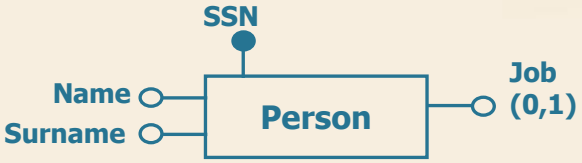
Relational logic design

Translation in relational model: entities and many to many relationships

Translation in relational model

- It is executed on the restructured ER schema
  - i.e., the schema without hierarchies, multivalued attributes and compounds attributes
- Transformations
  - Each entity is translated into a table with the same attributes
  - For relations we need to consider the maximum cardinality

Entities translation



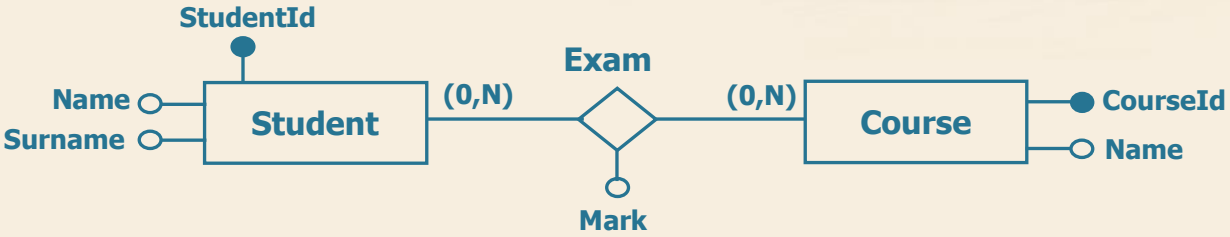
Person(SSN, Name, Surname, Job*)

- Primary key underlined
- Optional attributes indicated by * (asterisk)

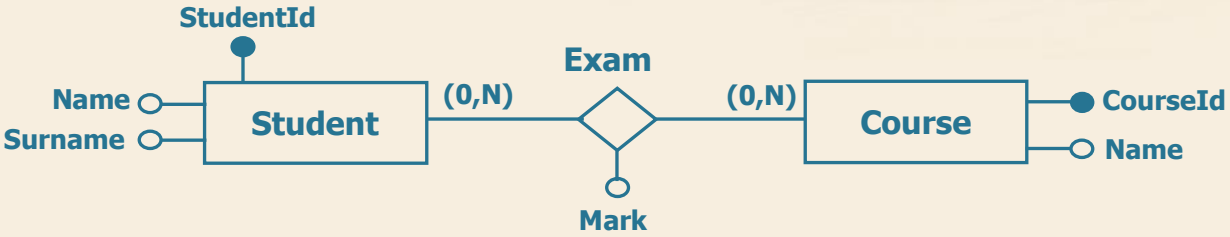
Translation of many to many binary relations

- Each many to many relationship is translate into a table
  - The primary key is the combination of the identificators of all the linked entities
  - It is possible to rename the attributes of the table that corresponds to the relation (needed in case of recursive relations)

Many to many binary relationship

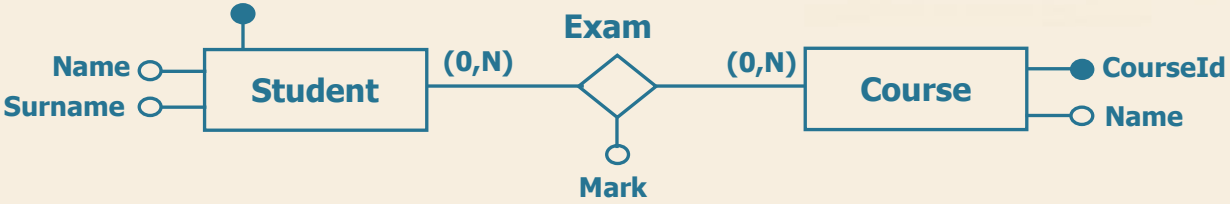


Many to many binary relationship: entity



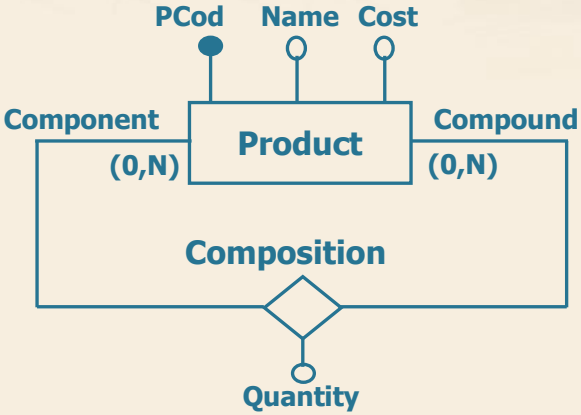
Student(<u>StudentId</u>, Name, Surname)
Course(<u>CourseId</u>, Name)

Many to many binary relationship

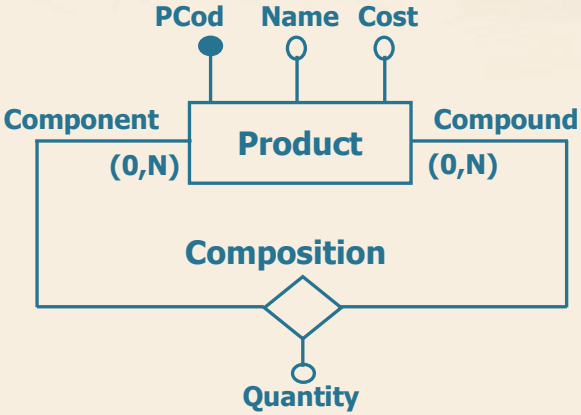


Student(StudentId, Name, Surname)
Course(CourseId, Name)
Exam(StudentId, CourseId, Mark)

