Database Management Systems (DBMS)

Lec 10: Entity-Relationship Model (cont.)

Ramesh K. Jallu

IIIT Raichur

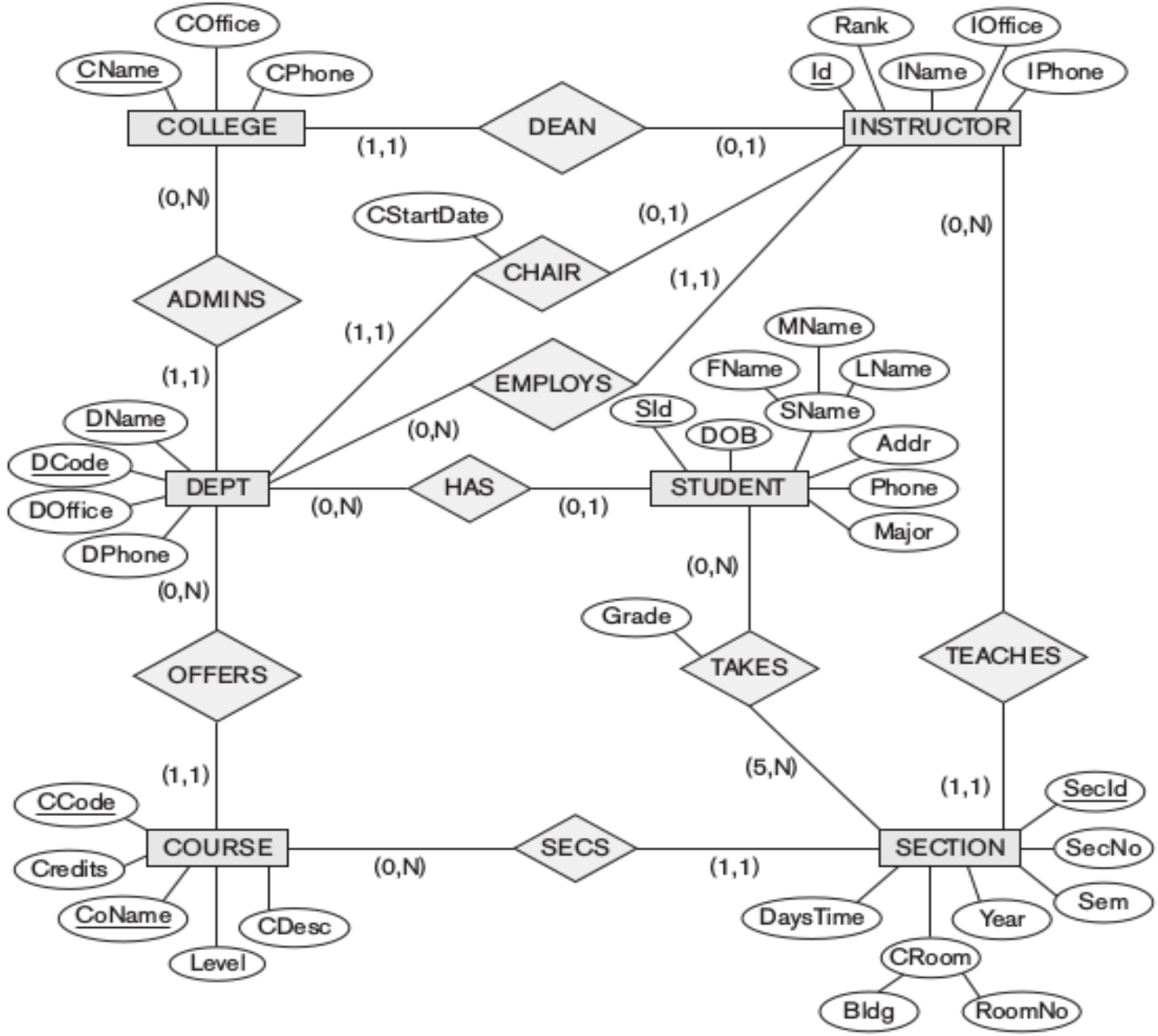
Date: 2/3/21

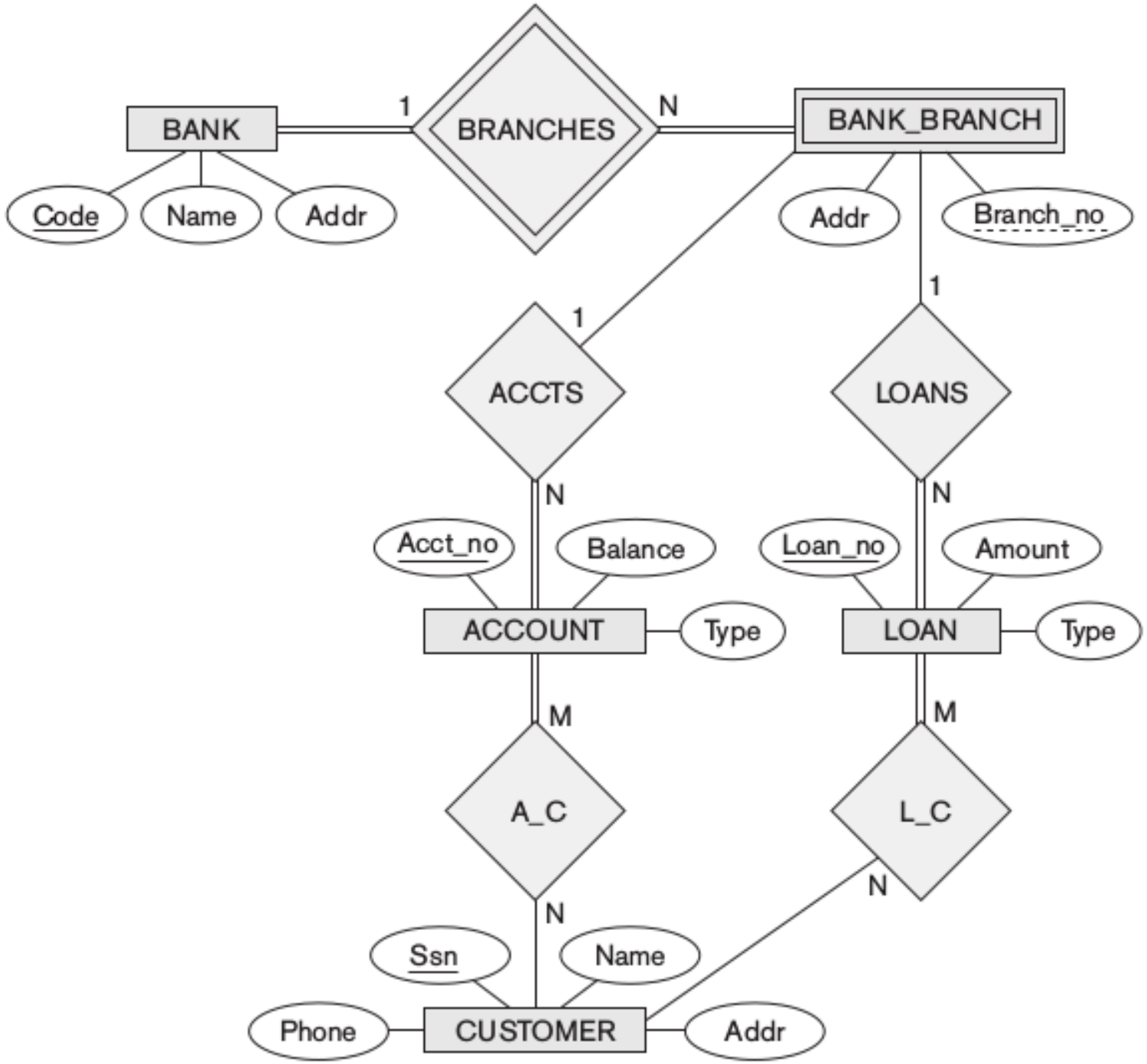
Recap

- Relationship sets
- Structural constraints
- Identifying relationships

Today's plan

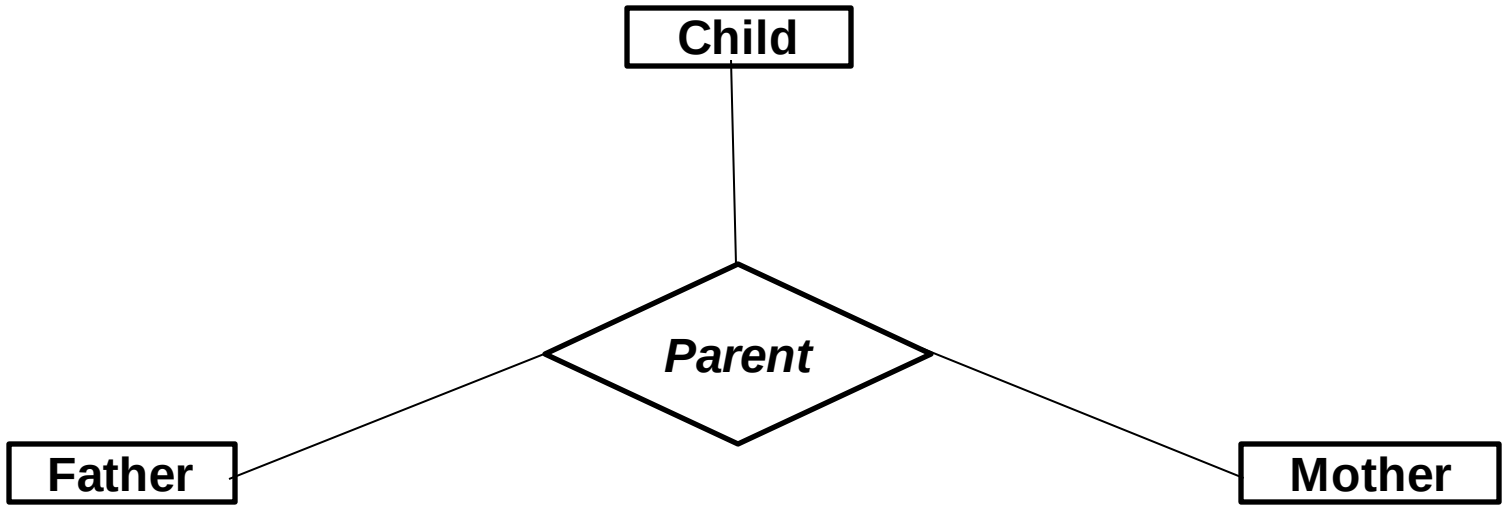
- Two more examples
  - University database
  - Bank database
- Relationship types of higher degree
- Common mistakes in designing ER diagrams


COffice
CName
CPhone
COLLEGE
DEAN
(1,1)
(0,1)
Rank
IOffice
Id
IName
IPhone
INSTRUCTOR
