

# Databáze I

## 4. přednáška

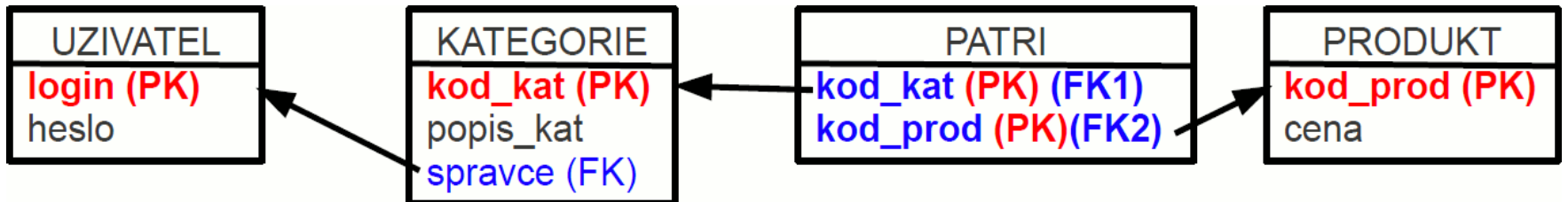
Helena Palovská  
palovska@vse.cz

# Mapování ER modelu do relačního DB schématu

- Od 80. let 20. stol. znám „algoritmus“, implementován v CASE nástrojích
- Rutinní postup s volbami rozhodnutí o tom,
  - co se vztahy 1:1
  - co s ISA vztahy

# Zjednodušený postup

- Každý entitní typ → relační tabulka
- Každý vztah n:m → relační tabulka
- Každý vztah 1:n → cizí klíč v odpovídající tabulce



# Co zbývá vyřešit?

- Entitní podtypy (dědičnost)
  - 3 základní možnosti řešení,  
eventuálně kombinované s view
- Vztahy 1:1
  - 3 možnosti řešení
- Polyární vztahy
  - vždy další relační tabulka

# Celé mapování - postupně

0. Rozhodněte o všech odvozených typech entit, atributů a vztahů, jež mají být ukládány.
  - Odvození se mapují do integritních omezení.
1. Rozhodněte o mapování všech ISA vztahů (*viz další snímek...*)

# Mapování ISA vztahů – možnosti

- **Absorpce** všeho do jedné tabulky
  - atributy podtypů tvoří nepovinné sloupce
- **Separace** do více tabulek
  - společné je ve společné tabulce
  - specifické je mapováno do samostatných tabulek
- **Rozklad** do samostatných tabulek
  - společná tabulka není

# Další kroky mapování

2. Rozhodněte o mapování 1:1 vztahů
  - rozhoduje **povinné/nepovinné členství**, event. které členství je **pravděpodobnější**
    - cizí klíč na tu stranu, kde je vztah pravděpodobnější
  - při oboustranně povinných
    - přehodnotit analýzu
  - při oboustranně nepovinných a málo pravděpodobných
    - možno mapovat i do samostatné tabulky

# Další kroky mapování

3. Rozhodněte o primárních klíčích
  - sémantické business identifikátory mapovat na UNIQUE
4. Každý entitní typ mapujte do samostatné tabulky
  - atomické atributy do polí této tabulky
5. Každý vztah n:m:... do samostatné tabulky.
  - Její klíč bude složen z cizích klíčů ~ rolí  
(viz další snímek...)
  - Případné atributy vztahu tvoří další pole.



# Mapování více-árních vztahů

- Klíče tabulky vztahu jsou určeny integritními omezeními:
  - někdy může být více alternativních klíčů
  - cizí klíče každé z rolí jsou součástí nějakého alternativního klíče

## Příklady:

- *čerpání služeb* <http://nb.vse.cz/~palovska/bivs/obr6.jpg>
- *rozvrh* <http://nb.vse.cz/~palovska/bivs/obr5.jpg>

# Další kroky mapování

6. Vztahy 1:n mapujte do cizích klíčů.

7. Vícehodnotové atributy:

- jako vztah k hodnotovému typu

8. Možnosti pro složené atributy:

- vytvoření umělého entitního typu
- rozdělení na jednotlivé složky

**Příklad:**

- *internetový obchod* <http://nb.vse.cz/~palovska/bivs/obr1.jpg>

<http://krokodata.vse.cz/DM/Mapovani>

# Je výsledné databázové schéma správně?

- Nehrozí *nekonzistence* údajů umožněná tím, že něco je v databázi dvakrát?
- Nebude obtížné zapisovat data proto, že tutéž skutečnost musíme zapsat *na dvě místa*?
- Nehrozí *nezamýšlená ztráta informace*, pokud smažeme určité záznamy?

