



Modelování dat



- Definuje neměnné atributy a strukturu dat.
- Slouží jako stabilní základ procesního modelu.
- Data, která systém uchovává, jsou relativně konstantní. V čase se daleko více mění způsob jejich zpracování, než jejich struktura.
- Datový model vyjadřuje vztahy, které nejsou zachyceny v procesních modelech.
- Vztahy v datovém modelu nesmí být porušeny při transformaci dat jednotlivými procesy.

Komponenty datového modelu



- **Entita** - objekt, o němž uchováváme informace
 - Je identifikovatelná (existuje způsob, jak odlišit jednotlivé entity od sebe).
 - Je potřebná , hraje významnou roli ve vytvářeném systému.
 - Je popsána nejméně jedním, ale zpravidla více atributy.
- **Entitní množina**
 - Množina entit stejného druhu, typu.
- **Vztah** - vztah mezi entitami, který evidujeme a o němž uchováváme informace

MUDr. Spaček je ošetřujícím lékařem pana Vopičky. }

Na zájezd se přihlásili pan Mňouk a pan Muk. }

Host si rezervoval pokoj. ←

vztah

typ vztahu



Komponenty datového modelu

- **Vztahová množina**
 - Množina vztahů stejného druhu, typu.
- **Atribut**
 - Vlastnost entity nebo vztahu, jejíž hodnotu chceme uchovat a používat v systému.
- **Doména atributu**
 - Obor hodnot atributu.

Terminologické posuvy v některé literatuře:

typ entity - entita

entita - instance (výskyt) entity

typ vztahu - vztah

vztah - instance (výskyt) vztahu



Datový slovník - notace

- = skládá se z
- + a zároveň
- () nepovinná část (0 nebo 1)
- [|] výběr jedné z možností
- { } iterace (1 nebo více) $a=_1\{b\}_{15}$
- * * komentář
- @ identifikátor (klíč)

Příklad - Objednávka



Objednávka

č. 2006-007-24

Sportovní potřeby, Mírotvůrců 17, 602 00 Brno

Datum vystavení: 23.4.2006

Datum dodání: 30.4.2006

Zákazník: č. 007,
Dr. Jaroslav Ráček

Zboží:

Číslo	Název	Kusů	Cena za kus
PB666	Baseballová pálka Magnum	30	499
H46	Helma Tvrďák	30	---

Příklad - Objednávka



OBJEDNÁVKA = @číslo objednávky + datum vystavení
objednávky + číslo zákazníka + jméno zákazníka + (datum
dodání) + { číslo výrobku + název výrobku + počet objednaných
výrobků + (cena za výrobek) }

jméno = titul + křestní jméno + (prostřední jméno) + příjmení

titul = [Mr. | Mrs. | Miss. | Dr. | Prof.]

křestní jméno = { přípustný znak }

přípustný znak = [A - Z | a - z | ' |]



Definice datového elementu - DE

Definice zachycuje zjištěná fakta o DE

- Význam DE v kontextu aplikace
- Kompozice DE, pokud je složen z jiných DE
- Hodnoty, kterých může DE nabývat

Příklady:

váha = * váha pacienta při přijetí *,
* jednotky: kg; rozsah: 1 - 200 *

pohlaví = * hodnoty [M | F] *

zákazníková adresa = [doručovací adresa | účtovací adresa |
doručovací adresa + účtovací adresa]



Entity

Datové modelování hledá entity jako popisy

- konkrétních věcí - *zaměstnanec, součástka*
- událostí - *dopravní nehoda, operace*
- pojmů - *pojistka, smlouva*

Pro každý *objekt, událost, pojem* zkoumáme

- k čemu slouží (účel)
- z čeho se skládá

„Top-down“ řešení datového modelu

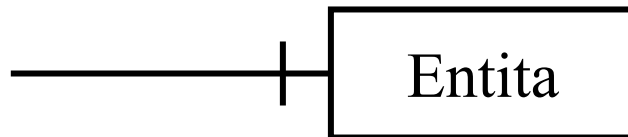
1. výběr (nalezení) entit
2. návrh diagramu entit a vztahů (ERD)
3. přidání atributů



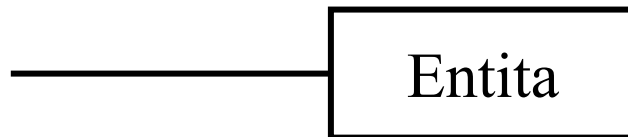
Komponenty ERD:

- **Entity** (objektové typy, objekty, entitní množiny)
- identifikovatelné, potřebné, více atributů.
- **Relace, vztahy**
- vztah, který je nutné pamatovat, nelze odvodit nebo mechanicky vypočítat
- **Indikátory vztažených entit**
- **Indikátory nadtypů/podtypů**

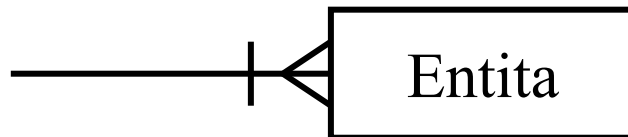
Grafické prvky ERD



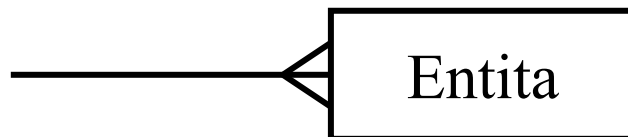
Právě jeden výskyt



Žádný nebo jeden výskyt

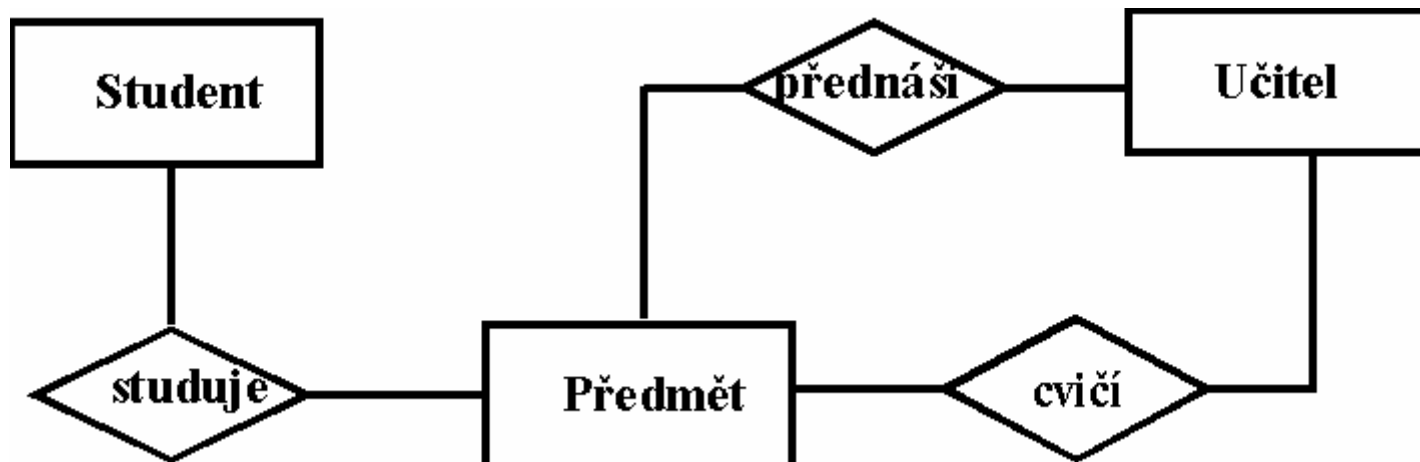


Jeden nebo více výskytů



Žádný, jeden nebo více výskytů

Příklad - Studium



Entity: *student, předmět, učitel*

Vztahy: *student - studuje - předmět*
učitel - přednáší - předmět
učitel - cvičí - předmět

Arity - násobnosti vztahů



**Každý vedoucí vede právě jedno oddělení.
Každé oddělení je vedeno právě jedním vedoucím.**



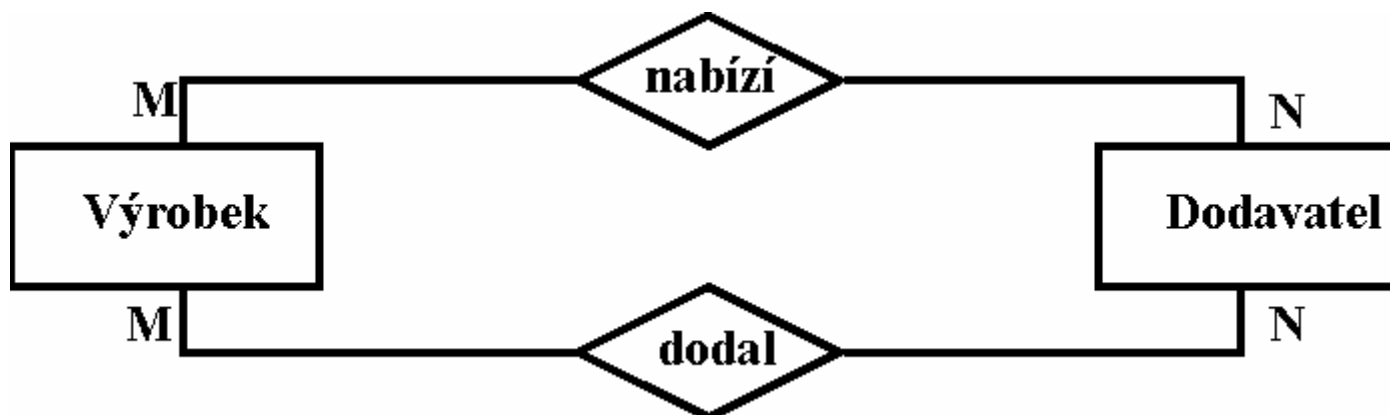
**Každý ediční plán obsahuje jeden nebo více titulů.
Každý knižní titul je zařazen právě v jednom edičním plánu.**