(0,N)
CStartDate
(0,1)
CHAIR
(1,1)
(0,N)
ADMINS
(1,1)
(1,1)
MName
FName
LName
EMPLOYS
SName
SId
DOB
Addr
DName
(0,N)
DCode
DEPT
HAS
STUDENT
Phone
DOffice
(0,N)
(0,1)
Major
DPhone
(0,N)
Grade
(0,N)
TAKES
TEACHES
OFFERS
(5,N)
(1,1)
(1,1)
SecId
CCode
COURSE
SECS
SECTION
SecNo
Credits
(0,N)
(1,1)
Sem
DaysTime
Year
CoName
CDesc
CRoom
Level
Bldg
RoomNo

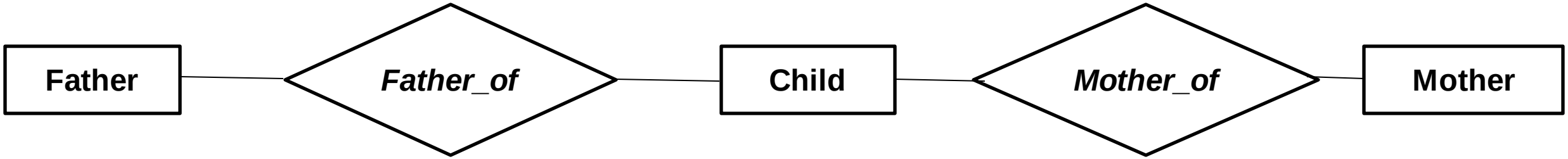

BANK
BRANCHES
BANK_BRANCH
1
N
Code
Name
Addr
Addr
Branch_no
1
1
ACCTS
LOANS
N
N
Acct_no
Balance
Loan_no
Amount
ACCOUNT
Type
LOAN
Type
M
M
A_C
L_C
N
N
Ssn
Name
Phone
CUSTOMER
Addr

