

Kapitola 2: Entitně-vztahový model (Entity-Relationship model)

- Množiny entit
- Množiny vztahů
- Otázky návrhu
- Plánování mezí
- Klíče
- E-R diagram
- Rozšířené E-R rysy
- Návrh E-R databázového schématu
- Redukce E-R schématu na tabulky

Množiny entit (entitní množiny)

- *Databáze* může být modelována jako:
 - množina entit
 - vztahy mezi entitami
- *Entita* je objekt, který existuje a je odlišitelný od ostatních objektů.
Např.: nějaká osoba, společnost, událost, rostlina
- *Množina entit* je skupina entit stejného typu, které sdílejí stejné vlastnosti.
Např.: skupina všech osob, firem, stromů

Atributy

- Entita je reprezentována množinou atributů, to jsou popisné vlastnosti všech členů množiny entit.
Např.:
 - zákazník* = (*jméno*, *rodné_číslo*, *ulice*, *město*)
 - účet* = (*číslo účtu*, *zůstatek*)
 - tj. *entita* = (*atributy*, ...)
- *Doména* – množina povolených hodnot pro každý atribut
- Typy atributů:
 - *Jednoduché* atributy (*jméno*) a *složené* atributy (*datum*).
 - Atributy s *jednoduchou hodnotou* (*single-valued*) (např. *jméno*) a s *násobnou hodnotou* (*vícehodnotové*) (*multi-valued*) (např. *telefonní čísla*)
 - *Nulové* atributy (např.: *nemá telefon*) (**null**)
 - *Odvozené* atributy (např.: *věk*)

Množiny vztahů

- *Vztah* je spojení mezi několika entitami

Např.:

Novák	<i>vkladatel</i>	A-102
entita <i>zákazník</i>	množina vztahů	entita <i>účet</i>

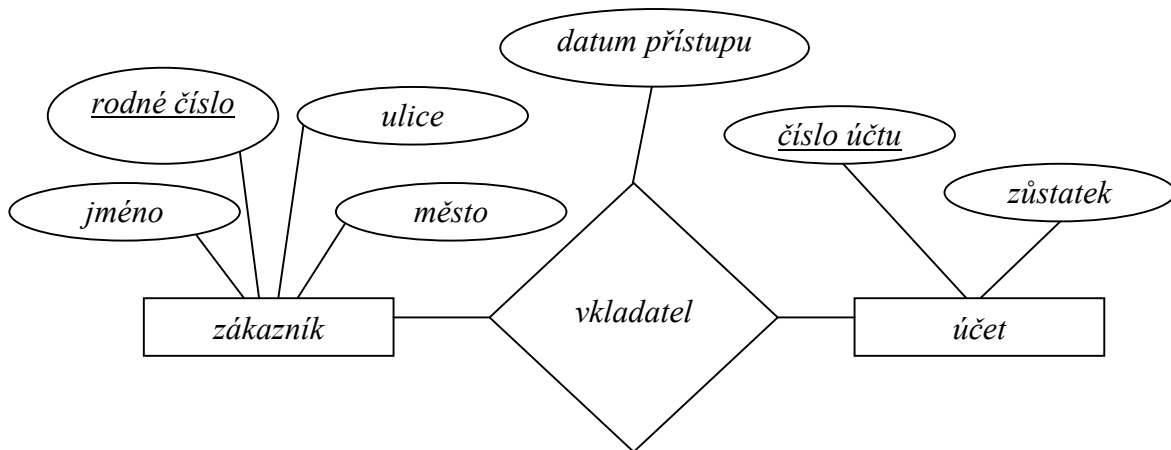
- *Množina vztahů* je matematická relace mezi $n \geq 2$ entitami, každá je braná z konkrétní množiny entit

$$\{(e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$$

kde (e_1, e_2, \dots, e_n) je *vztah*, e_1, e_2, \dots, e_n jsou entity a E_1, E_2, \dots, E_n entitní množiny např.:

$$(Novák, A-102) \in vkladatel$$

- Množina vztahů může mít také atributy. Např. množina vztahů *vkladatel* mezi množinami entit *zákazník* a *účet* může mít atribut (*poslední*) *datum přístupu*.

