Lecture 3 Entity-Relationship Model Constraints

Outline

- Cardinality Constraints
- Participation Constraints
- Constraints for Ternary Relationship Sets

Relationship Constraints

- In most cases, database designers want to include some additional information about relationship sets to answer the questions like:
  - “how many entities can be associated with one entity at most (or at least)?”
- For example, we add the relationship set “major” to model “students have majors” in the exercise of the last lecture.
- But the description is not accurate.
  - Can one student have multiple majors?
  - Can one program have multiple students?
  - Does every student have a major?
  - Is every program the major for some students?

Relationship Constraints

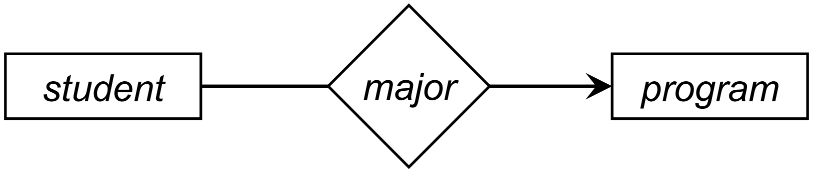
- To express the answers, ER diagrams have ***constraints*** on relationship sets.
- Constraints indicate some conditions under the context of the modeled problem.
- Two types of constraints
  - ***Cardinality constraints***
  - ***Participation constraints***
- “Cardinality” is a term from set theory. It is the number of items in a set.

Cardinality Constraints

- First, let's consider the questions:
- Can one student have multiple majors?
  - No, one student can only associate with at most one program as his/her major.
- Can one major program have multiple students?
  - Yes, one major program can have many students.
- This "major" relationship set is a ***many-to-one*** relationship set.
- The entity set "student" is on the many side, while the entity set "program" is on the one side.
- This type of constraints is called ***cardinality constraint***.

Cardinality Constraints

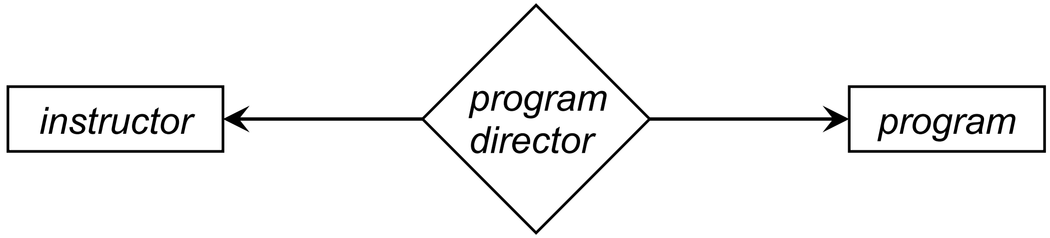
- To express the cardinality constraints, ER diagrams use an ***arrow*** (→) pointing to the one side.
- For the many side, the links simply have no arrow (–).



- A ***one-to-many*** relationship is the reverse of many-to-one.

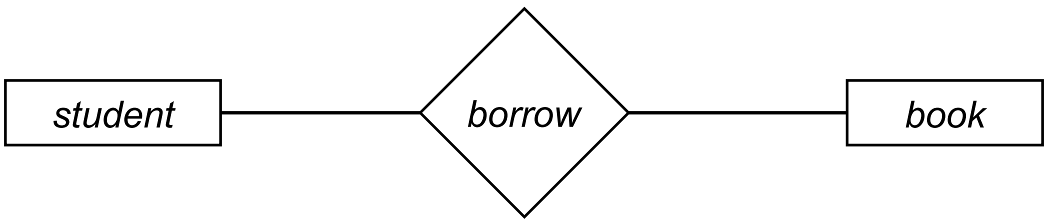
Cardinality Constraints

- ***One-to-one*** relationship:
  - One entity from one entity set is associated with at most one entity from the other entity set and vice versa.
  - For example, one instructor can be the program director of at most one program, and one program has at most one program director.



Cardinality Constraints

- ***Many-to-many*** relationship:
  - One entity from one entity set can be associated with multiple entities from the other entity set and vice versa.
  - For example, one student can borrow multiple books, and one book can be borrowed by multiple students.

