

Supplementary Notes for CA218

Part 2

Dr. Martin Crane

Correspondence Btw ER & Relational Models

Table 9.1 Correspondence between ER and Relational Models

ER Model	Relational Model
Entity type	“Entity” relation
1:1 or 1:N relationship type	Foreign key (or “relationship” relation)
M:N relationship type	“Relationship” relation and two foreign keys
n-ary relationship type	“Relationship” relation and n foreign keys
Simple attribute	Attribute
Composite attribute	Set of simple component attributes
Multivalued attribute	Relation and foreign key
Value set	Domain
Key attribute	Primary (or secondary) key

Steps in the Process

1. Create a relation for each strong entity type
2. Create a relation for each weak entity type
 - include primary key of owner (an FK - foreign key)
 - owner's PK + partial key becomes PK
3. For each binary 1:1 relationship choose an entity and include the other's PK in it as an FK. Include any attributes of the relationship
4. For each binary 1:n relationship, choose the n-side entity and include an FK w.r.t the other entity. Include any attributes of the relationship

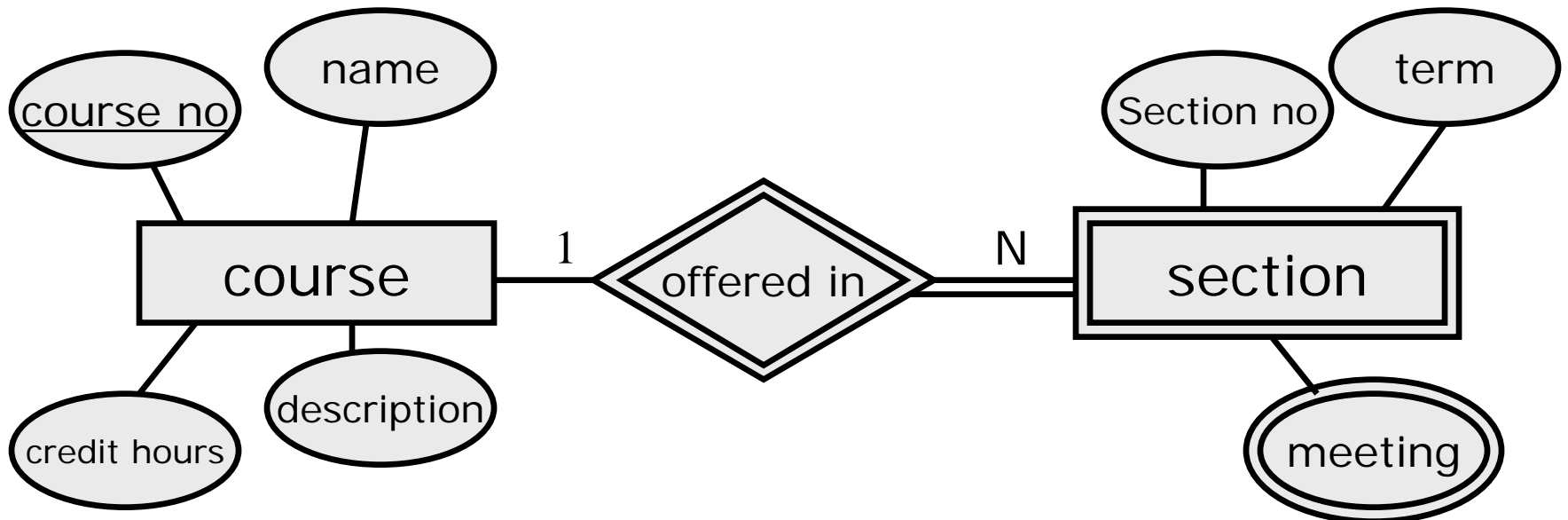
Steps in the Process (Cont'd)