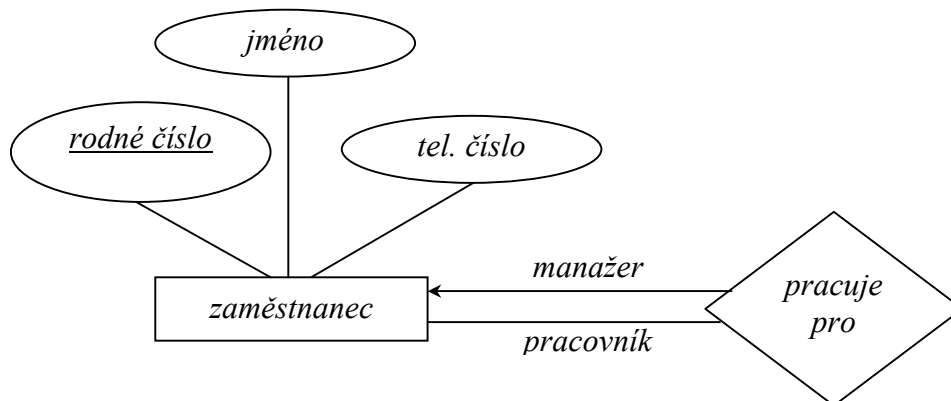


Stupeň vztahu

- Ukazuje počet množin entit, které jsou součástí množiny vztahů.
- Množiny vztahů, které zahrnují 2 množiny entit, se nazývají *binární* (nebo stupně 2). Obecně, většina vztahů v databázovém systému je binární.
- Množiny vztahů mohou zahrnovat více než 2 množiny entit. Např. množiny entit *zákazník*, *půjčka* a *pobočka* mohou být spojeny ternární (stupně 3) množinou vztahů *CLB*.

Role

Množiny entit u vztahů nemusí být rozdílné



- Popisky *manažer* a *pracovník* jsou nazývány *role*; specifikují, jak na sebe entity typu *zaměstnanec* vzájemně působí přes množinu vztahů *pracuje pro*.
- Role jsou v E-R diagramech znázorněny popisky u čar, které spojují kosočtverce s obdélníky.
- Popisky rolí jsou dobrovolné a jsou používány pro zvýraznění *sémantiky* (významu) vztahu

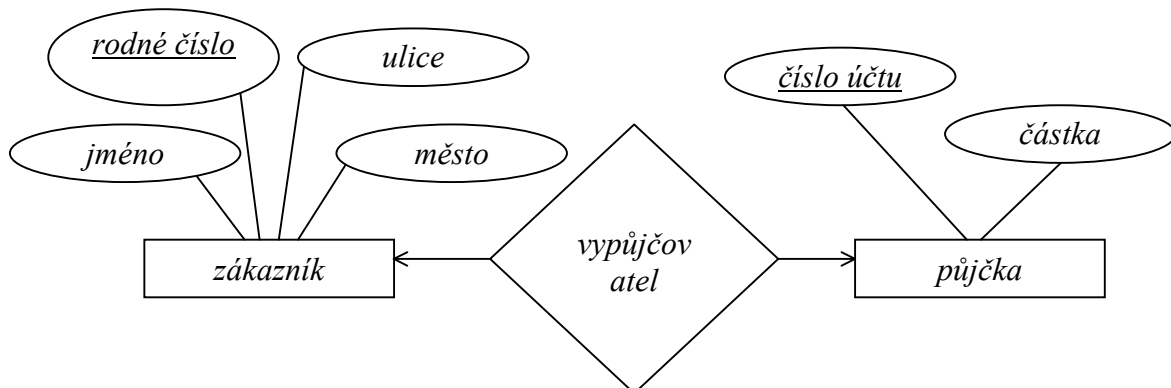
Otázky návrhu (Design Issues)

- Použití *entitní množiny* vs. *atributu*
Výběr závisí zejména na struktuře podniku a na významu daného atributu. Např.: *student – ročník*.
- Použití *entitní množiny* vs. *množiny vztahů*
Možným vodítkem může být sestavení množiny vztahů pro popis akce, která se odehrává mezi entitami. Např.: *vkladatel*
- *Binární* vs. *n-ární* množiny vztahů
Přestože je možné nahradit ne-binární (*n-ární*, pro $n > 2$) množinu vztahů několika různými binárními množinami vztahů, *n-ární* ukazuje mnohem jasněji, že několik entit je součástí jednoho vztahu. Např. *zákazník, půjčka, pobočka*.