# Při dokonalé analýze...

- Při bezchybném zachycení v ER modelu,
- po použití rutinního mapování do relačního schématu,
- netrpí výsledné schéma zmíněnými problémy.

Když nejsme dokonalí ... co potom?

# Lze výsledek zkontrolovat?

- Pomůckou jsou "normální formy"
  - 1. normální forma
  - 2. normální forma
  - 3. normální forma
  - Boyce-Coddova normální forma
  - 4. normální forma
  - 5. normální forma

# Normalizace relačního DB schéma

- 1. NF
  - Doména každého atributu obsahuje pouze jedno-hodnotové atomické hodnoty

Není v 1.NF

Vysoká škola ekonomická	W.Churchilla 4, 130 00 Praha 3	224-095-111
Vysoká škola finanční a správní	Estonská 500, 101 00 Praha 10	210-088-800 271-741-597

**Vnitřní struktura významů,  
neatomická (strukturovaná) hodnota.**

**Více hodnot  
(se stejným významem).**

# První normální forma

- Doména každého atributu obsahuje pouze jedno-hodnotové atomické hodnoty
- Dva různé atributy v téže relační tabulce nemají stejný význam

Není v 1.NF

	telefon1	telefon2
VŠE	224-095-111	
VŠFS	210-088-800	271-741-597



# Normalizace relačního DB schéma

- 2.NF
  - Je v 1.NF,
  - a každý neklíčový atribut závisí na celém klíči, ne pouze na části klíče

## Není v 2.NF



**Závisí jen na názvu kategorie,  
opakuje se u každého produktu v dané kategorii!**

# Normalizace relačního DB schéma

- 3.NF
  - Je v 2.NF.
  - a každý neklíčový atribut je závislý přímo, a nikoli tranzitivně, na každém kandidátním klíči

Nej v 3.NF

KATEGORIE
<b>kod_kat (PK)</b>
popis_kat
login_spravce
heslo_spravce

**Závisí na loginu správce, který je určen pro kód kategorie. Opakuje se u každé kategorie, již je tento správcem!**

# Normalizace relačního DB schéma

- Porušení 2., 3., ... NF:
  - Tatáž informace v různých místech databáze.
  - Hrozí nekonzistence zápisu této informace v těch různých místech
  - Zatěžuje se DB nutností zapisovat více, než je třeba
  - V případech mazání nenormalizovaných záznamů hrozí nezamýšlená ztráta informace (např. jaké heslo má ten uživatel)
  - Hrozí nepochopení schématu a jeho nesprávné užití v dotazech, programech

# Další normální formy

- Boyce-Coddova normální forma (BCNF)
  - jako 3.NF, ale podmínka netranzitivnosti platí pro jakýkoli, i klíčový, atribut
    - Například tabulka poštovních směrovacích čísel není v BCNF:  
  
POST\_SMER\_CIS(PSC, mesto, ulice)  
  
Dva kandidátní klíče (mesto, ulice) a (PSC, ulice), tedy žádný atribut není neklíčový.  
Ale PSC→mesto není závislost na celém klíči.
- BCNF se obvykle nepožaduje, vyžaduje „nepřirozený“ rozklad relačních tabulek. Někdy je dokonce nedosažitelná.

# Další normální formy

- 4.NF

- Týká se tzv. multizávislostí.

- Například relační tabulka PREDMET s atributy

- (ident\_predmetu, skupina\_predmetu, vyucujici\_ucitel)

není v 4.NF,

vyucujici\_predmetu i skupina\_predmetu *multizávisí na* ident\_predmetu (předmět je ve více skupinách a může mít více vyučujících),

pokud je předmět v 5 skupinách a má 4 vyučující, tabulka obsahuje pro tento předmět 5 x 4 záznamů.