5. For each binary $M:N$ relationship, create a relation for the relationship
 - include PKs of both participating entities and any attributes of the relationship
 - PK is the concatenation of the participating entity PKs
6. For each multivalued attribute create a new relation
 - include the PK attributes of the entity type
 - PK is the PK of the entity type and the multivalued attribute
7. For each n -ary relationship, create a relation for the relationship
 - include PKs of all participating entities and any attributes of the relationship
 - PK may be the concatenation of the participating entity PKs

ER to Relational Example: Step 1

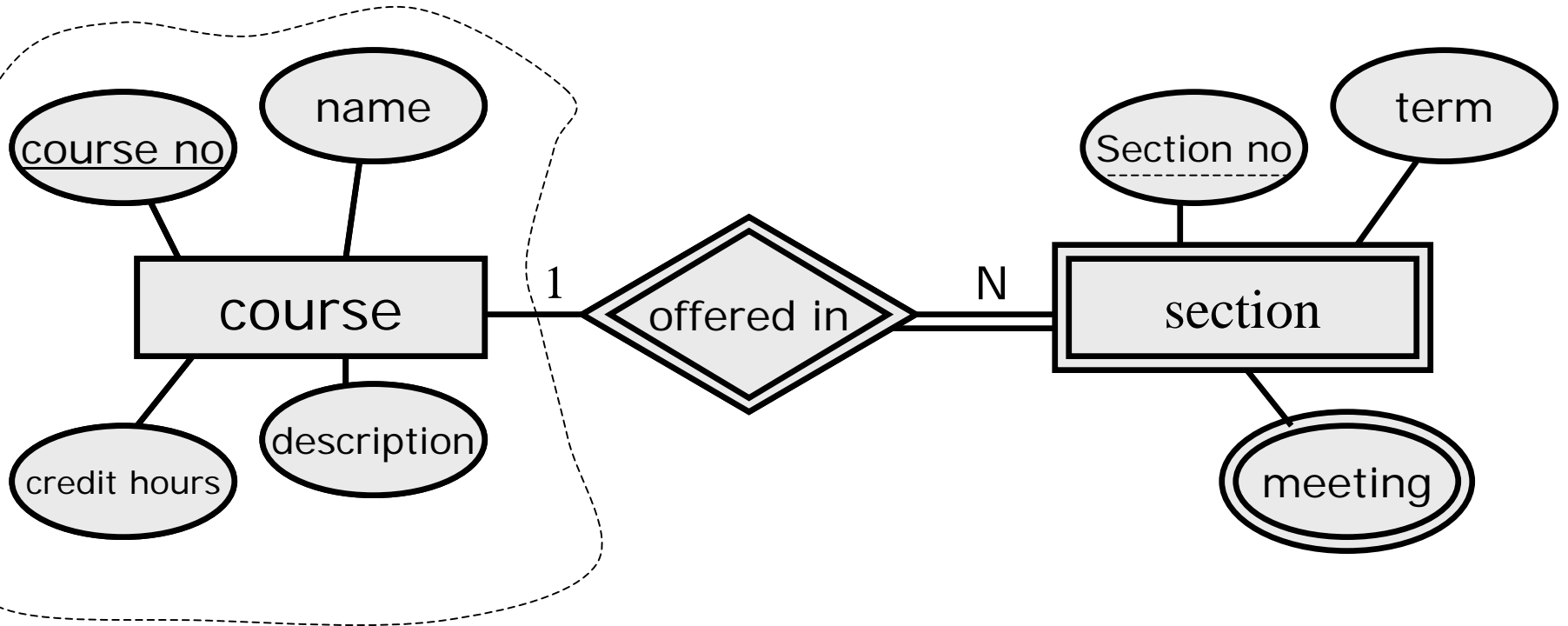
1. Create a relation for each strong entity type
 - include all simple attributes
 - choose a primary key

Suppose we have:



ER to Relational Example: Step 1 (cont'd)

Create a relation for *Course* – with four attributes, *course_no* is the PK.