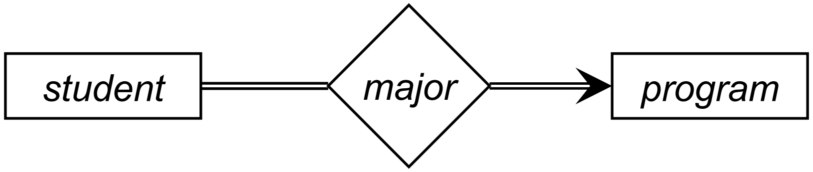


Participation Constraints

- To express the answers to the other two questions:
  - Does every student have a major?
  - Is every program the major for some students?
- ER diagrams have ***participation constraints***.
- ***Total participation***:
  - Every entity participates a relationship.
  - The link is a ***double line*** (=).
- ***Partial participation***:
  - There are some entities do not participate any relationship.
  - The link is a ***single line*** (–).

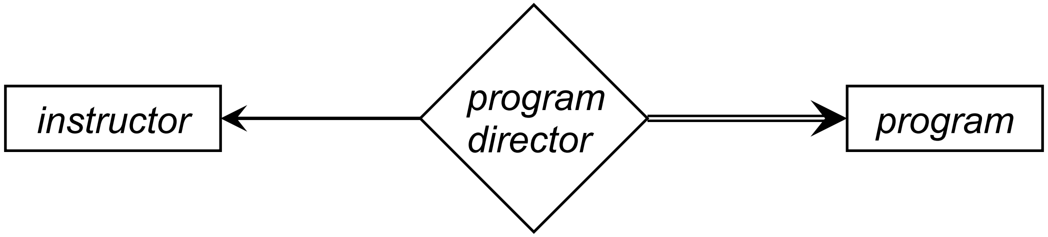
Participation Constraints

- Back to the example,
  - Every student has a major. (Students totally participate in the relationship set.)
  - Every program is the major for some students. (Programs also totally participate.)
- Combining the cardinality constraints,



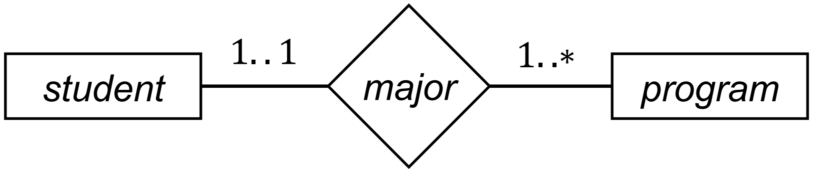
Participation Constraints

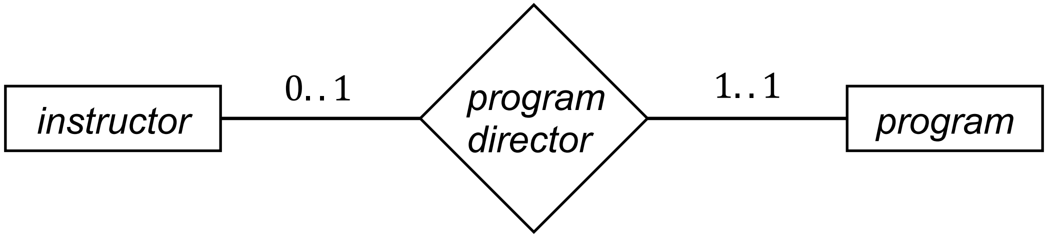
- Consider another example, the relationship set “program director”.
  - It’s possible that an instructors is not a program director for any program.
  - But every program has a program director.
- Thus, “instructor” is on the partial side, while “program” is on the total side.
- Combining the cardinality constraints,



Alternative Notations

- Instead of using lines and arrows, relationship constraints can be expressed by numbers in the form $a..b$ besides links.
- a is the minimum number of entities associated with.
- b is the maximum number of entities associated with.
- $*$ means multiple.
- For example,



