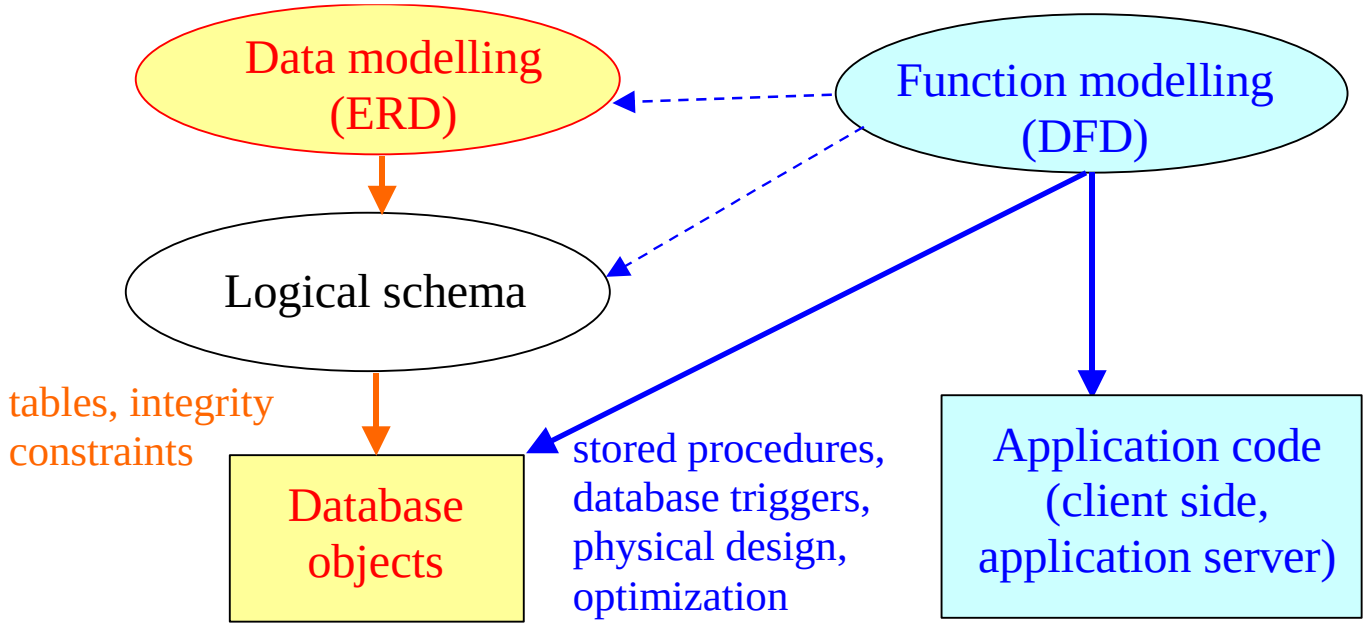
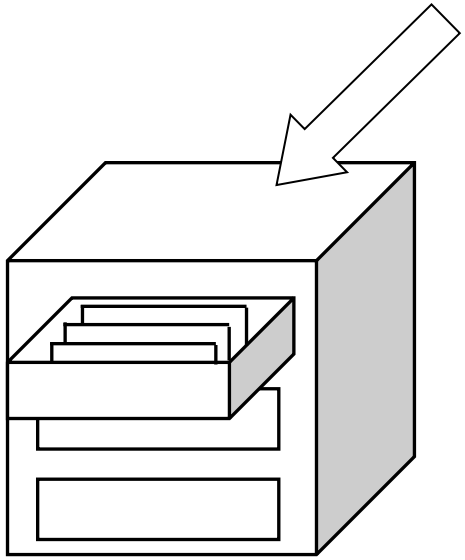
4 Mapping ER diagrams to tables