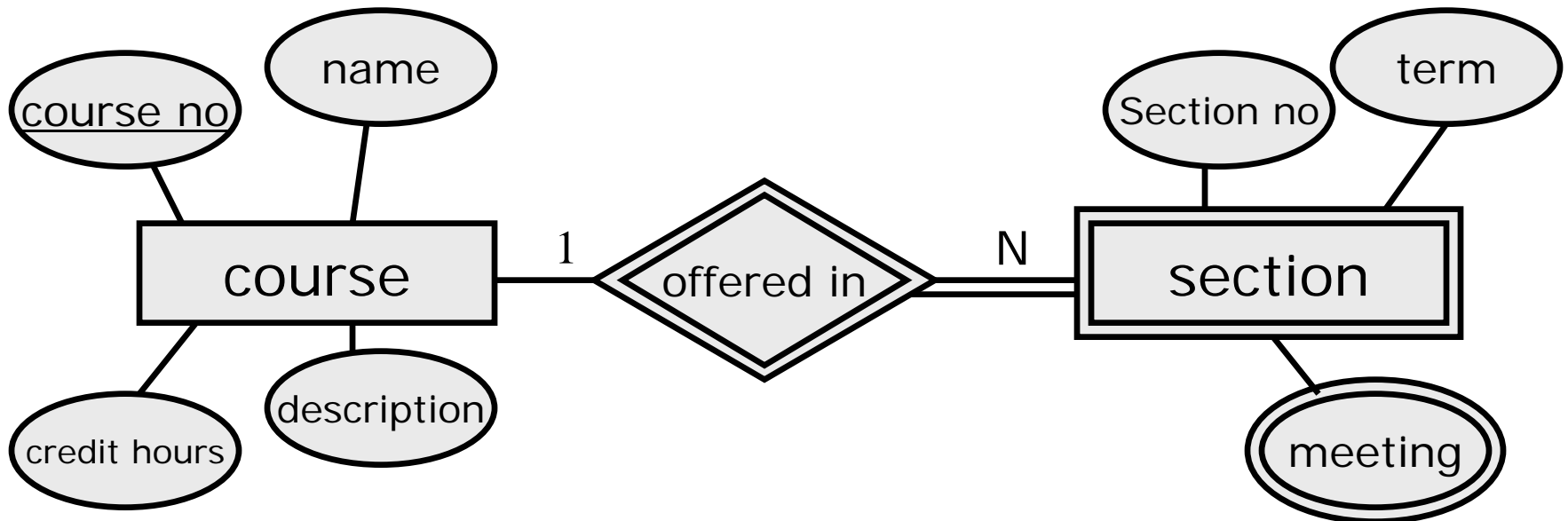
Course

<u>Course_no</u>	name	credit_hours	description
------------------	------	--------------	-------------