**Výrobce vyrábí jeden nebo více výrobků.
Výrobek je vyráběn jedním nebo více výrobci.**



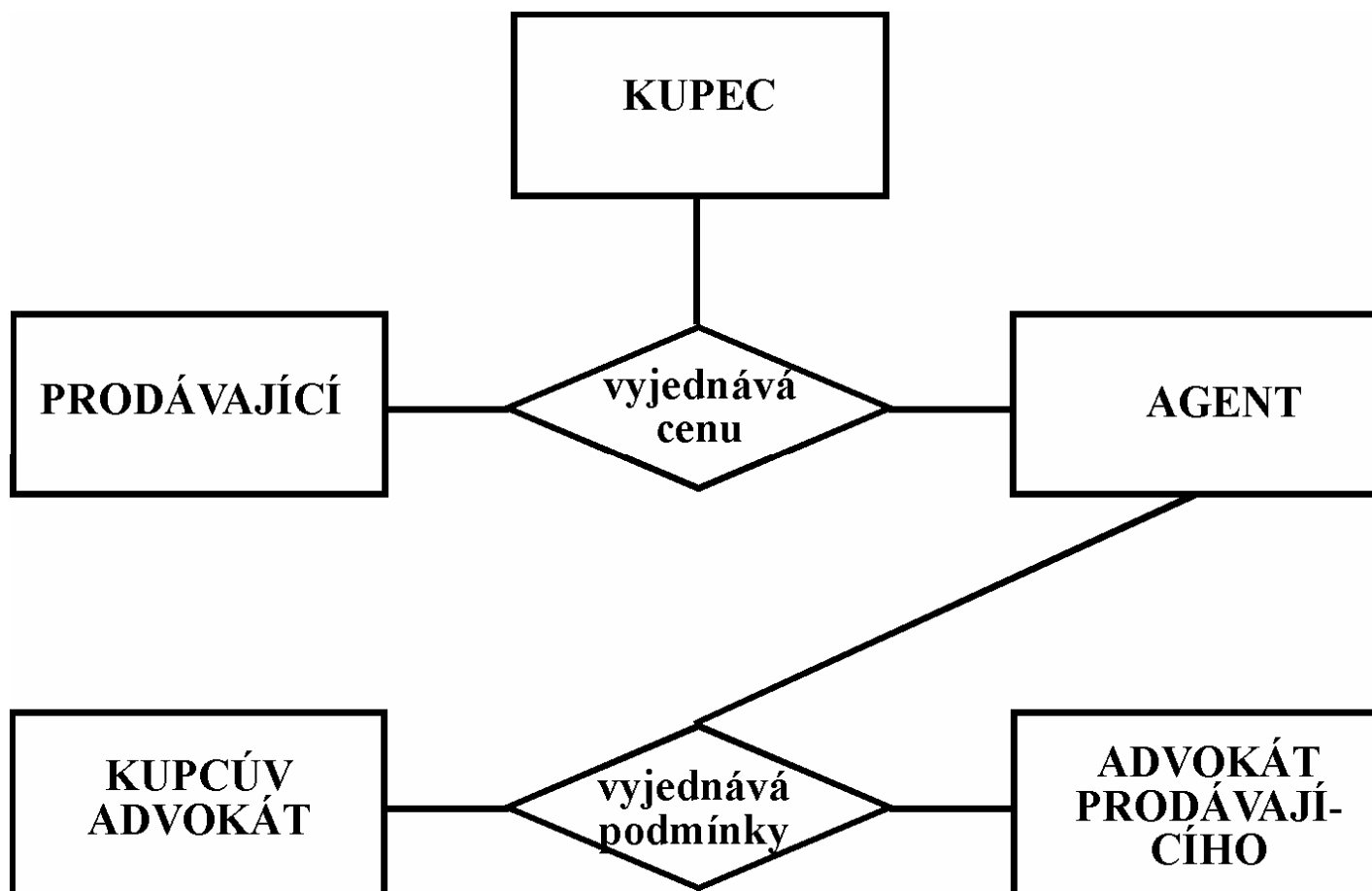
ERD s vícenásobnými vztahy



Vztah *nabízí* má atributy: *platební podmínky, termíny*.

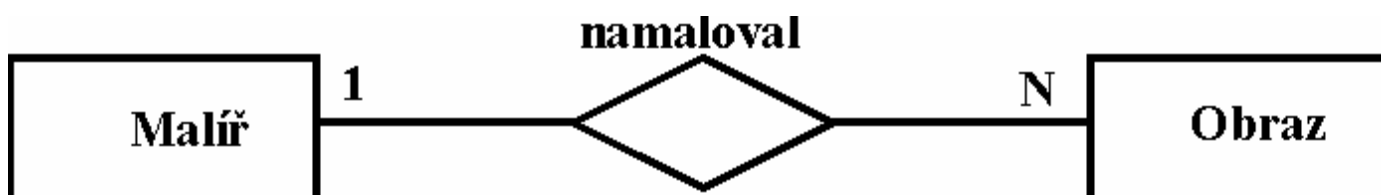
Vztah *dodal* má atributy: *údaje z dodacího listu*.

ERD s vícenásobnými vztahy





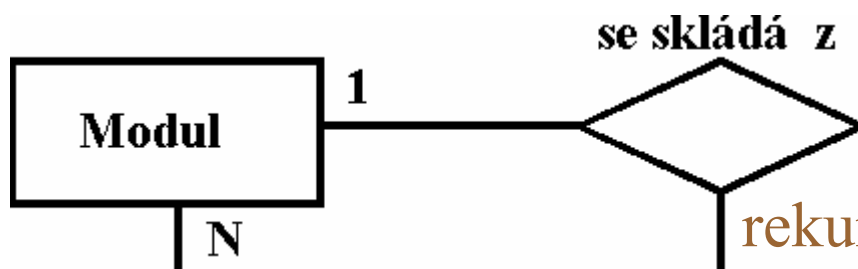
Povinné, nepovinné a rekurzivní vztahy



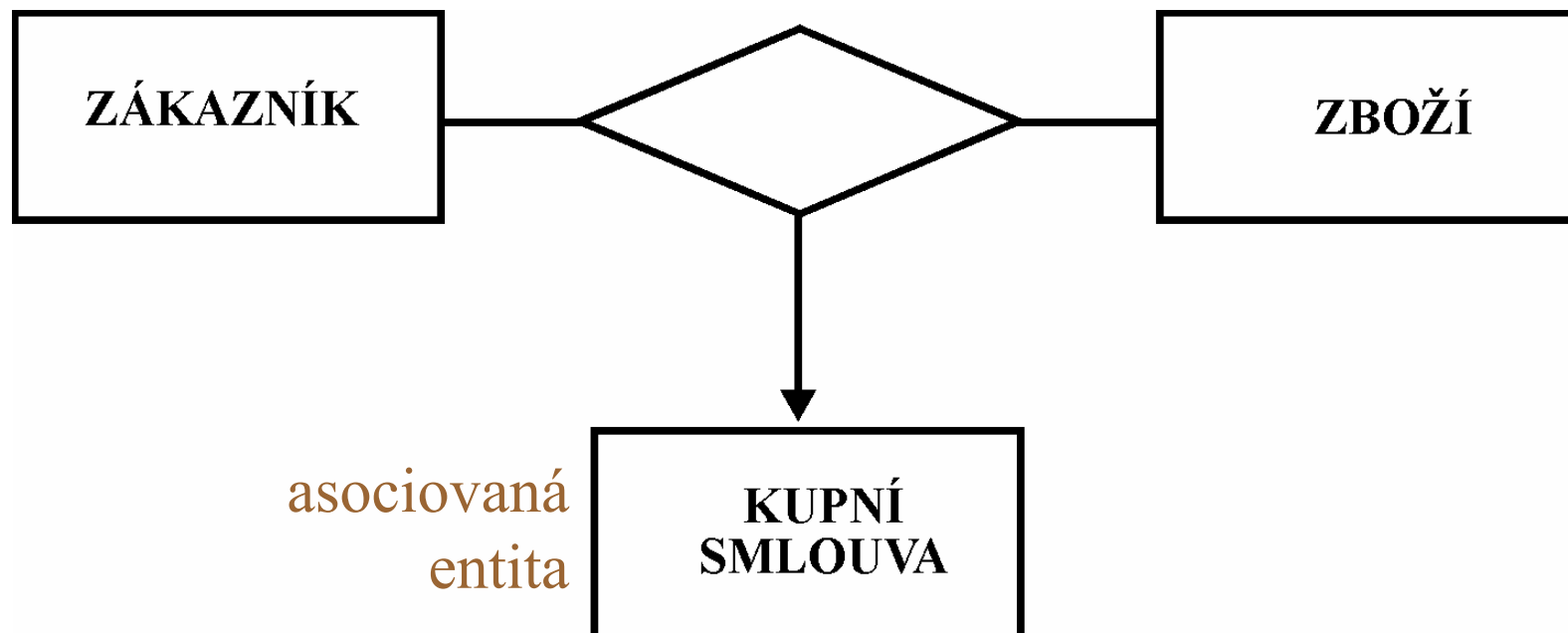
povinný vztah



nepovinný vztah

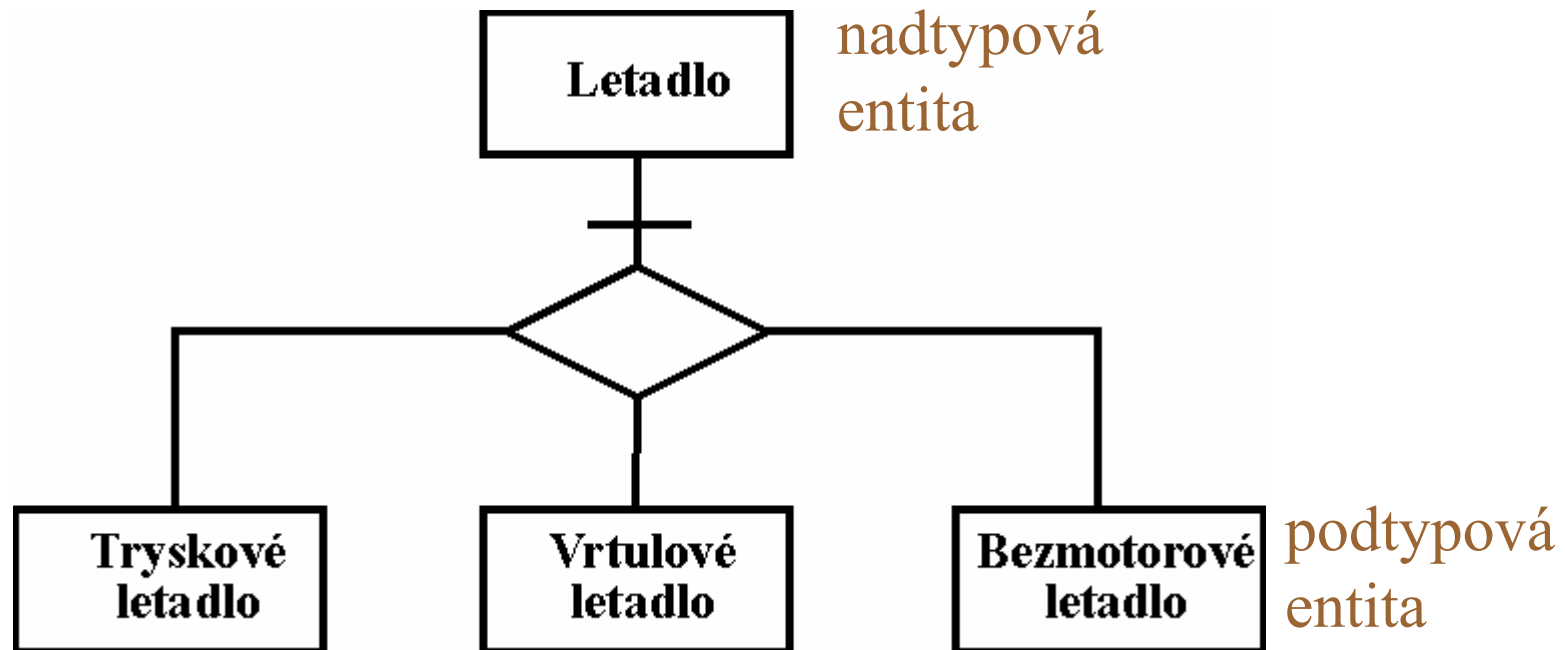


rekurzivní vztah



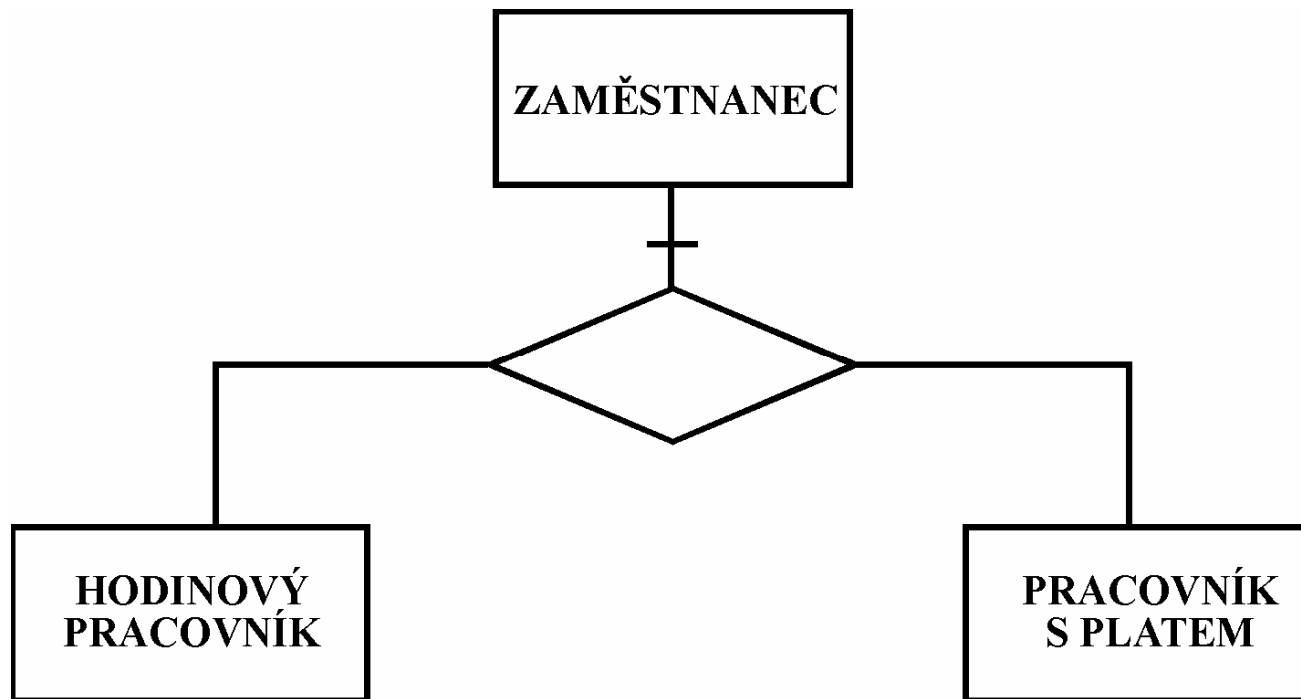
Kupní smlouva existuje pouze jako výsledek vztahu mezi jinými objekty, k nimž je připojena.