Četnosti vztahů (Mapping Cardinalities)

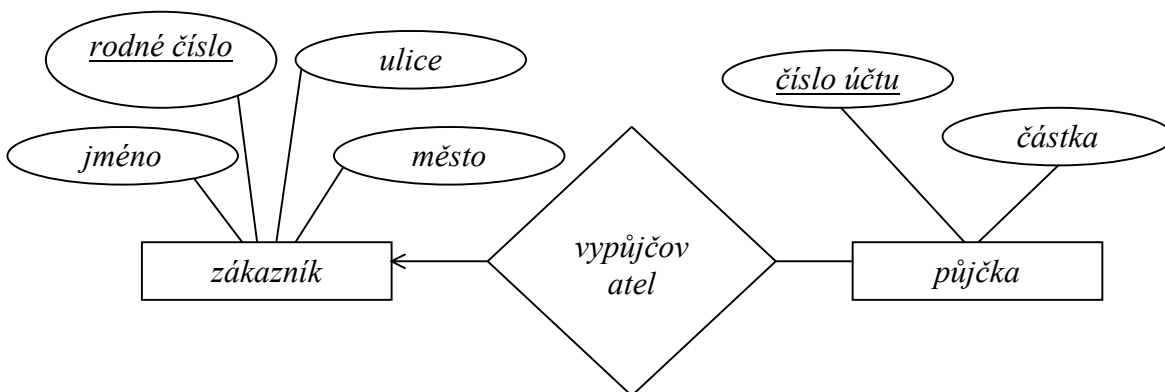
- Označuje počet entit, se kterými mohou být ostatní entity propojeny pomocí množiny vztahů.
- Nejužitečnější je v popisu binárních množin vztahů.
- Pro binární množinu vztahů musí být četnost jednoho z následujících typů:
 - jedna na jednu
 - jedna na mnoho
 - mnoho na jednu
 - mnoho na mnoho
- Mezi těmito typy rozlišujeme kreslením buď šipky (\rightarrow) značící *jeden* nebo normální čáry ($-$) značící *mnoho* mezi množinou entit a vztahů.

Vztah jedna na jednu (One-to-one)

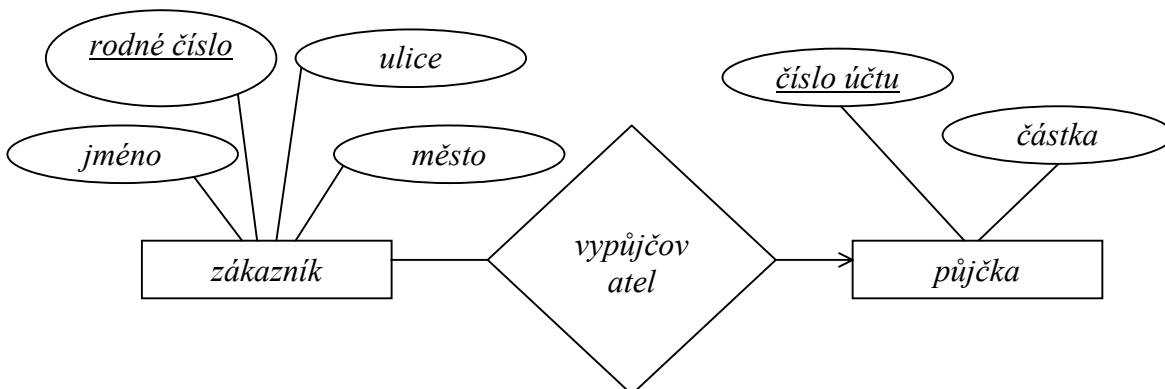


- Zákazník je spojen s nejvýše jednou půjčkou vztahem *vypůjčovatel*.
- Půjčka je spojena s nejvýše jedním zákazníkem vztahem *vypůjčovatel*.