ER to Relational Example: Step 2

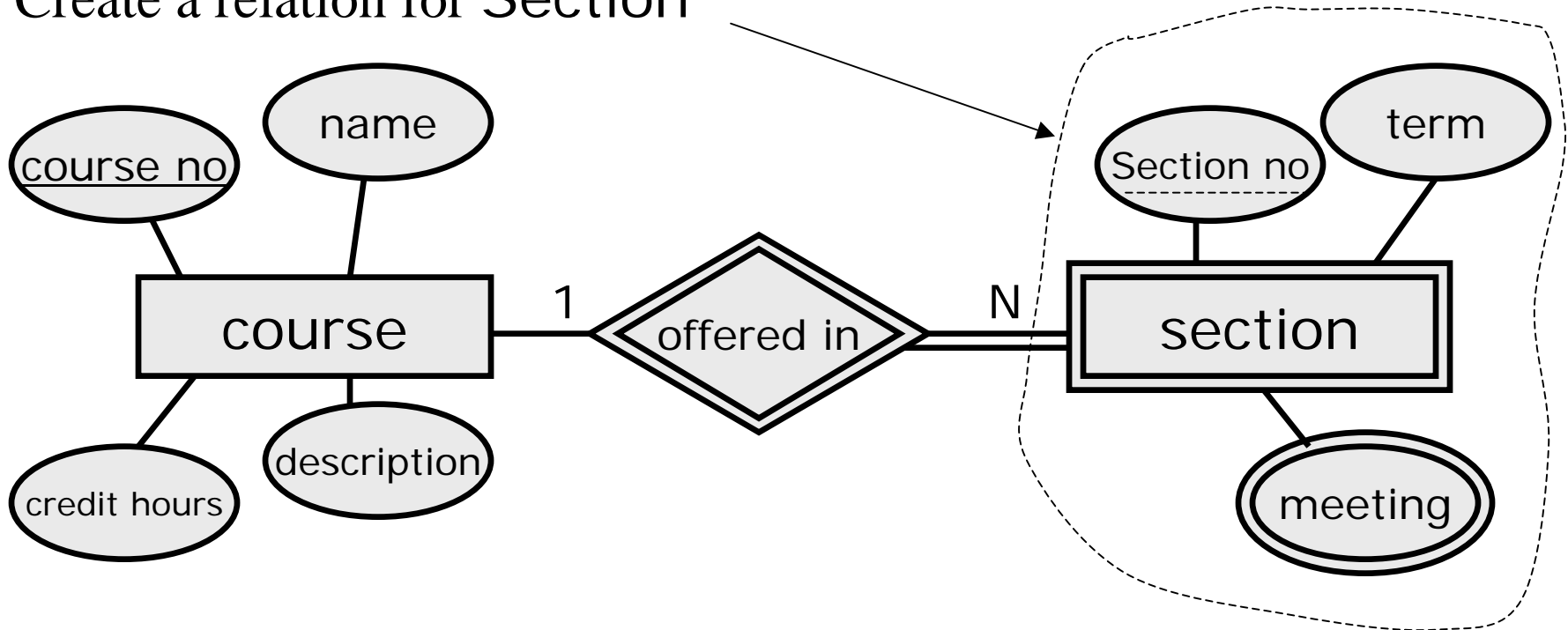
2. Create a relation for each weak entity type
 - include primary key of owner (an FK)
 - Owner's PK + partial key become the PK

Suppose we have:



ER to Relational Example: Step 2 (Cont'd)

Create a relation for Section



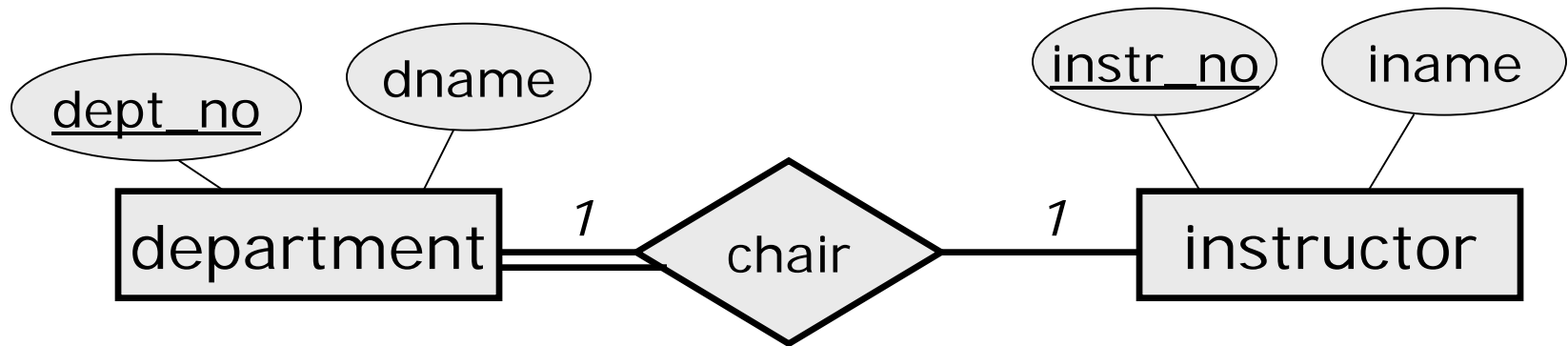
Section

<u>Course_no</u>	<u>Section_no</u>	Term
------------------	-------------------	------

- PK is {course_no, section_no}.
- course_no is an FK.
- meeting is not a simple attribute, so it's not included.