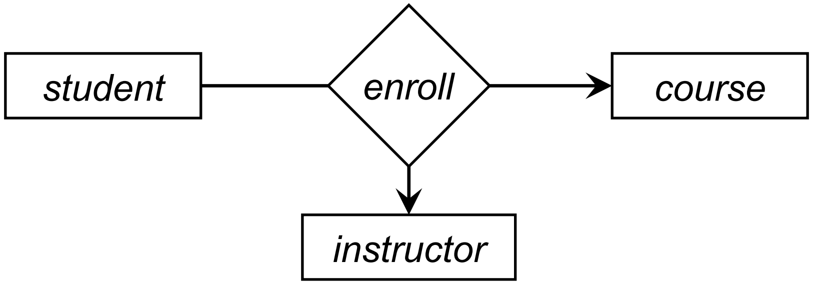


Constraints for Ternary Relationship Sets

- Constraints for ternary relationship sets are not easy.
- They easily creates ambiguous, meaning that an ER diagram can be understood in different ways.
- Thus, we usually avoid using constraints on ternary relationship sets.
- If expressing constraints is important, ternary relationship sets can always be converted into several binary ones.

Constraints for Ternary Relationship Sets

- Consider the ternary relationship “enroll” associating “student”, “instructor”, and “course”.
- Assume we want to express that
  - one course is instructed by at most one instructor; and
  - one instructor instructs at most one course at a time. (Let’s just assume this for the example purpose. Even it is not the real case.)
- So, both of “instructor” and “course” are pointed by arrows.
- One may draw the ER diagram naively as follows.

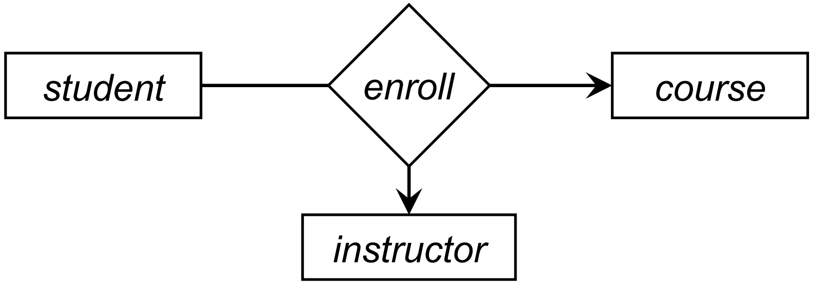


Constraints for Ternary Relationship Sets

- However, this ER diagram can be understood in different ways.
- Meaning 1:
  - One student is only associated with one combination of a course and an instructor. But, no constraints is between course and instructor.
- Meaning 2:
  - One course is instructed by at most one instructor.
  - One instructor can instruct at most one course.
  - But there is no constraints on students.

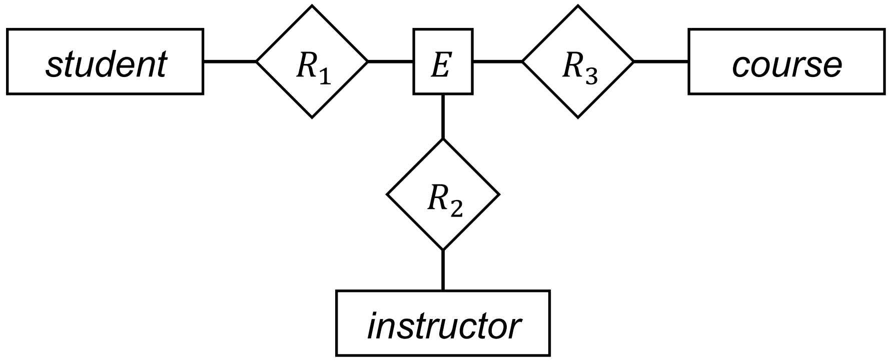
Relationships	Meaning 1	Meaning 2
(s_1, i_1, c_1)		
(s_2, i_2, c_1)		Not allowed
(s_3, i_1, c_2)		Not allowed
(s_1, i_2, c_1)	Not allowed	Not allowed
(s_2, i_1, c_2)	Not allowed	Not allowed
(s_4, i_1, c_1)		

- Assuming s_n, i_n, and c_n are students, instructors, and courses respectively.