Vztahy jedna na mnoho a mnoho na jednu (One-to-many a many-to-one)

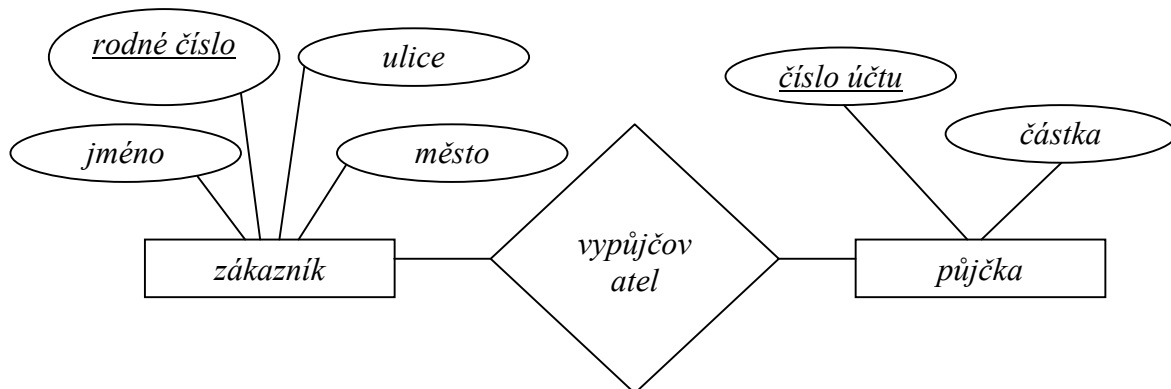


- Ve vztahu jedna na mnoho je půjčka spojena s nejvýše jedním zákazníkem a zákazník je spojen s žádnou nebo několika půjčkami vztahem *vypůjčovatel*.



- Ve vztahu mnoho na jednu je půjčka spojena s žádným nebo několika zákazníky a zákazník je spojen s nejvýše jednou půjčkou vztahem *vypůjčovatel*.

Vztah mnoho na mnoho (Many-to-many)



- Zákazník je spojen s žádnou nebo několika půjčkami vztahem *vypůjčovatel*
- Půjčka je spojena s žádným nebo několika zákazníky vztahem *vypůjčovatel*

Existenční závislost

- Závisí-li existence entity *x* na existenci entity *y*, pak *x* se nazývá existenčně závislé (*existence dependent*) na *y*.
 - *y* je *dominantní entita* (v příkladu níže *půjčka*)
 - *x* je *podřízená entita* (v příkladu níže *splátka*)



- Je-li entita *půjčka* smazána, pak všechny s ní spojené entity *splátka* musí být smazány také.

Klíče

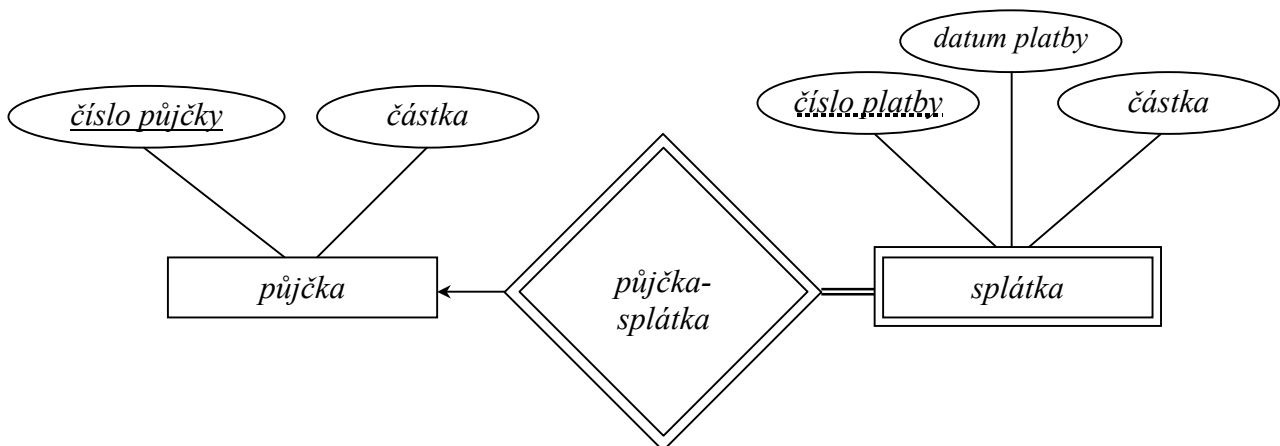
- *Super klíč* množiny entit je množina jednoho nebo více atributů, jejichž hodnoty jednoznačně určují entitu (tedy *klíč* je *podmnožina atributů* – např. všechny atributy).
- *Kandidátní klíč* množiny entit je minimální super klíč.
 - *rodné číslo* je kandidátní klíč entity *zákazník*
 - *číslo účtu* je kandidátní klíč je kandidátní klíč entity *účet*
- Protože může existovat několik kandidátních klíčů, jeden z nich je vybrán jako *primární klíč*.
- Sjednocení primárních klíčů zúčastněných entitních množin určuje kandidátní klíč pro množinu vztahů.
 - při výběru *primárního klíče* musíme dávat pozor na četnosti vztahů a sémantiku množiny vztahů
 - např. (*rodné číslo*, *číslo účtu*) je primární klíč množiny vztahů *kladatel*