4.1 Mapping ER diagrams to tables of a relational database



Ex) See Introduction



MAR

ECO

Name: Economy section

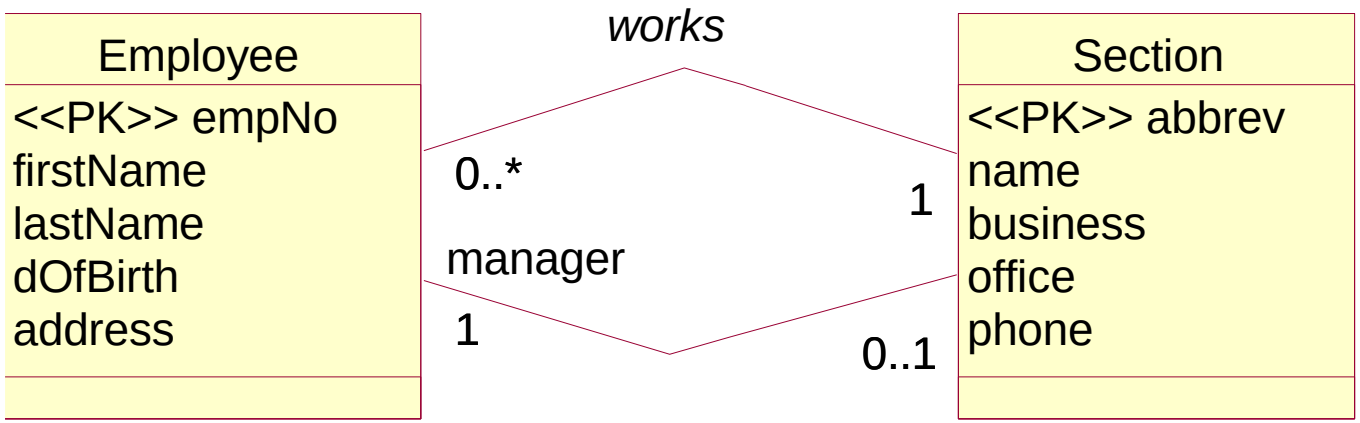
Business: Economy affairs

Section manager: Jana Nováková

Office: A-216

Phone: 253

empNo	firstName	lastName	dOfBirth	address
30	Jana	Nováková	12.3.1950	Brno
58	Karel	Veselý	23.6.1958	Brno
80	Eva	Pokorná	3.9.1965	Blansko


Employee
<<PK>> empNo
firstName
lastName
dOfBirth
address
works
0..*
manager
1
1
0..1
Section
<<PK>> abbrev
name
business
office
phone

SECTION

abbrev	name	business	manager	office	phone
ECO	Economy section	Economy affairs, ...	30	A-216	253
MAR	Marketing section	Marketing affairs, …	10	A-320	301
…	…	…			

values from another column (a foreign key)

EMPLOYEE

empNo	firstName	lastName	dOfBirth	address	section
...	...	...	...	...	...
...	...	...	...	...	...
10	Josef	Floryán	18.3.1945	Brno	MAR
...	...	...	...	...	...
30	Jana	Nováková	12.3.1950	Brno	ECO
...	...	...	...	...	...
58	Karel	Veselý	23.6.1958	Brno	ECO
...	...	...	...	...	...
80	Eva	Pokorná	3.9.1965	Blansko	ECO
...		...	...	...	...

unique values (a primary key)

- **Basic problems of bad design**
  - repetition of information (redundancy)
  - inability to represent certain information
  - complex integrity constraints checking

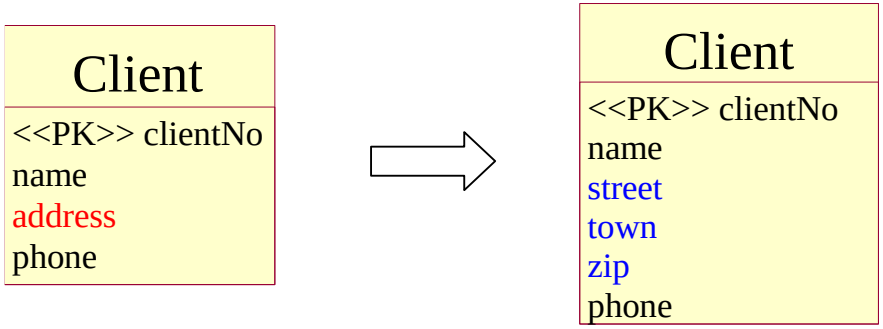
  } formally BCNF or 3NF

Ex)Employee (it is not well designed)