Converting Non-binary to Binary

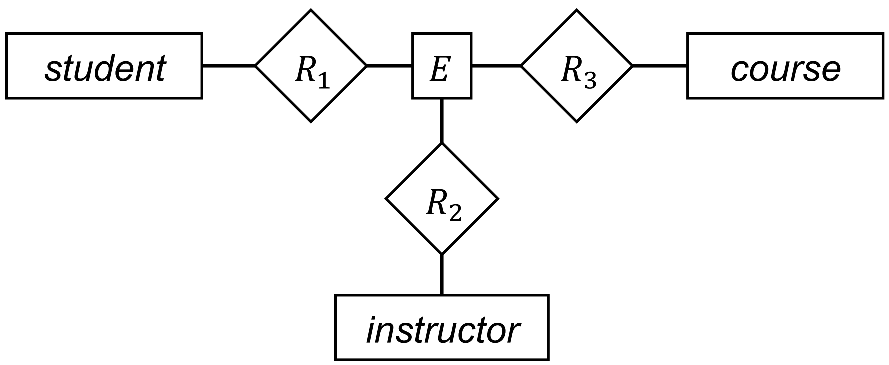
- To express complex constraints on a ternary relationship, one can always convert the ternary relationship to binary ones by introducing artificial entity sets and relationship sets.
- For meaning 1, one entity is associated with at most one combination of two entities.
- This type of many-to-one constraints (the many side is a combination of entities) can be converted into



- E is an artificial entity set.
- R_1, R_2, and R_3 are artificial relationship sets.
- Constraints are on the next page.

Converting Non-binary to Binary

- First, for each relationship (s, i, c) in “enroll”, the entity set E contains an artificial entity e.
- Then, to represent the relationship (s, i, c),
  - (s, e) to the relationship set R_1,
  - (i, e) to the relationship set R_2,
  - (c, e) to the relationship set R_3.



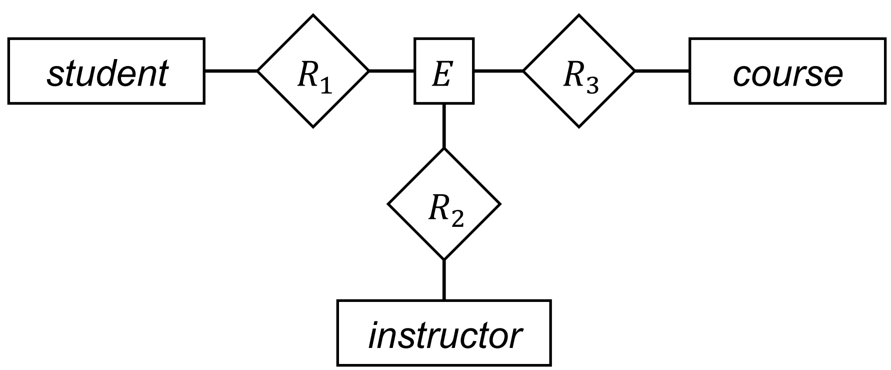
Converting Non-binary to Binary

- To represent Meaning 1.

- $$E = \{e_1, e_2, e_3, e_6\}$$
  $$R_1 = \{(s_1, e_1), (s_2, e_2), (s_3, e_3), (s_4, e_6)\}$$
  $$R_2 = \{(i_1, e_1), (i_2, e_2), (i_1, e_3), (i_1, e_6)\}$$
  $$R_3 = \{(c_1, e_1), (c_1, e_2), (c_2, e_3), (c_1, e_6)\}$$

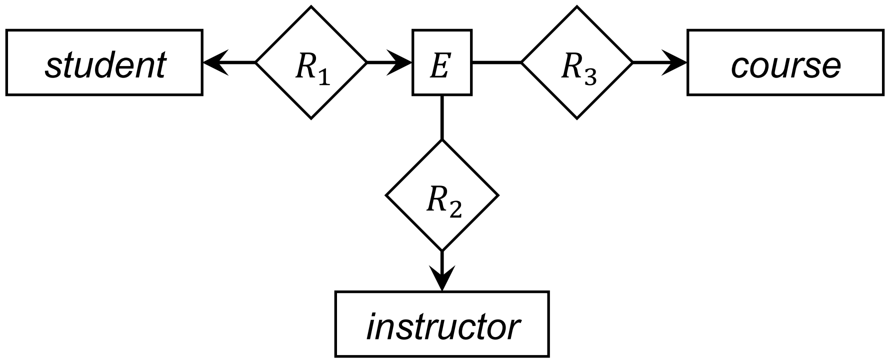
Relationships	Meaning 1
$e_1 = (s_1, i_1, c_1)$	
$e_2 = (s_2, i_2, c_1)$	
$e_3 = (s_3, i_1, c_2)$	
$e_4 = \cancel{(s_1, i_2, c_1)}$	~~Not allowed~~
$e_5 = \cancel{(s_2, i_1, c_2)}$	~~Not allowed~~
$e_6 = (s_4, i_1, c_1)$	

- Therefore, R_2 and R_3 are one-to-many relationship sets ("instructor" and "course" are on the "one" side.) and R_1 is a one-to-one relationship set.



