Database Lecture 7
DB Design Exercises

Lecture Addendum

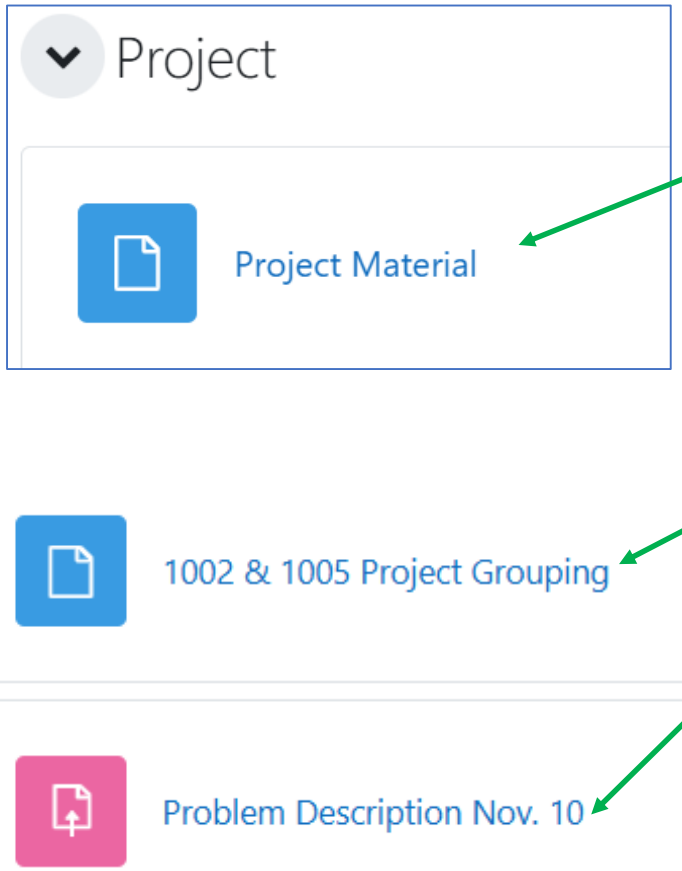
Dr. Jefferson Fong

Announcements

Midterm Test

- Time: Wed 6 Nov, 5:00-7:00 pm.
  - No one has requested the alternate time of 3:30-5:30 pm.
  - Anyone who cannot take the test at the regular time must get official approval from CST office at T3-602.
- Venue: T7-603
- Covers Lecture 1 to 7 and Lab 1 to 7
- Closed book, no notes or any electronic devices allowed.
- Bring a pen (or two) and you UIC ID card.
  - If don’t have your UIC ID card, bring your government issued photo ID.

Announcements



- Group Project info is in iSpace under Project tab.
  - Project Material > COMP3013 Project Requirement 24F.pdf
    - Quickly go over COMP3013 Project Requirement 24F.pdf.
  - Timeline sections shows when requirements are due.
  - Group members listing.
  - Problem description is due on Nov 10.
    - Describe the website you will build.
      - E.g. can be website from SWDW: online second hand store, or any online store in the Intro.
    - Some of the things mentioned in the document will be covered in future lectures.
  - Focus of the project is on the backend database.
    - We don't care how pretty your frontend looks, but it has to work well enough to handle the backend database.

- In previous lectures, we spent lots of time giving you theory.
- In this lecture, we do exercises for you to practice applying the theory.

Exercise 1

- Construct an ER diagram for a car-insurance company. The model includes
  - each customer is described by a unique id, a name, and an address;
  - each car is described by a unique vehicle identification number, the manufacturer, and the year of production;
  - each accident is described by a unique report number, the date and the location when the accident happens;
  - each customer owns one or more cars;
  - every car is owned by at exactly one customer;
  - each accident is only responsible to the owner of the car, no matter who drives;
  - it is possible that multiple cars involve in a same accident and the amount of insurance claim for different cars are different.
- List all assumptions that you make.
- And convert your ER diagram to a logical design.

Step1

- Highlight (or circle) important nouns that might be entities.
- Highlight (or underline) verbs that might be relationships.
  - Sometimes we convert a noun into a verb, e.g. Instructor PD Program.
- Do obvious ones first; do the rest later.
- Use a pencil and eraser, or do in scratch paper first.