Nadtypové a podtypové entity

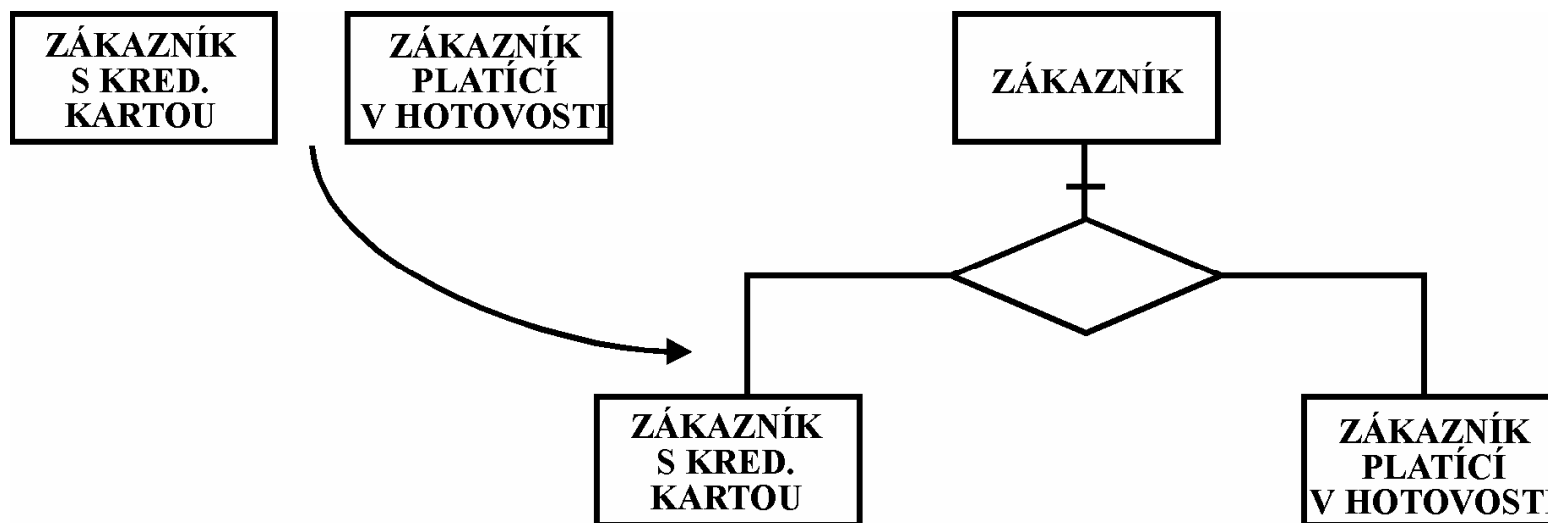


Vztah „generalizace - specializace“, „dědičnost“

ERD s nadtypovým vztahem



ERD s nadtypovým vztahem





1. Vytvoření iniciálního ERD

- Pochopení uživatelské aplikace
- Interview s uživatelem
- Jiné techniky získání dat

2. Přidání entitních množin

- Dále použít metodu postupného zjemňování
- Po vytvoření prvotního datového modelu nalézt atributy entit na základě
 - existujícího funkčního (procesního) modelu
 - seznamu datových elementů od zákazníka
 - existujícího datového slovníku od zákaznickových „databázistů“



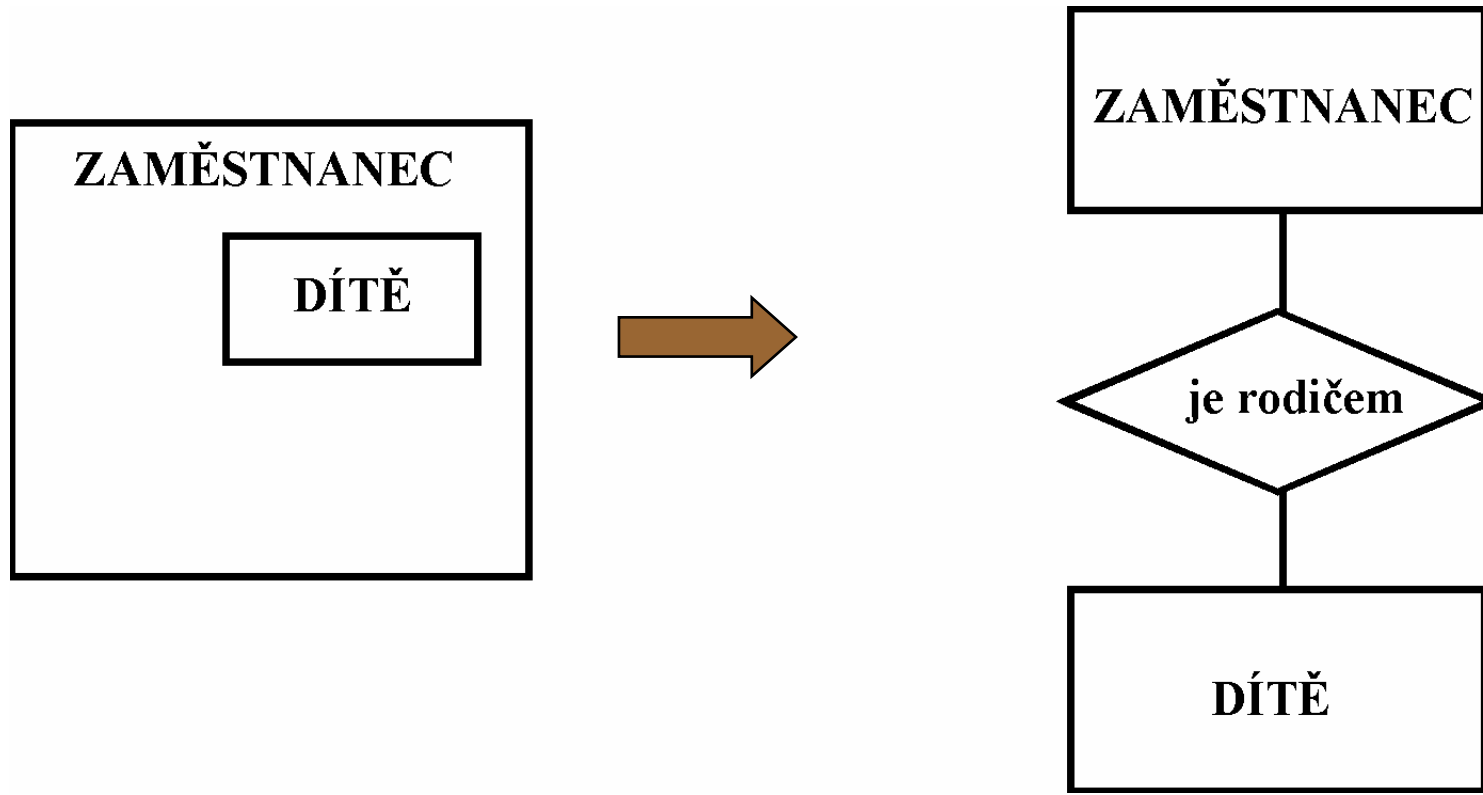
3. Prověrka entit na základě seznamu datových elementů

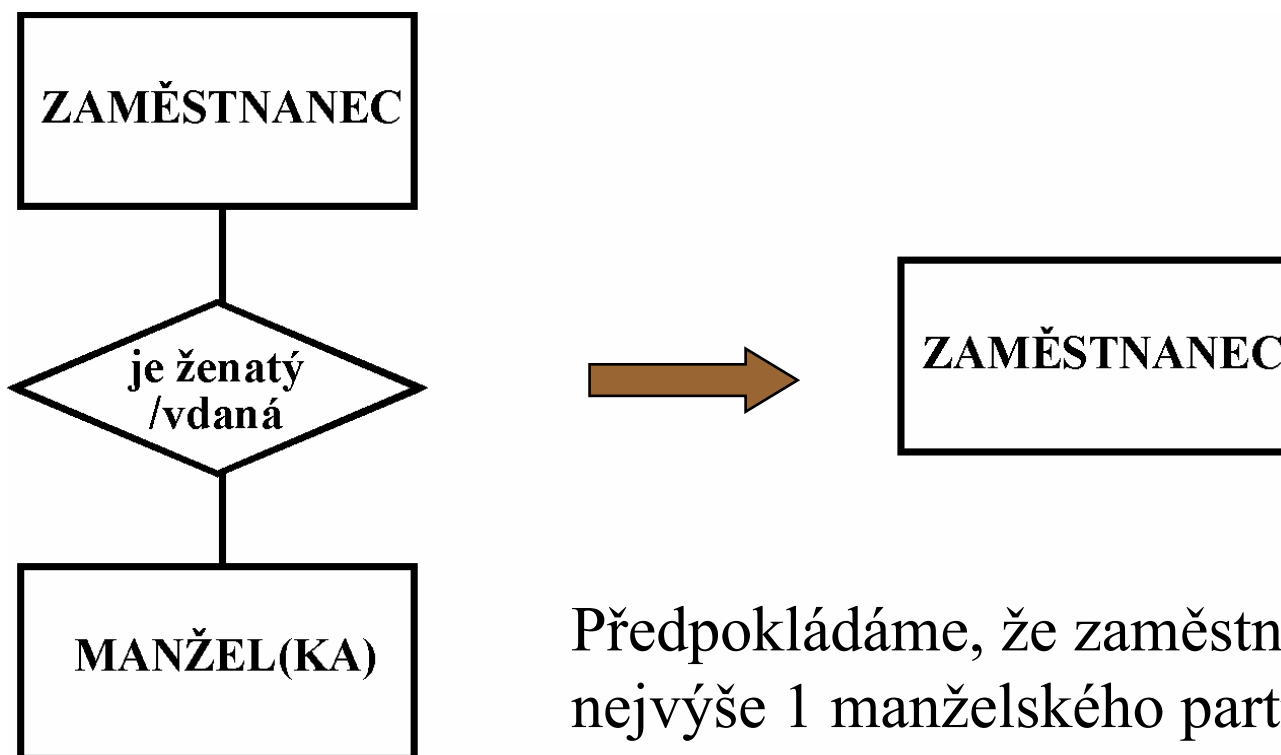
- Některé datové elementy (DE) lze přidat jen k některým entitám, u jiných entit stejné entitní množiny nemají smysl.
- Některé DE lze aplikovat na všechny entity dvou různých entitních množin.
- Některé DE uplatněné u entitní množiny popisují vztahy mezi jinými entitními množinami.

4. Odstranění entitních množin

- Entity tvořené pouze identifikátorem.
- Entitní množiny s jedinou entitou.
- Volně zavěšené asociativní entity.
- Odvozené vztahy.