# Další normální formy

- 5.NF
  - Například když
    - Každý technik umí nějaké úkony
    - Na každém produktu se provádějí nějaké úkony
    - Technik může pracovat na nějakých produktech
  - pak tabulka  
MUZE\_VYKONAVAT(technik, produkt, ukon)  
není v 5.NF.
  - Jsou třeba tyto tabulky:  
UMI(technik, ukon)  
SE\_DELA(produkt, ukon)  
PRACUJE\_NA(technik, produkt)

# Normální formy relací

- První tři jsou často požadovány.
- Existují i další normální formy.
- Smyslem je zabránit redundanci informace, nekonzistenci informací, eventuálně nezamýšlené ztrátě informace.
- Standardní transformace ER do relačního DB schéma produkuje normalizovanou databázi *za předpokladu bezchybné analýzy* vtělené do ER modelu.

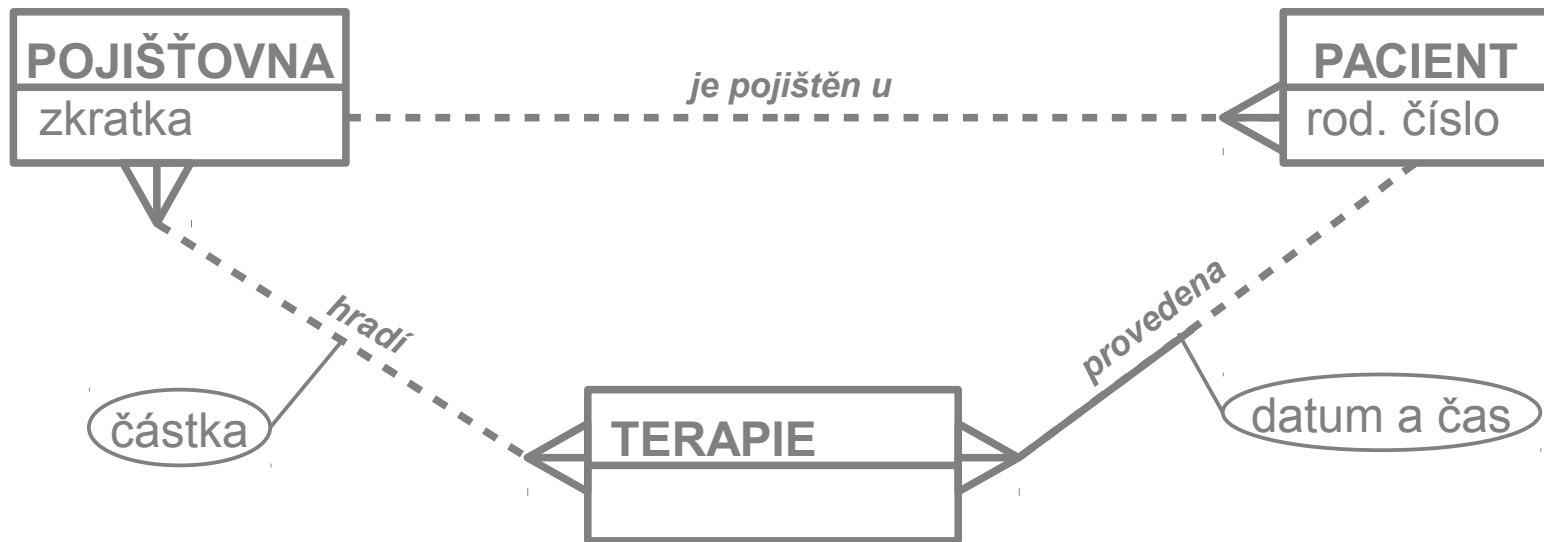
# Nejčastější chyby konceptuální analýzy informací

- Dvoutvářné entity
- Dědičnost namísto vztahu typu a výskytu
- Ternární či více-ární vztah namísto několika binárních
- et vice versa
- Chybějící vztahová entita při „opakování vztahu“



