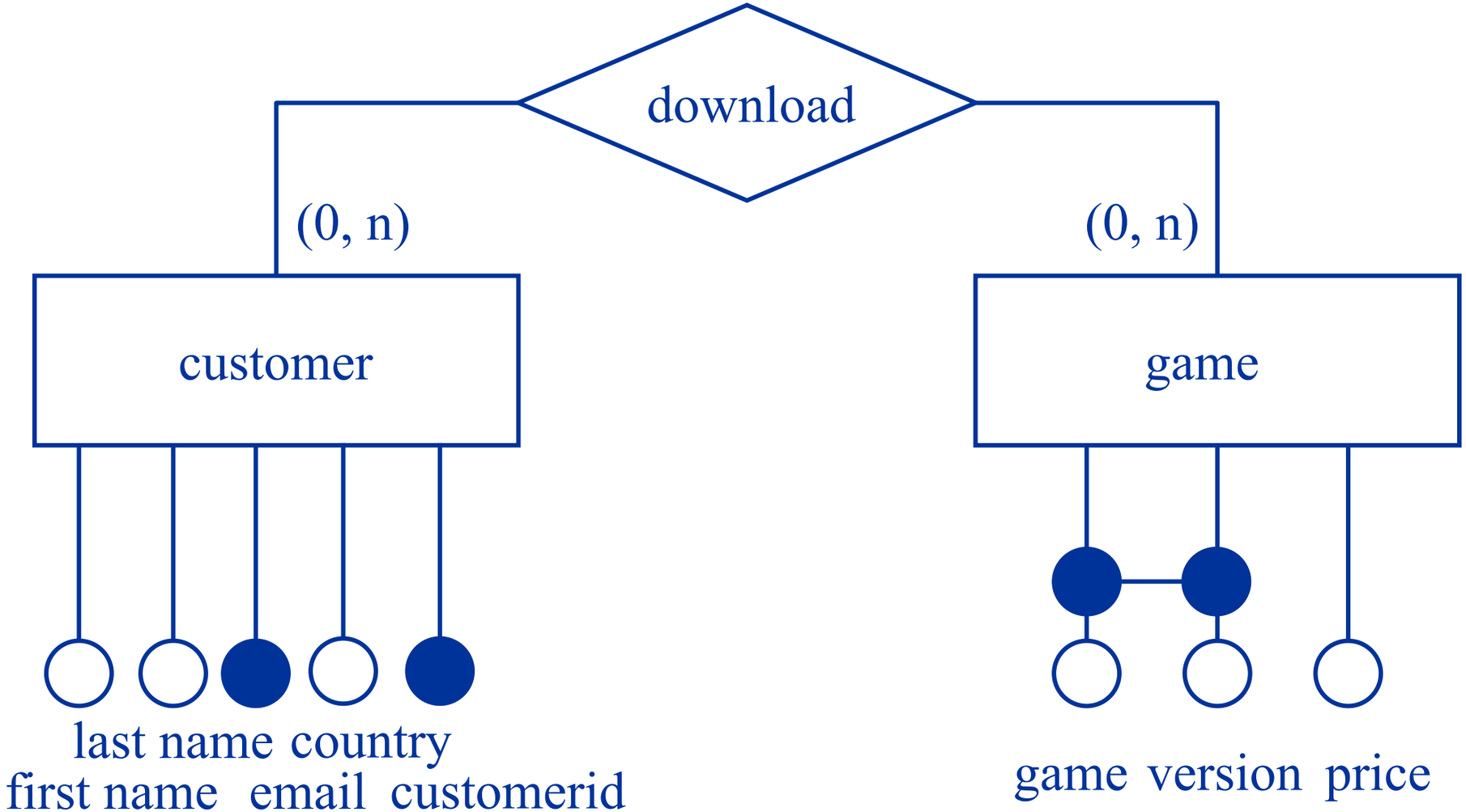
Conceptual Modelling with the Entity-relationship Model and Diagrams

Stéphane Bressan



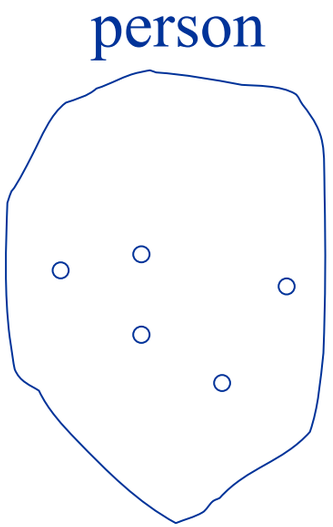


We want to develop an application for managing the data of our online app store. We would like to store several items of information about our customers such as their first name, last name, date of birth, e-mail, date and country of registration to our online sales service and the customer identifier that they have chosen. We also want to manage the list of our products, games, their name, their version and their price. The price is fixed for each version of each game. Finally, our customers buy and download games. We must remember which version of which game each customer has downloaded. It is not important to keep the download date for this application.


download
(0, n)
(0, n)
customer
game
last name country
first name email customerid
game version price

Entities and Entity Sets

Entities are identifiable “things”. The named box represents a set of entities or entity set.

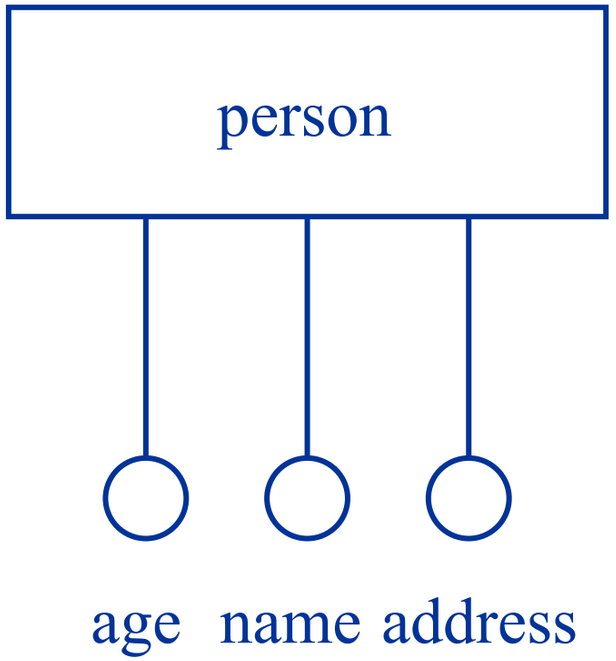


Attributes,Values and Value Sets

The E-R model is value-oriented. Values can be integer, strings, or atoms.

Attributes of Entities

Entities can have attributes. All entities in one entity set have the same attributes. However the attributes take different values for each entity.

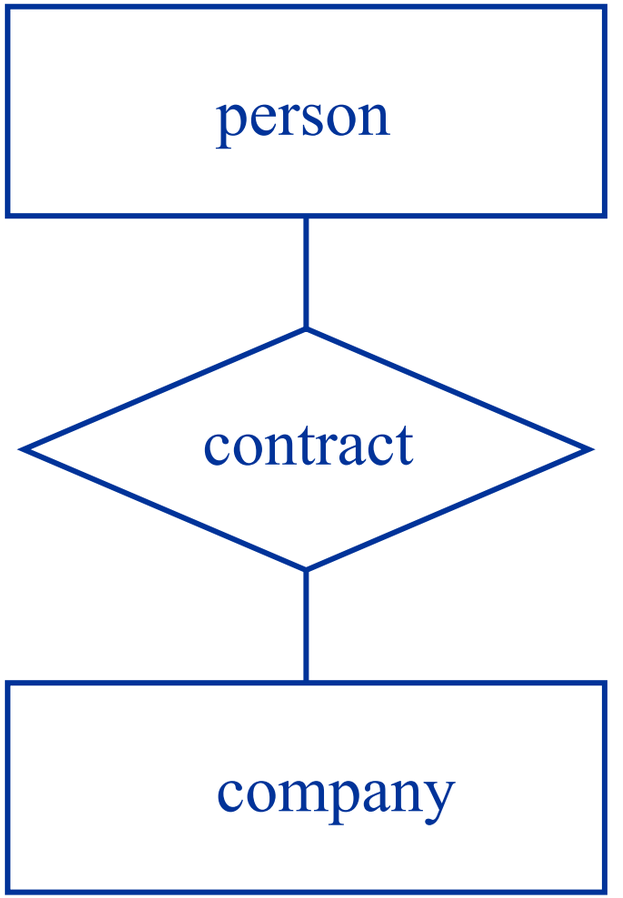


Relationships and Relationship Sets

A lozenge represents a set of relationships
or a relationship set.