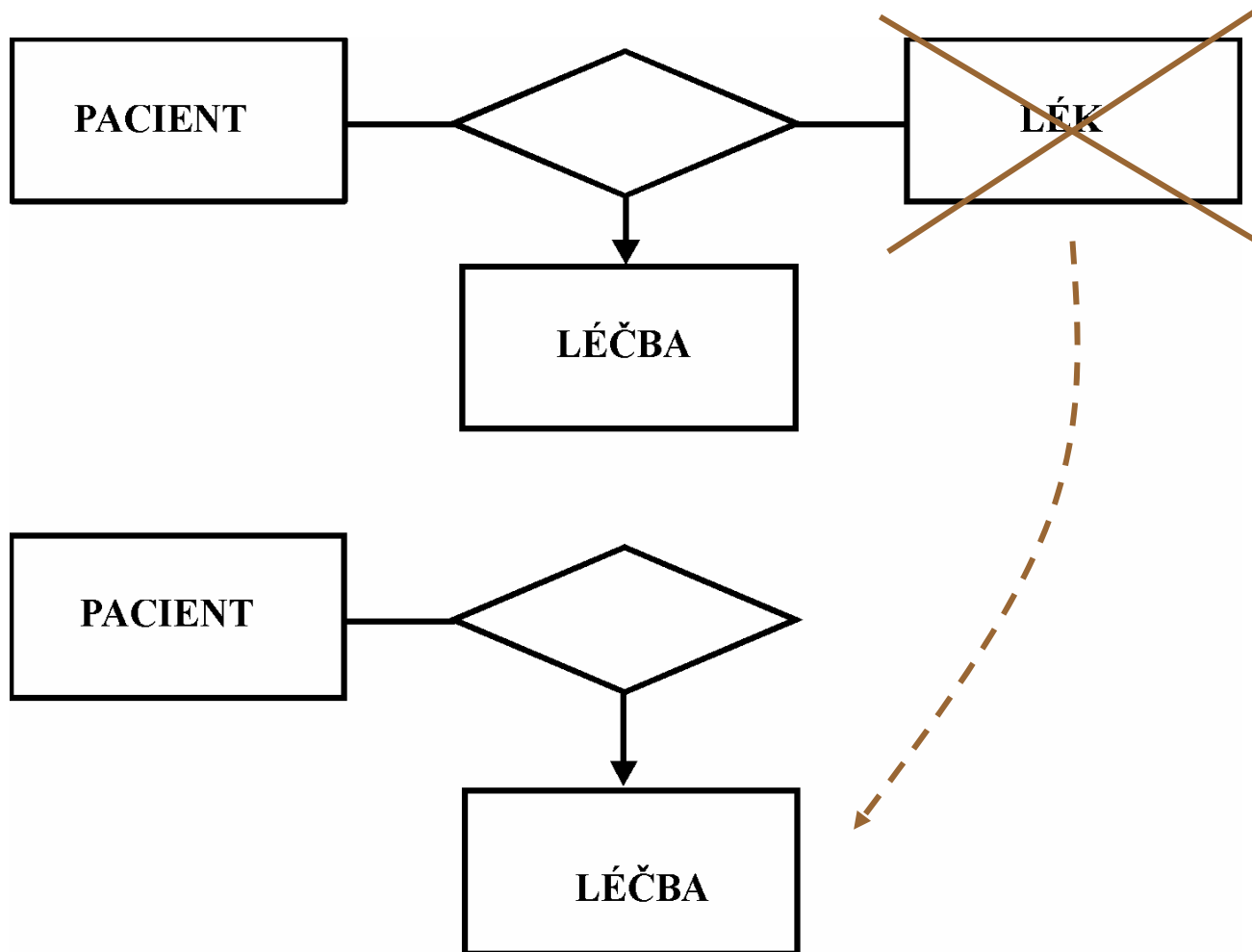
Vnořené entity



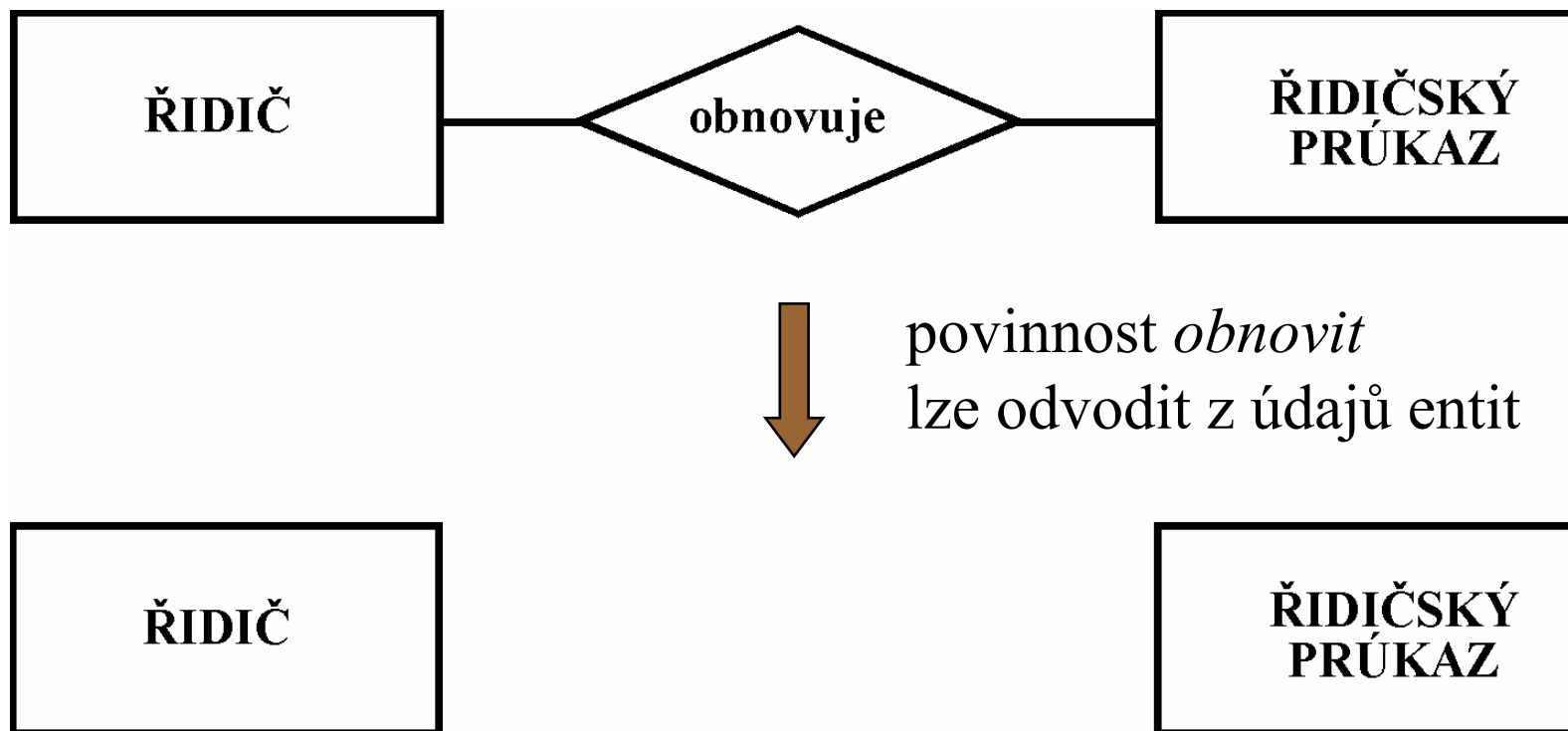


Předpokládáme, že zaměstnanec má nejvýše 1 manželského partnera.

Zrušení entity



Zrušení nepotřebného vztahu





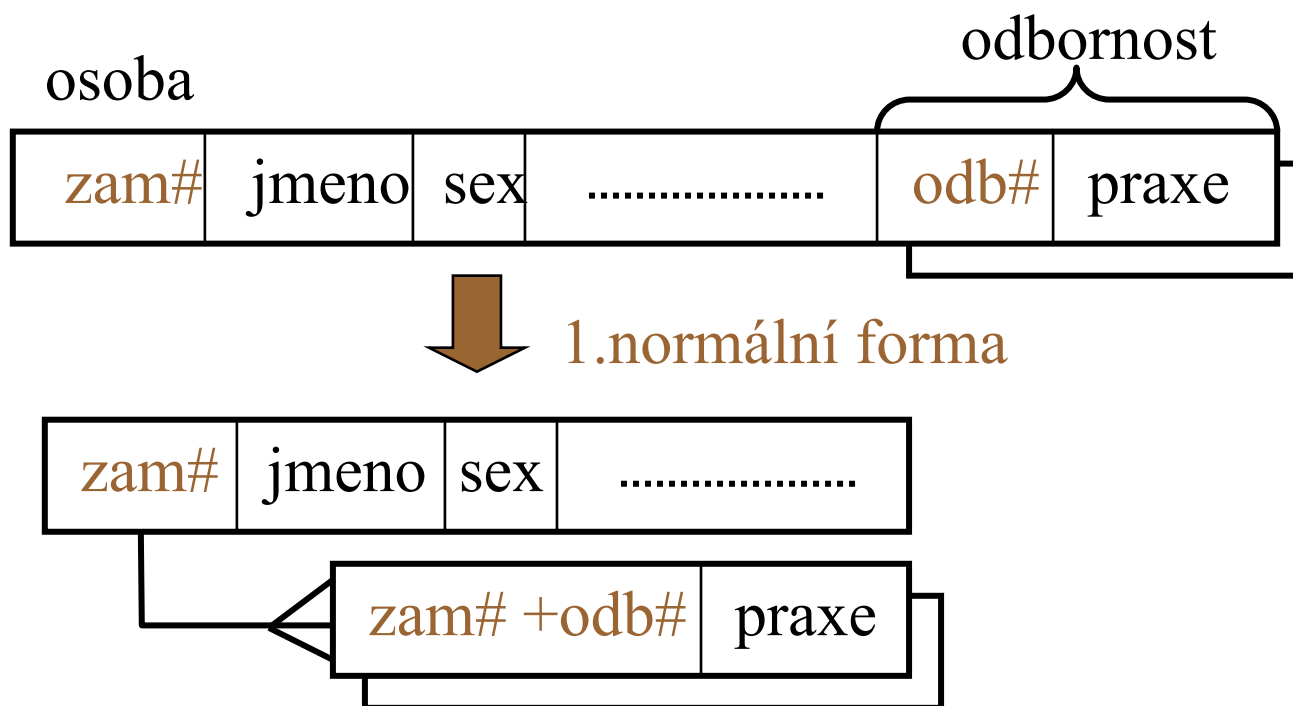
- **Jednoznačná identifikace**
 - Každá entita je unikátně identifikována klíčem.
- **Úspornost (nonredundance)**
 - Zrušením libovolné položky klíče je porušeno pravidlo jednoznačné identifikace.
- **Kandidátní klíče**
 - Více položek může být použito pro jednoznačnou identifikaci ve funkci klíče.

↓ výběr
primární klíč#



Normalizace dat (E.F.Codd) - 1. normální forma

Uspořádání dat takovým způsobem, abychom se vyhnuli možným anomáliím a problémům, které mohou vzniknout při práci s daty.



Def.1.NF: V záznamech se nevyskytují opakující se skupiny položek.