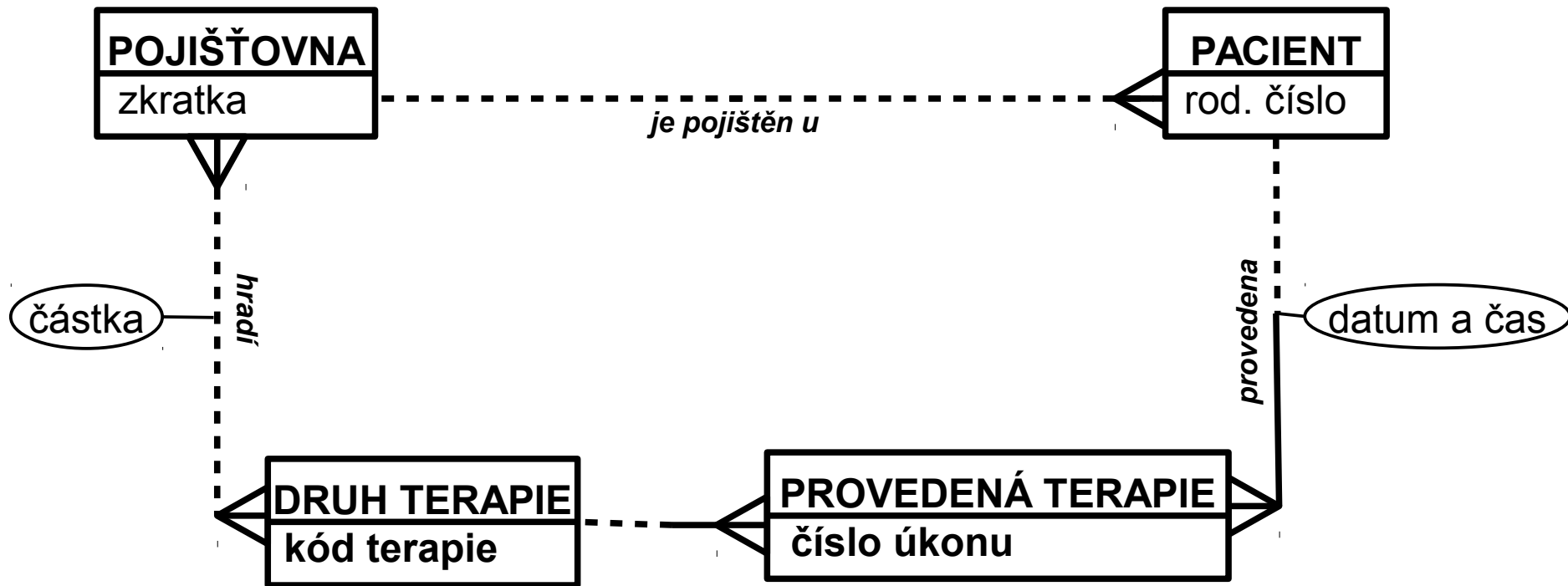
# Dvoutvářné entity

- Je toto schéma správně?



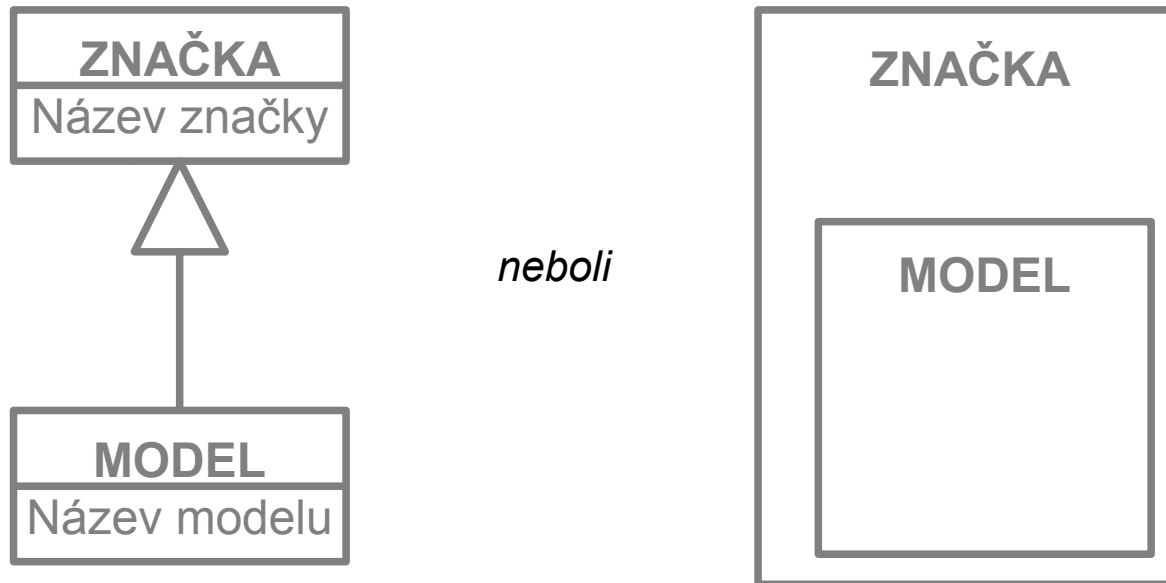
Co je klíč TERAPIE?