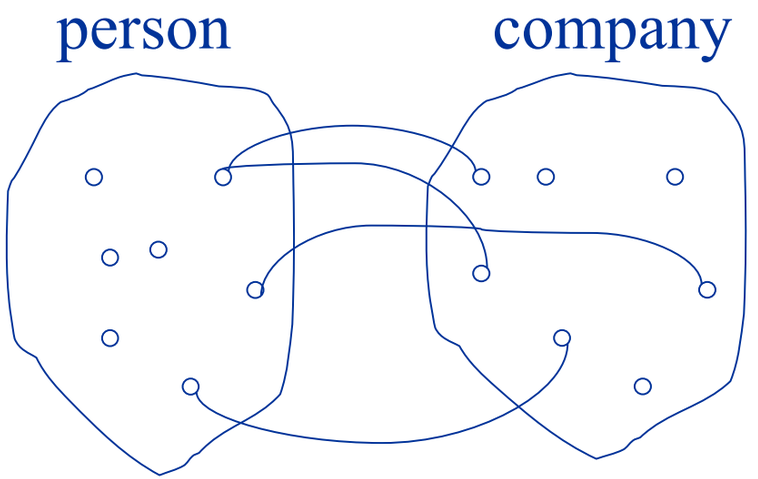
Relationships and Relationship Sets

A relationship associates two entities (can also be 0 or more). A relationship set is a set of relationships associating entities from the same entity sets.



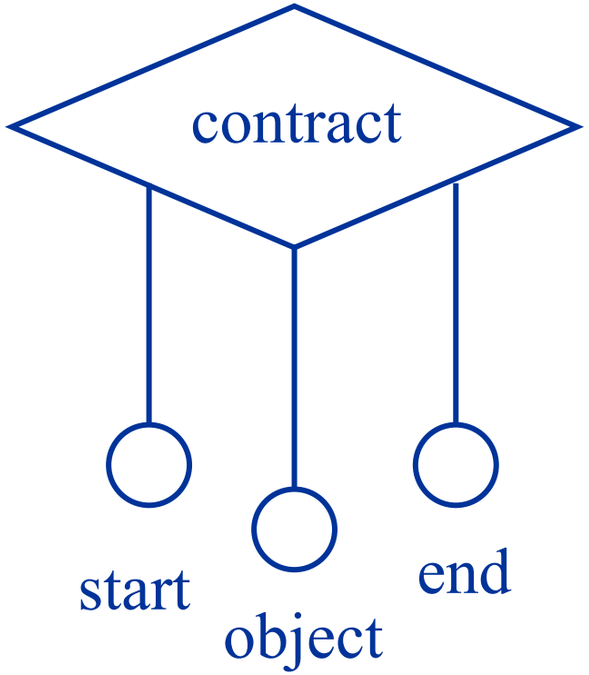
Relationships and Relationship Sets

A relationship associates 2 or more entities.
A relationship set is a set of relationships associating entities from the same entity sets.



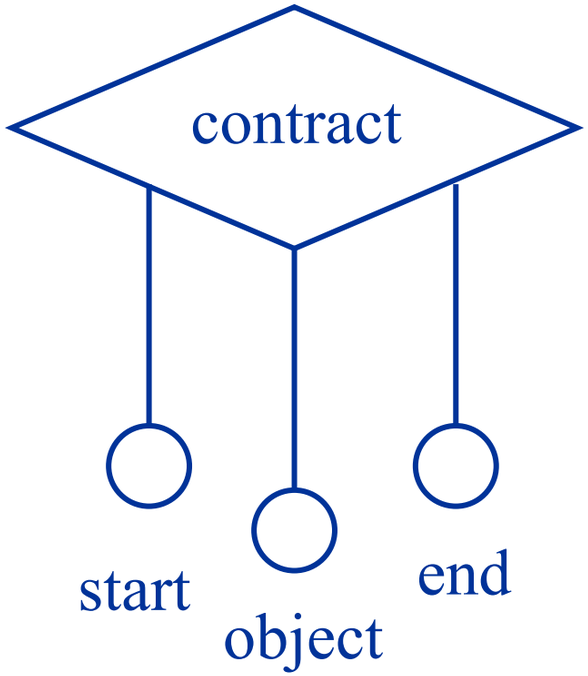
Attributes of Relationships

Relationship can have attributes. All relationships in one relationship set have the same attributes.



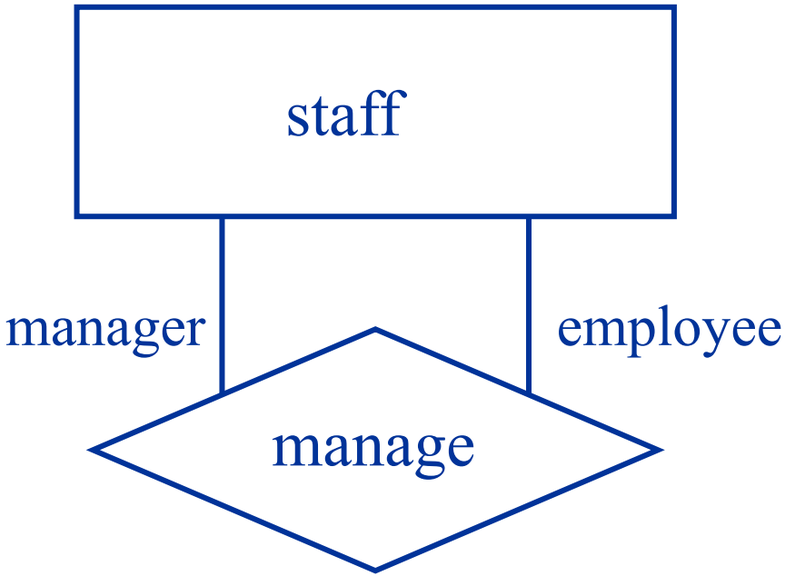
Attributes of Relationships

Relationships are distinguished not by their attributes but by their participating entities.



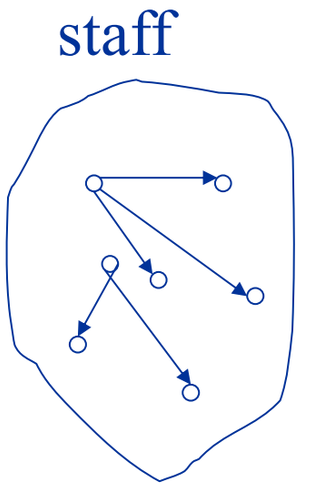
Relationships and Relationship Sets

Relationships can associate entities from the same entity set. In this case and in general, participation, or role, in the relationship can be named.