ER to Relational Example: Step 3

3. For each binary *1:1* relationship choose an entity and include the other's PK in it as an FK.



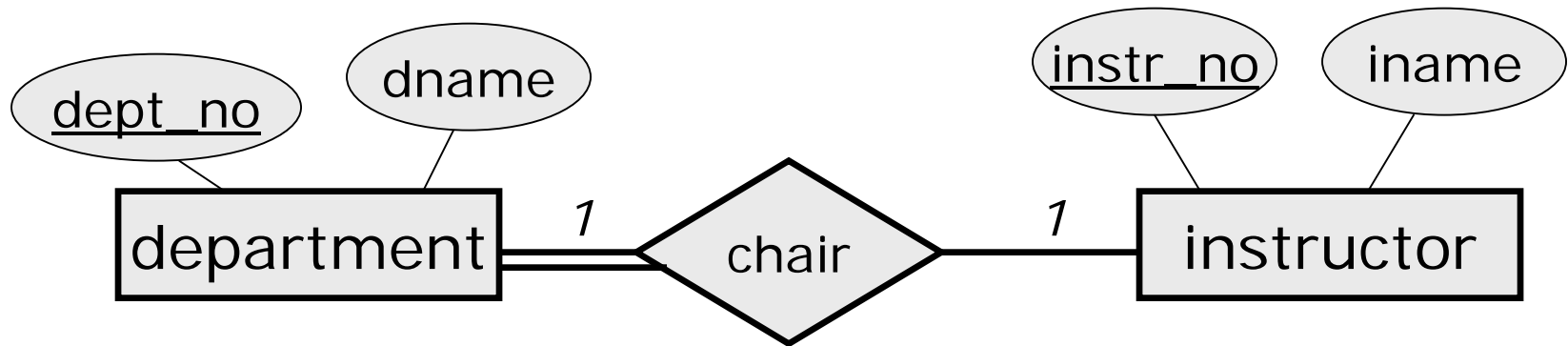
There are two choices here

- choose department, or
- choose instructor

Which is the better choice?

ER to Relational Example: Step 3 (Cont'd)

department is the better choice since it must participate in relationship.



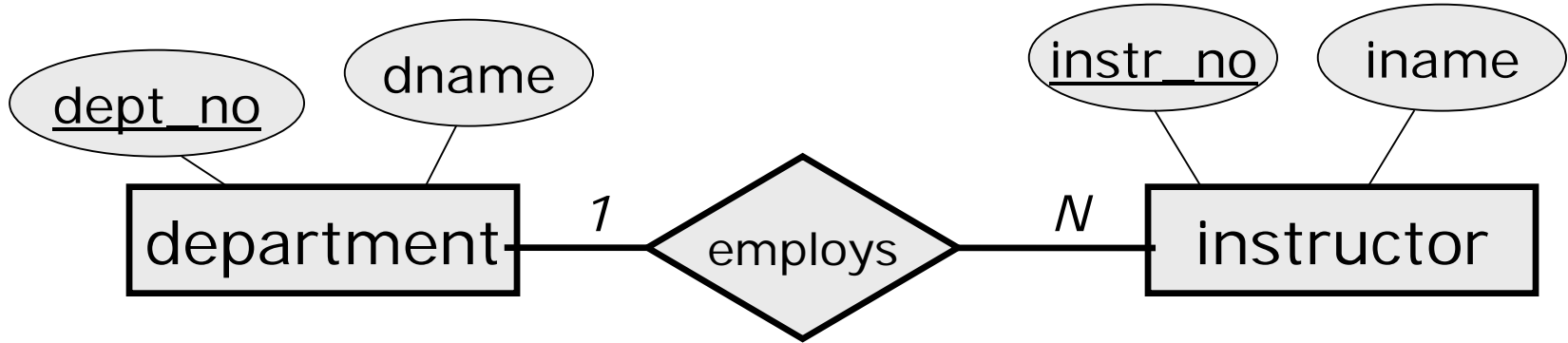
Department

<u>dept_no</u>	dname	chair
----------------	-------	-------

If we choose department then instr_no is included as, of course, an FK. Note that instr_no must have a value.