# Dvoutvářné entity



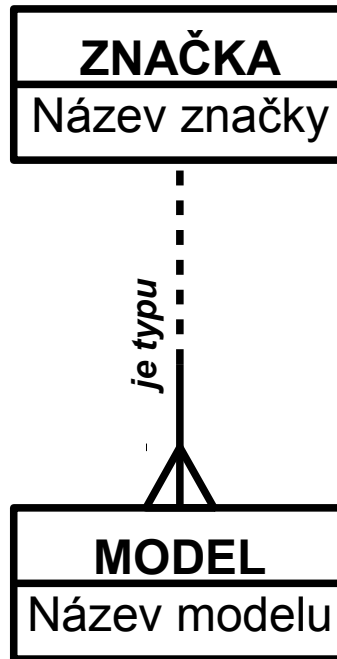
# Dědičnost namísto vztahu typu a výskytu

- Je toto schéma správně?



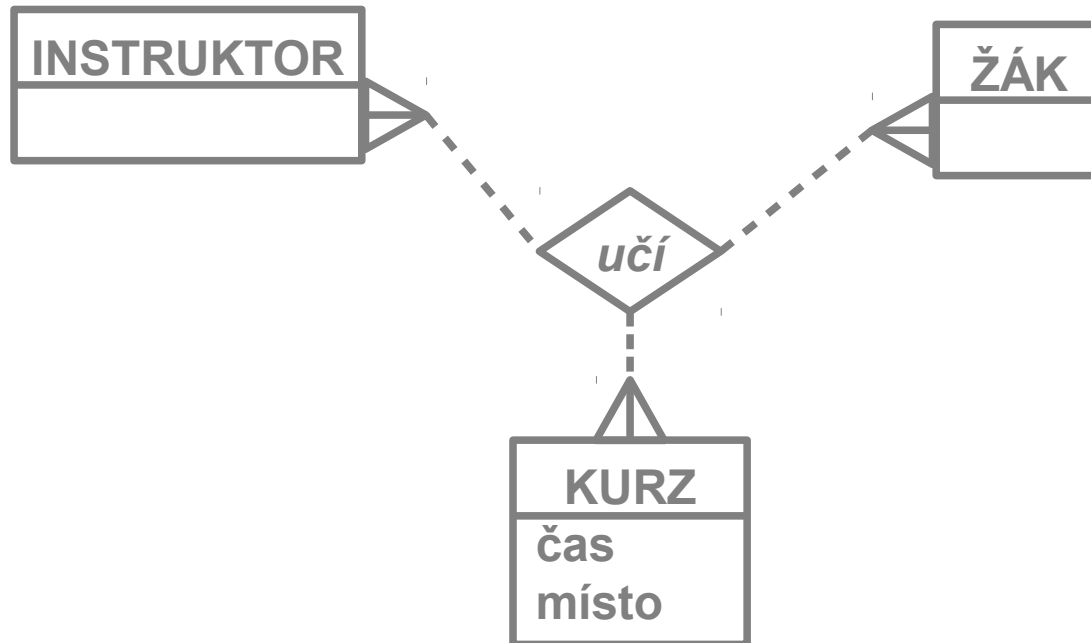
**Co je klíč MODELu?  
Kolik MODELů náleží k jedné ZNAČCe?**