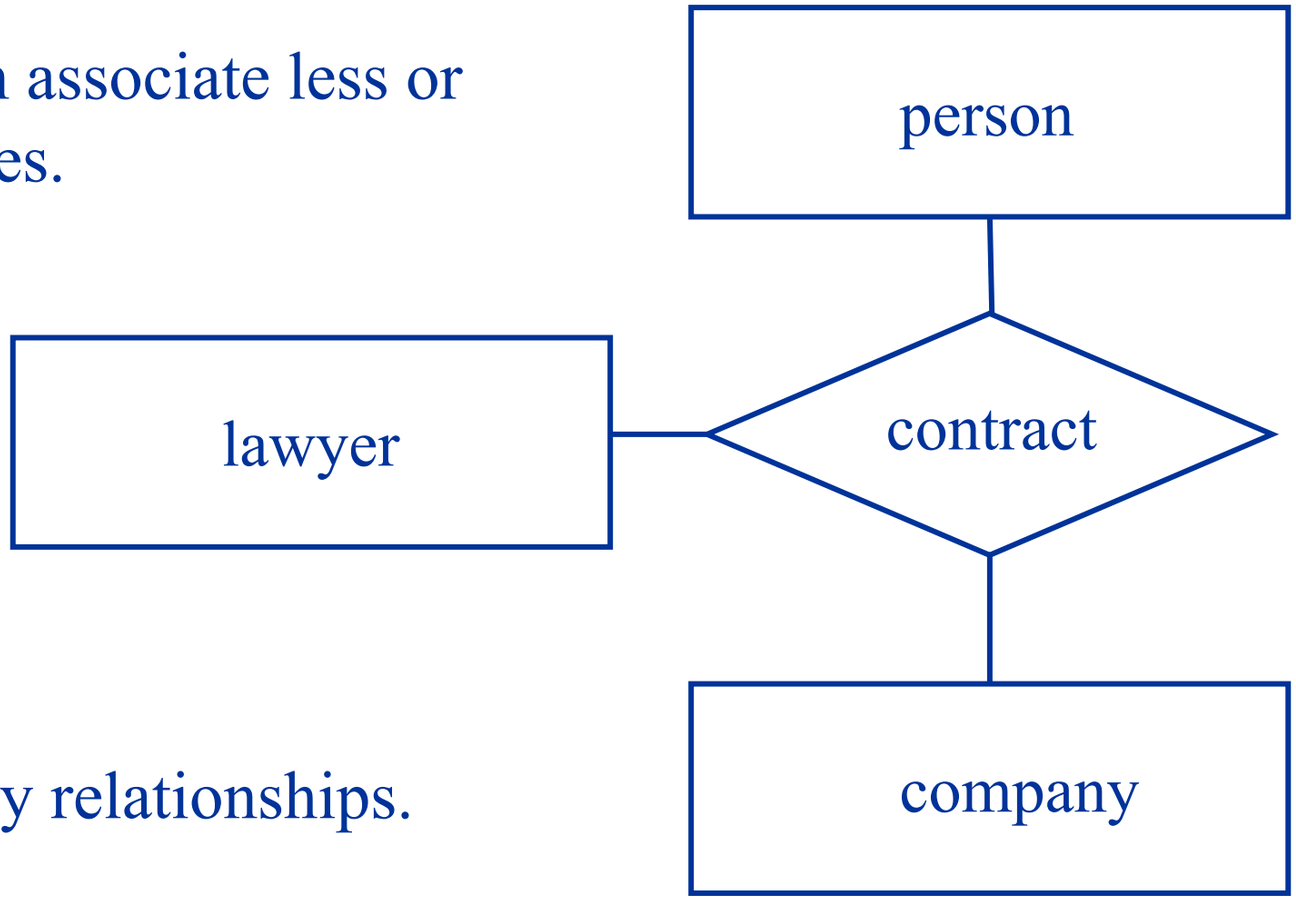
Relationships and Relationship Sets

Relationships can associate entities from the same entity set. In this case and in general, participation, or role, in the relationship can be named.



Relationships and Relationship Sets

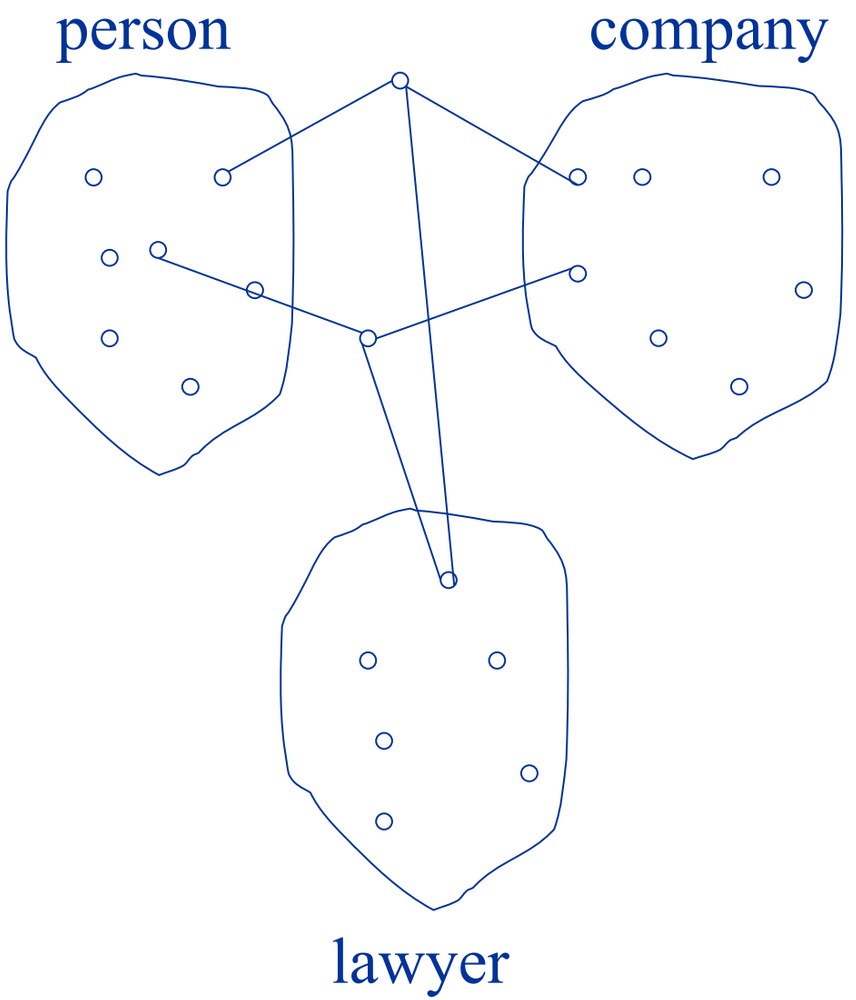
A relationship can associate less or more than 2 entities.



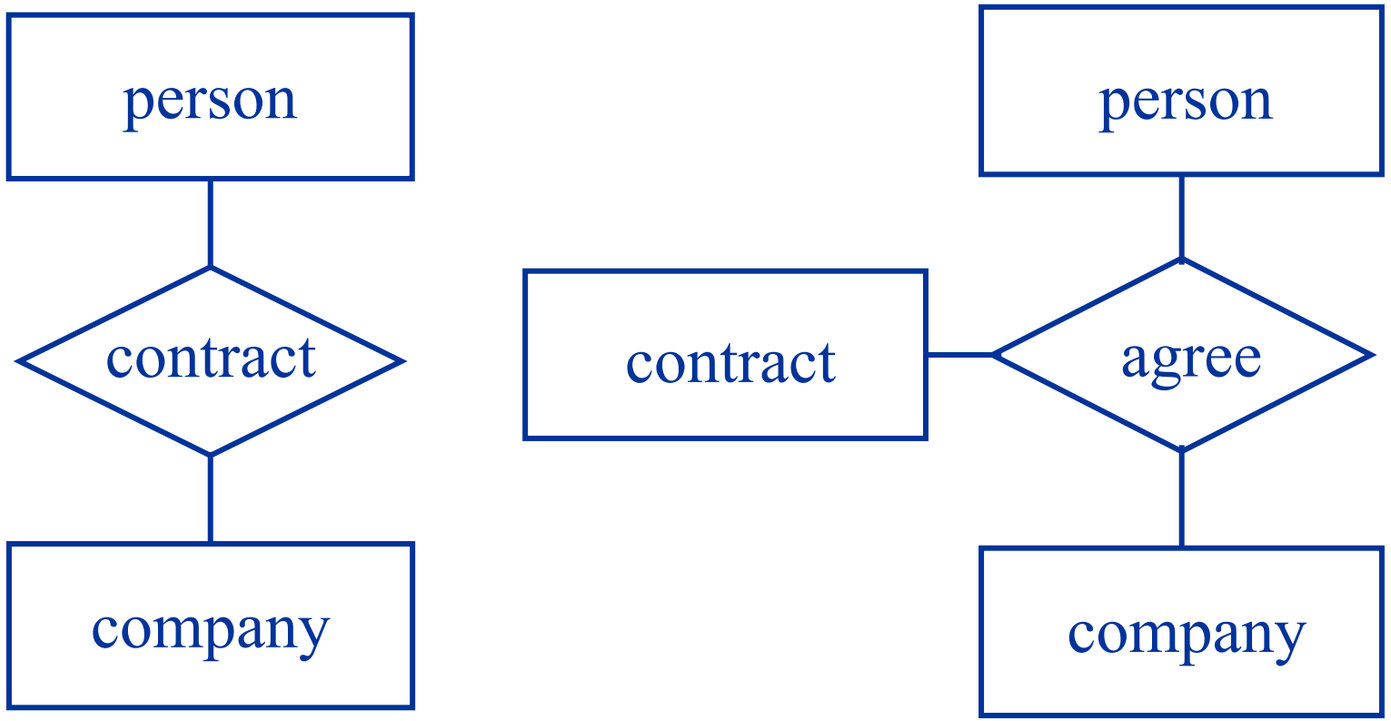
We call them n-ary relationships.

Relationships and Relationship Sets

Relationships can associate entities from the same entity set. In this case and in general, participation, or role, in the relationship can be named.



Entity or Relationship?