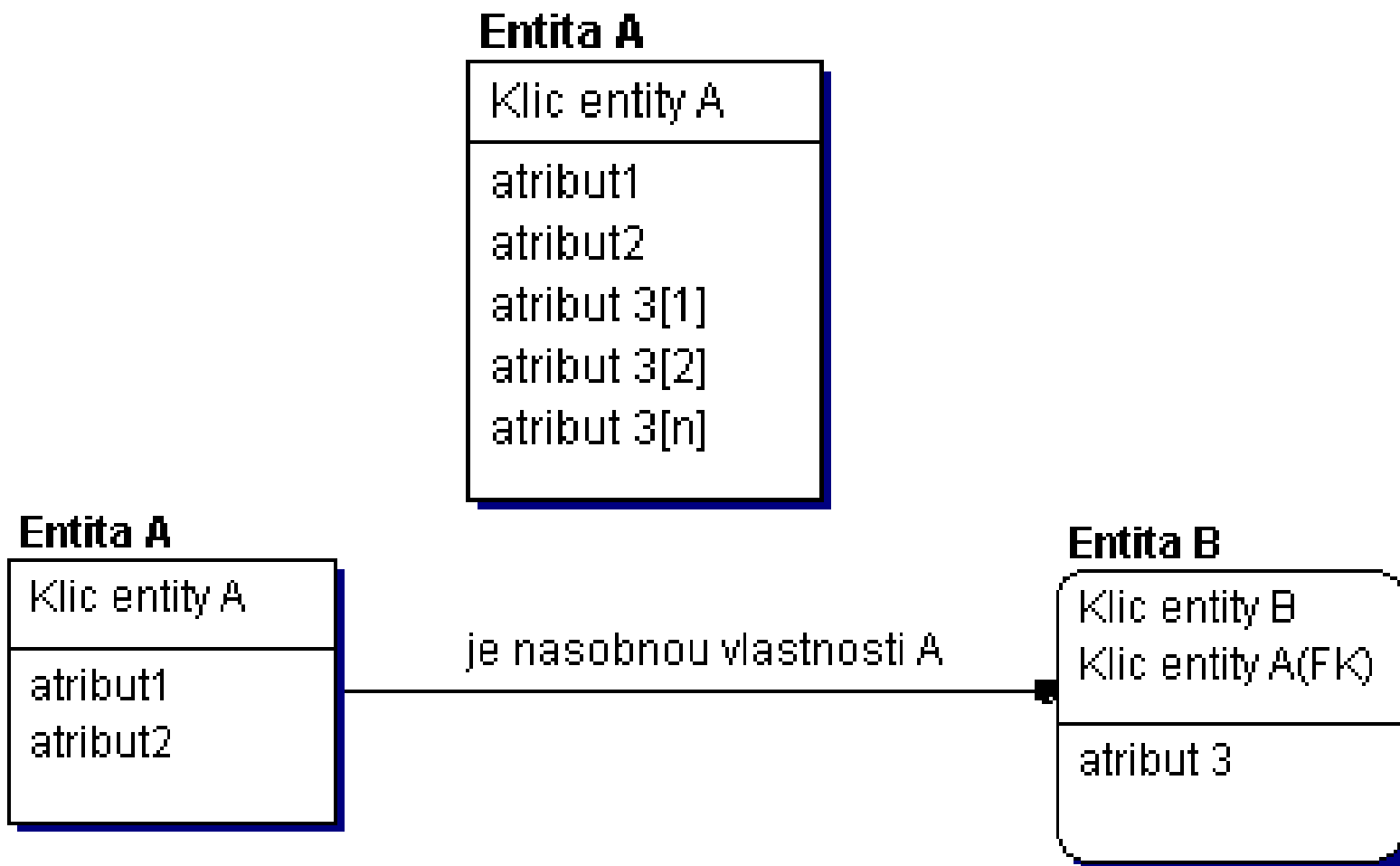
1. normální forma

2 definice 1.NF:

- Datový záznam je v 1. NF, pokud všechny jeho komponenty jsou atomické.
- Datový záznam je v 1. NF pokud se v něm nevyskytují opakující se skupiny položek

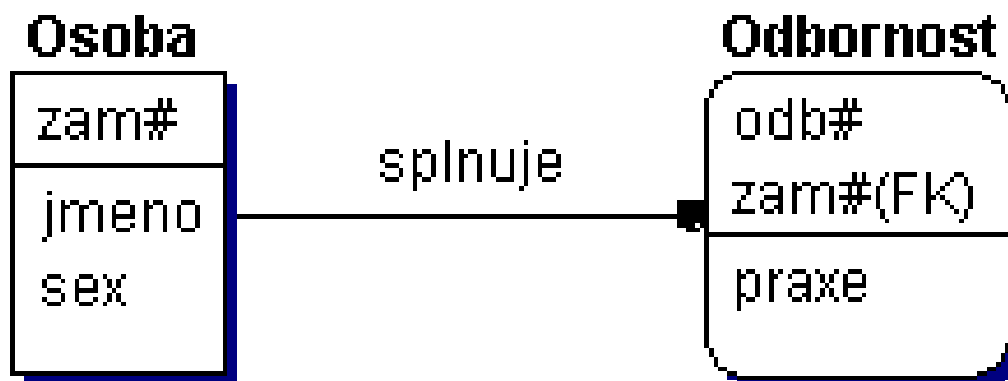
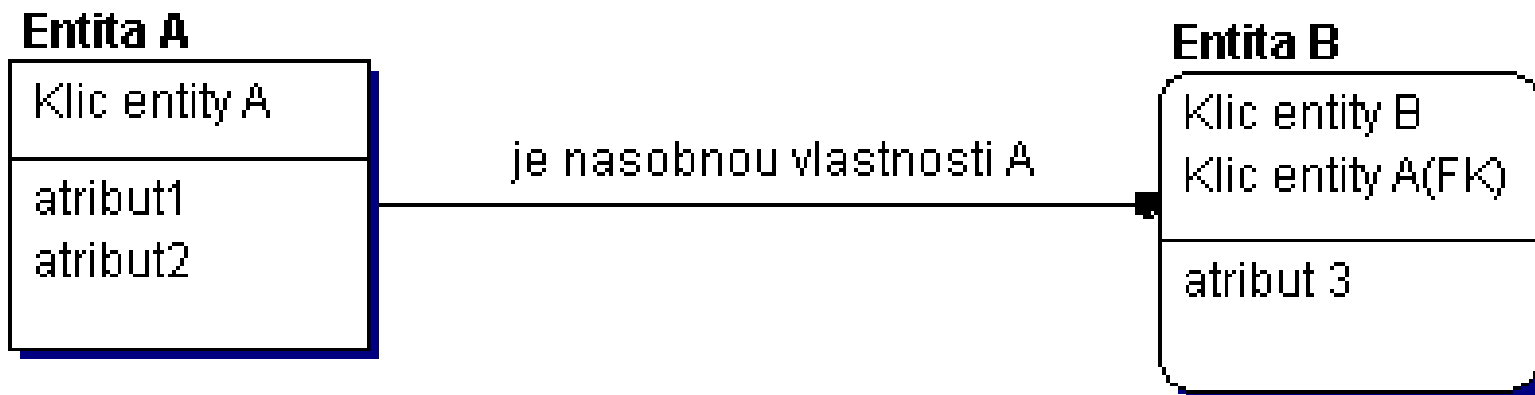


1. normální forma





1. normální forma





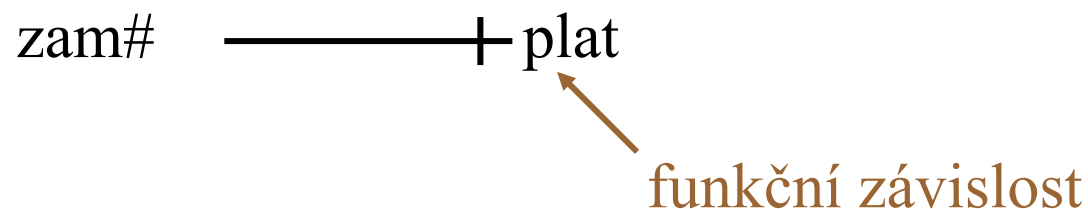
2. normální forma

Požaduje plnou funkční závislost všech neklíčových atributů na celém klíči.

Datová položka B záznamu R funkčně závisí na datové položce A z R, pokud v libovolném časovém okamžiku každé hodnotě A odpovídá nejvýše jedna sdružená hodnota B v záznamu R.

A identifikuje B.

Př.: Zaměstnanec bere plat. Klíč zam# jednoznačně určí plat.



Plná funkční závislost

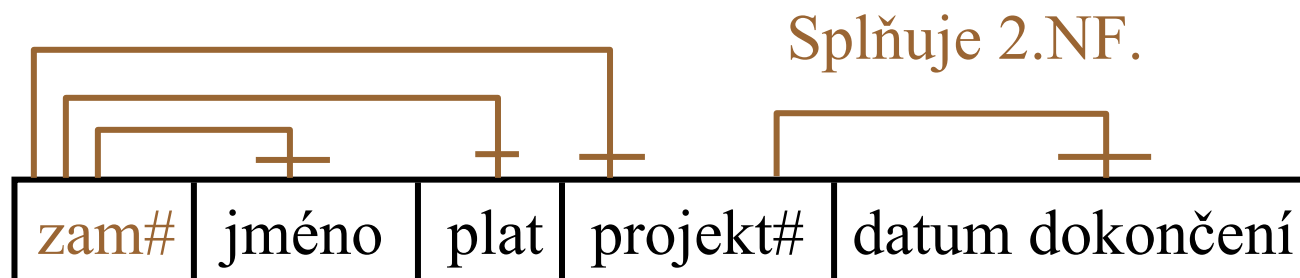


Datová položka nebo kolekce datových položek B záznamu R jsou plně funkčně závislé na kolekci datových položek A záznamu R , pokud je B funkčně závislé na celém A, ale nikoliv na vlastní podmnožině A.