# Dědičnost namísto vztahu typu a výskytu



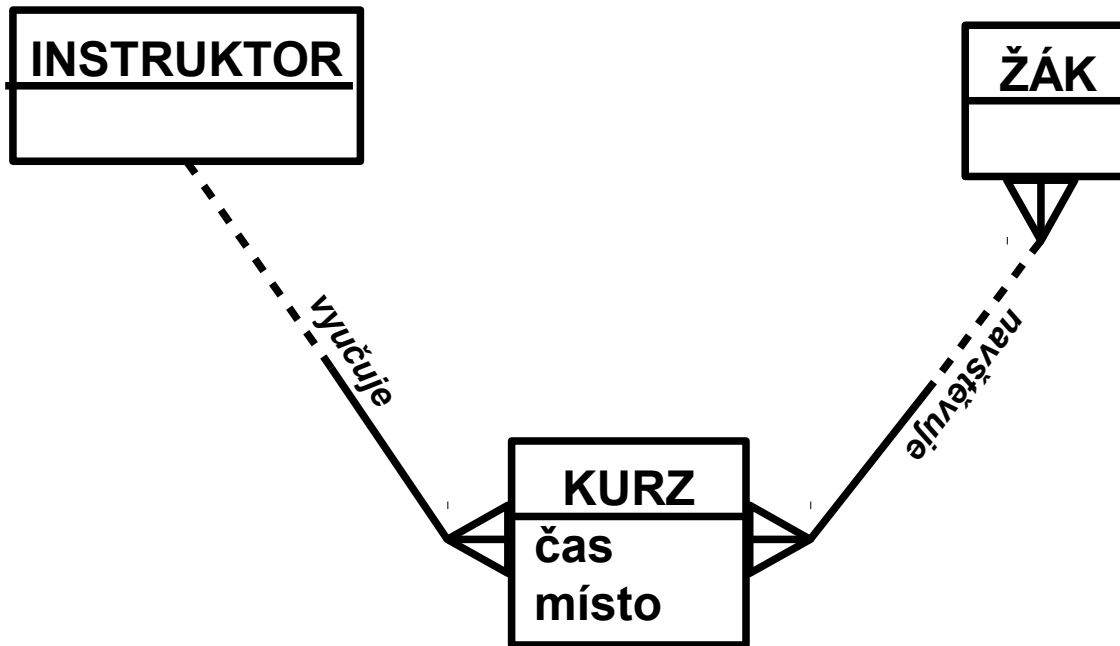
# N-arní vztah či několik binárních?

- Je toto schéma správně?



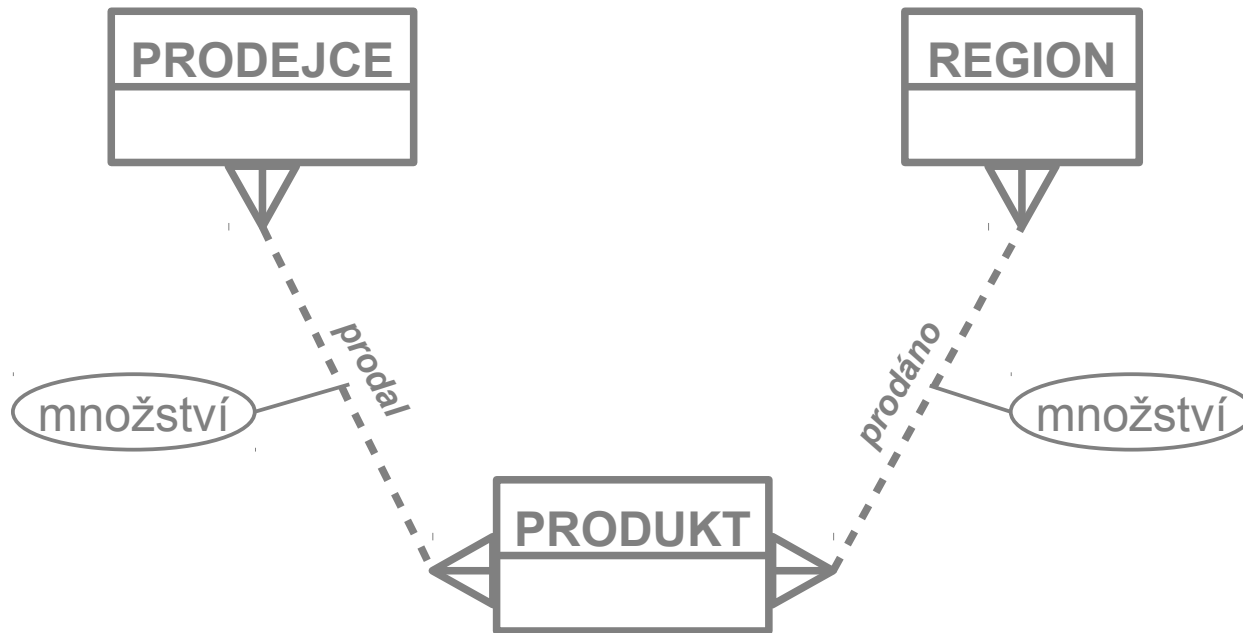
**Co je klíč tabulky vztahu „učí“?  
Není možno vztah rozložit bez ztráty informace?**