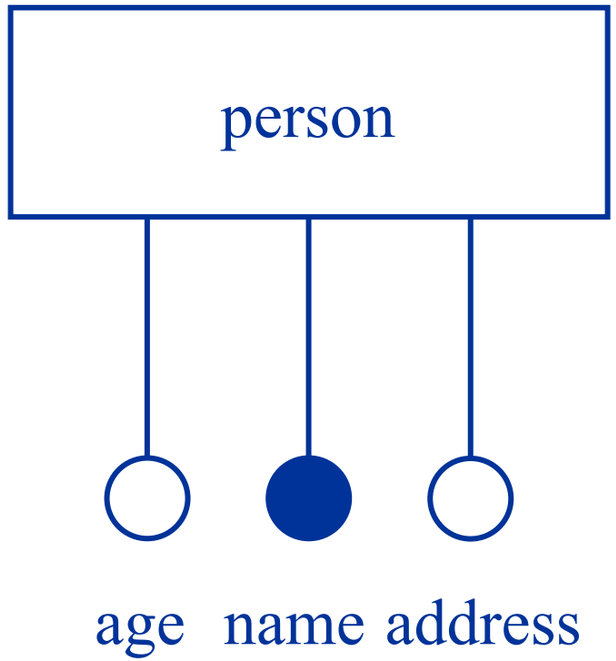



KEEP
CALM
AND
FIND
YOUR
KEY

Entities' Identity

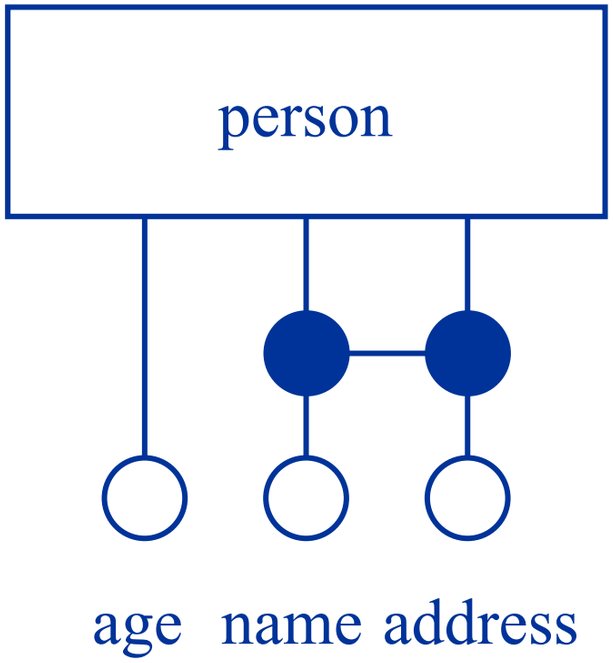
One attribute can identify the entity. This is a property of all entities in an entity set

Notice: at least all attributes identify the entity.



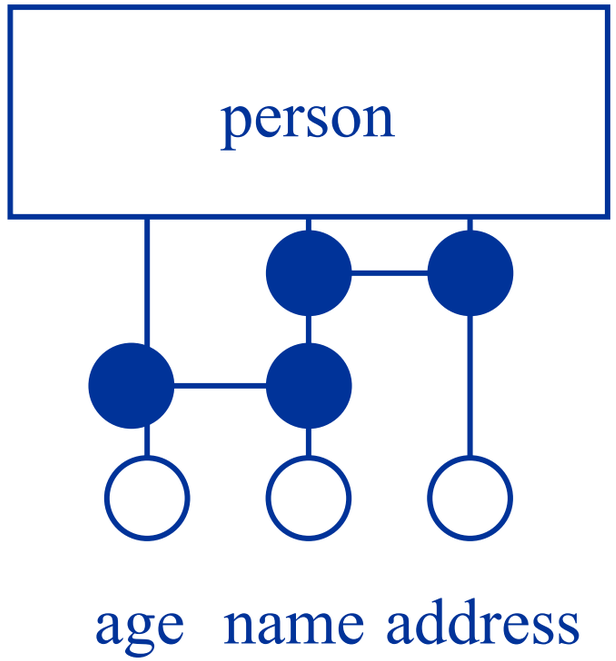
Entities’ Identity

A combination of attributes can identify the entity.



Entities’ Identity

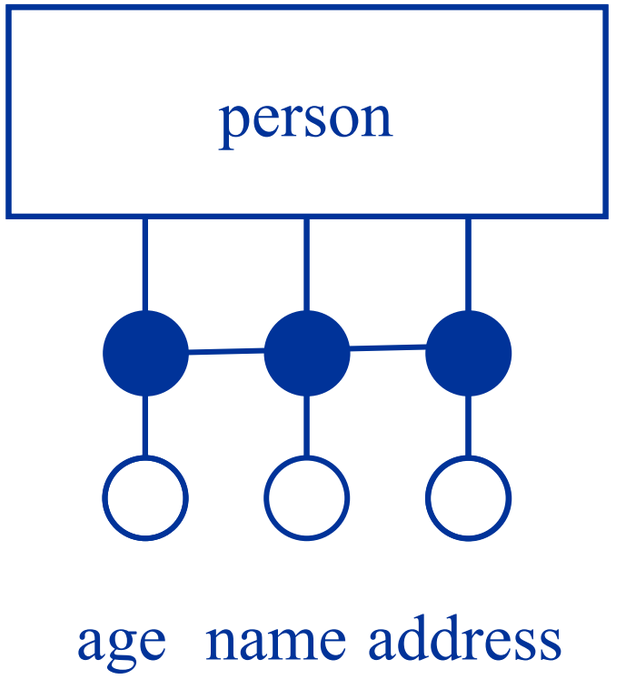
There might be several possible combination of attributes to identify an entity.



Entities' Identity

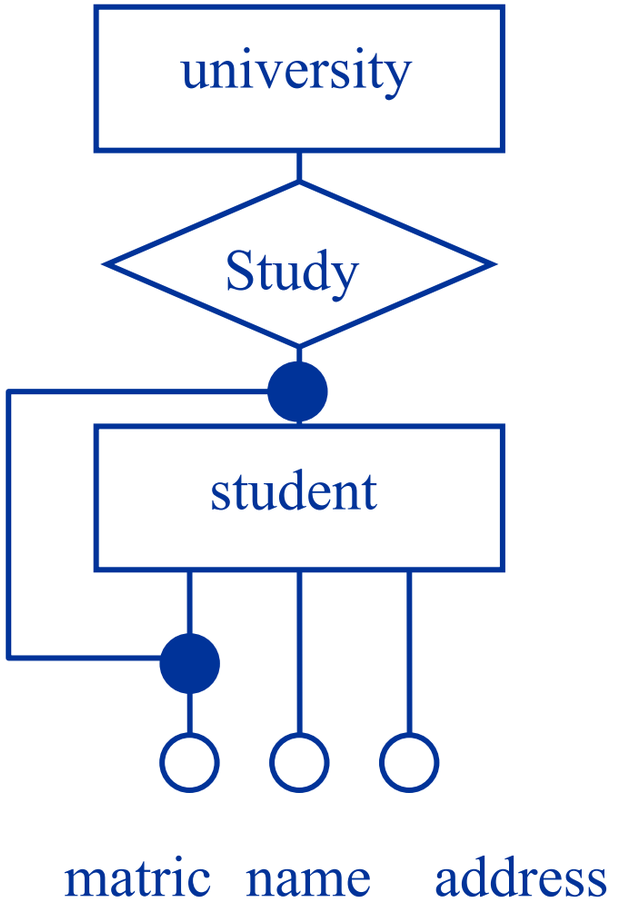
Notice: at least all attributes identify the entity

But we might prefer a minimum set of attributes.



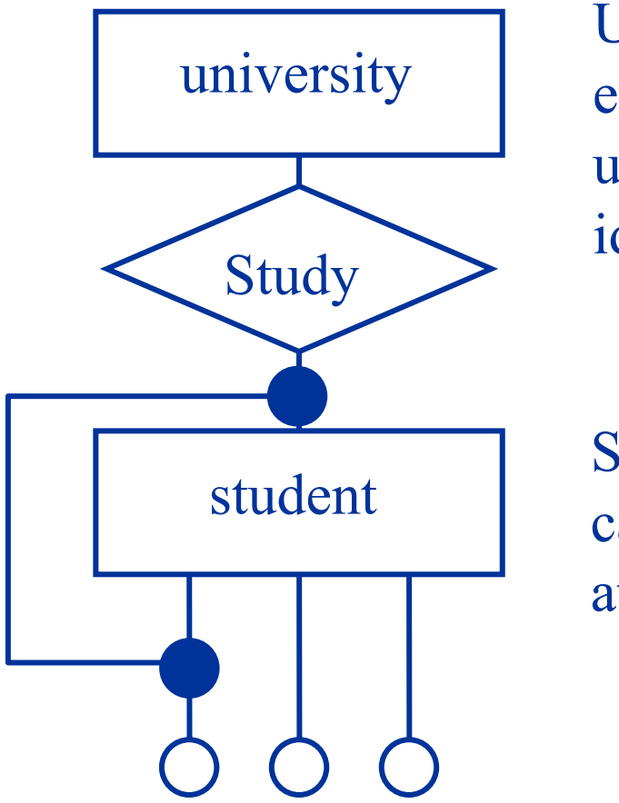
Weak Entities

Some entities can only be identified within the scope of a relationship with another entity set. Notice that the relationship must exist and be unique for each entity in the set.