Binary vs *n*-ary relationships

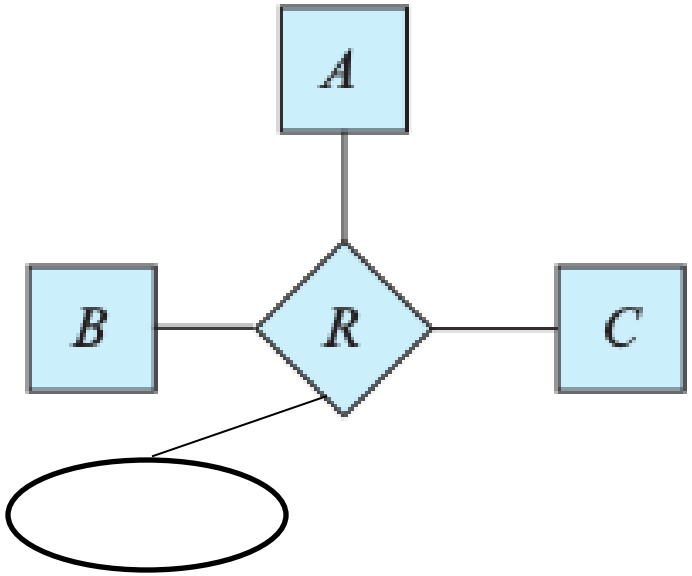
- Some relationships that appear to be nonbinary could actually be better represented by several binary relationships

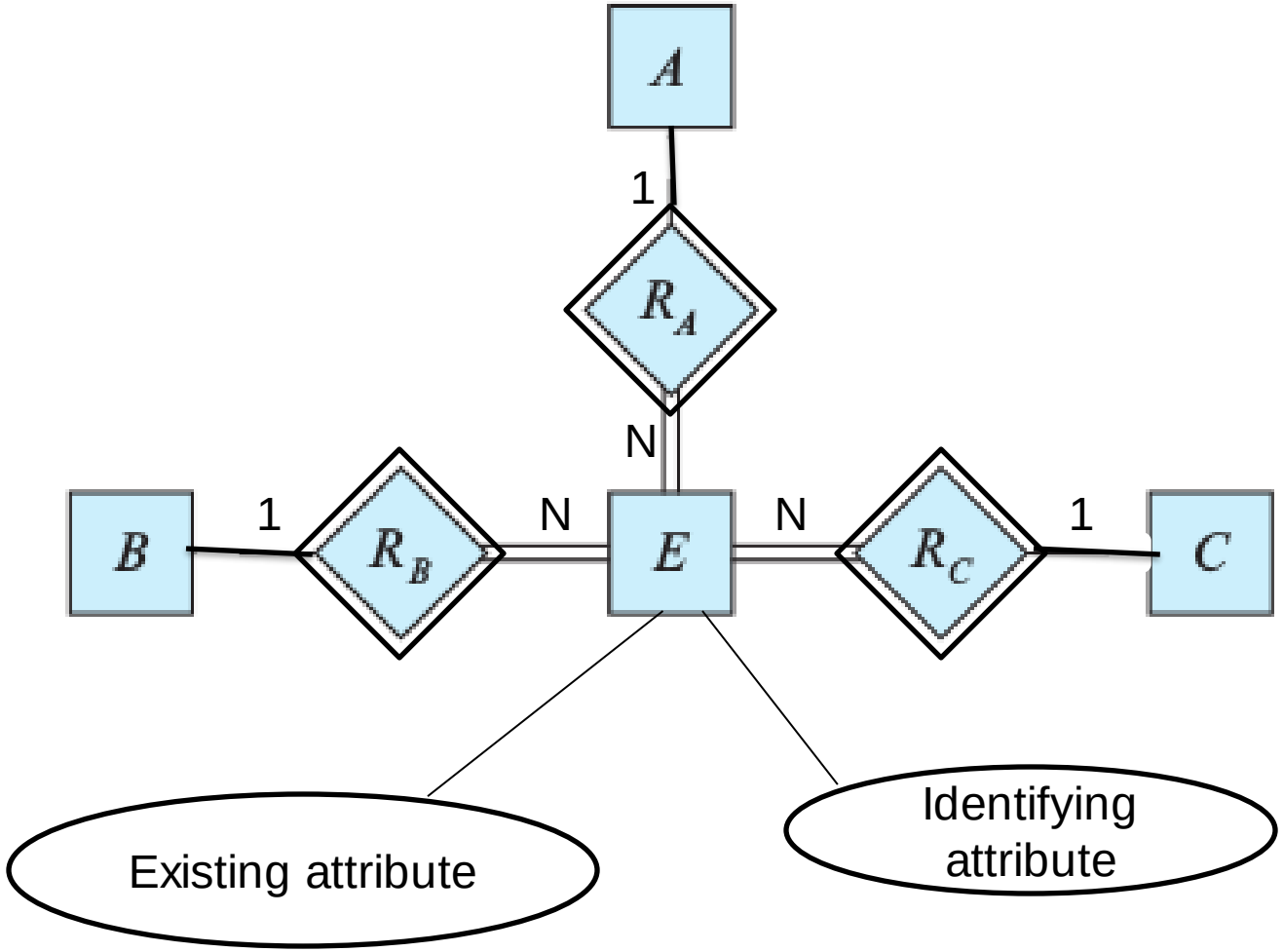


Binary vs *n*-ary relationships



Converting an *n*-ary relationship to a set of binary relations





Rules for converting

- We replace the relationship set R with an entity set E, and we create three relationship sets
  - R_A, R_B, and R_C are many-to-one relationships from E to A, E to B, and from E to C, resp.
- E is required to have total participation in each of R_A , R_B , and R_C
- If R had any attributes, these are assigned to entity set E; further, a special identifying attribute is created for E

Rules for converting (Cont.)