# N-arní vztah či několik binárních?



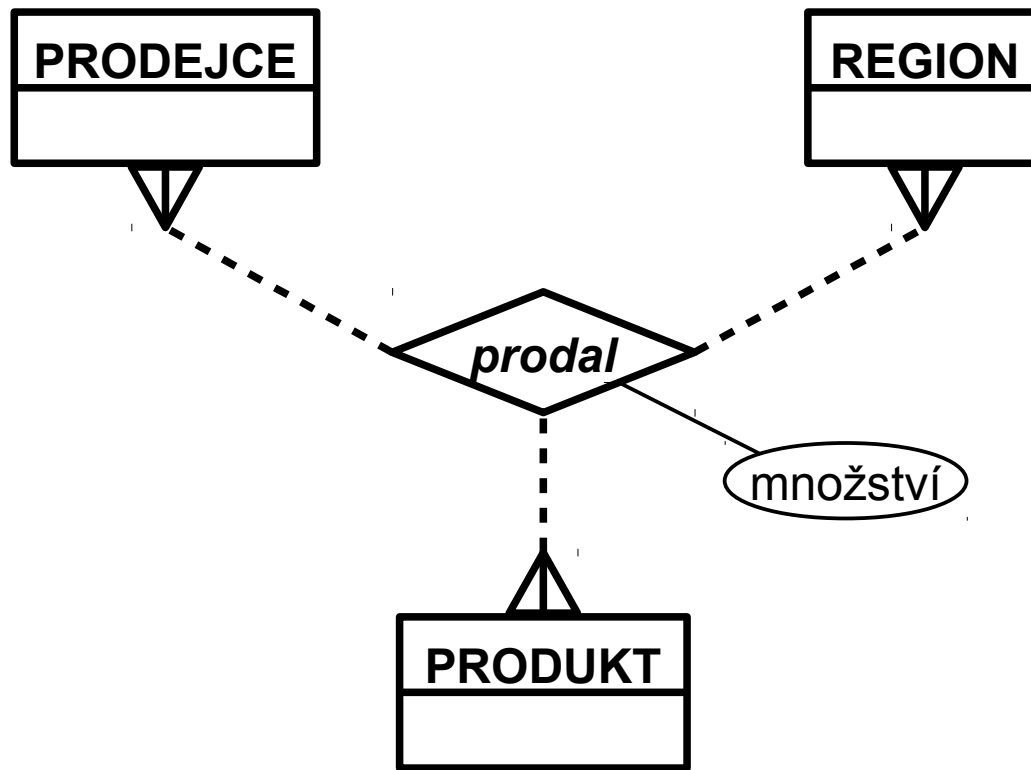
# N-arní vztah či několik binárních?

- Je toto schéma správně?



Zjistíme, kolik kterého PRODUKTu prodal který PRODEJCE v kterém REGIONu?

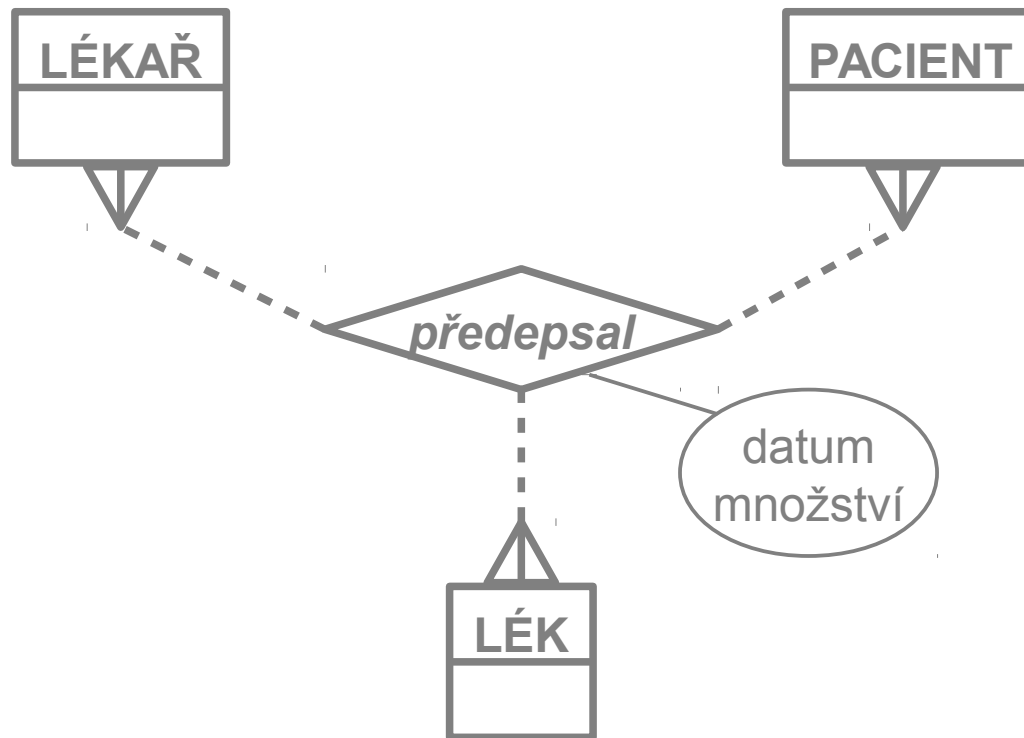
# N-ární vztah či několik binárních?