Recursive many to many binary relationship

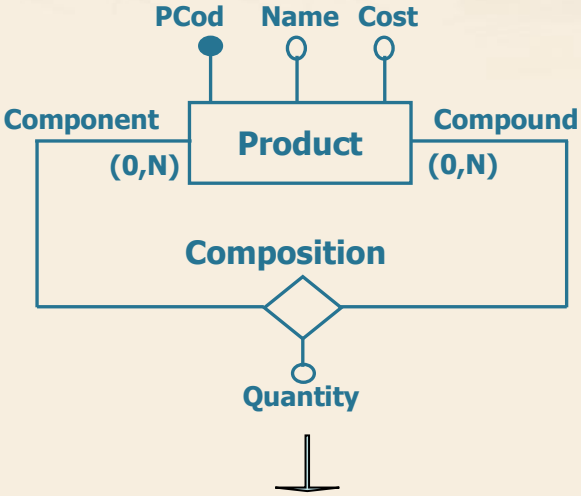


Recursive many to many binary relationship



Product(PCod, Name, Cost)

Recursive many to many binary relationship



Product(PCod, Name, Cost)
Composition(CompoundCod, ComponentCod, Quantity)

Relational logic design

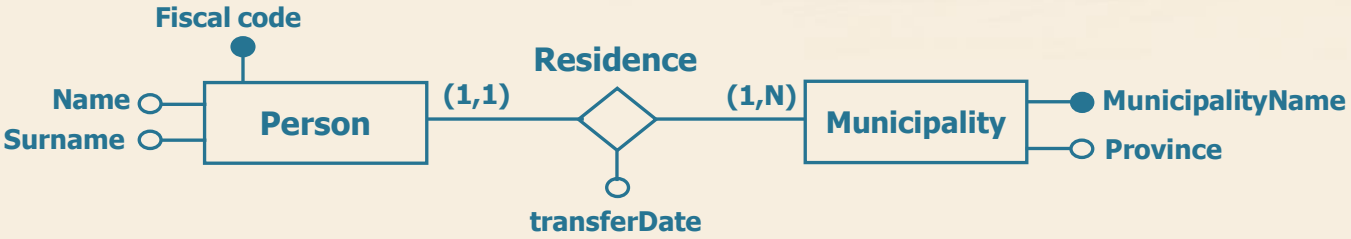
Translation in relational model: one to many relationship

DBMG

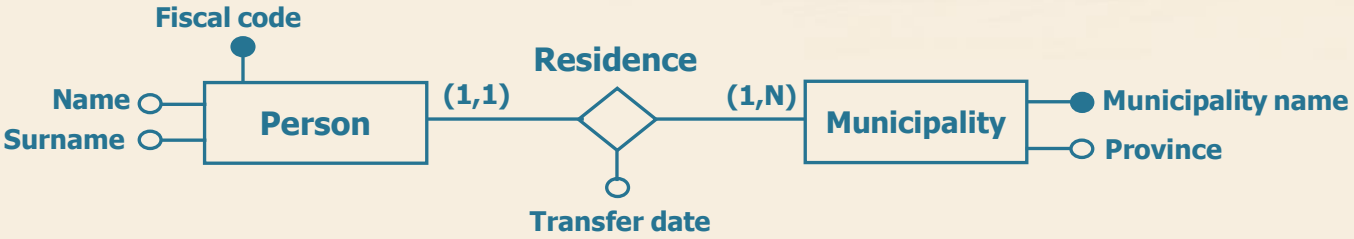
One to many binary relationship

- There are two translation modes
  - Through attributes
  - Through a new table

One to many binary relationship



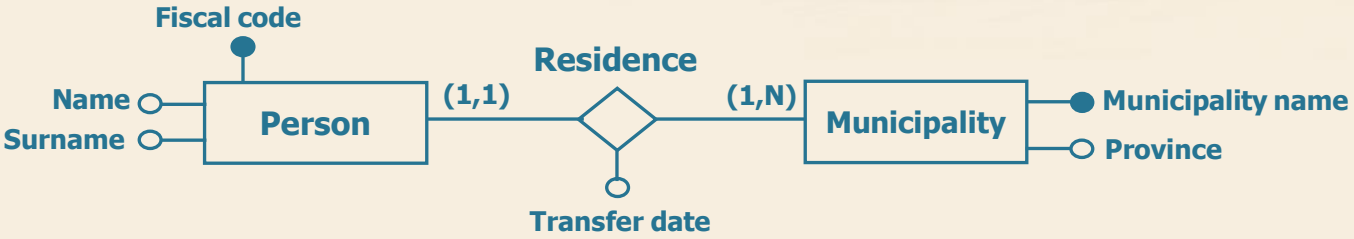
One to many binary relationship: entity



Person(FiscalCode, Name, Surname)

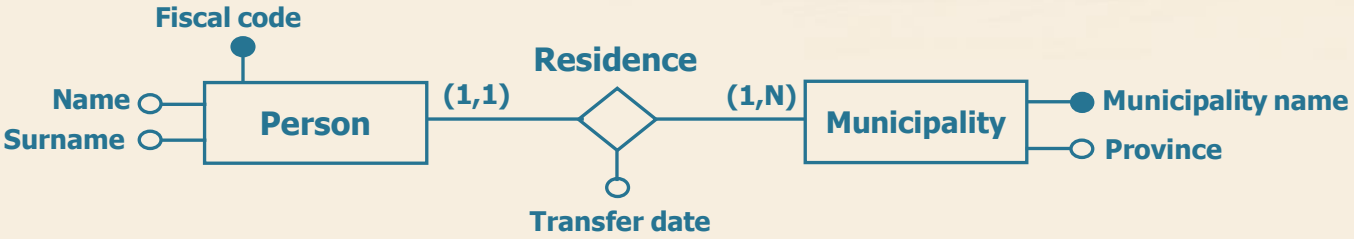
Municipality(MunicipalityName, Province)