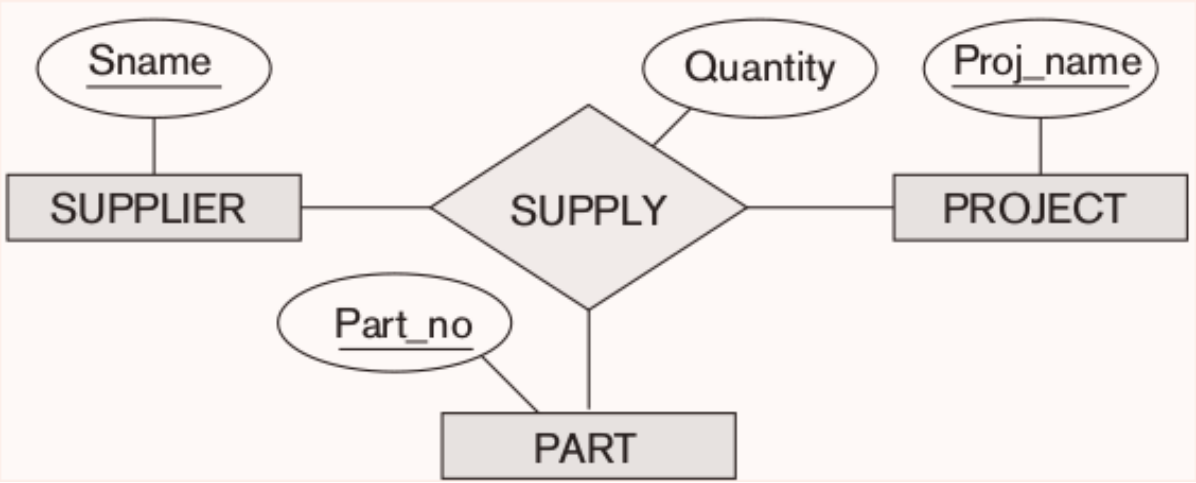
- For each relationship (a_i, b_i , c_i) in the relationship set R, we create a new entity e_i in the entity set E
- In each of the three new relationship sets, we insert a relationship (e_i, a_i) in R_A , (e_i, b_i) in R_B, and (e_i, c_i) in R_C
- We can generalize this process in a straightforward manner to n-ary relationship sets
- Thus, conceptually, we can restrict the ERmodel to include only binary relationship sets

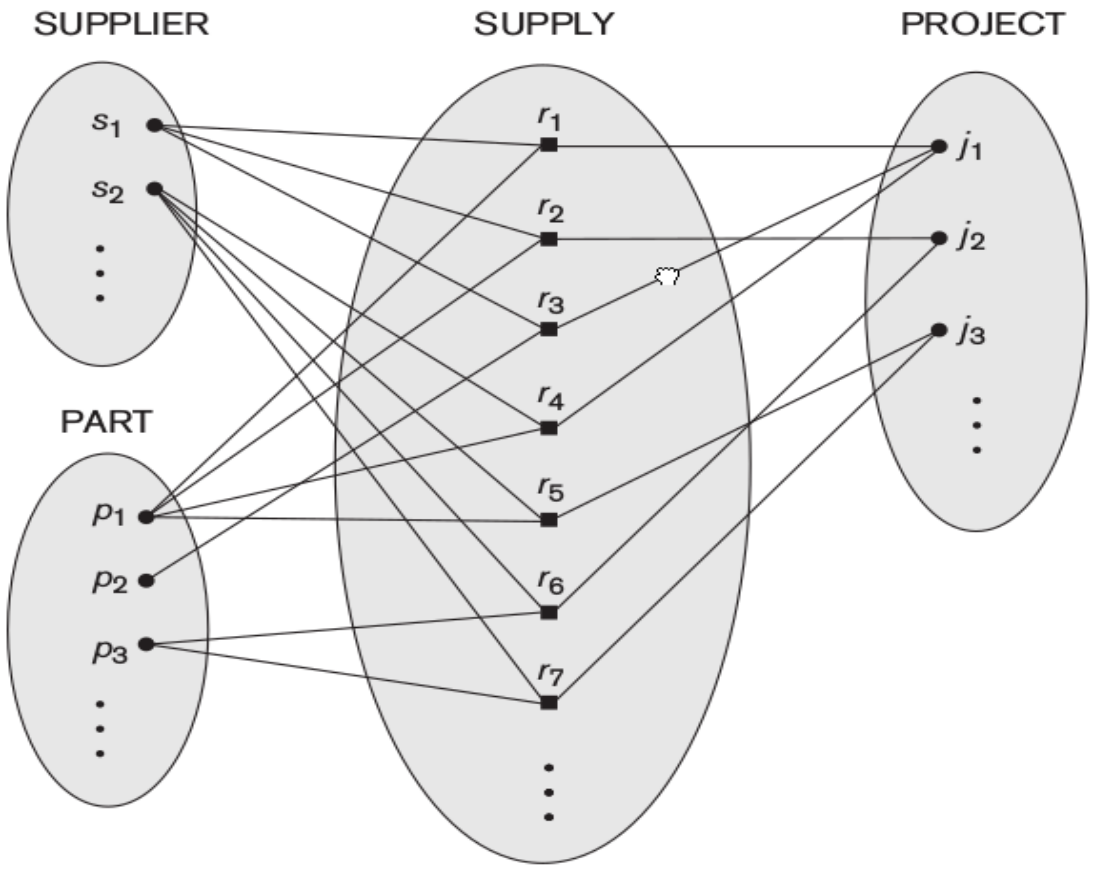
Not always desirable

- An identifying attribute may have to be created for the entity set created to represent the relationship set. This attribute, along with the extra relationship sets required, increases the complexity of the design