Komponenty E-R diagramu

- **Obdélníky** reprezentují množiny entit.
- **Elipsy** reprezentují atributy.
- **Kosočtverce** reprezentují množiny vztahů.
- **Čáry** spojují atributy s množinami entit a množiny entit s množinami vztahů.
- **Dvojité elipsy** reprezentují vícehodnotové atributy.
- **Čárkované elipsy** označují odvozené atributy.
- Atributy primárního klíče jsou podtržené.

Slabé množiny entit

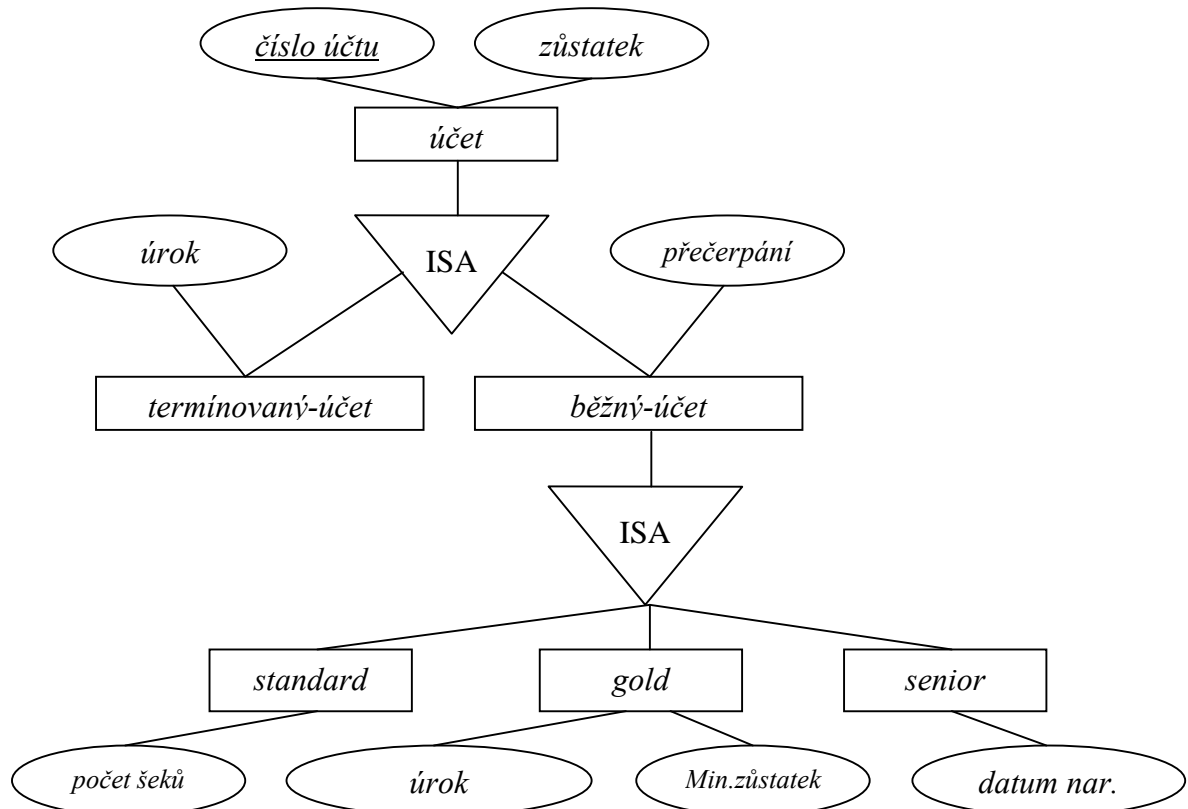
- Množina entit, která nemá primární klíč, se nazývá *slabá množina entit*.
- Existence slabé množiny entit závisí na existenci silné množiny entit; musí být spojena se silnou množinou vztahem mnoho na jednu.
- *Diskriminátor (parciální klíč)* slabé množiny entit je množina atributů, která od sebe odlišuje entity slabé množiny
- Primární klíč slabé množiny je tvořen *primárním klíčem silné množiny*, na níž je tato množina závislá a *parciálním klíčem* této slabé množiny.