Converting Non-binary to Binary

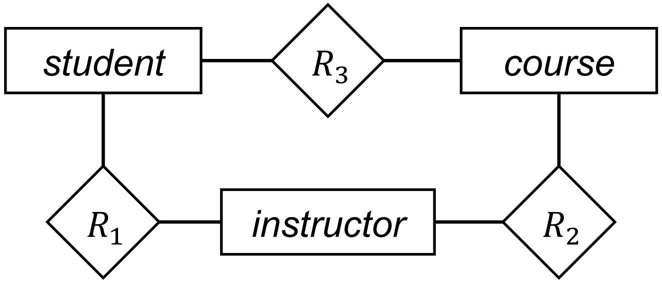
- Adding the constraints to the ER diagram,



- Finally, the entity set E has at least one attribute for the key and inherits all attributes from the original relationship set, if any.
- The above conversion can be generalized to all other similar cases.

Converting Non-binary to Binary

- Next, we move on to the meaning 2.
  - One course is instructed by at most one instructor.
  - One instructor can instruct at most one course.
  - But there is no constraints on students.
- However, the above construction does not work for this case. (Please have a try.)
- This constraint is between two entity sets and free from other entity sets.
- For this type of constraints, the structure is like this.



- Only three relationship sets are added.

Converting Non-binary to Binary

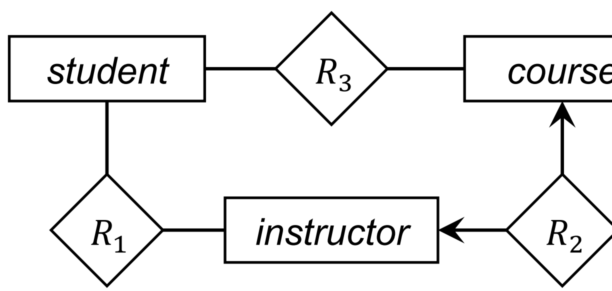
- For each relationship (s_n, i_n, c_n), we add
  - (s_n, i_n) to the relationship set R_1,
  - (i_n, c_n) to the relationship set R_2, and
  - (c_n, s_n) to the relationship set R_3.

Relationships	Meaning 2
$e_1 = (s_1, i_1, c_1)$	
$e_2 = \cancel{(s_2, i_2, c_1)}$	~~Not allowed~~
$e_3 = \cancel{(s_3, i_1, c_2)}$	~~Not allowed~~
$e_4 = \cancel{(s_1, i_2, c_1)}$	~~Not allowed~~
$e_5 = \cancel{(s_2, i_1, c_2)}$	~~Not allowed~~
$e_6 = (s_4, i_1, c_1)$	

- Thus, Meaning 2 is represented as

$$R_1 = \{(s_1, i_1), (s_4, i_1)\},\ R_2 = \{(i_1, c_1)\},\ R_3 = \{(s_1, c_1), (s_4, c_1)\}$$

- R_2 is a one-to-one relationship set.
- R_1 and R_3 are many-to-many. (Try more examples, then you will see.)