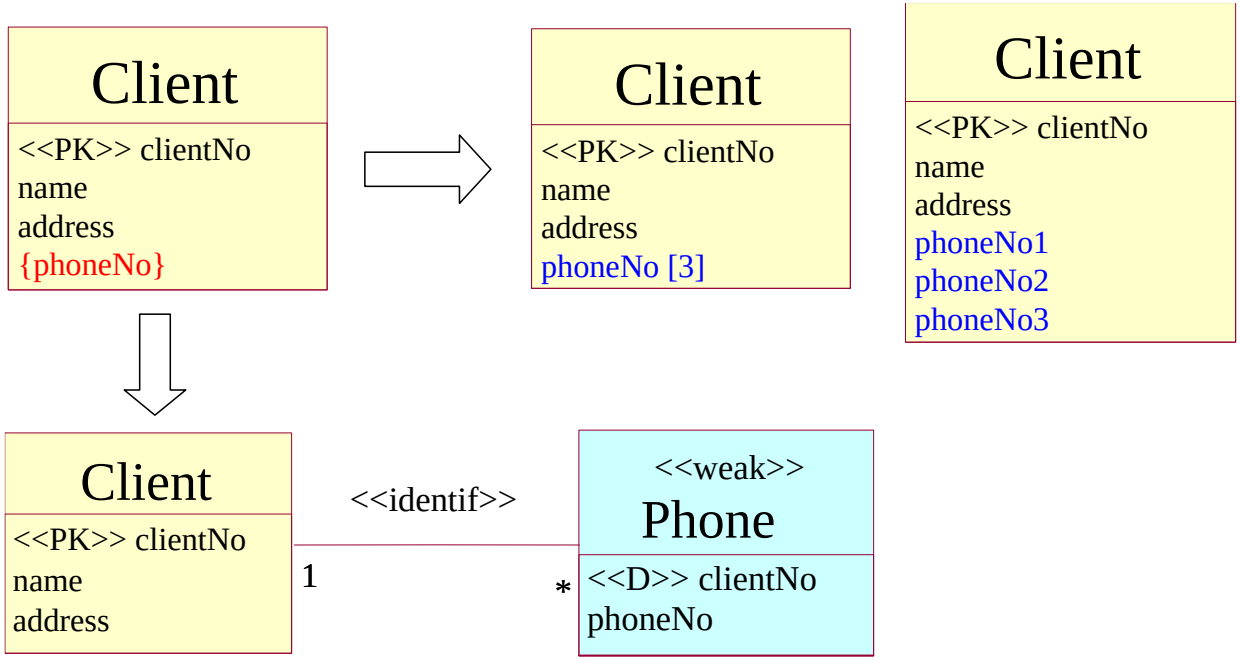
empNo	firstName	lastName	section	name
10	Josef	Floryán	MAR	Marketing section
30	Jana	Nováková	EKO	Economy section
58	Karel	Veselý	EKO	Economy section
80	Eva	Pokorná	EKO	Economy section

- **Design objectives**
  - to avoid problems of bad design
  - to meet other criteria, concerning performance mainly (do not create unnecessary tables!)

- **Transformation rules**
  - **·Remove composite and multiple-value attributes (convert to 1NF)**
    - **Composite attribute → several simple ones (components)**

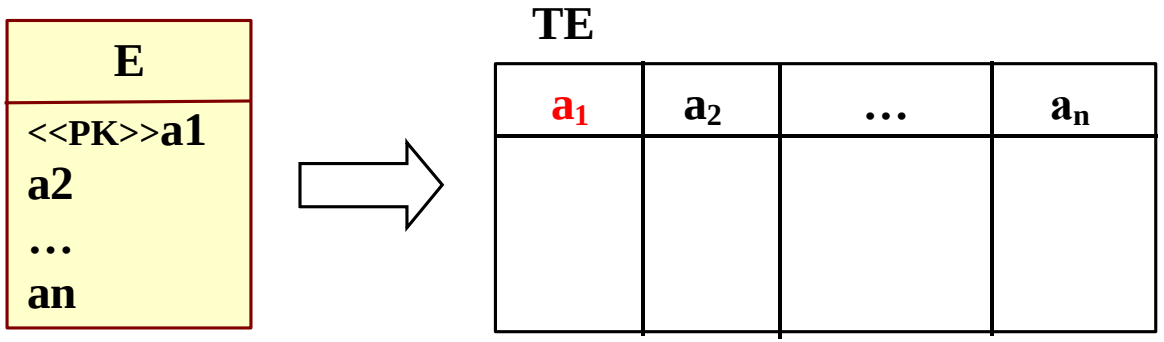


- **Multiple-valued attribute → another entity set or by a fixing the number of values**