Weak Entities

Matric numbers are given by the universities. The same number can be used by different universities.

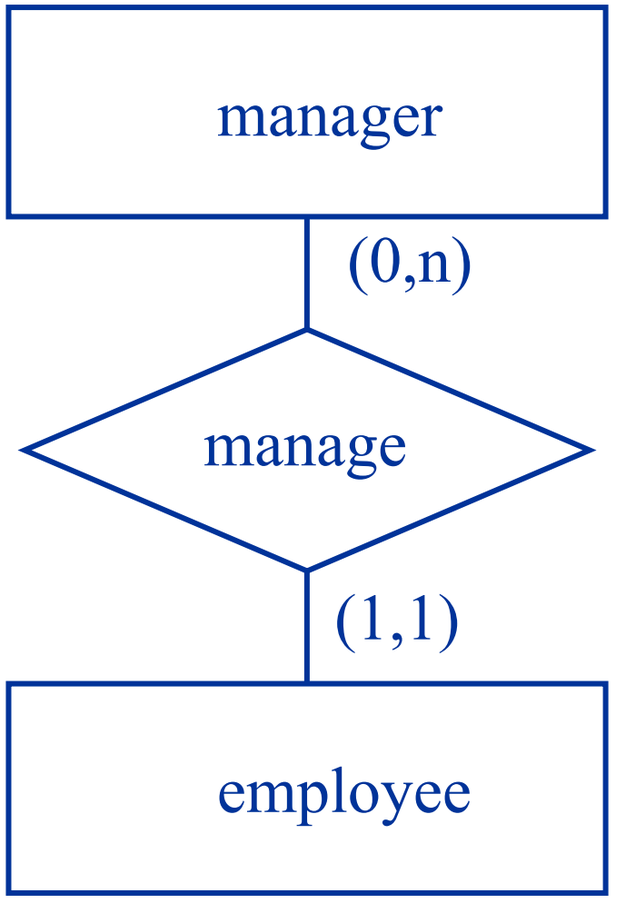


University is a dominant entity. We need to know the university in order to identify the student.

Student is a weak entity. It can be identified by its attributes alone.

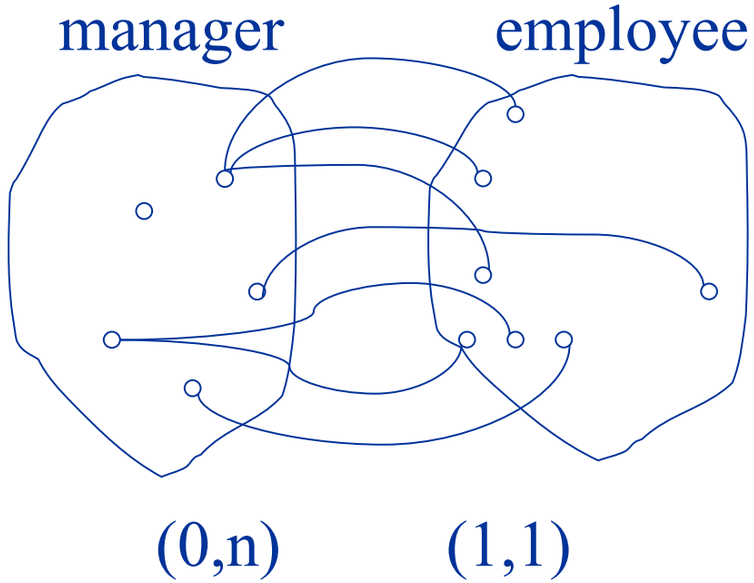
Relationships' Cardinality

The cardinality of the participation in a relationship can be constrained by a minimum and maximum value: (1,1), (0, n), (2, 5).



Relationships' Cardinality

The cardinality of the participation in a relationship can be constrained by a minimum and maximum value: (1,1), (0, n), (2, 5).



Relationships Participation and Cardinality

(1, x) mandatory participation.

(0, x) optional participation.

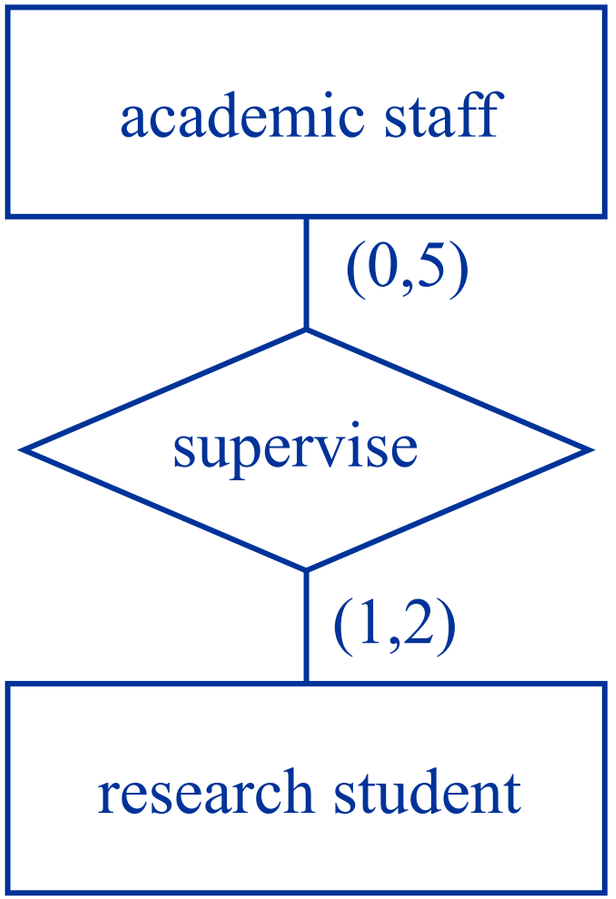
(x, 1) for all entities involved characterizes a one-to-one relationship.

(x, 1) for one entity involved and (x, N) or (x, y) $y > 1$ for the others characterizes a one-to-many relationship.

(x, N) or (x, y) $y > 1$ for all entities involved characterizes a many-to-many relationship.

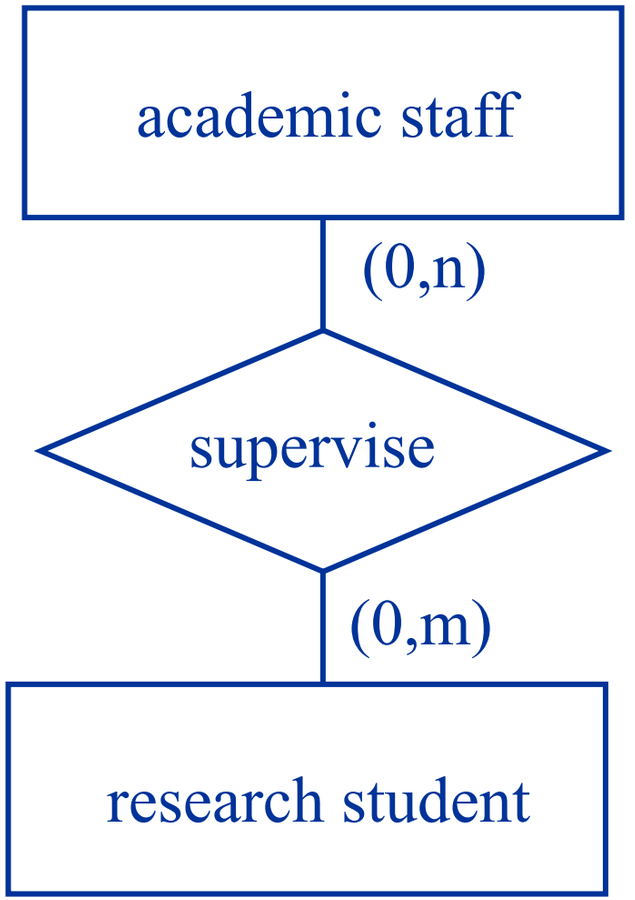
Relationships' Cardinality

Another example: academic staff can supervise up to 5 research students. Some staff do not supervise students. Research students can have one or two supervisors.