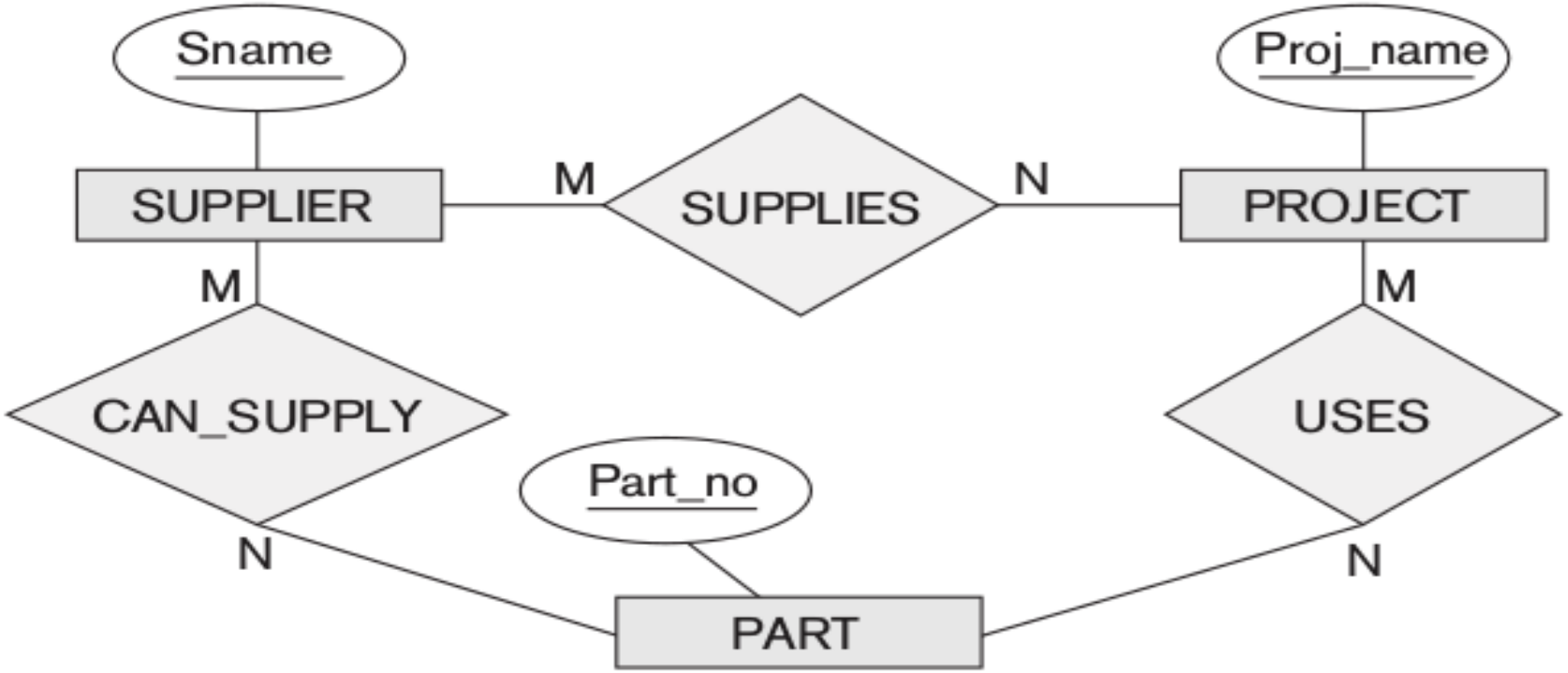
- An *n*-ary relationship set shows more clearly that several entities participate in a single relationship.
- There may not be a way to translate constraints on the ternary relationship into constraints on the binary relationships. For example, consider a constraint that says that R is many-to-one from A, B to C; that is, each pair of entities from A and B is associated with at most one C entity. This constraint cannot be expressed by using cardinality constraints on the relationship sets R_A, R_B, and R_C.