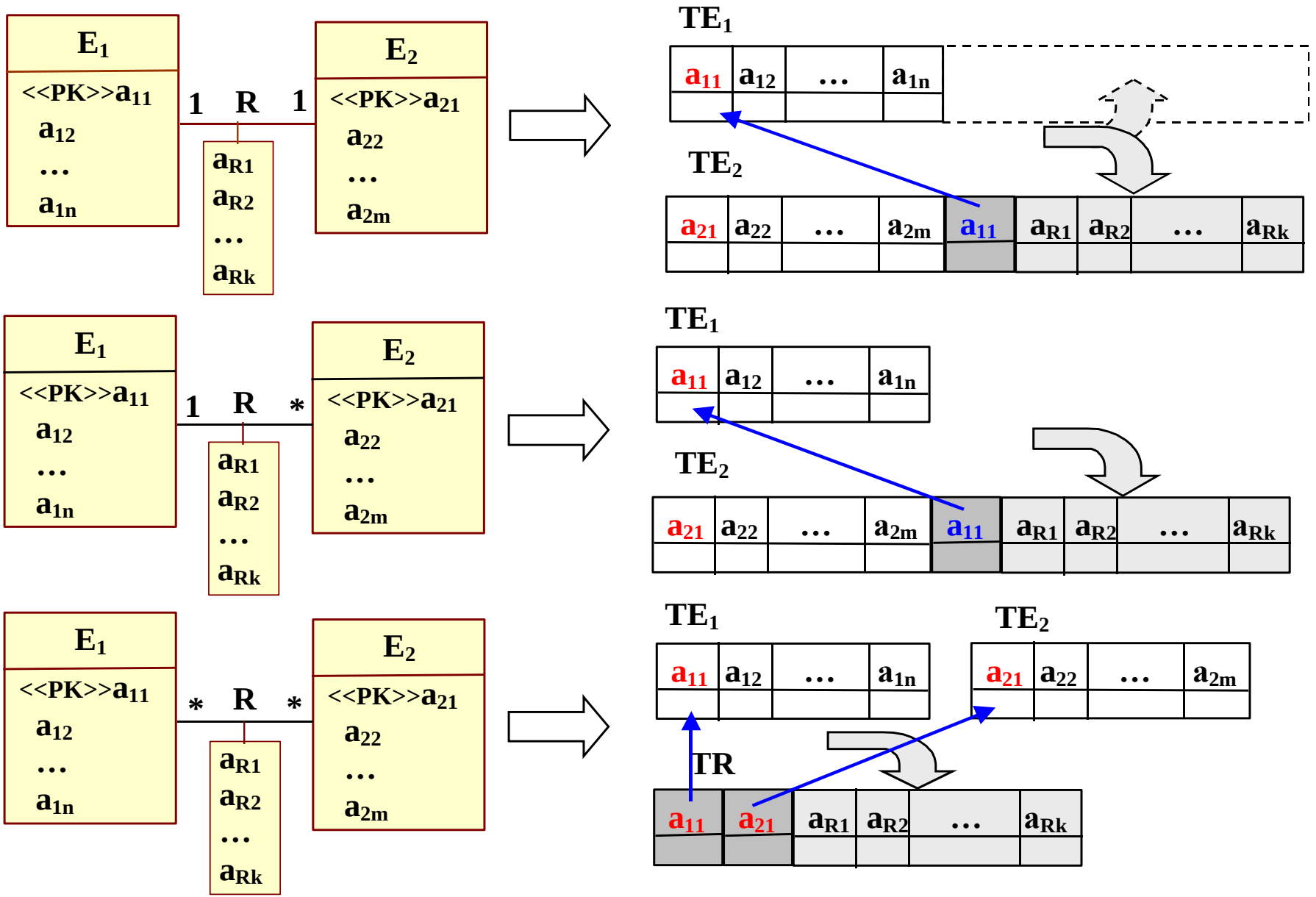


- it is also possible to transform first and only then normalize

➢ Strong entity sets