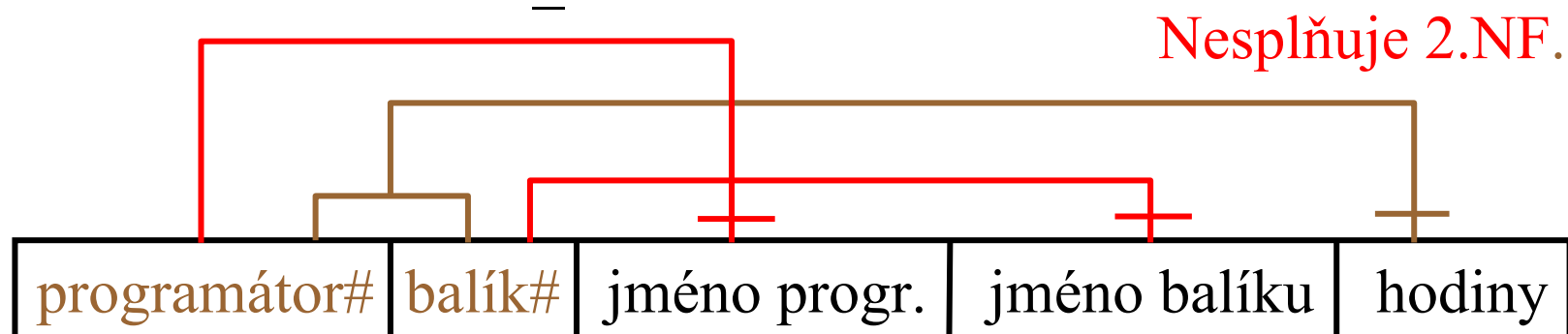


2. Normální forma - příklady

Př.: Entita ZAMESTNANEC

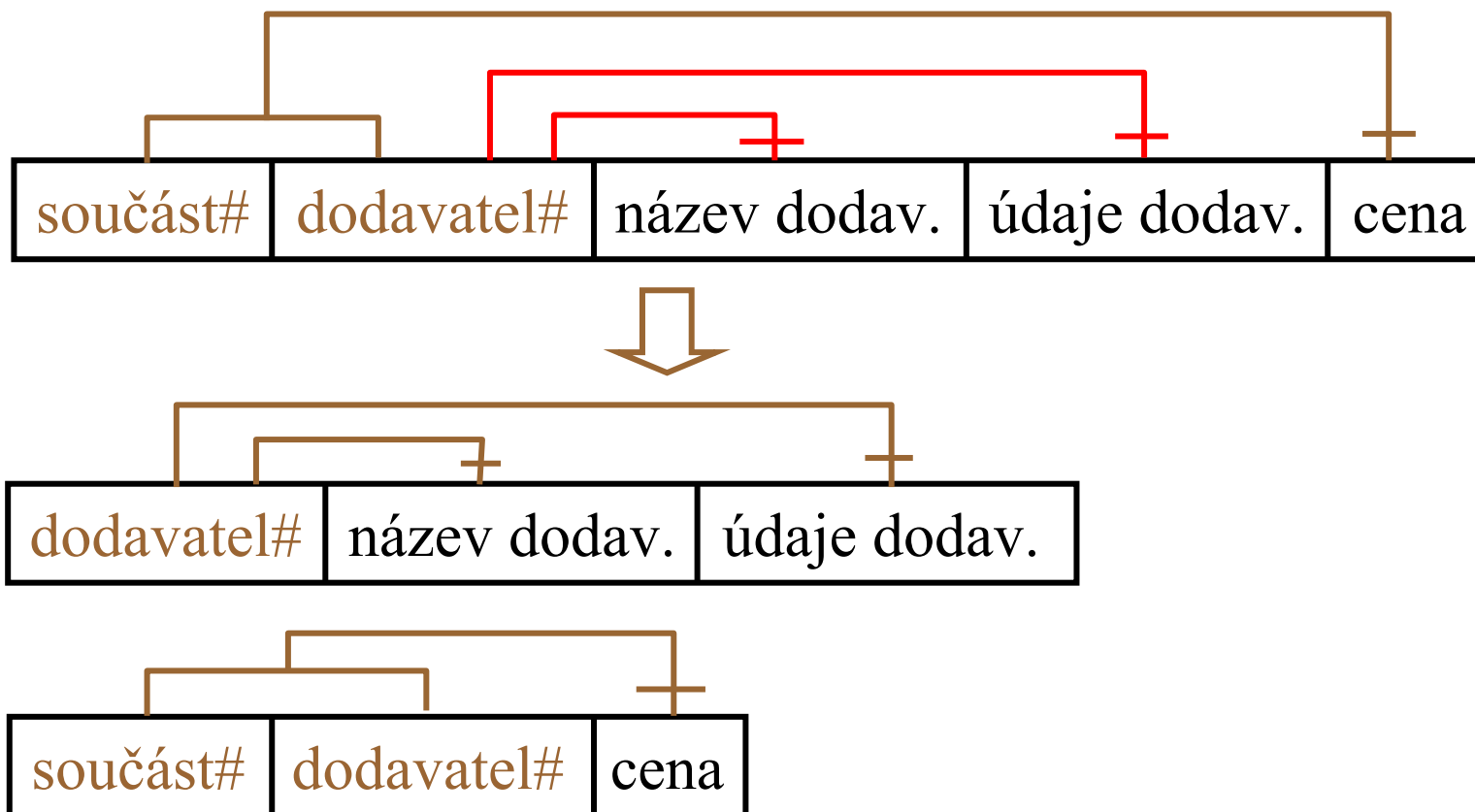


Př.: Entita AKTIVITA_PROGRAMATORA





2. Normální forma - příklad konverze



Problémy „nebytí“ ve 2. normální formě - příklady



Dokud nám dodavatel nedodá součást, nemůžeme zapsat jeho adresu a další údaje.

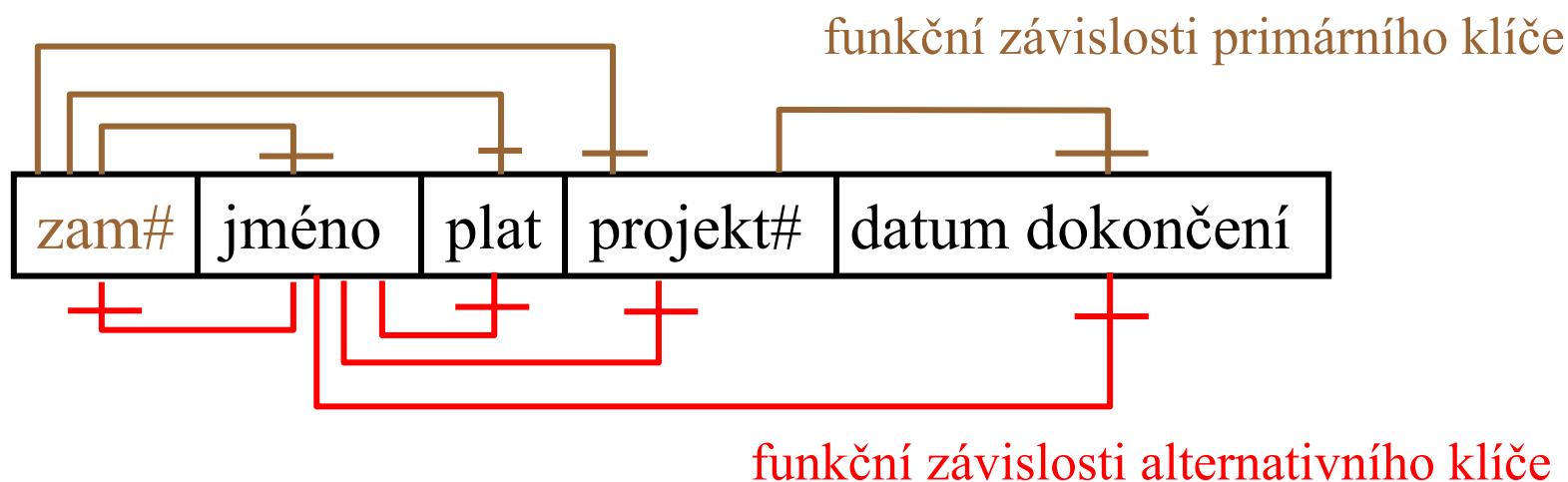
Pokud přestane dodavatel dočasně zásobovat, pak zrušení záznamu o součásti zruší i jeho údaje.

Jakákoliv změna v údajích o dodavatelích je komplikovaná (vyhledání a oprava více záznamů).



2. Normální forma - jiná definice

Př.: Entita ZAMESTNANEC

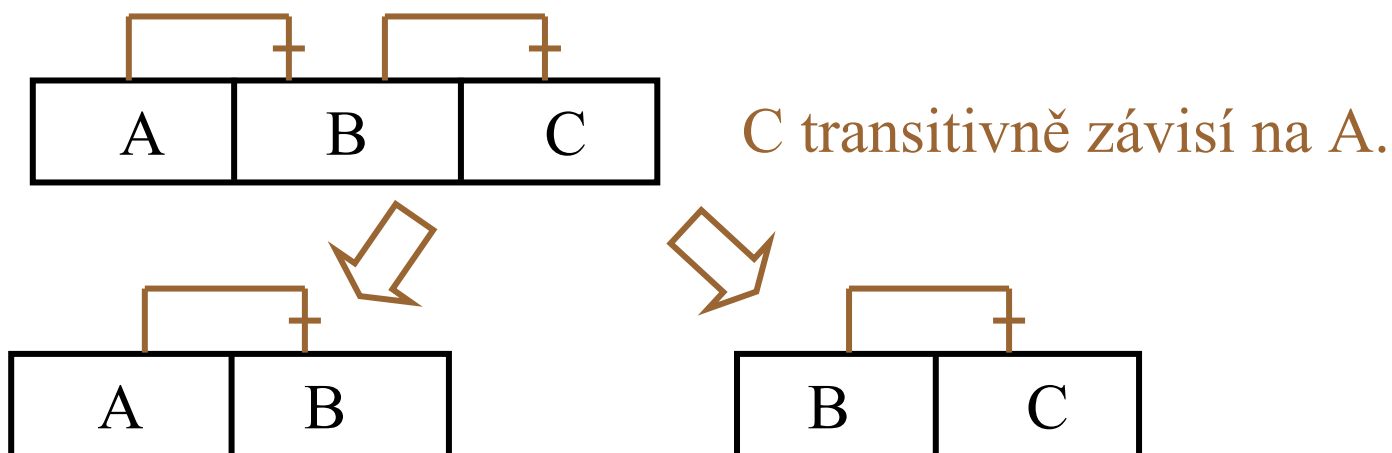


Def.: 2.NF:

Záznam R je ve 2.NF, pokud je v 1.NF a každá neklíčová položka v R je plně funkčně závislá na každém kandidátním klíči R.



3. normální forma - tranzitivní závislost



Def. 3NF (jednoduše):

Záznam je ve 3.NF, pokud je ve 2.NF a každý atribut je funkčně závislý na klíči a pouze na klíči.

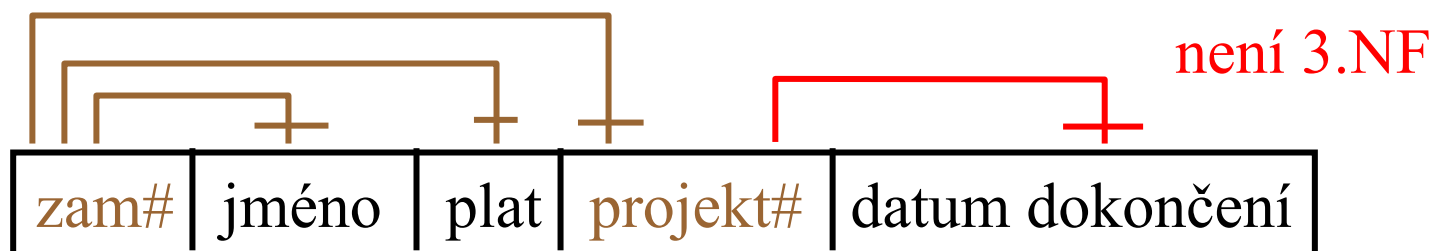
Def. 3.NF (přesněji):

Záznam R je ve 3.NF, pokud je ve 2.NF a každá neklíčová položka R je netranzitivně závislá na každém kandidátním klíči z R.



Problémy „nebytí“ ve 3. normální formě

Př.: ZAMESTNANEC



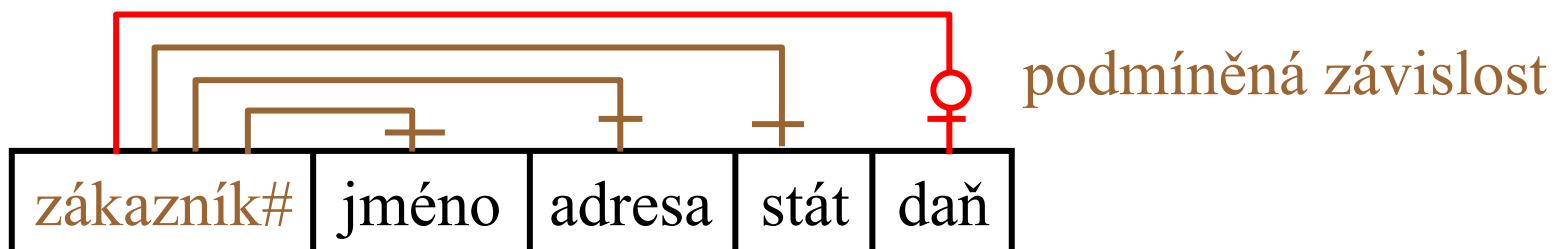
Problémy „nebytí“ ve 3.NF

- Dokud nepřidělíme pracovníky na projekt, nemůžeme zapsat datum ukončení.
- Jestliže všichni opustí projekt, zrušíme veškerou informaci o datu ukončení.
- Změnu data ukončení je nutné provést na mnoha místech.



4. normální forma - podmíněná závislost

4.NF odstraňuje podmíněné funkční závislosti.



Daň strháváme těm, kteří sídlí ve stejném státě jako naše firma.

ZÁKAZNÍCI

zákazník#	jméno	adresa	stát
-----------	-------	--------	------

ZÁKAZNÍCI DOMÁCÍ

zákazník#	daň
-----------	-----



Příklad - Normalizace do 4.NF

[0] SET 003 (Pracovnik) [???]

▶Cislo pracovníka	↑
Jmeno pracovníka	
Cislo oddeleni	
Vedouci oddeleni	
Cislo projektu	
Nazev projektu	
Pocet stravenych hodin	

Indicate key

primární klíč

[0] SET 003 (Pracovnik) [???]

Cislo pracovníka	↑
Jmeno pracovníka	
Cislo oddeleni	
Vedouci oddeleni	
▶Cislo projektu	
▶Nazev projektu	
▶Pocet stravenych hodin	

Indicate repeating set

opakující se skupiny



Příklad - Normalizace do 4.NF

[0] SET 004 (Pracovník) IN?