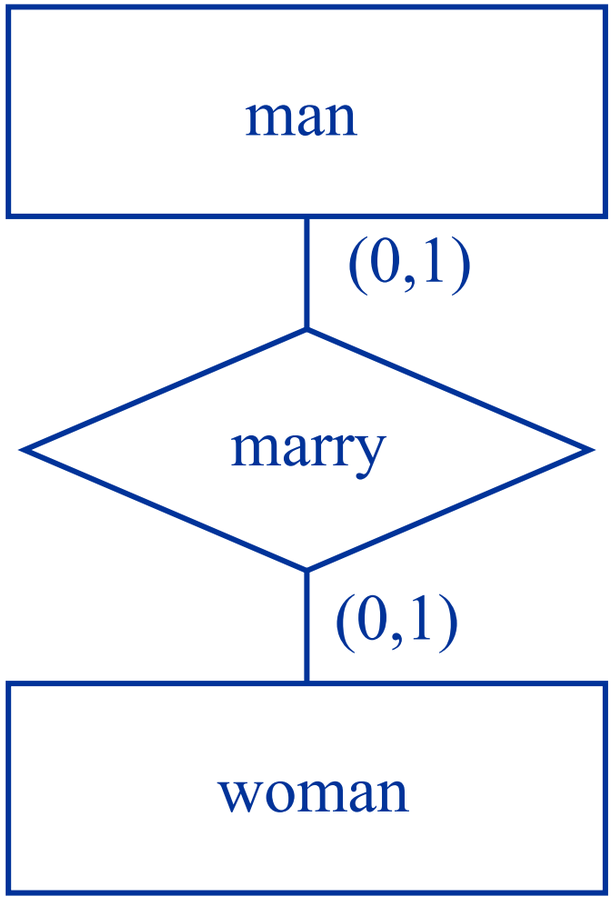
Relationships' Cardinality

By default we have many-to-many relationships.



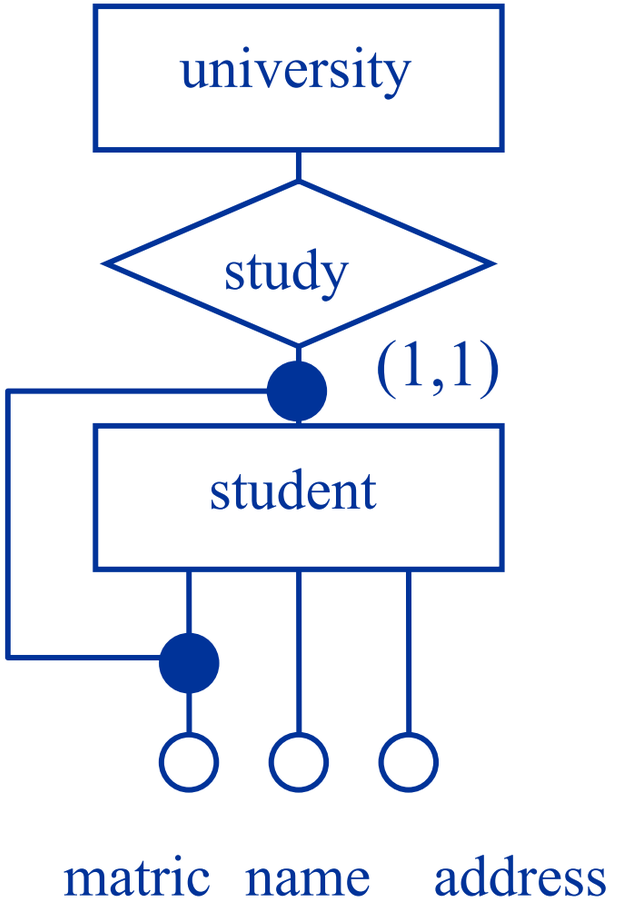
Relationships' Cardinality

Example of a one-to-one relationship.



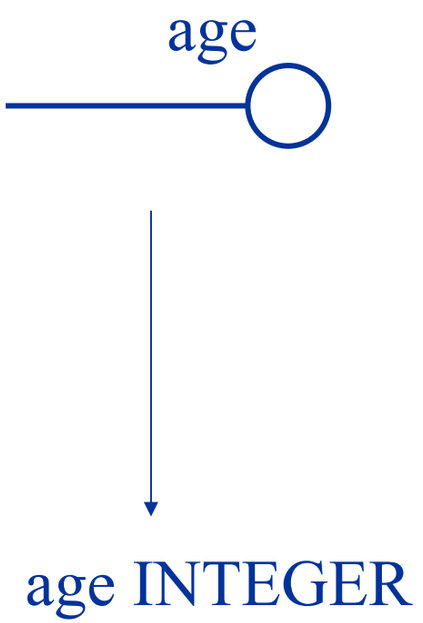
Weak Entities

Weak entities can only be defined for a participation constrained by (1,1) cardinalities. Also called mandatory one-to-many relationships.



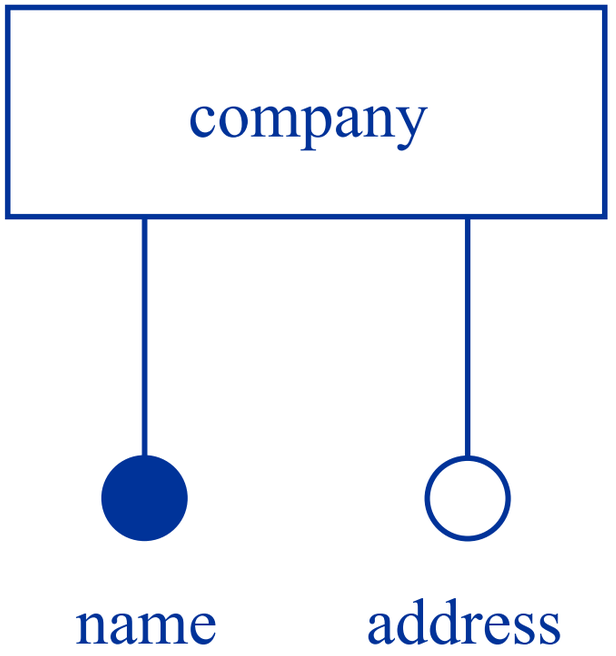
Rule 1: Value Sets

Value sets are mapped to domains. In practice this is a first step towards the physical design. E-R attributes are mapped to attributes of relations.

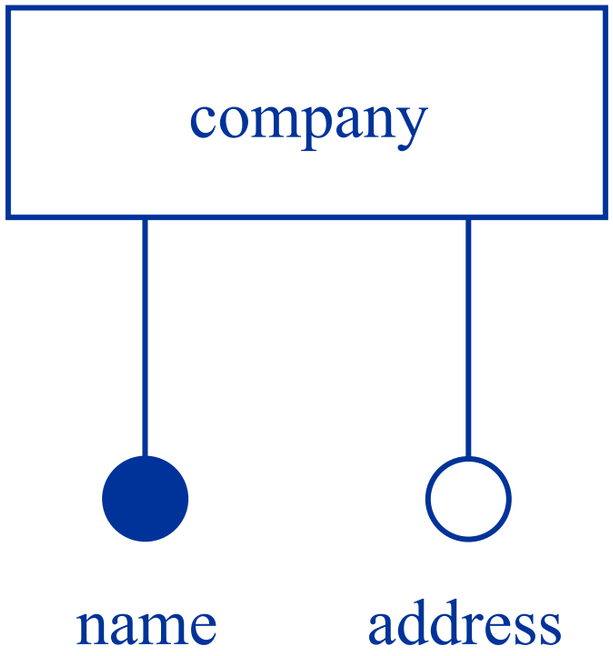


Rule 2: Entity Sets

Entity sets are mapped to relations. The entity set attributes are mapped to attributes of the relation. The keys are mapped to primary key



Rule 2: Entity Sets



Rule 2: Entity Sets

```
CREATE TABLE company(
name VARCHAR(64) PRIMARY KEY,
address VARCHAR(128) NOT NULL);
```