Relationship types of degree higher than two

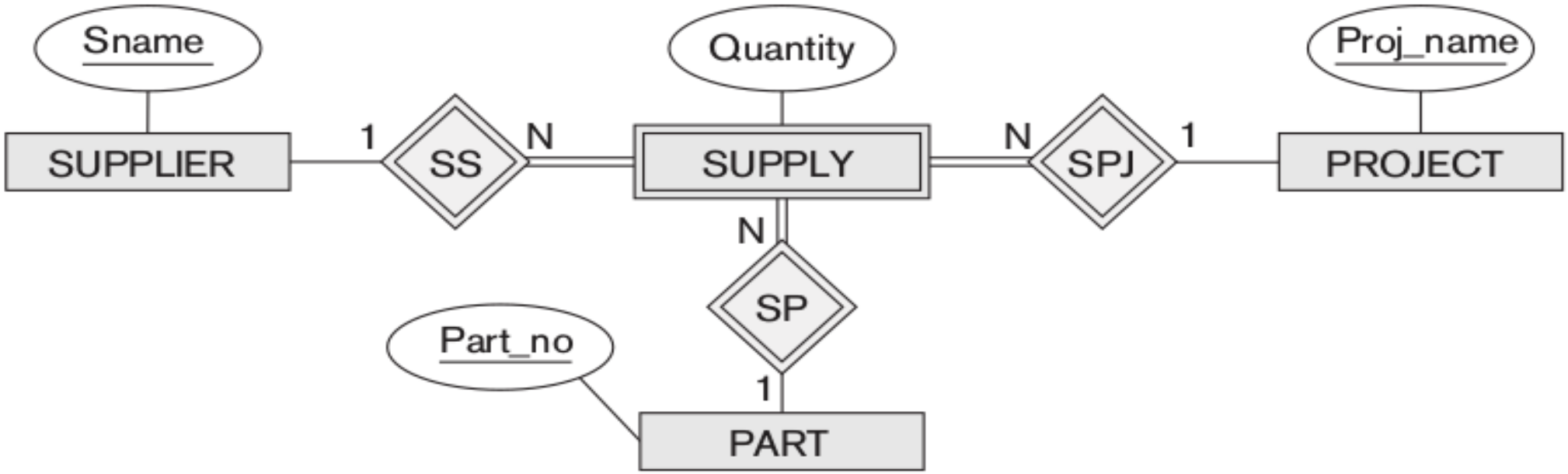




Relationship types of degree higher than two



Example

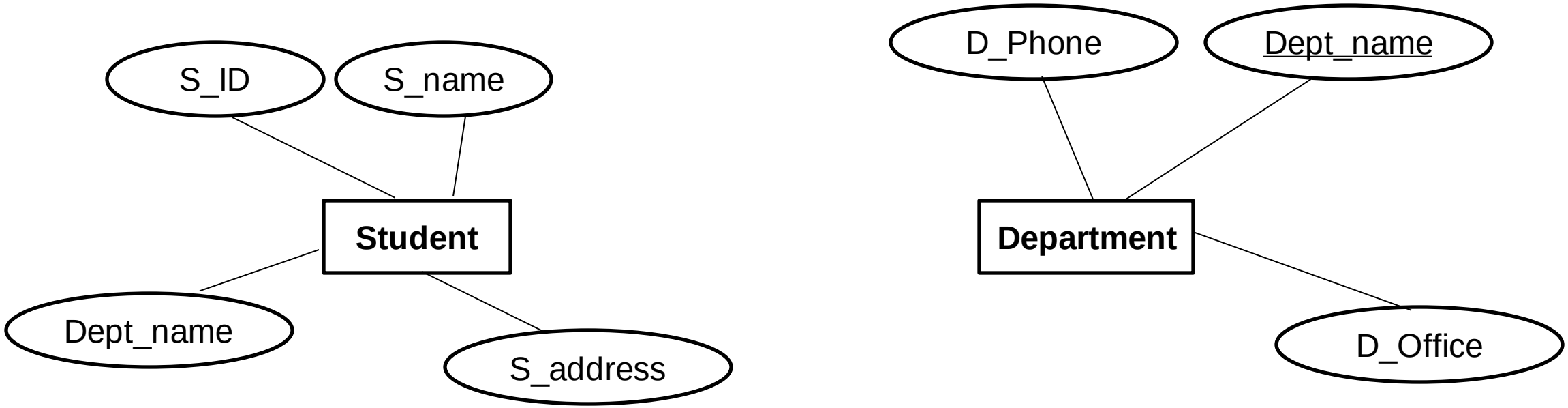

Sname
Quantity
Proj_name
SUPPLIER
1
SS
N
SUPPLY
N
SPJ
1
PROJECT
N
SP
Part_no
1
PART

Key points

- It is often tricky to decide whether a particular relationship should be represented as a relationship type of degree *n* or should be broken down into several relationship types of smaller degrees
- The designer must base this decision on the semantics or meaning of the particular situation being represented
- The designer must base this decision on the semantics or meaning of the particular situation being represented
- If the ER model that permit only binary relationships, a ternary relationship must be represented as a weak entity type, with no partial key and with three identifying relationships

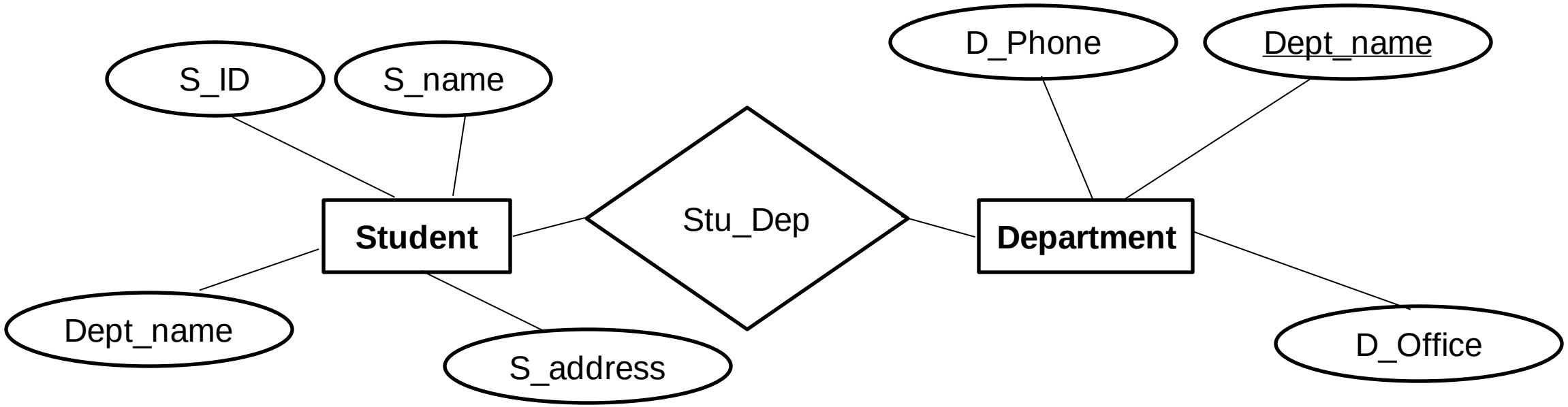
Common mistakes in ER diagrams

- Use of the primary key of an entity type as an attribute of another entity type, instead of using a relationship



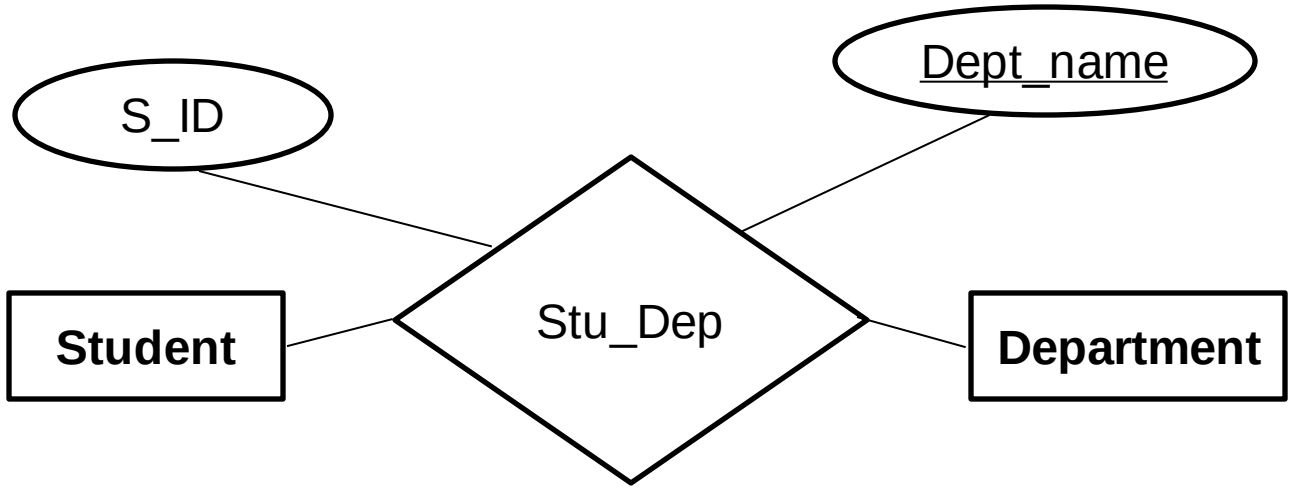
Common mistakes in ER diagrams

- Use of the primary key of an entity type as an attribute of another entity type, instead of using a relationship