Exercise 1, **Step 2**

- First draw the entities (nouns): **customer**, **car**, **accident**.
- Then add in relationships (verbs): **own**, **involve** (participate)

- Add in attributes.
  - Cross them out in the description after you added them in the diagram.
- Add constraints.
- Iterate – add more entities, relationships, etc. from the description.
- Adjust the locations of the elements in diagram.

Step 3: After finished ER diagram, do logical design (schema)

- Schema for customer entity: **customer = (cid, name, address)**
  - List the key cid first with underline.
- "Customer own car" is a 1-to-many relationship.
  - Don't need a table for the relationship own.
  - Put the cid, the key for customer (the "one") into the car table as the foreign key.
  - Schema for car entity: **car = (vid, model, year, cid).**
  - The tables customer and car can be linked together using a join on cid.
  - Draw the tables with sample values to help you figure out the constraints.
    - Customer C1 has many cars.

Customer (one)

cid	name	address
C1		
C2		
C3		

C for customer

Car (many)

vid	model	year	cid
V1			C1
V2			C1
V3			C3

V for vehicle

Step 3 (continue)

"Car participate accident" is a many-to-many relationship.

- Need a table for the relationship **participate** = (vin, rnumber, amount)
  - Typo in the answer (given next week); diagram and table should have rnumber rather than report_number or r_number.
- A car can participate in many accidents (with different time or location).
  - Car V1 participates in accident R1 and R2.
- An accident can involve many cars.
  - Accident report R1 involves car V1 and V3.

- Car (many)

vid	model	year	cid
V1			
V2			
V3			

Participate (involve)

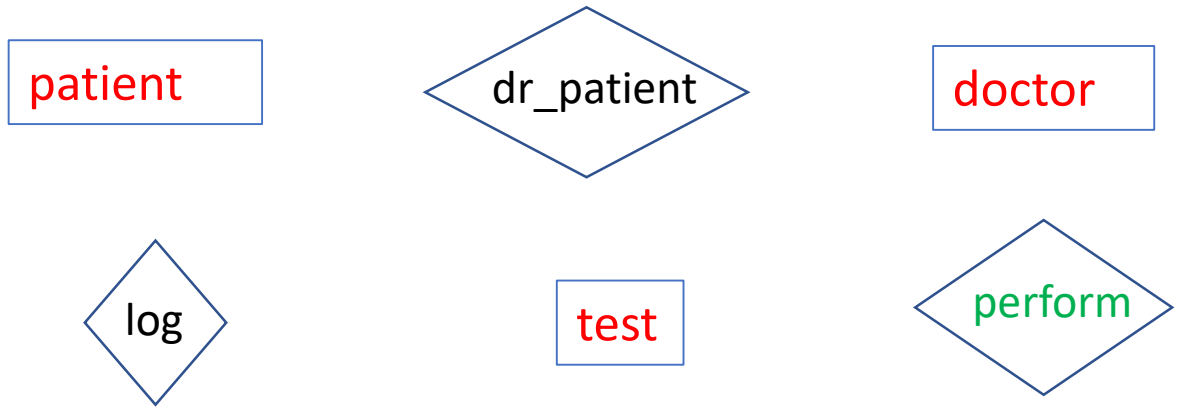
vid	rnumber	amount
V1	R1	
V1	R2	
V3	R1	

Accident (many)

rnumber	location	date
R1		
R2		
R3		

Exercise 2

- Highlight entity nouns: patient, doctor, test
- Highlight relationship verbs: **perform**, **dr_patient (care for)**, **log**
  - Not sure about **log**; do that later.



- Add in attributes and adjust the locations of the elements in the diagram.
- Add in more stuff from the description; cross out stuff used already.
- “every patient is **cared** by one or multiple doctor”
  - Should **cared** be a relationship? Yes, use **dr_patient** as the relationship for now.