One to many binary relationship



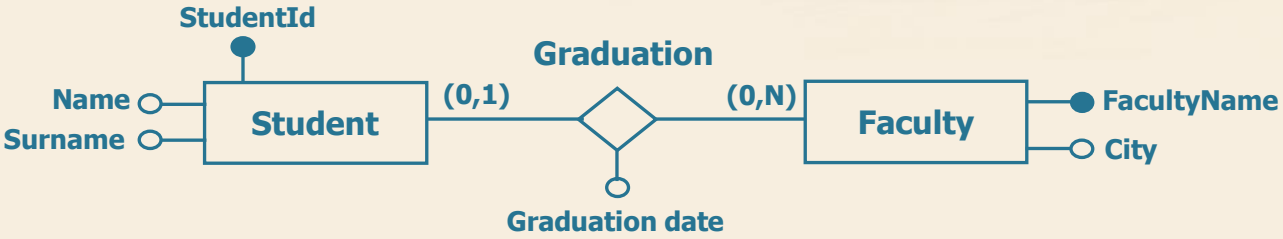
Person(<u>FiscalCode</u>, Name, Surname,
MunicipalityName)
Municipality(<u>MunicipalityName</u>, Province)

One to many binary relationship

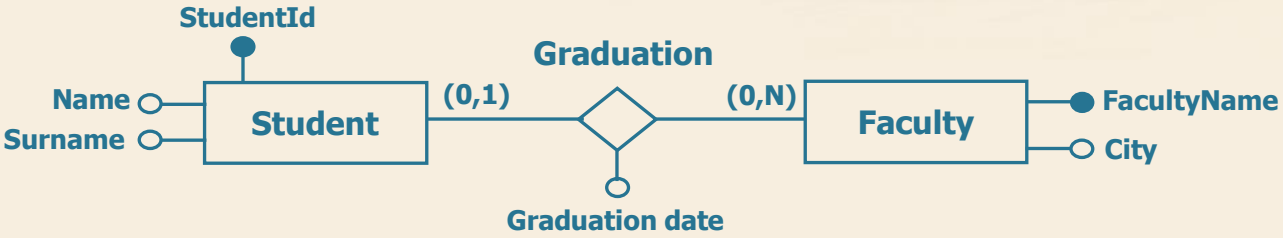


Persona(<u>FiascalCode</u>, Name, Surname,
MunicipalityName, *TransferDate*)
Municipality(<u>MunicipalityName</u>, Province)

One to many binary relationship

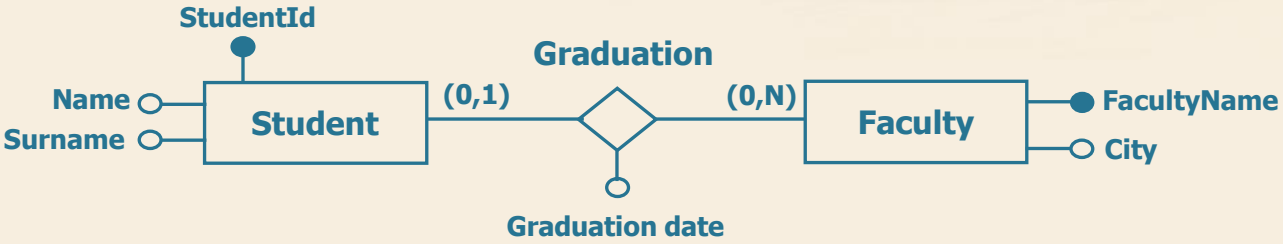


One to many binary relation: alternative n.1



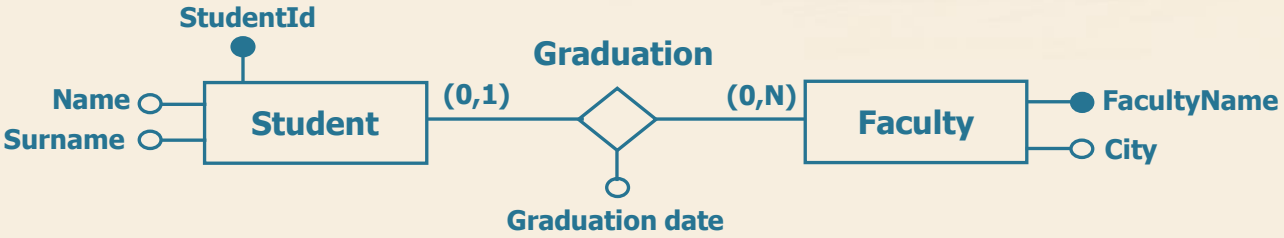
Student(StudentiId, Name, Surname)
Faculty(FacultyName, City)

One to many binary relation: alternative n.1



Student(StudentiId, Name, Surname)
Faculty(FacultyName, City)
Graduation(StudentId, FacultyName, GraduationDate)

One to many binary relation: alternative n.2



Student(StudentId, Name, Surname, FacultyName*, GraduationDate*)
Faculty(FacultyName, City)