ER to Relational Example: Step 4

4. For each binary $1:n$ relationship, choose the n -side entity and include an FK w.r.t the other entity.



We must choose instructor

We end up with:

instructor

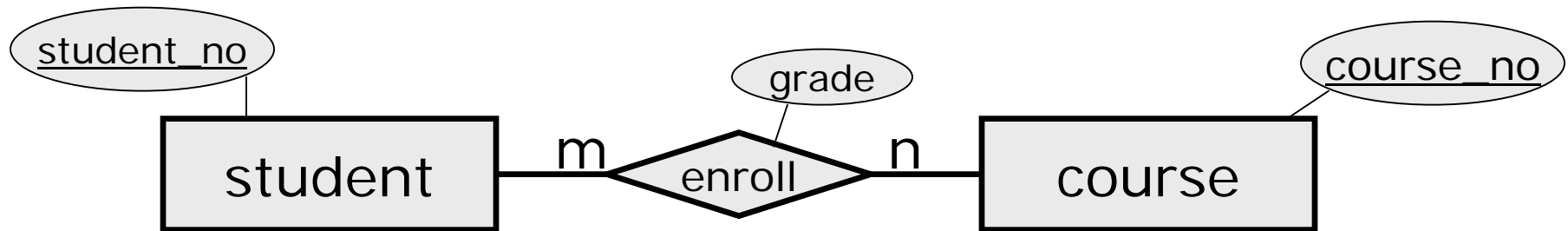
<u>instr_no</u>	iname	dept_no
-----------------	-------	---------

Note that Step 1 would lead to the instructor relation - we have now augmented instructor with the dept_no attribute.

- PK is instr_no
- dept_no is an FK

ER to Relational Example: Step 5

5. For each binary $M:N$ relationship, create a relation for the relationship
- include PKs of both participating entities and any attributes of the relationship
 - PK is the catenation of the participating entities' PKs