Schema Exercise 2

- Schema for entities are straight forward, e.g.

  patient = (pid, pname, insurance), doctor = (did, dname, specialization)

- Patient – doctor is a many-to-many relationship
  - Need a table for the relationship **dr_patient** = (pid, did).
- "Doctor perform test" is a many-to-many relationship
  - Need a table for the relationship **perform** = (did, tid).
- Patient – test is a 1-to-many relationship.
  - One patent can have many tests.
  - Don't need a table for "log"
  - Put pid, the key for patient (the "one") as a foreign key in the table test, so

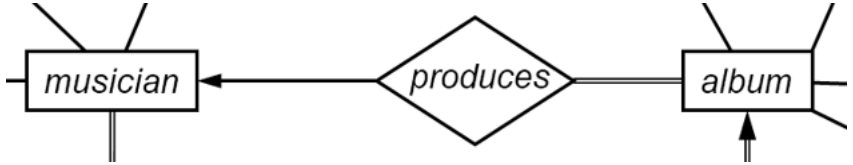
    test = (tid, tname, date, time, result, pid)

Exercise 3

- Create an ER diagram for a recording company with the following requirements:
  - Each musician has an unique SSN (key for musician table), a name, an address, and a phone number.
  - Each instrument has a name (e.g., guitar) and a musical key (e.g., C, just to confuse you).
  - Each album that has a title, a copyright date, a format (e.g., CD), and an album identifier.
  - Each song has a title and an author.
  - Musicians may play many instruments, and an instrument may be played by many musicians.
  - Each album has a number of songs on it, but no song may appear on more than one album.
  - Each song is performed by one or more musicians, and a musician may perform a number of songs.
  - Each album has exactly one musician who acts as its producer. A musician can of course produce several albums.

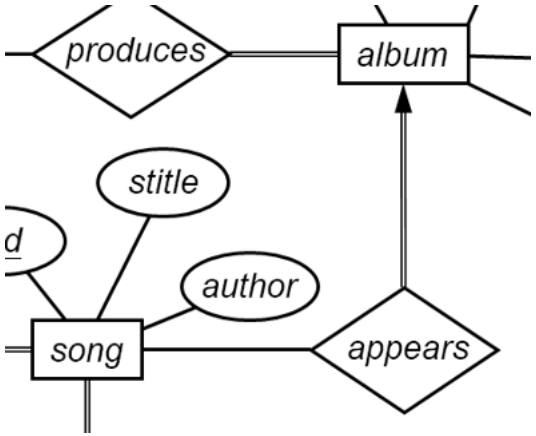
Schema Exercise 3

- "Each album has exactly **one musician** who acts as its producer. A musician can of course produce several albums."



  - This is case 7 in Lecture 6; album = (aid, atitle, crdate, format, ssn)
  - No schema is needed for relationship "produces", with the ssn (key for musician) appearing in the table album.

- "Each (one) album has a number of (many) songs on it, but no song may appear on more than one album."



  - This is case 6 in Lecture 6.
  - Need a schema for relationship "appears", with appears = (sid, aid)
  - If we try song = (sid, stitle, author, aid) like we did in case 7, some row can have a null value for aid (foreign key) if that song is not in any album.

Summary of ER Diagram Design

entity set

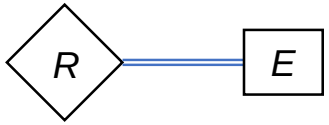
relationship set

attribute

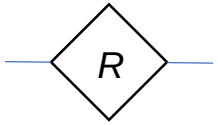
weak entity set

identifying relationship set

multivalued attribute

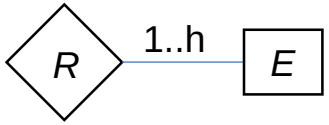


total participation

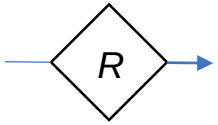


many-to-many relationship

derived attribute

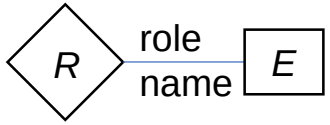


cardinality limits

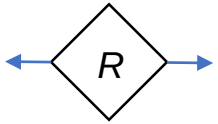


many-to-one relationship

primary key

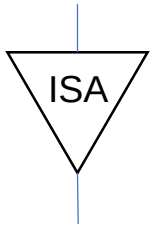


role indicator

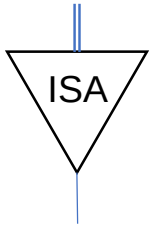


one-to-one relationship

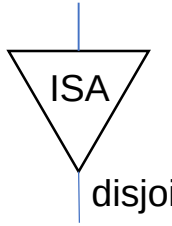
discriminator



ISA



total generalization



disjoint generalization

- Intro to this week's lab.

Midterm course evaluation

- See email from AR