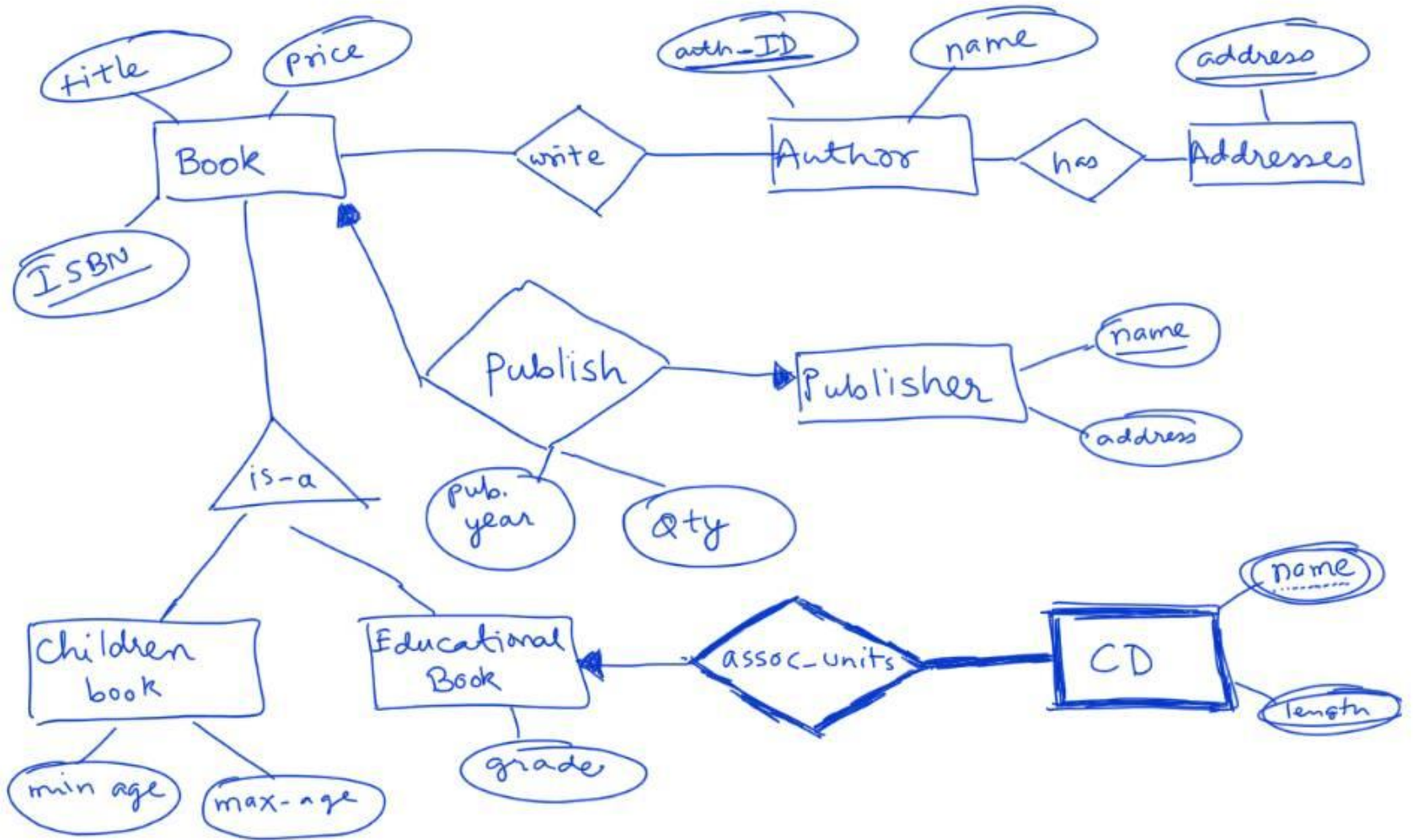
EducationalProduct(ID:String,
ageGroup:String, topic:String)

SoftwareProduct(ID:String,
platforms:String,
requiredMemory:String)

Same ID may appear in several relations

Walkthrough Example: Book Problem

- We want to store information about the books. The **book** has a **title**, **price** and an **ISBN** no. (which uniquely identifies the book). Some books are **children's books**. Each children's books also specifies the **minimum age** and the **maximum age** (indicating the ideal age range for the reader). Some books are **educational books**. Each educational books also specifies the **grade** (i.e. the school grade that book is most suited for). **Each educational book** may also be associated with one or multiple exercises **CDs**. Each CD has a **name** and **a length** (in min). We can uniquely identify each CD using its **name** and the information of the book that the CD is associated with. Each book is written by one or many **authors**. Each author has an **author ID**, **a name**, and **multiple addresses**. Each book is published by only one **publisher**. Each publisher has a **name** (which uniquely identifies the publisher) and a **single address**. When a publisher publishes a book, we want to record the **publishing year** and the **quantity** (no. of copies that the publisher produces). **EDIT: A publisher can publish only one book.**
- **Question: How to convert this information into ER diagram, and later into relational schema?**



Relational Schema

- Book(ISBN, title, price, pub_name, pub_addr, pub_year, quantity)
- ChildrenBook (ISBN, min_age, max_age)
- EducationBook (ISBN, grade)
- CD (ISBN, cd_name, cd_length)
- Author (authorID, name)
- Address(address)
- write (ISBN, authorID)
- has(authorID, address)

Q&A

Thanks!