Rule 2: Entity Sets

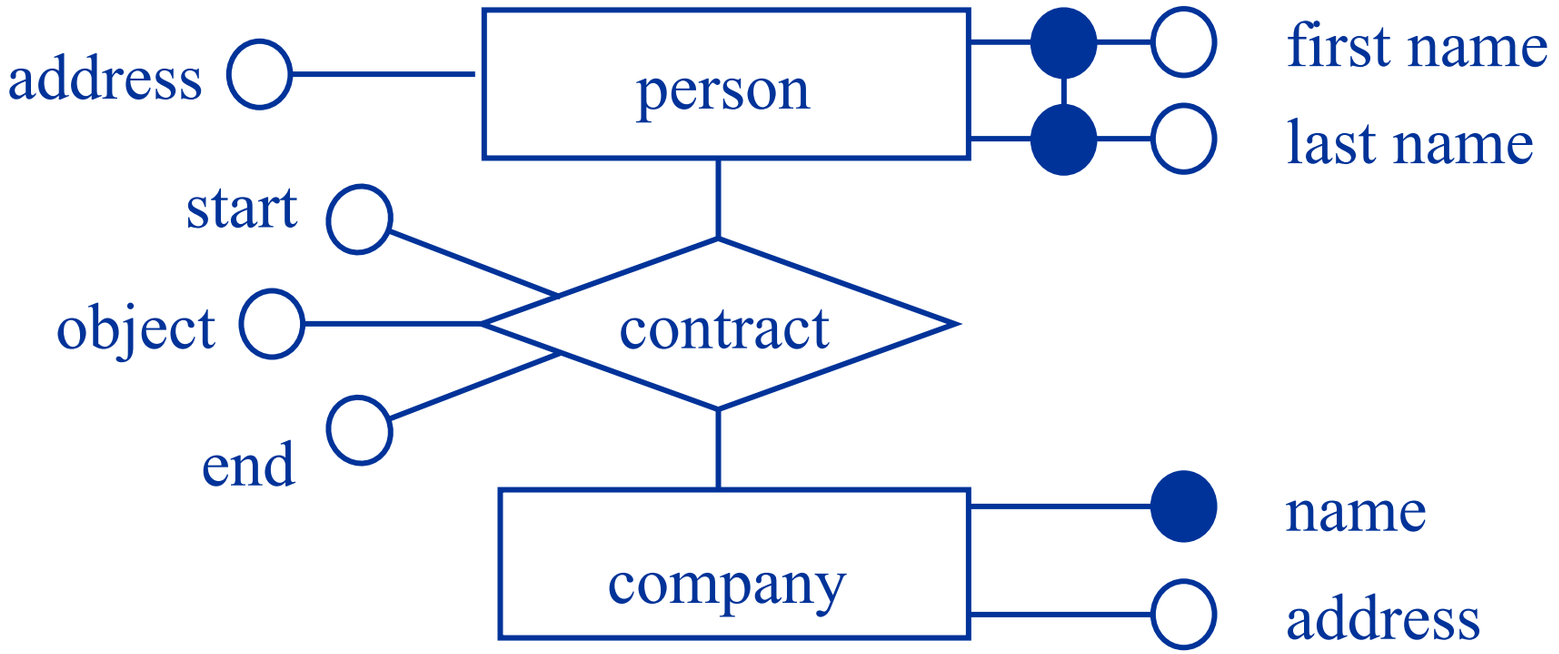
Rule 2: Entity Sets

```
CREATE TABLE person (
first name VARCHAR(32),
last_name VARCHAR(32),
address VARCHAR(128) NOT NULL,
PRIMARY KEY (first_name, last_name));
```

Rule 3: Relationship Sets

Relationship sets are mapped to relations. The attributes of the relation consist of the attributes of the relationship set. As well as of the keys of the participating entities.

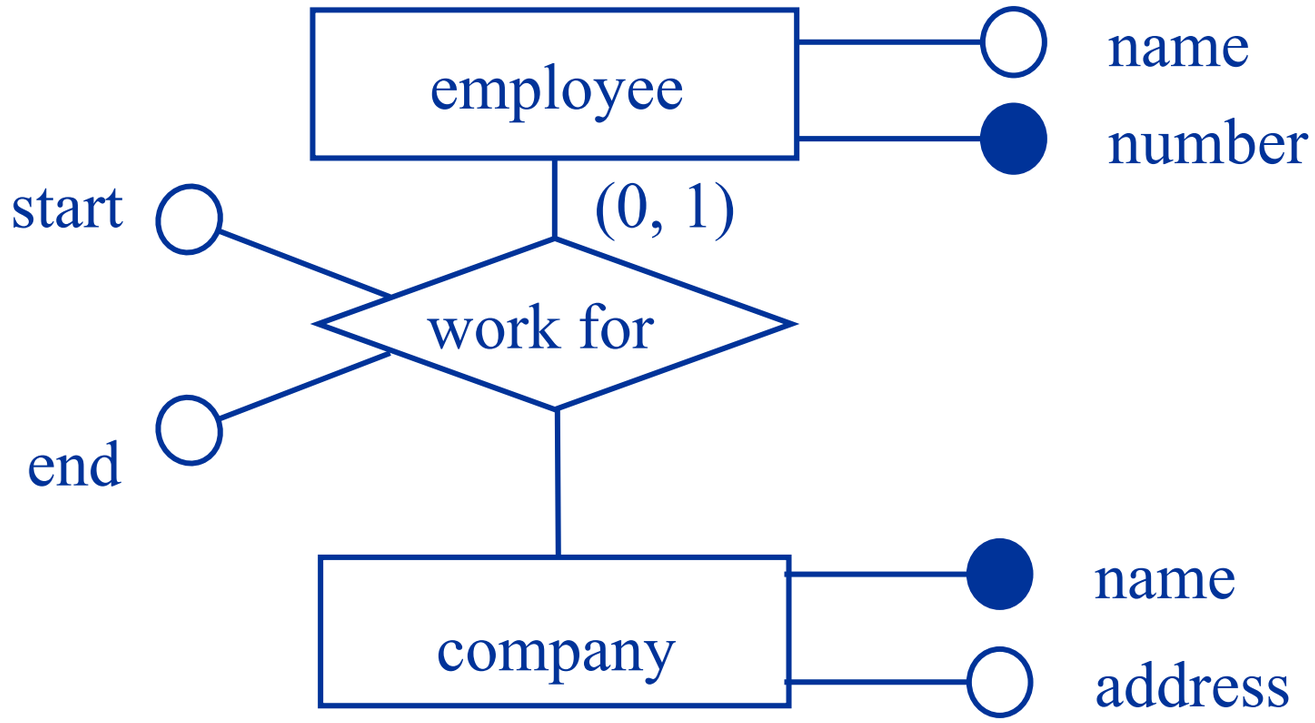
Rule 3: Relationship Sets



Rule 3: Relationship Sets

```
CREATE TABLE contract(
start DATE NOT NULL,
end DATE NOT NULL,
object VARCHAR(128) NOT NULL,
pfirst_name VARCHAR(32),
plast_name VARCHAR(32),
cname VARCHAR(64),
PRIMARY KEY (pfirst_name, plast_name, cname),
FOREIGN KEY (pfirst_name , plast_name)
REFERENCES person(first_name, last_name),
FOREIGN KEY (cname) REFERENCES company(name));
```

Exception 1: One-to-many Relationships

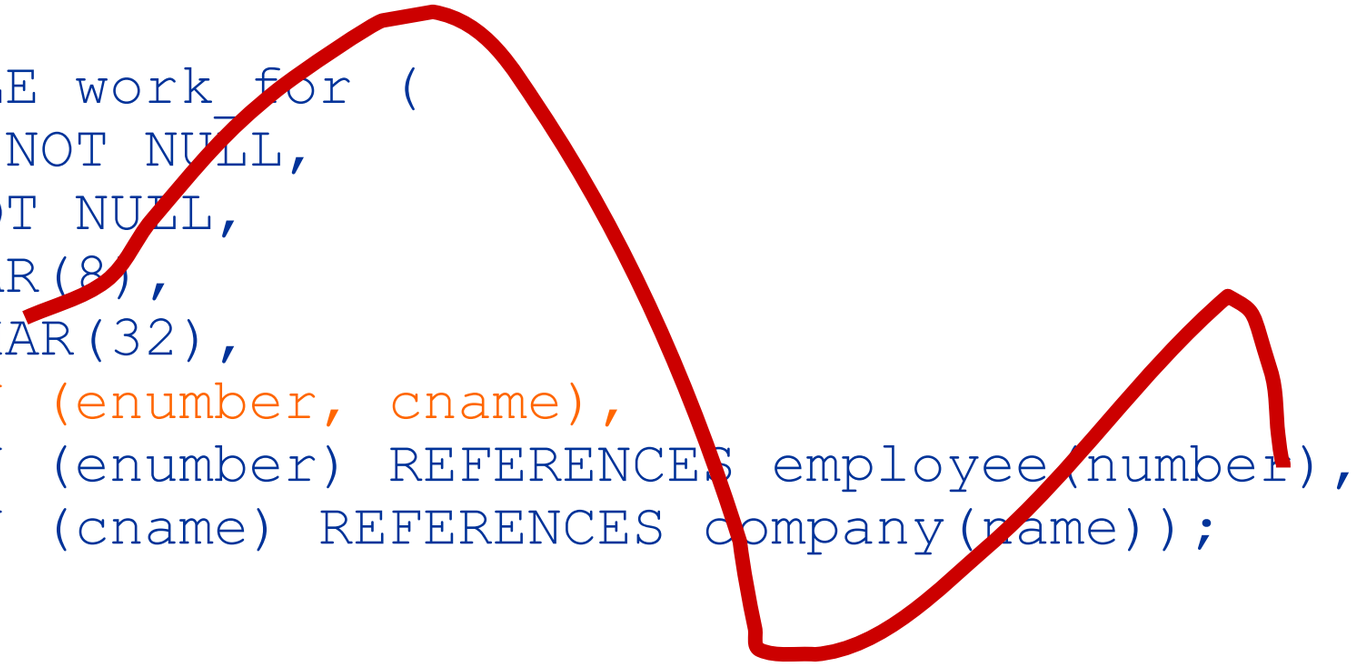


A one-to-many relationship indicates a key constraint

Exception 1: One-to-many Relationships

The primary key of the relationship table is inadequate...

```
CREATE TABLE work_for (
start DATE NOT NULL,
end DATE NOT NULL,
enumber CHAR(8),
cname VARCHAR(32),
PRIMARY KEY (enumber, cname),
FOREIGN KEY (enumber) REFERENCES employee(number),
FOREIGN KEY (cname) REFERENCES company(name));
```