- Cislo pracovníka ↑
- Cislo projektu
- Nazev projektu
- Pocet stravenych hodin

[2] SET 003 (Pracovník) ????

- Cislo pracovníka ↑
- Jmeno pracovníka
- ▶Cislo oddeleni
- ▶Vedouci oddeleni

tranzitivně závislé atributy

SDW DA Indicate transitive dependent attr.



Příklad - Normalizace do 4.NF

The screenshot shows a database normalization tool interface with three tables:

- [0] SET 004 (Pracovnik)**:
 - Cislo pracovnika
 - Cislo projektu
 - Nazev projektu
 - Pocet stravenych hodin
- [4] SET 005 (Pracovnik)**:
 - Cislo oddeleni
 - Vedouci oddeleni
- [4] SET 003 (Pracovnik)**:
 - Cislo pracovnika
 - Jmeno pracovnika
 - Cislo oddeleni

Arrows point from a thought bubble containing the word "hotovo" to the tables. A status bar at the bottom reads "Set has been normalized".



Příklad - Normalizace do 4.NF

[1] SET 004 (Pracovnik) [???]?

Cislo pracovnika
Cislo projektu
▶ Nazev projektu
Pocet stravenych hodin

atribut závislý
na části klíče

Indicate attr. dependent on part of key

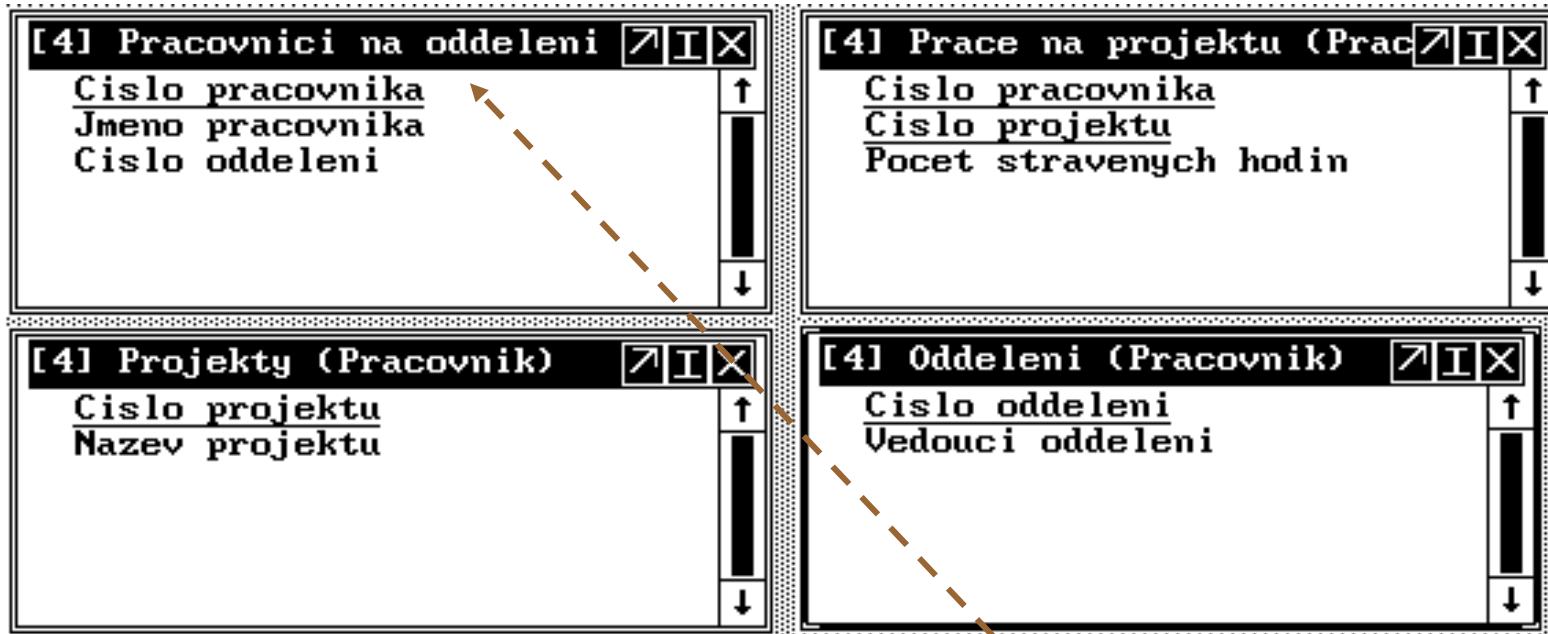
[1] SET 004 (Pracovnik) [???]?

Cislo pracovnika
> Cislo projektu
▶ Nazev projektu
Pocet stravenych hodin

část klíče

Indicate part of key

Příklad - Normalizace do 4.NF



Závěr: Upravené entitní množiny byly vhodně pojmenovány, všechny jsou ve 4.NF. (Pracovník) je jméno výchozí množiny před normalizací.

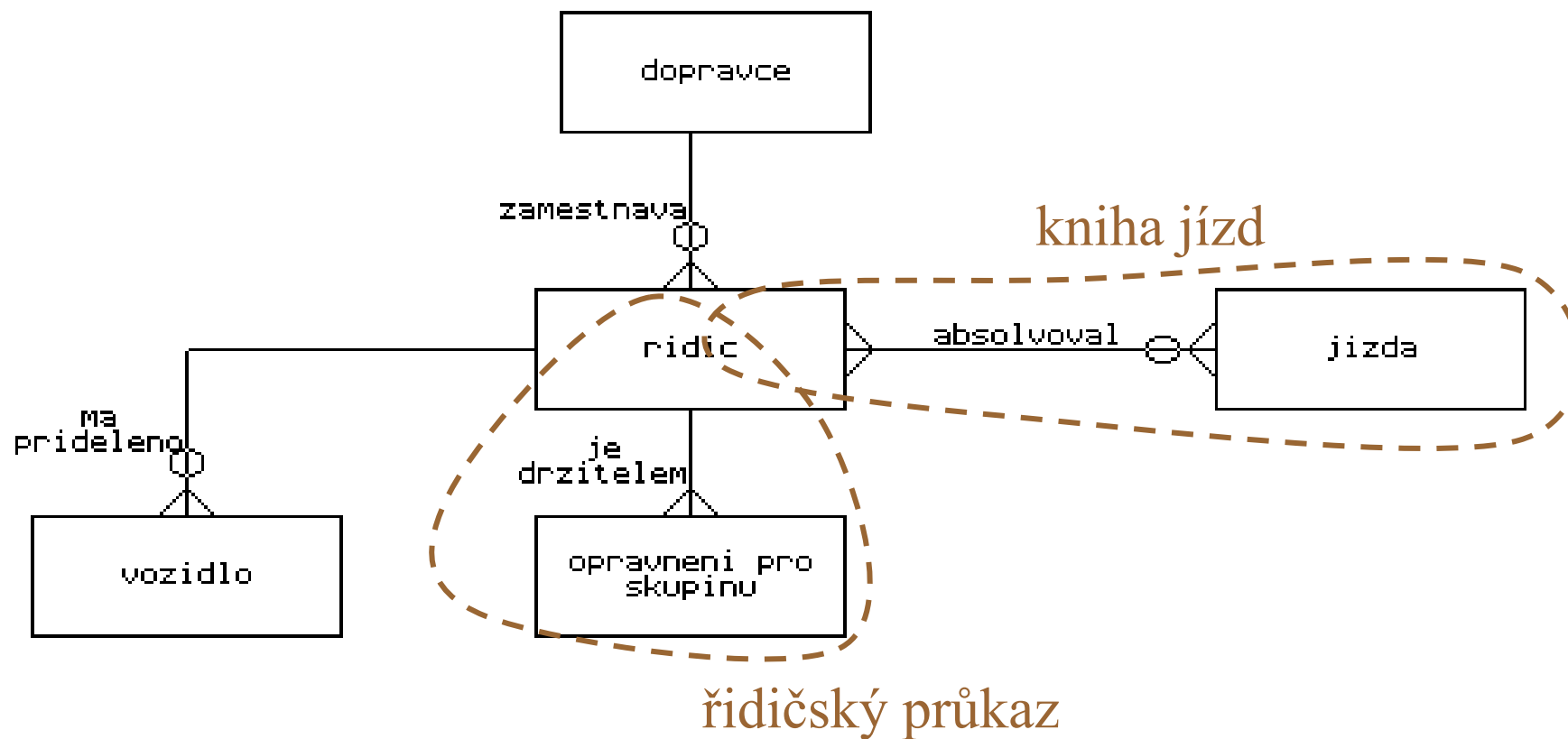


Příklad - Normalizace do 4.NF - Vygenerování ERD

[4] Pracovníci na oddeleni [I] [X] <u>Cislo pracovníka</u> Jmeno pracovníka Cislo oddeleni	[4] Prace na projektu (Prac [I] [X] <u>Cislo pracovníka</u> <u>Cislo projektu</u> Pocet stravenych hodin
[4] Projekty (Pracovník) [I] [X] <u>Cislo projektu</u> Nazev projektu	[4] Oddeleni (Pracovník) [I] [X] <u>Cislo oddeleni</u> Vedouci oddeleni