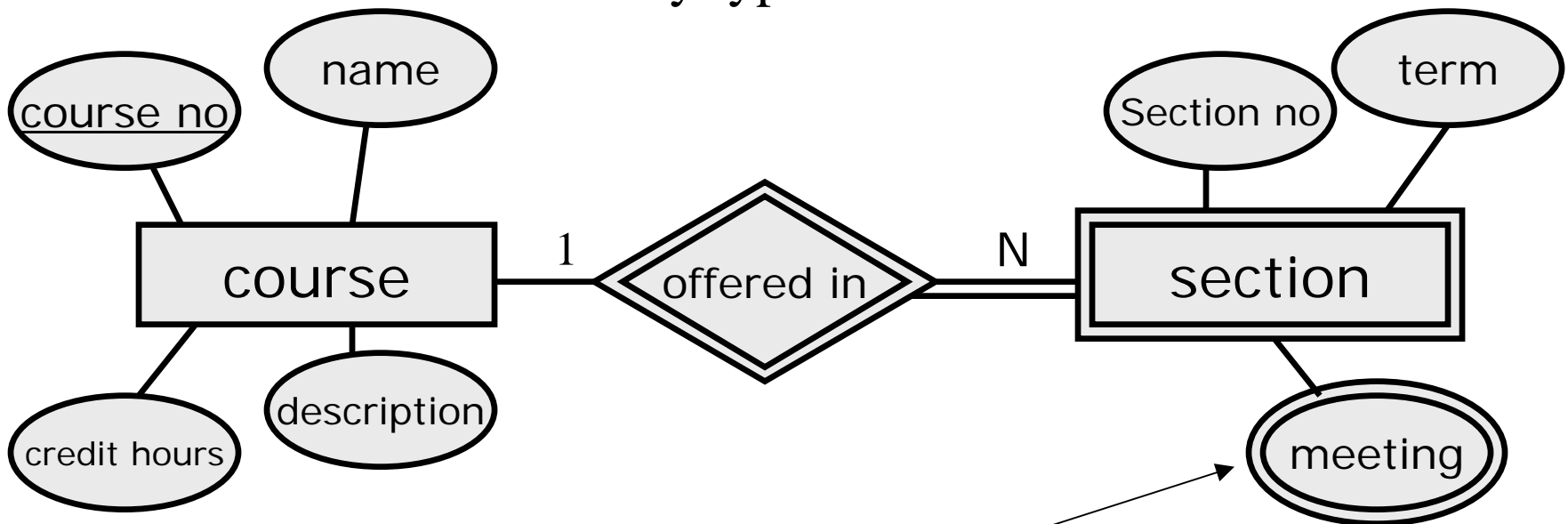
Enroll

<u>student_no</u>	<u>Course_no</u>	grade
-------------------	------------------	-------

- PK is {student_no, course_no}
- student_no is a FK
- course_no is a FK
- grade is an attribute of Enroll

ER to Relational Example: Step 6

6. For each multi-valued attribute create a new relation
- include the PK attributes of the entity type
 - PK is the PK of the entity type and the multi-valued attribute



meeting is a multi-valued attribute

ER to Relational Example: Step 6 (Cont'd)