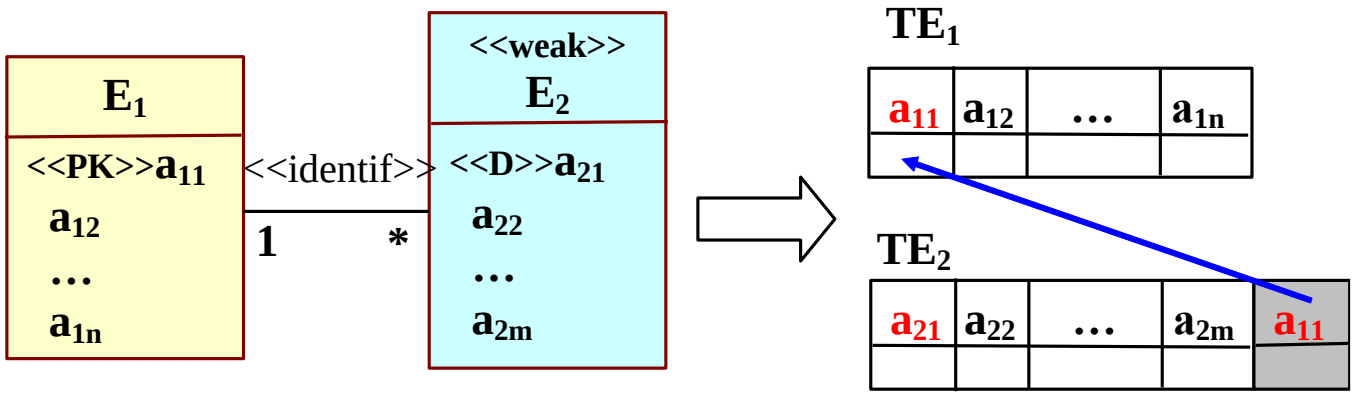


➢ Relationship sets

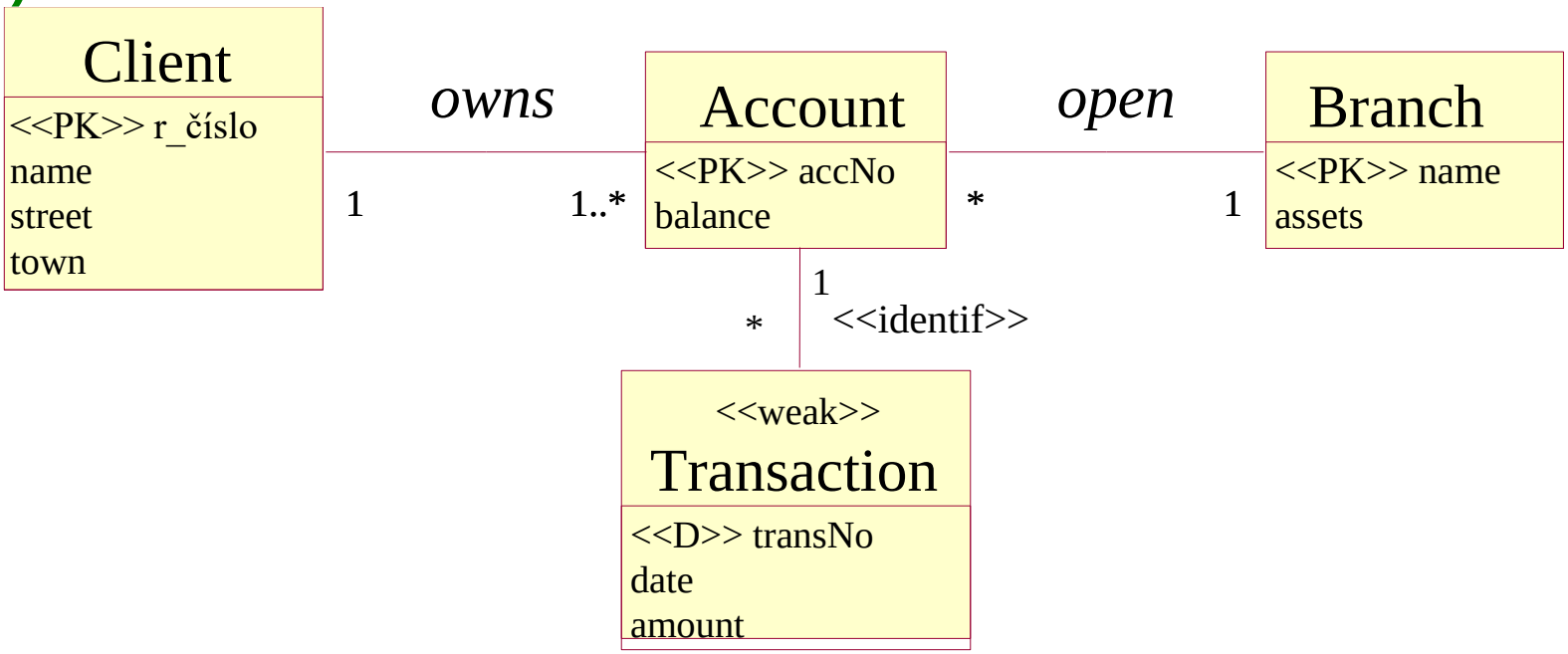


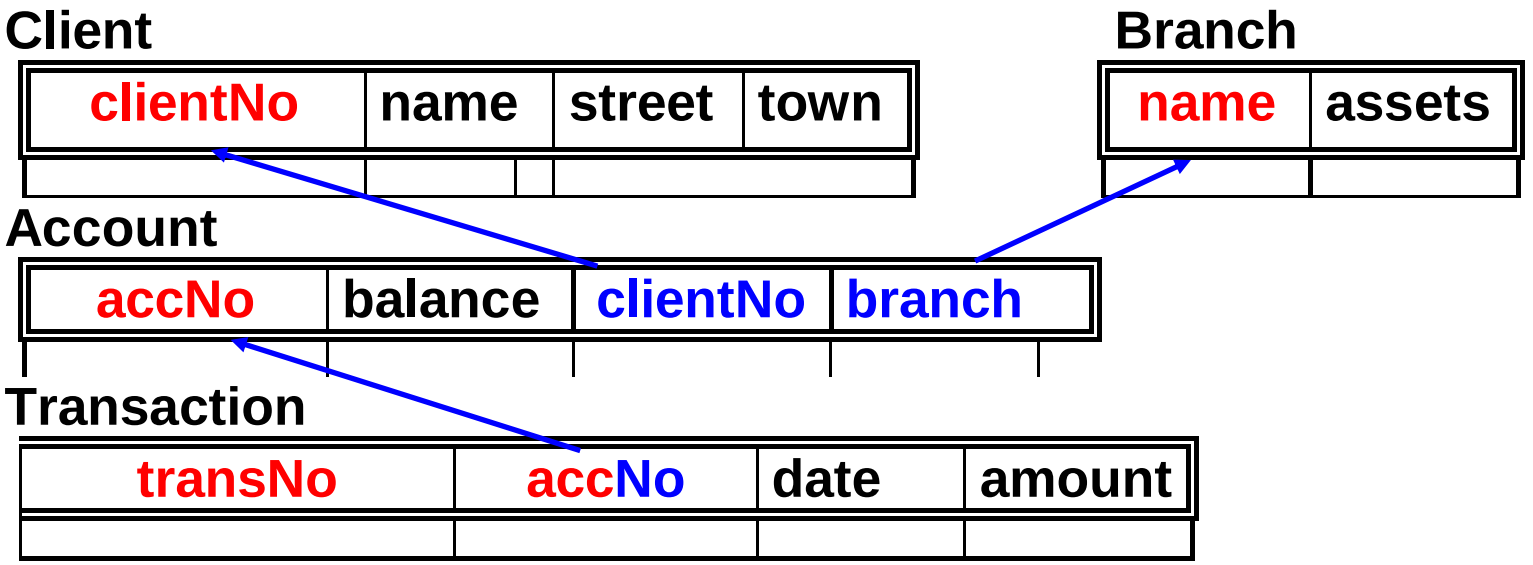
➢ Weak entity sets

- **Actually, it is a combination of transformation of an entity set and a relationship set of 1:M type**