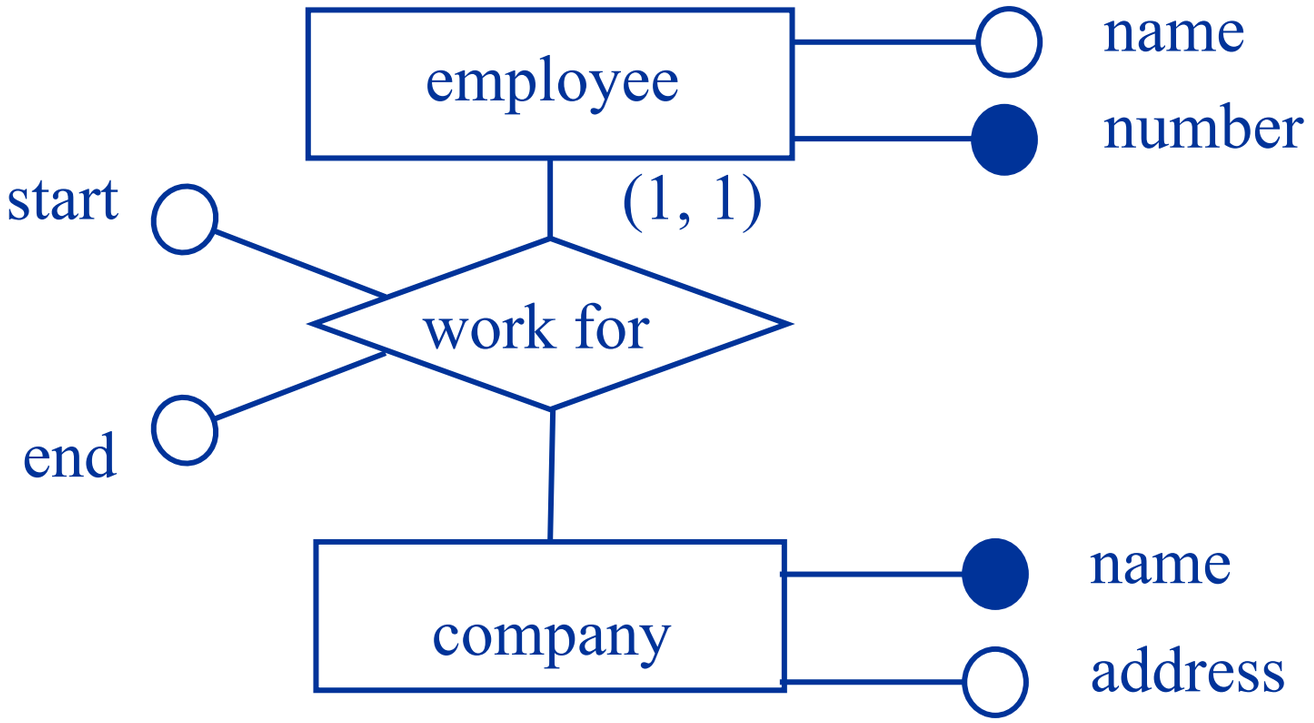


Exception 1: One-to-many Relationships

We change the primary key of the relationship table or add UNIQUE constraints.

```
CREATE TABLE work_for(
start DATE NOT NULL,
end DATE NOT NULL,
enumber CHAR(8) PRIMARY KEY,
cname VARCHAR(32) NOT NULL,
FOREIGN KEY (enumber) REFERENCES employee(number),
FOREIGN KEY (cname) REFERENCES company(name));
```

Exception 2: (1, 1) Participation Constraints

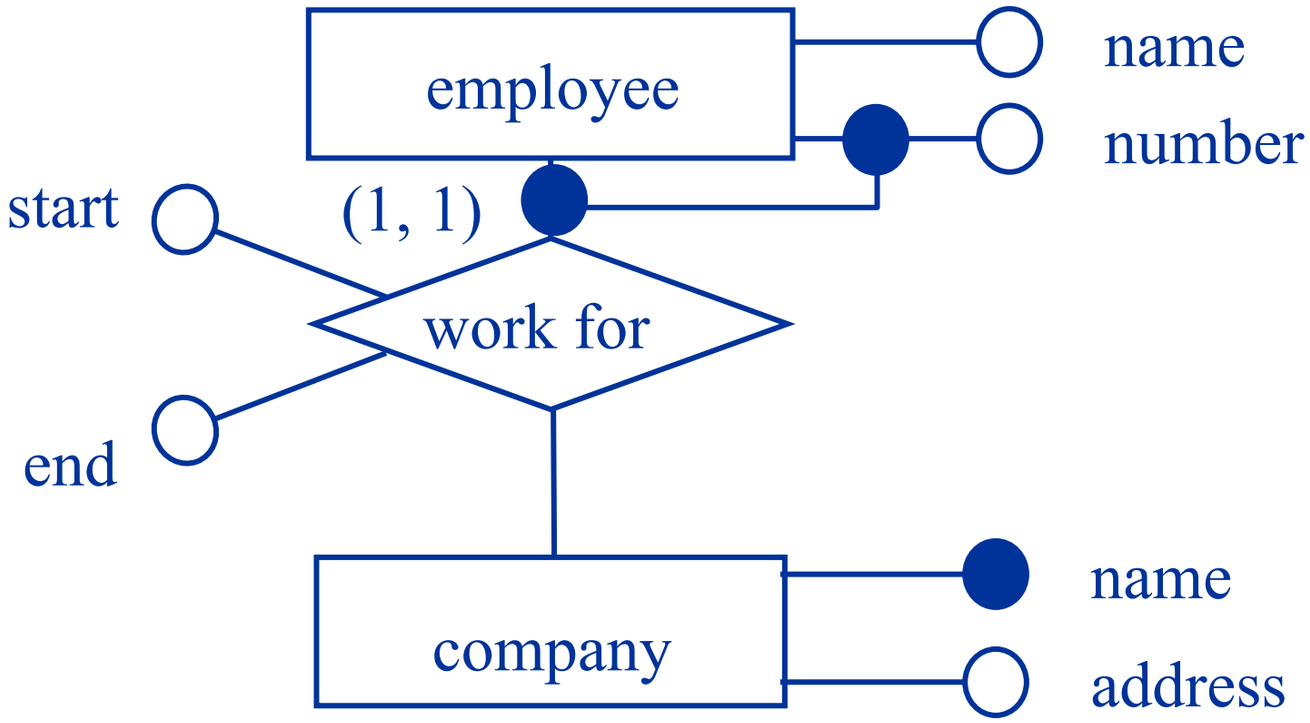


Exception 2: (1, 1) Participation Constraints

We merge the table employee and the table work_for and use the primary key of the employee table.

```
CREATE TABLE employee_work_for (
start DATE NOT NULL,
end DATE NOT NULL,
enumber CHAR(8) PRIMARY KEY,
ename CHAR(32) NOT NULL,
cname VARCHAR(32) NOT NULL,
FOREIGN KEY (cname) REFERENCES company(name));
```

Exception 3: Weak Entity



Exception 3: Weak Entity

The primary key of the emplyee table is not enumber because it is a weak entity.

```
CREATE TABLE employee_work_for(
start DATE NOT NULL,
end DATE NOT NULL,
enumber CHAR(8) PRIMARY KEY,
ename CHAR(32) NOT NULL,
cname VARCHAR(32) NOT NULL,
FOREIGN KEY (cname) REFERENCES company(name));
```

Exception 3: Weak Entity

We merge the table employee and the table work_for and use the primary key of the weak entity.

```
CREATE TABLE employee_work_for(
start DATE NOT NULL,
end DATE NOT NULL,
enumber CHAR(8),
ename CHAR(32) NOT NULL,
cname VARCHAR(32),
PRIMARY KEY (enumber, cname),
FOREIGN KEY (cname) REFERENCES company(name));
```

Credits

The content of this lecture is based on chapter 1 of the book "Introduction to database Systems" By S. Bressan and B. Catania, McGraw Hill publisher

Images and clips used in this presentation are licensed from Microsoft Office Online Clipart and Media

For questions about the content of this course and about copyrights, please contact Stéphane Bressan

steph@nus.edu.sg