Create a relation for meeting

Section was created because of Step 2 - its PK is {course_no, section_no}



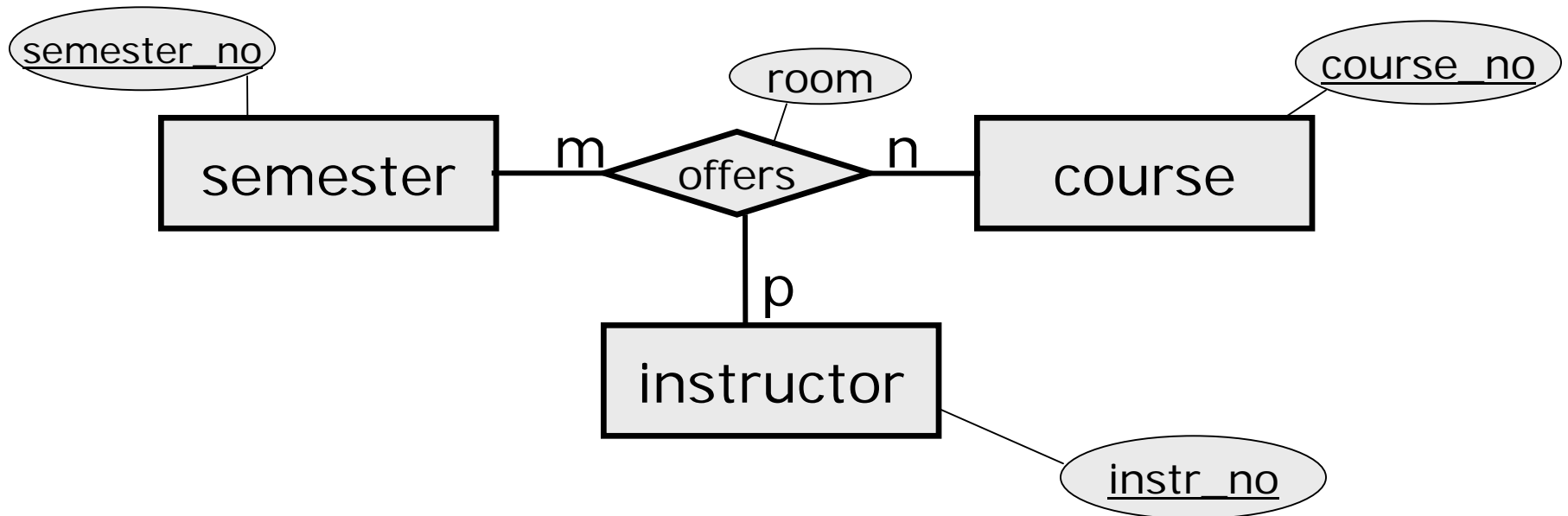
Meeting

<u>course_no</u>	<u>section_no</u>	<u>meeting</u>
------------------	-------------------	----------------

- PK is {course_no, section_no, meeting}.
- Meeting is an all-key relation.

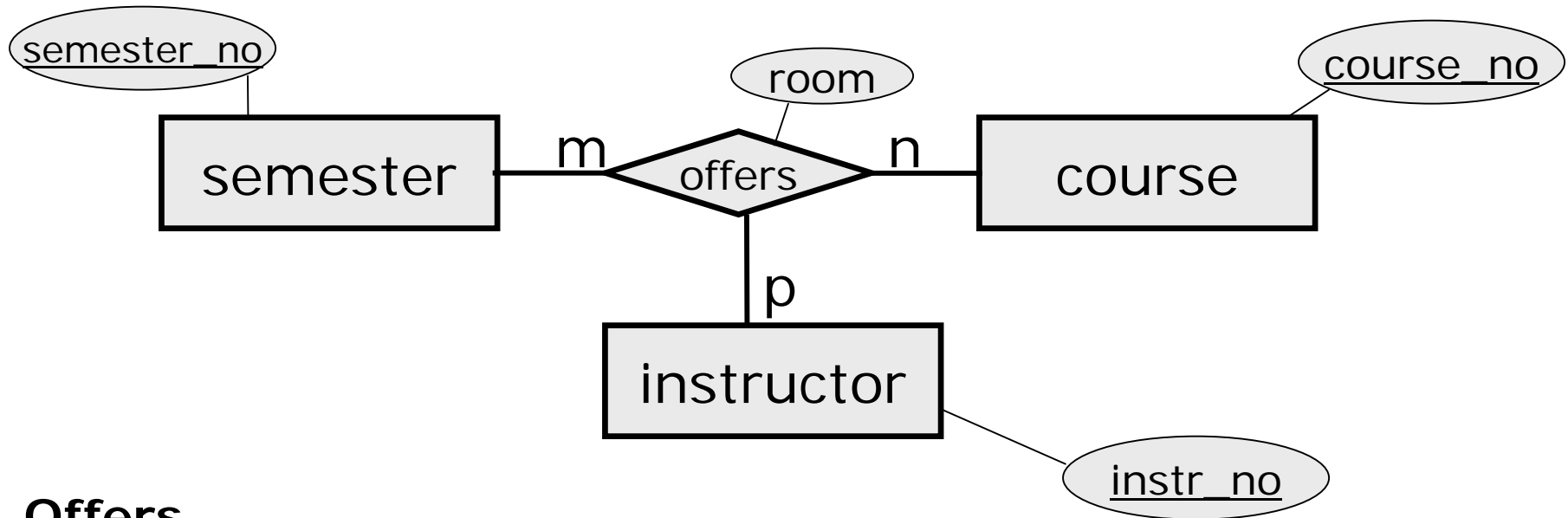
ER to Relational Example: Step 7

7. For each n -ary relationship, create a relation for the relationship
- include PKs of all participating entities and any attributes of the relationship
 - PK may be the catenation of the participating entity PKs (depends on cardinalities)



ER to Relational Example: Step 7 (Cont'd)

Need one relation, offers, with PK of {semester_no, course_no, instr_no}



Offers

<u>course_no</u>	<u>instr_no</u>	<u>semester_no</u>	Room_no
------------------	-----------------	--------------------	---------