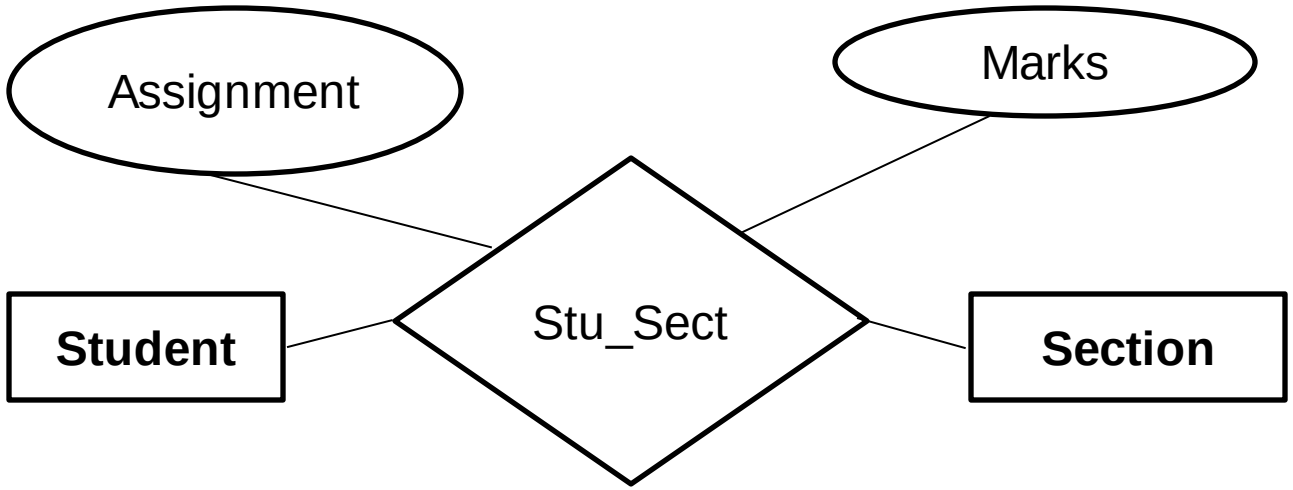
Common mistakes in ER diagrams

- To designate the primary key attributes of the related entity sets as attributes of the relationship



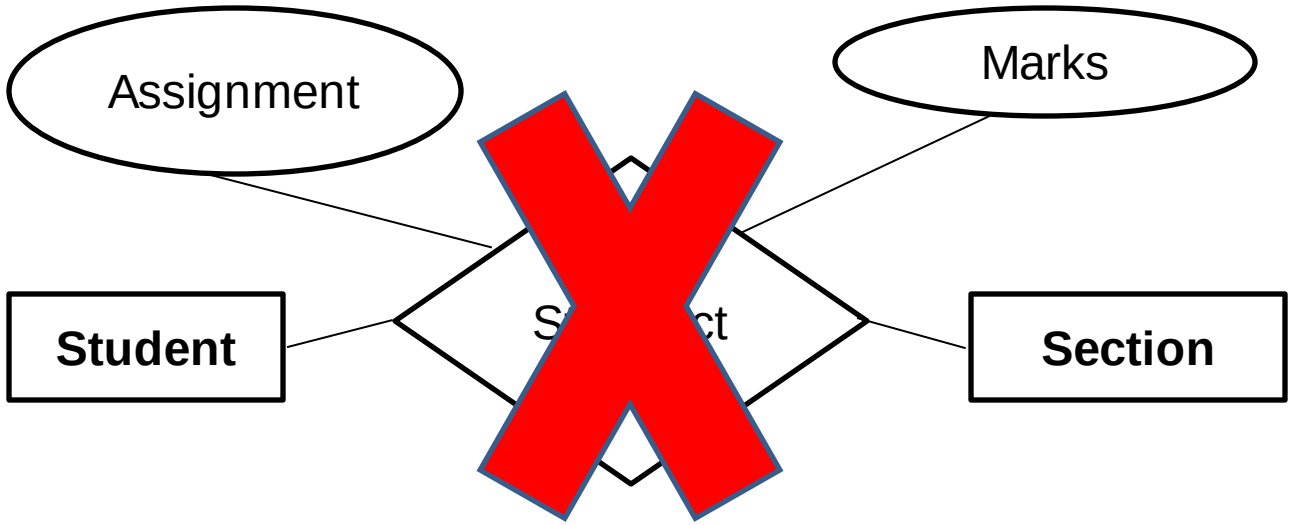
Common mistakes in ER diagrams

- To use a relationship with a single-valued attribute in a situation that requires a multivalued attribute
  - e.g., store the marks that a student gets in different assignments of a course