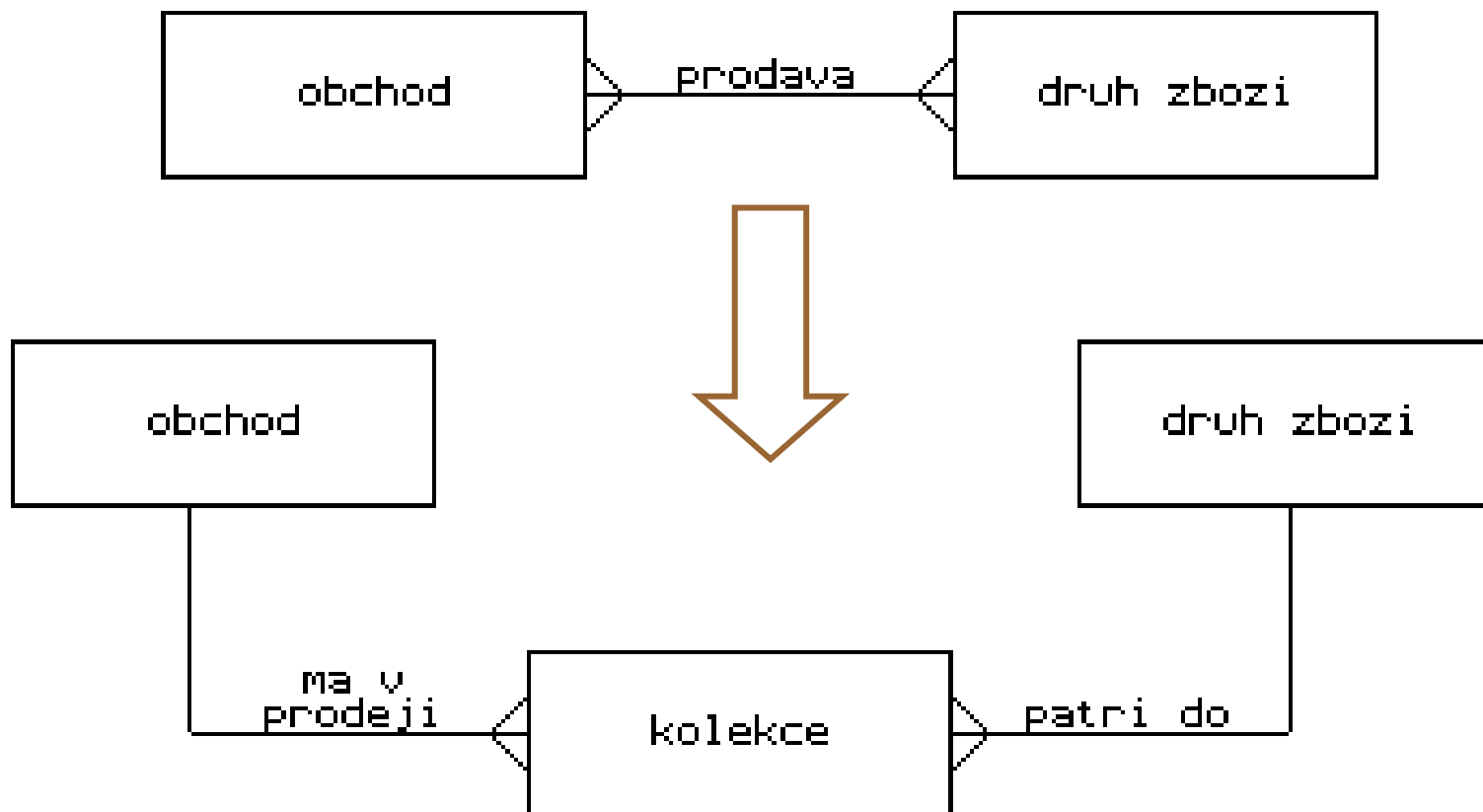


Příklad ERD - Doprava

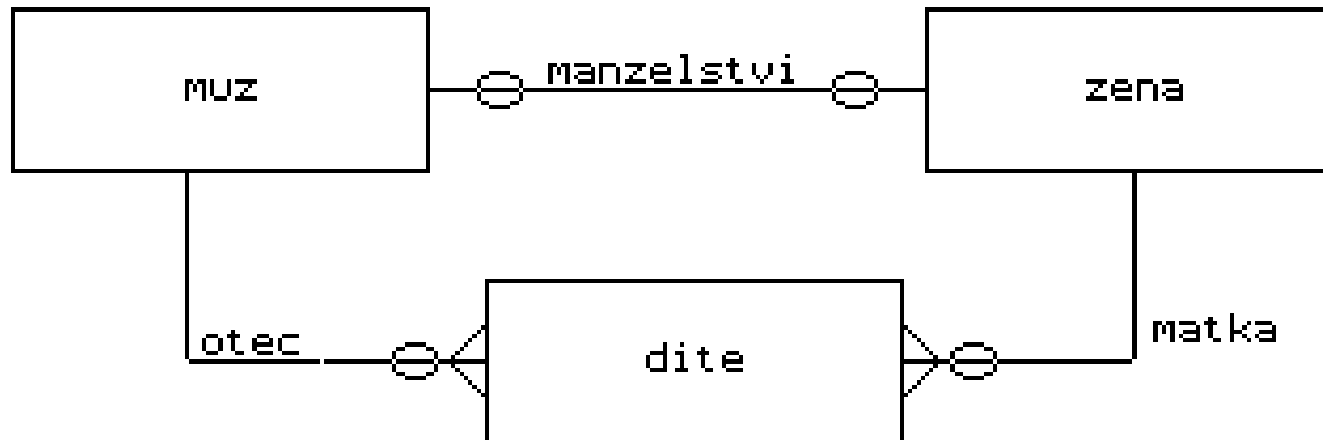




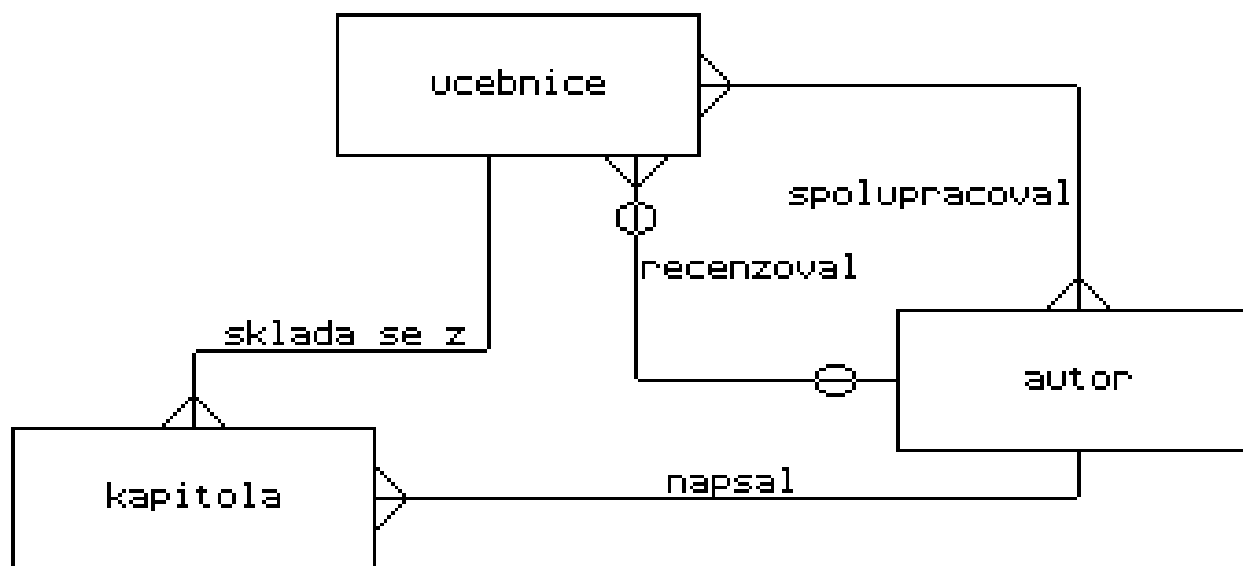
Příklad ERD - Vztahy M x N



Příklad ERD - Osobní záznamy

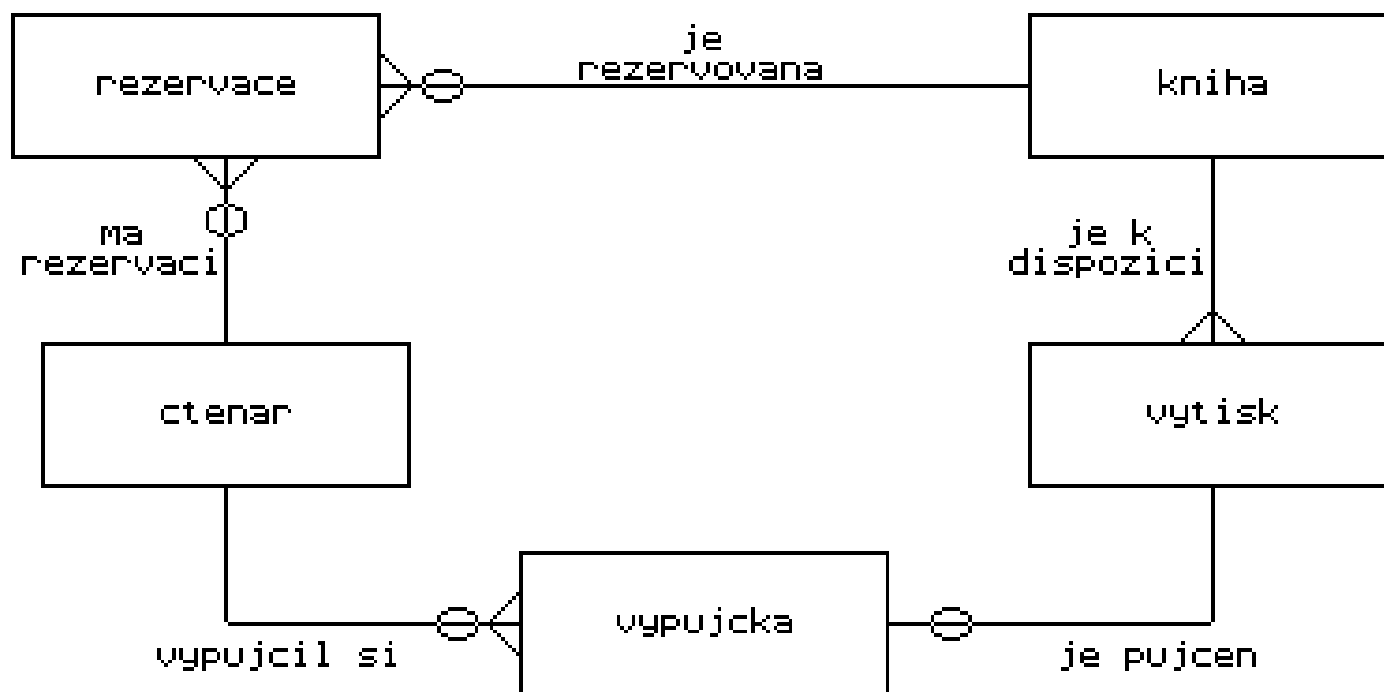


Příklad ERD - Učebnice



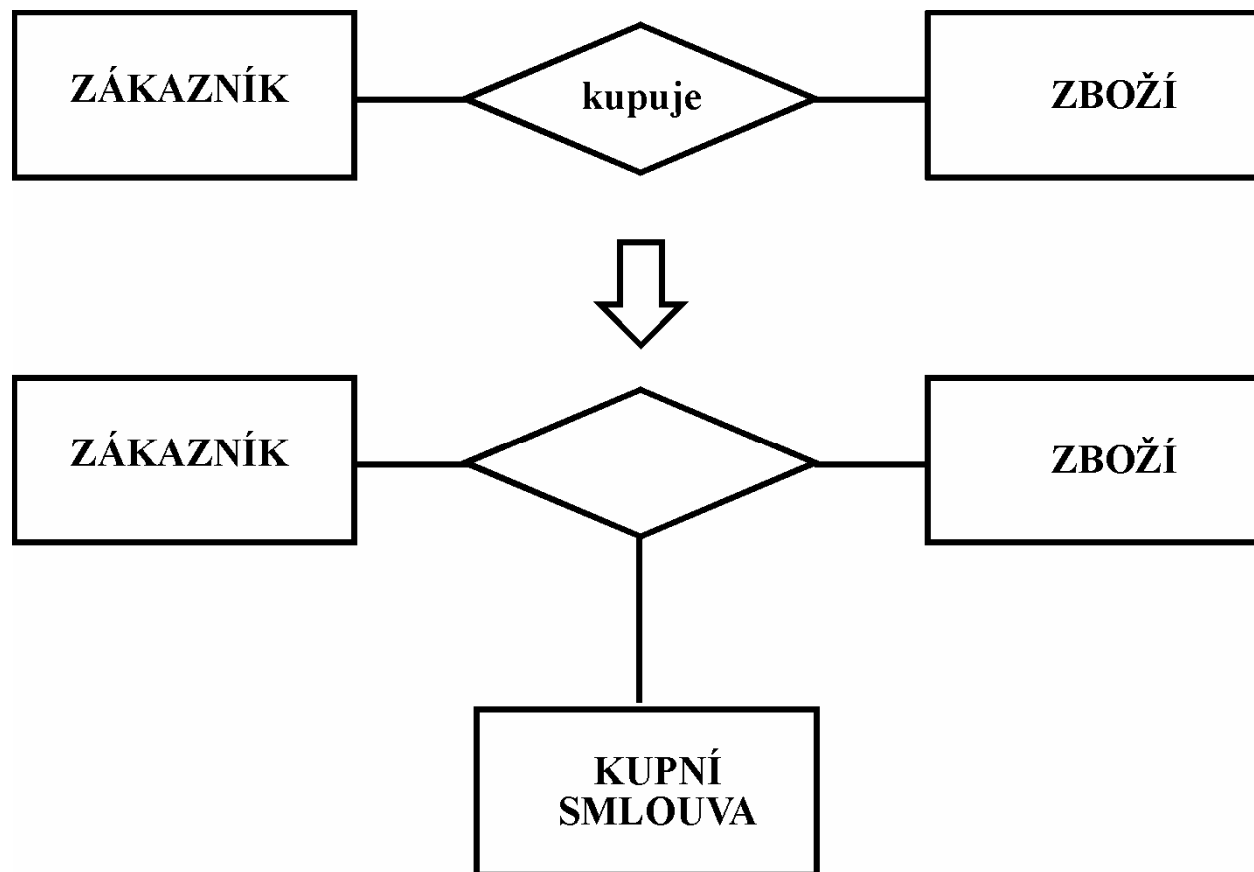


Příklad ERD - Veřejná knihovna





Příklad ERD - Nahrazení vztahu asociovanou entitou



Příklad - Úprava n-ární relace