Relational logic design

Translation in relational model: one to one relationship

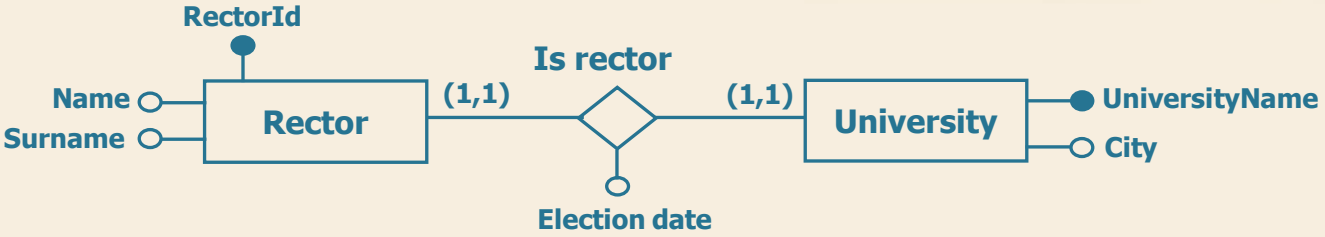
DBMG

One to one binary relationship

- Different translations are possible
  - It depends on the minimum cardinality value

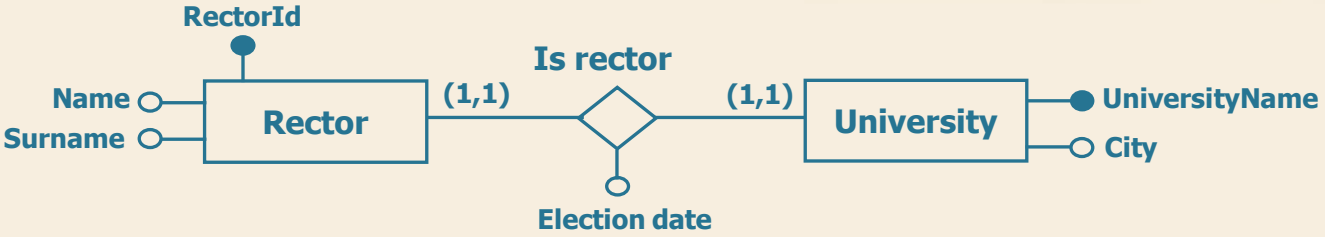
One to one binary relationship: case 1

- Mandatory partecipation from both sides



One to one binary relation: alternative n.1

- Mandatory partecipation from both sides

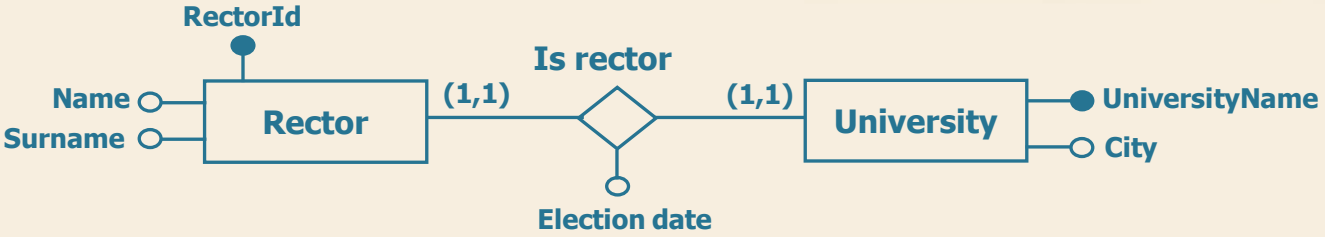


Rector(<u>RectorId</u>, Name, Surname)

University(<u>UniversityName</u>, City)

One to one binary relation: alternative n.1

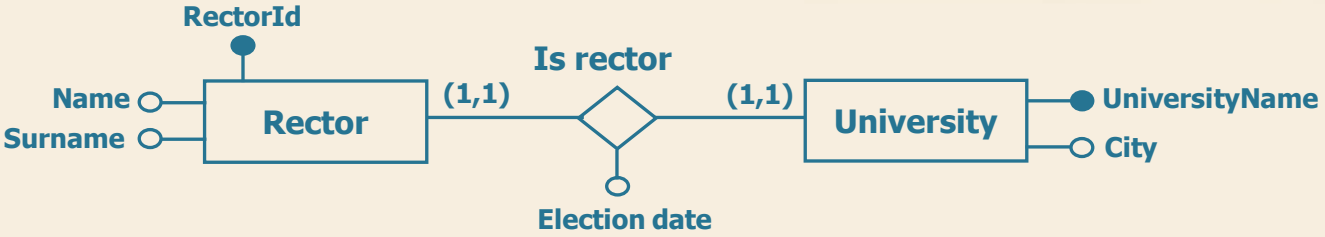
- Mandatory partecipation from both sides



Rector(<u>RectorId</u>, Name, Surname, *UniversityName, ElectionDate*)
University(<u>UniversityName</u>, City)

One to one binary relation: alternative n.2

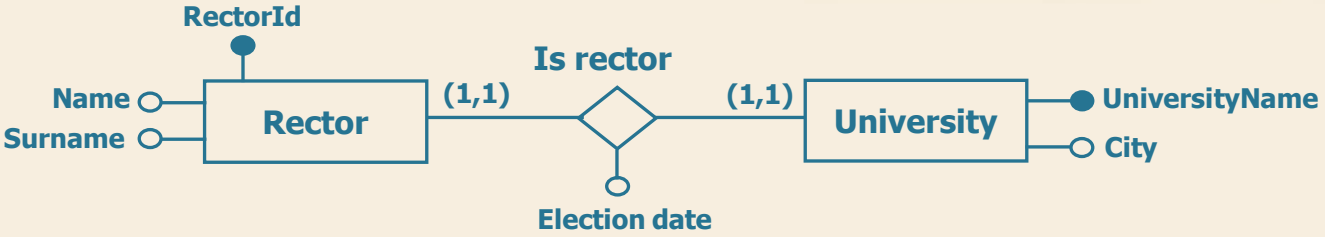
- Mandatory partecipation from both sides



Rector(<u>RectorId</u>, Name, Surname)
University(<u>UniversityName</u>, City)