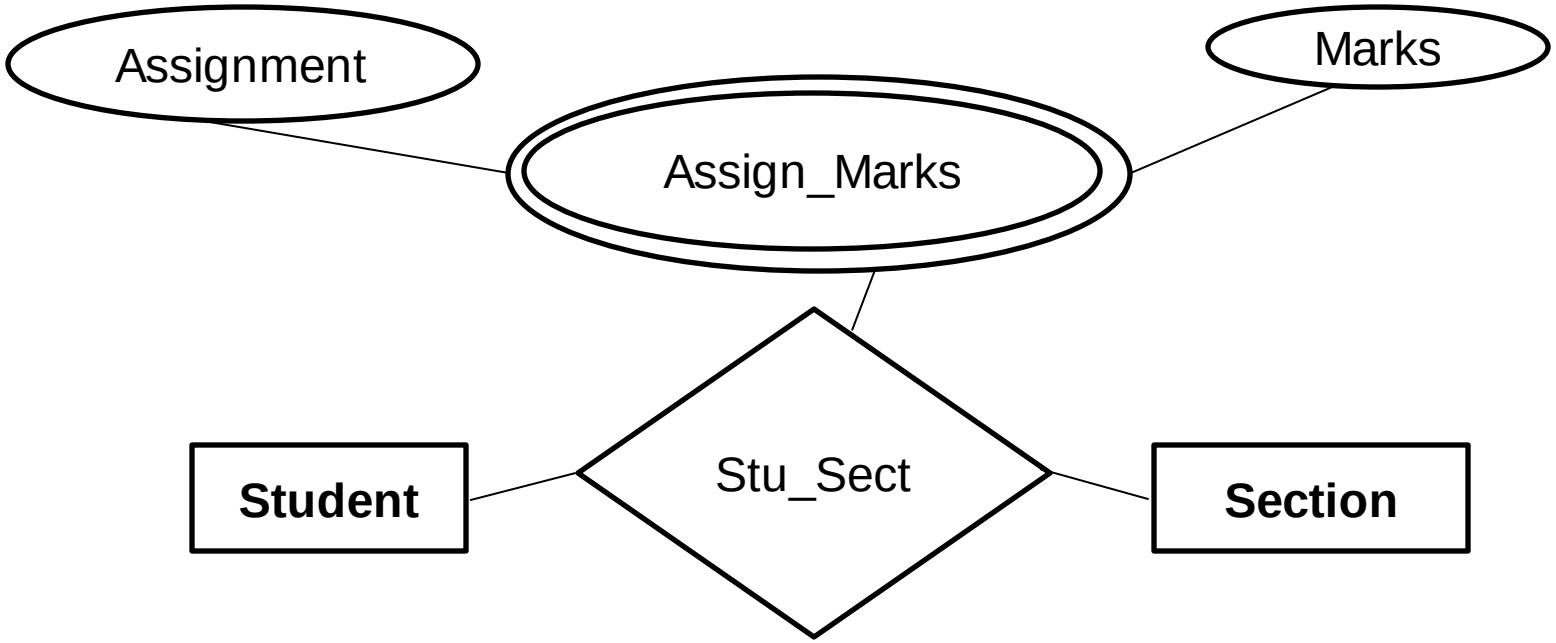
Common mistakes in ER diagrams

- To use a relationship with a single-valued attribute in a situation that requires a multivalued attribute
  - e.g., store the marks that a student gets in different assignments of a course



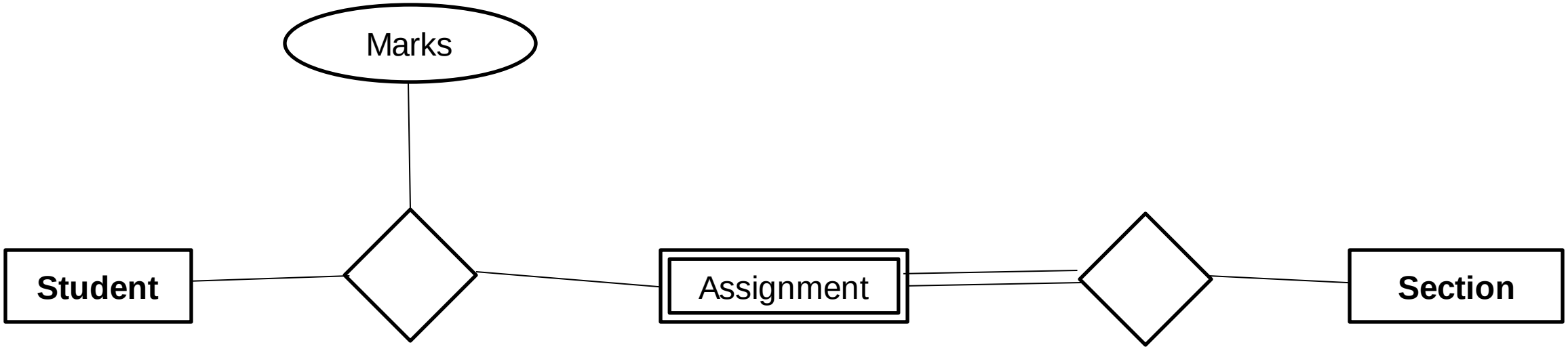
Common mistakes in ER diagrams

- To use a relationship with a single-valued attribute in a situation that requires a multivalued attribute
  - e.g., store the marks that a student gets in different assignments of a course



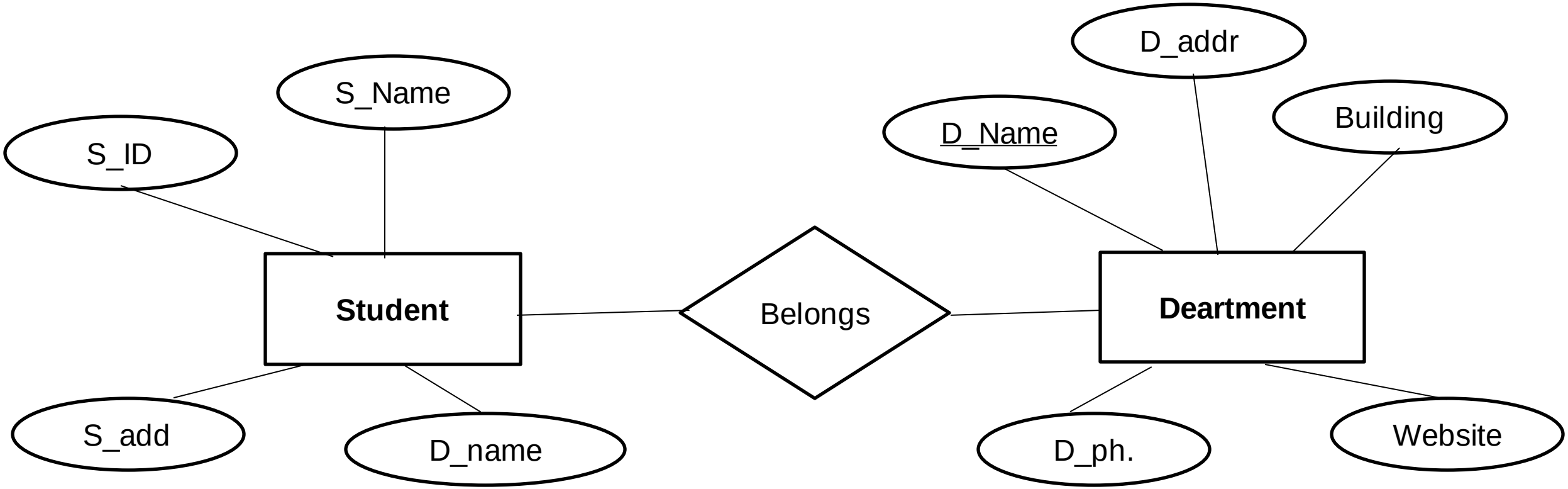
Common mistakes in ER diagrams

- To use a relationship with a single-valued attribute in a situation that requires a multivalued attribute
  - e.g., store the marks that a student gets in different assignments of a course



Removing redundant attributes

- The attribute D_name in **Student** replicates information present in the relationship, and is therefore redundant and needs to be removed



Thank you!