Converting Non-binary to Binary

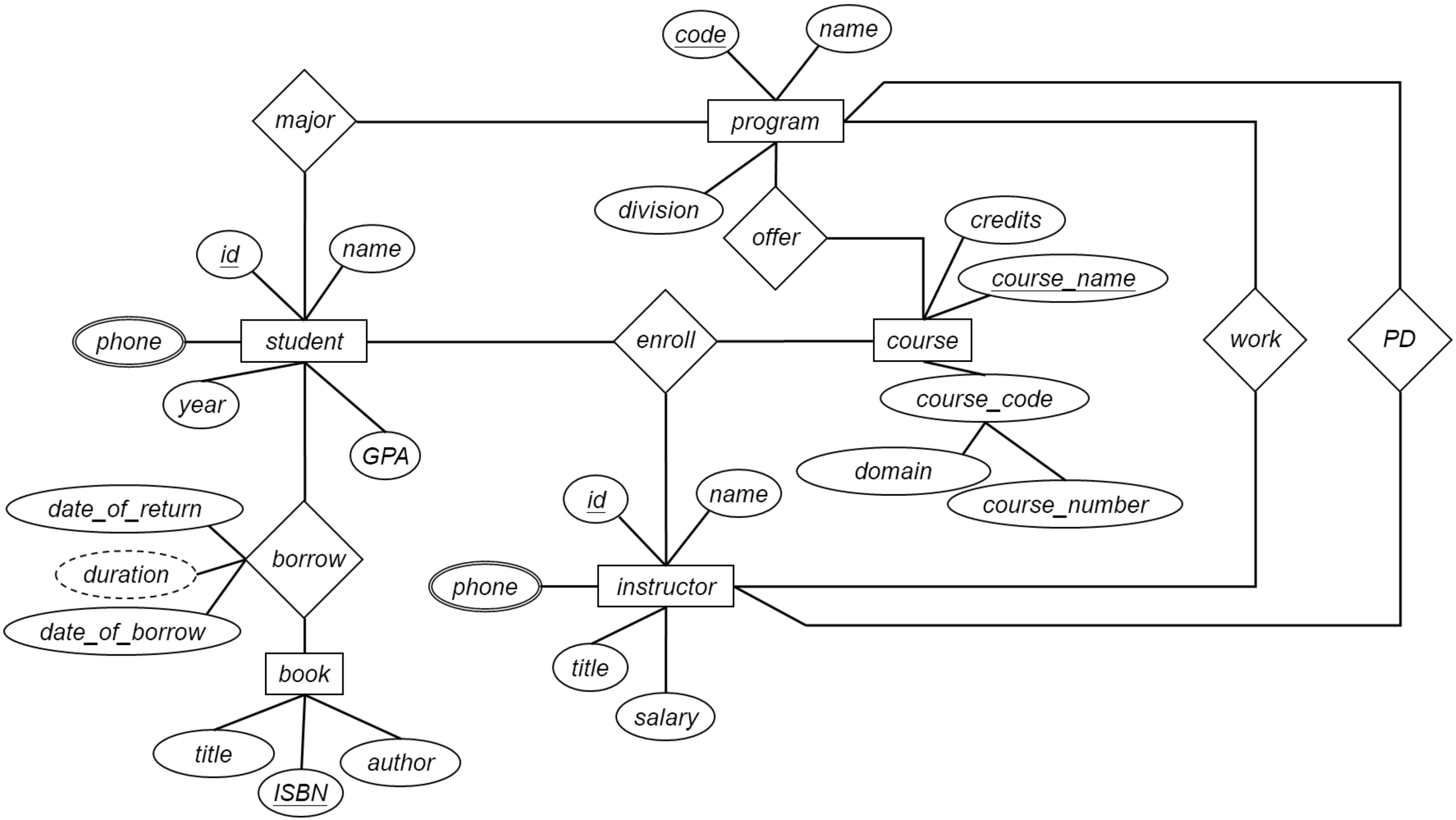
- Sometimes more complex constraints is possible in the applications.
- For example, combining Meaning 1 and 2,
  - One course is instructed by at most one instructor.
  - One instructor can instruct at most one course.
  - One student is associated with at most one combination of instructors and courses.
- The previous conversion cannot work. (Please also have a try)
- To handle this case, some extended features will be introduced in the next lecture.
- We have only discussed cardinality constraints for ternary relationship sets. Participation constraints are actually the same.

Exercises

- Combining the constraints that we have discussed.

1. Try to make reasonable constraints on the relationship sets “major” and “PD”.
2. Add a new entity set “transcript”.
3. Model some reasonable relationship sets among “student”, “transcript”, and “course”.

Exercises


code
name
major
program
division
offer
credits
course_name
id
name
phone
student
enroll
course
work
PD
year
GPA
course_code
domain
id
name
course_number
date_of_return
duration
borrow
phone
instructor
date_of_borrow
book
title
salary
title
ISBN
author

End of Lecture 3