One to one binary relation: alternative n.2

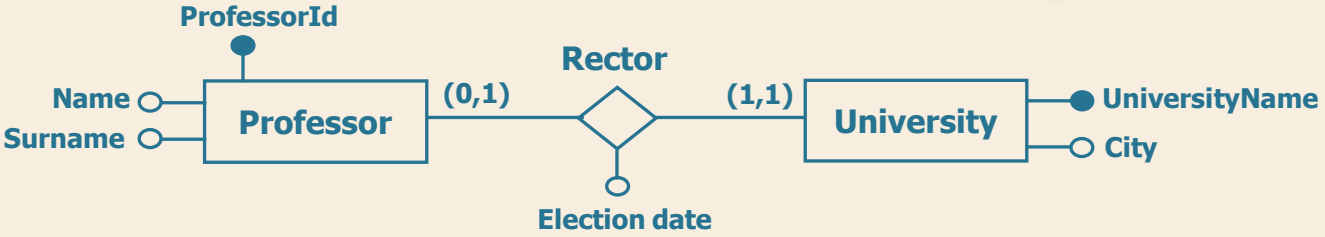
- Mandatory partecipation from both sides



Rector(<u>RectorId</u>, Name, Surname)
University(<u>UniversityName</u>, City, *RectorId*, *ElectionDate*)

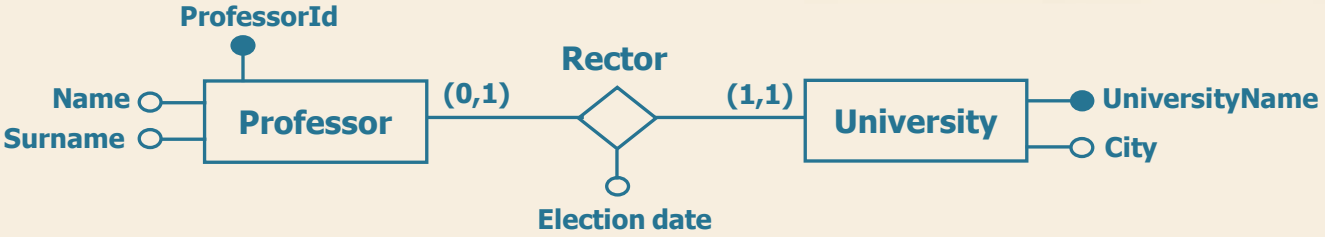
One to one binary relationship: case 2

- Optional partecipation on one side



One to one binary relationship: entity

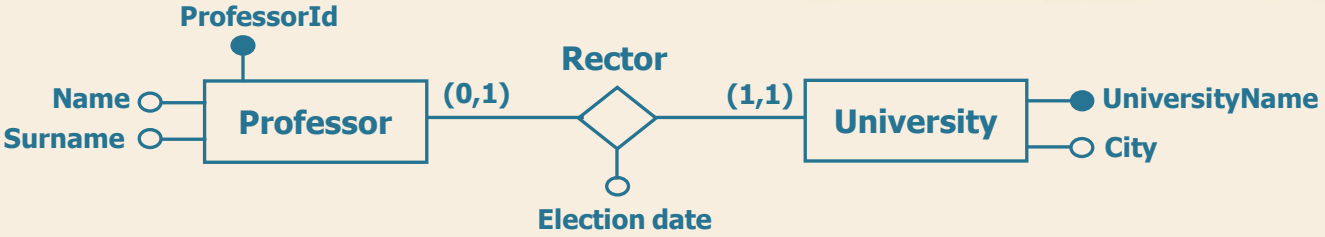
- Optional partecipation on one side



Professor(<u>ProfessorId</u>, Name, Surname)
University(<u>UniversityName</u>, City)

One to one binary relationship

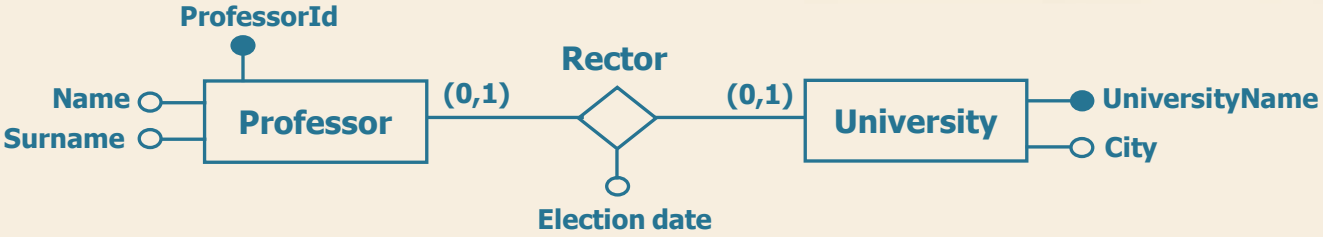
- Optional partecipation on one side



Professor(<u>ProfessorId</u>, Name, Surname)
University(<u>UniversityName</u>, City, *ProfessorId*, *ElectionDate*)