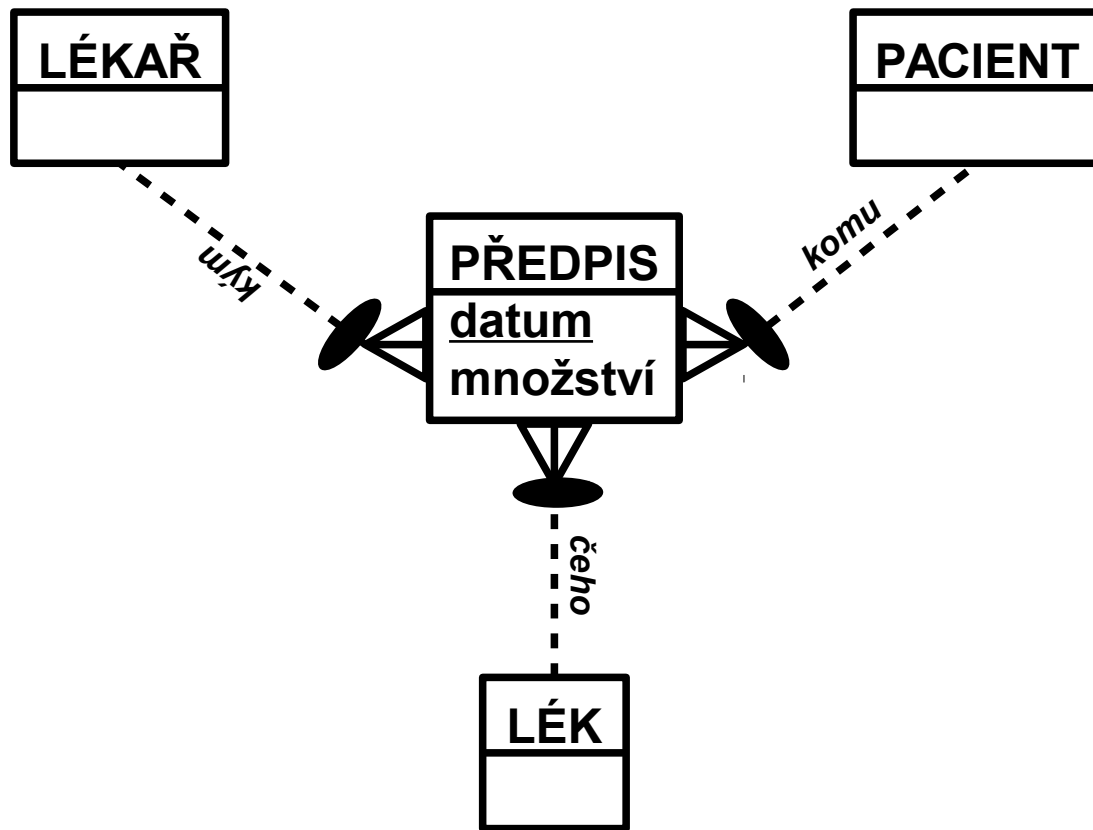
# Opakování vztahu

- Je toto schéma správně?



Co je klíč tabulky vztahu „předepsal“?  
Kolikrát mohl jeden LÉKAŘ jednomu PACIENTovi předepsat jeden LÉK?

# Opakování vztahu



# Obecné užitečné otázky informační analýzy

- Umožní nám navržené schéma zapsat **všechny potřebné informace?**
- Jak z navržené databáze získáme **danou konkrétní informaci?**
- Neumožňuje navržené schéma **nekonzistenci v poskytovaných informacích?**
- Nehrozí nezamýšlená **ztráta informace?**

# Další pomůcky

- Databázové návrhové vzory
  - Šablony „správných“ řešení
  - Znovupoužitelné
  - Ověřené

Konec