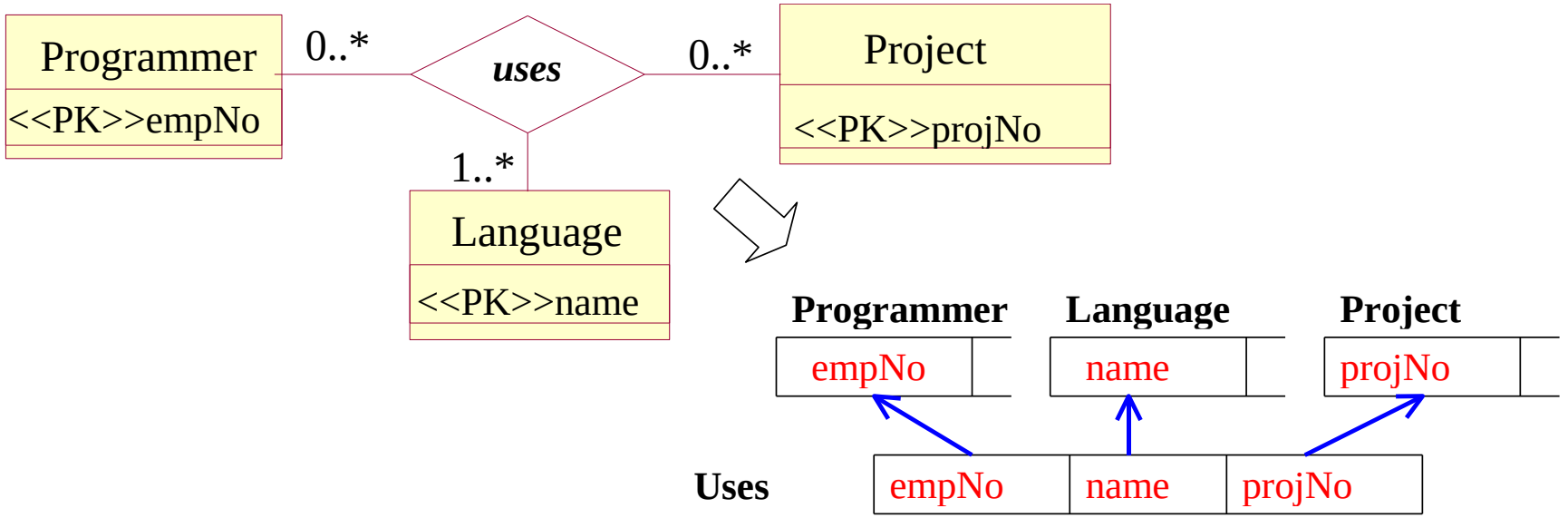


Ex) A bank


Client
<<PK>> r_číslo
name
street
town
owns
1
1..*
Account
<<PK>> accNo
balance
open
*
1
Branch
<<PK>> name
assets
1
*
<<identif>>
<<weak>>
Transaction
<<D>> transNo
date
amount


Client
clientNo
name
street
town
Branch
name
assets
Account
accNo
balance
clientNo
branch
Transaction
transNo
accNo
date
amount

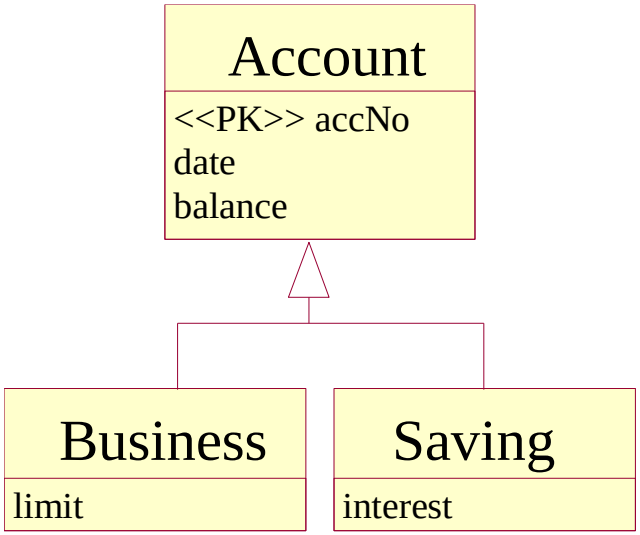
➢ Ternary relationship sets



➢ Generalization/specialization

- **Variants:**
  - **a table for the supertype + a table for each subtype with a primary key of the supertype**
  - **only tables for subtypes that include attributes of the supertype**
  - **a table for the supertype and a table for the subtypes**
  - **everything in one table**
    - **NULLs or so-called *discriminator* can be used to distinguish subtypes**

Ex)



1) Account(accNo, date, balance),
   Business(accNo, limit),
   Saving(accNo, interest)

2) Business(accNo, date, balance, limit),
   Saving (accNo, date, balance, interest)

3) Account(accNo, date, balance),
   Business _saving (accNo, type, limit, interest)

4) Account(accNo, date, balance, limit, interest) or
   Account(accNo, date, balance, type, limit, interest).

- It is necessary to take into account mainly:
  - whether the specializations are disjoint,
  - whether the specialization is total,
  - operations with what data (specializations only or generalization too) will be executed.

Bibliography

1. Silberschatz, A., Korth H.F, Sudarshan, S.:Database System Concepts. Fourth Edition. McGRAW-HILL. 2001, pp. 62 – 68.