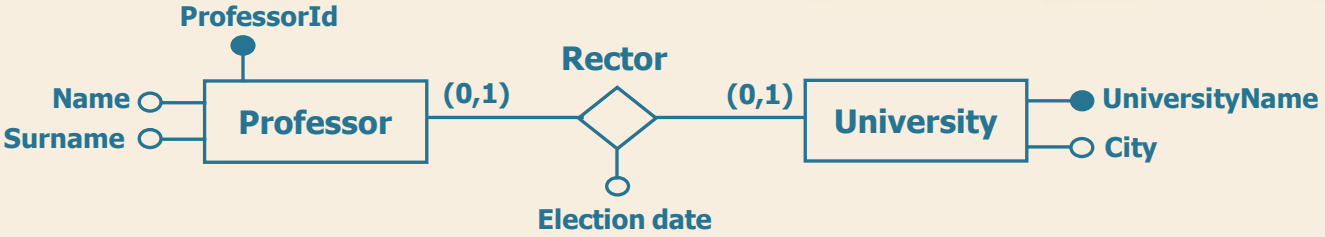
One to one binary relationship: case 3

- Optional partecipation from both sides



One to one binary relationship:alternative n.1

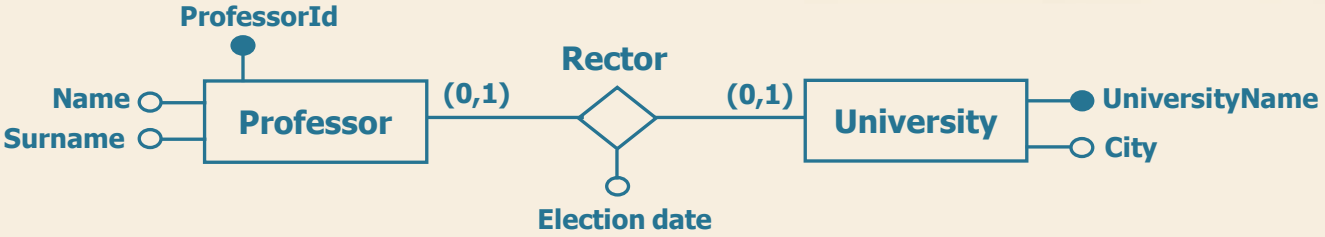
- Optional partecipation from both sides



Professor(<u>ProfessorId</u>, Name, Surname)
University(<u>UniversityName</u>, City)

One to one binary relationship:alternative n.1

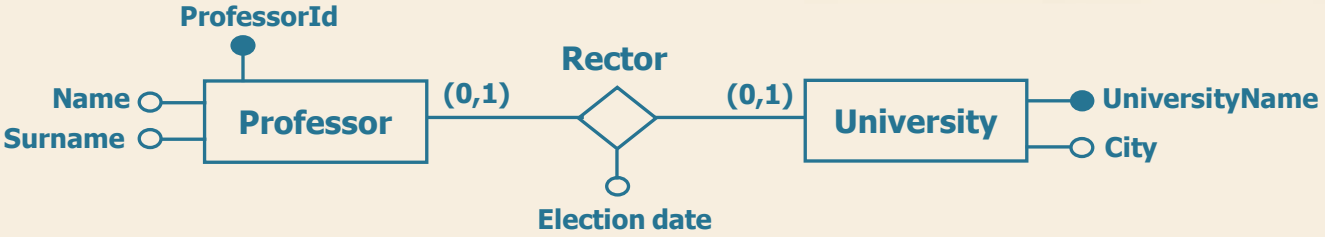
- Optional partecipation from both sides



Professor(<u>ProfessorId</u>, Name, Surname)
University(<u>UniversityName</u>, City)
Rector(<u>ProfessorId</u>, UniversityName, ElectionDate)

One to one binary relationship:alternative n.2

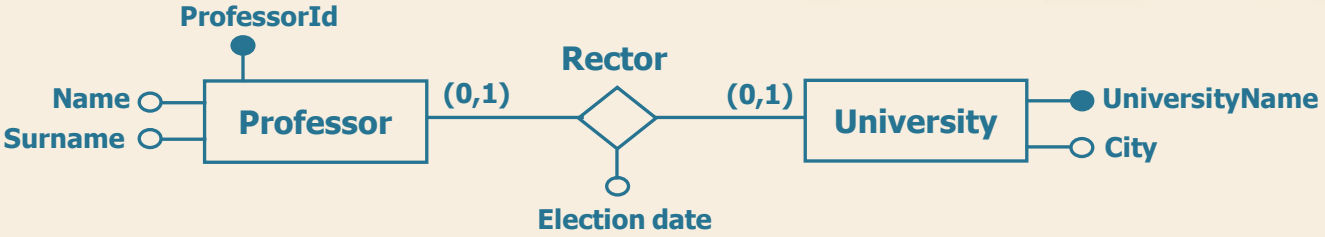
- Optional partecipation from both sides



Professor(<u>ProfessorId</u>, Name, Surname)
University(<u>UniversityName</u>, City)
Rector(ProfessorId, <u>UniversityName</u>, ElectionDate)

One to one binary relationship:alternative n.3

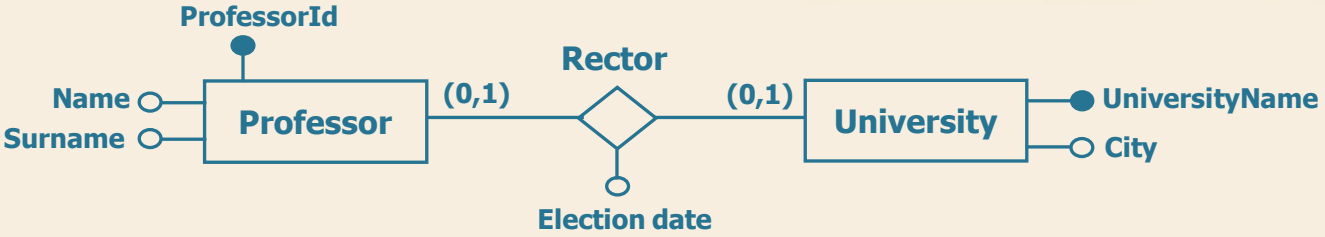
- Optional partecipation from both sides



Professor(<u>ProfessorId</u>, Name, Surname)
University(<u>UniversityName</u>, City)

One to one binary relationship:alternative n.3

- Optional partecipation from both sides



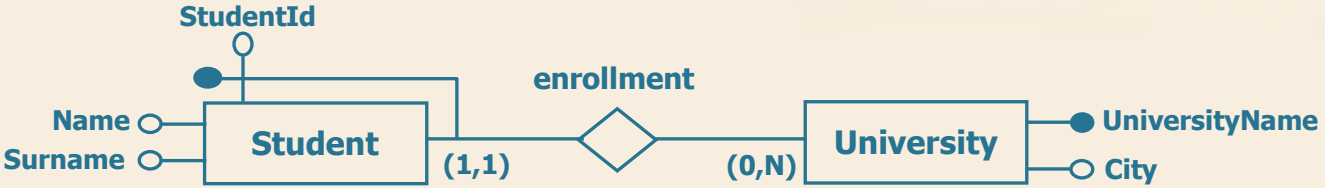
Professor(<u>ProfessorId</u>, Name, Surname)
University(<u>Name</u>, City, *ProfessorId** , *ElectionDate**)

Relational logic design

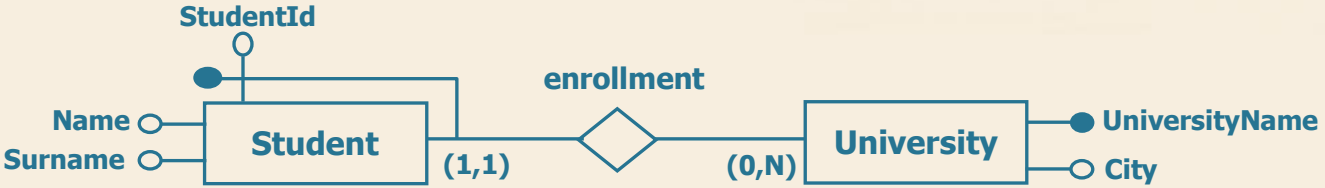
Translation in relational model: entity with external identificator

DBMG

Entity with external identificator

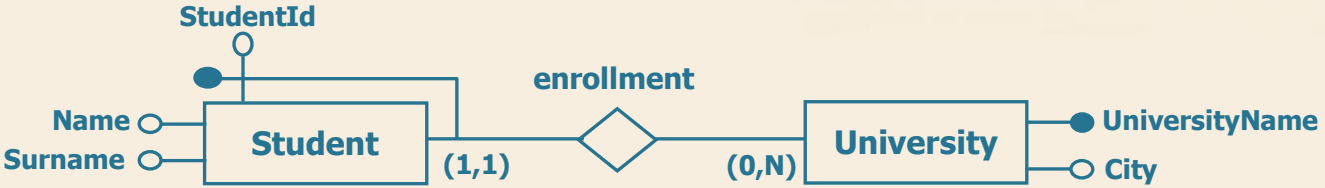


Entity with external identificator



University(<u>UniversityName</u>, City)
Student(<u>*StudentId*</u>, <u>*UniversityName*</u>, Name, Surname)

Entity with external identificator



University(<u>UniversityName</u>, City)
Student (<u>StudentId</u>, <u>UniversityName</u>, Name, Surname)

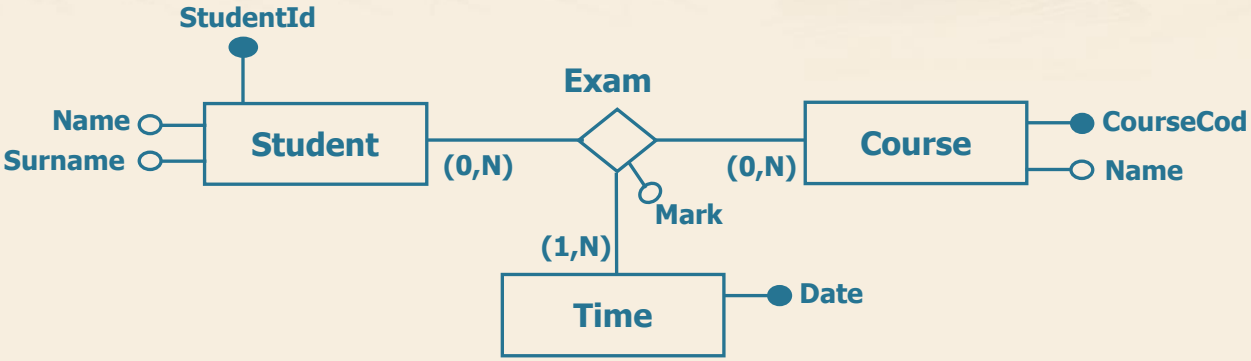
- The relationship is represented along with the identificator

Relational logic design

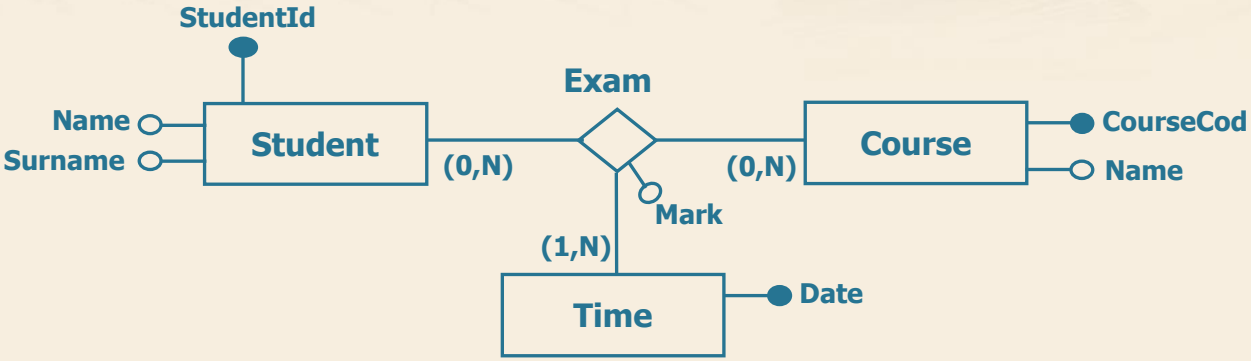
Translation in relational model: ternary relationships

DBMG

Ternary relationship

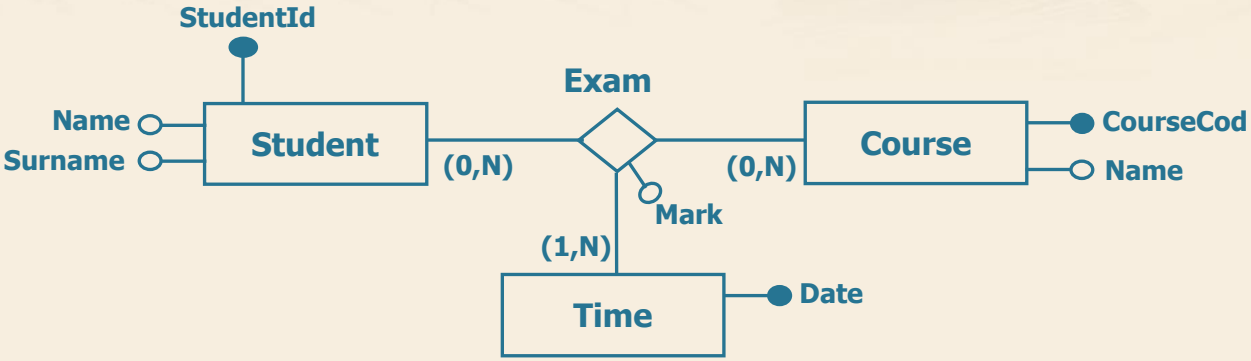


Ternary relationship: entity



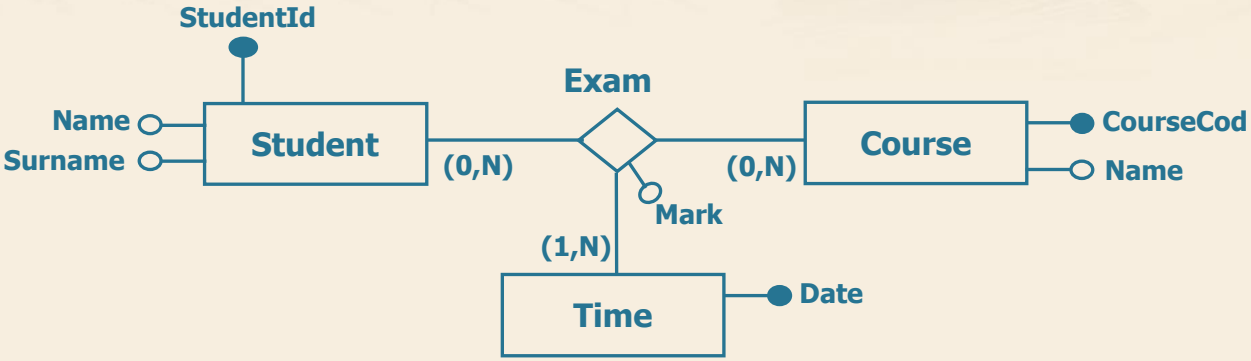
Student(<u>StudentId</u>, Name, Surname)
Course(<u>CourseCod</u>, Name)
Time(<u>Date</u>)

Ternary relationship: identificator



Student(<u>StudentId</u>, Name, Surname)
Course(<u>CourseCod</u>, Name)
Time(<u>Date</u>)
Exam(<u>StudentId, CourseCod, Date</u>)

Ternary relationship: attributes



Student(<u>StudentId</u>, Name, Surname)
Course(<u>CourseCod</u>, Name)
Time(<u>Date</u>)
Exam(<u>StudentId, CourseCod</u>, <u>Date</u>, Mark)

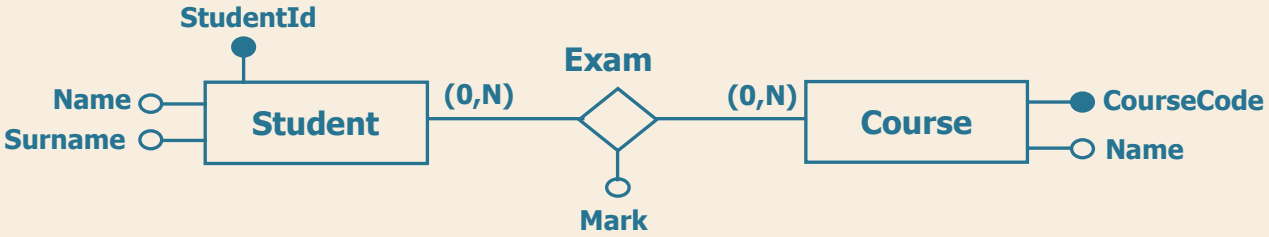
Relational logic design

Constraints of referential integrity

DBMG

Constraints of referential integrity

- Relationships represent referential constraints



Referential integrity: exam relationship

- Involved tables

Student(StudentId, Name, Surname)

Course(CourseCod, Name)

Exam(*StudentId*, CourseCod, Mark)

- Constraints of referential integrity

Exam(StudentId) REFERENCES Student(StudentId)

Referential integrity: exam relationship

- Involved tables

Student(StudentId, Name, Surname)

Course(CourseCod, Name)

Exam(StudentId, *CourseCode*, Mark)

- Constraints of referential integrity

Exam(StudentId) REFERENCES Student(StudentId)

Exam(CourseCod) REFERENCES Course(CourseCod)