- Slabé množiny entity znázorňujeme dvojitým obdélníkem.
- Parciální klíč slabé množiny entit se podtrhává přerušovanou čarou.
- *číslo splátky* – parciální klíč množiny entit *splátka*
- Primární klíč pro množinu *splátka* je (*číslo půjčky*, *číslo splátky*)

Specializace

- Tvoříme podskupiny v množině entit, které jsou různé od ostatních entit v množině (proces seshora dolů)
- Tyto podskupiny se stávají množinami entit nižší úrovně, které mají atributy nebo jsou součástí množin vztahů, které se nepromítají do množiny vztahů vyšší úrovně.
- Znázorňujeme trojúhelníkovou komponentou označenou ISA (*termínovaný vklad „je (is a)“ účet*)



Generalizace (Zobecnění)

- Kombinujeme několik množin entit, které sdílejí stejné rysy do množiny entit vyšší úrovně (proces zezdola nahoru)
- Specializace a generalizace jsou jednoduše vzájemně inverzní; jsou reprezentovány E-R diagramem stejným způsobem.
- **Dědičnost atributů** – entita nižší úrovně dědí všechny atributy a účasti ve vztazích z množiny entit vyšší úrovně.

Omezení pro specializaci/generalizaci

Omezení na to, které entity mohou být prvky entitní množiny na nižší úrovni:

- omezení daná nějakou *podmínkou*
- omezení definovaná *uživitelem* (pro každou entitu zvlášť)

Omezení na to, jestli může entita patřit jen do jedné nebo do více entitních množin na jedné úrovni jedné generalizace:

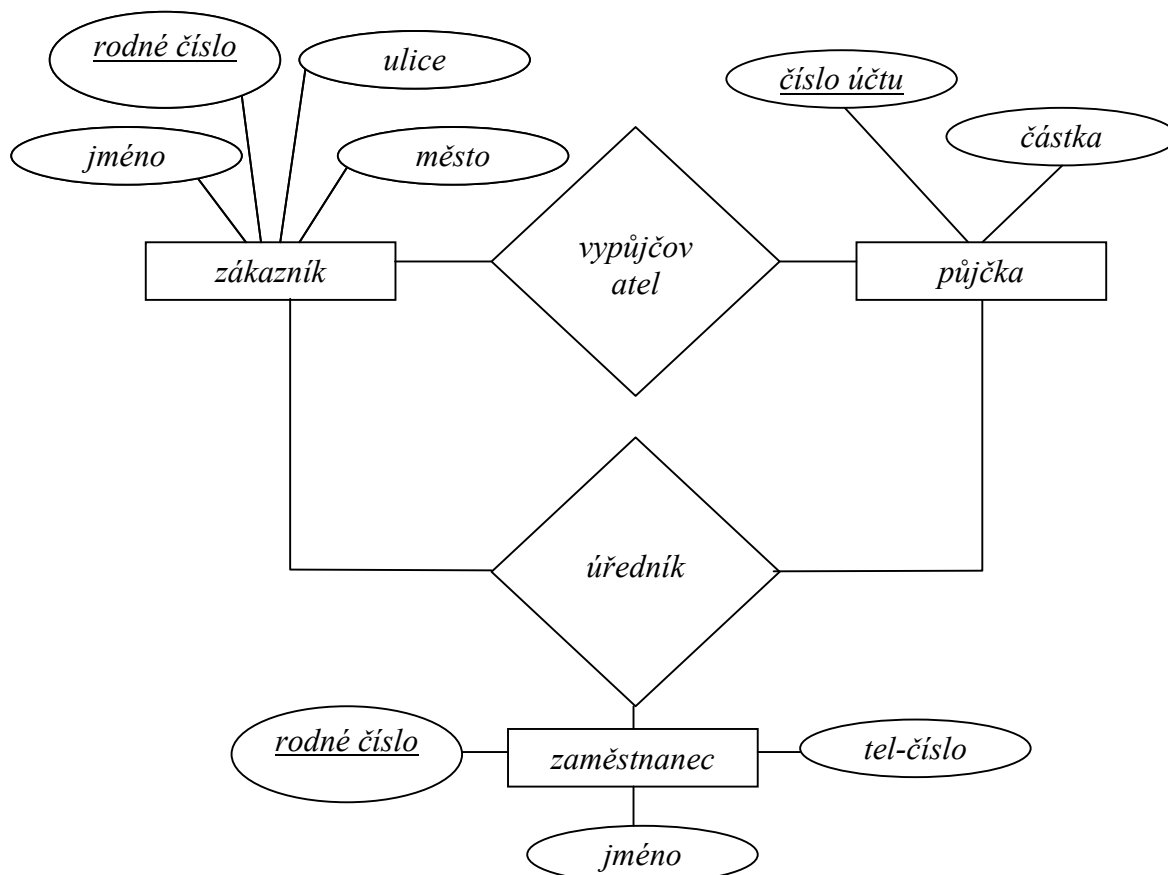
- disjunktí
- překrývající se

Omezení na to, jestli každá entita z vyšší třídy musí nebo nemusí patřit do jedné z entitních množin na nižší úrovni:

- úplná specializace
- částečná specializace

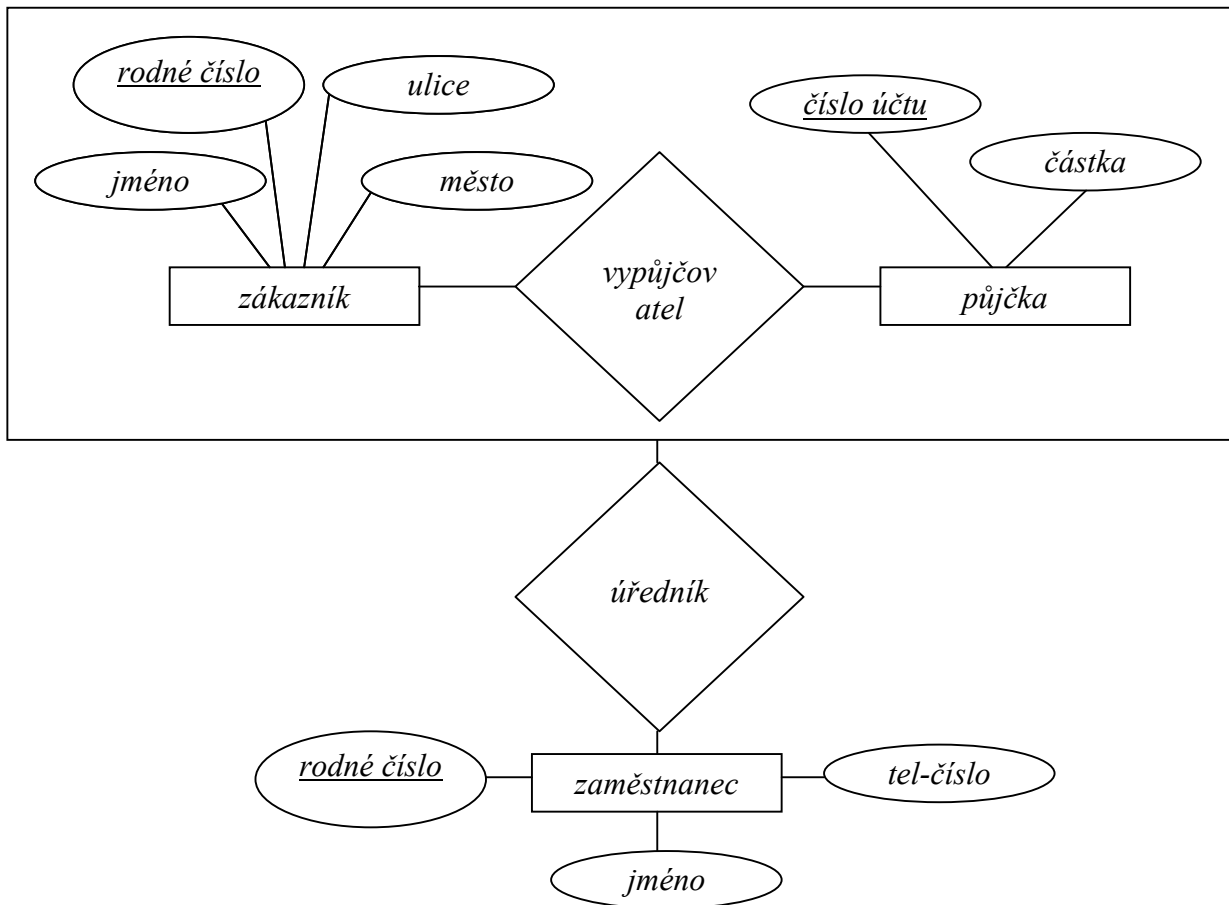
Agregace

- Dlužník (*zákazník*) může být kontrolován *úředníkem*.



- Množiny vztahů *vypůjčovatel* a *úředník* reprezentují stejnou informaci.
- Tuto redundanci eliminujeme agregací
 - Se *vztahem* zacházíme jako s abstraktní entitou
 - Umožňuje vztahy mezi vztahy
 - Abstrakce vztahu do nové entity

- Následující diagram reprezentuje:
 - Zákazník si vezme půjčku
 - Zaměstnanec může být úředníkem pro dvojici zákazník-půjčka



Rozhodnutí o návrhu E-R schématu

- Použití atributu nebo množiny entit pro reprezentaci objektu.
- Je význam reálného světa lépe vyjádřen entitní množinou nebo množinou vztahů?
- Použití ternárního vztahu vs. páru binárních vztahů.
- Použití silných nebo slabých množin entit.
- Použití generalizace – přispívá k modularitě návrhu.
- Použití agregace – můžeme zacházet s agregovanou množinou entit jako s jednotkou bez ohledu na podrobnosti a její vnitřní strukturu.

Převod E-R schématu na tabulky

- Primární klíče umožňují vyjádřit množiny entit a vztahů jako tabulky reprezentující obsah databáze.
- Databáze, která odpovídá E-R diagramu, může být reprezentována jako kolekce tabulek.
- Pro každou *množinu entit* a *vztahů* je jedinečná tabulka, která je spojená se jménem příslušné množiny entit nebo vztahů.
- Každá tabulka má počet sloupců odpovídající atributům dané množiny entit/vztahů. Sloupce mají jedinečná *jména* v rámci jedné tabulky.
- Převod E-R diagramu na tabulky je základ pro odvozování návrhu relační databáze z E-R diagramu.

Reprezentace množin entit tabulkami

- Silná množina entit se převede na tabulku se stejnými atributy.

<i>jméno</i>	<i>rodné číslo</i>	<i>ulice</i>	<i>město</i>
Starý	800327/6655	Lidická	Brno
Slavík	645326/1258	Úzká	Klatovy
Novák	891117/1111	Lidická	Brno

Tabulka *zákazník*

- Slabá množina entit se převede na tabulku, která obsahuje i sloupec pro primární klíč identifikační silné entitní množiny.

<i>číslo půjčky</i>	<i>číslo splátky</i>	<i>datum splátky</i>	<i>částka splátky</i>
L-17	5	10.5.1996	50
L-23	11	17.5.1996	75
L-15	22	23.5.1996	300

Tabulka *splátka*

Reprezentace množin vztahů tabulkami

- Množina vztahů *mnoho na mnoho* je reprezentována jako tabulka se sloupci pro primární klíče dvou zúčastněných entitních množin a sloupce pro popisné atributy množiny vztahů.

<i>rodné číslo</i>	<i>číslo účtu</i>	<i>datum přístupu</i>
...

Tabulka *kladatel*

- Tabulka odpovídající množině vztahů spojující slabou množinu entit s její identifikační silnou množinou je zbytečná. Tabulka *splátka* již obsahuje informace, které by se objevily v tabulce *půjčka-splátka* (tj. sloupce *číslo půjčky* a *číslo splátky*)

Reprezentace generalizace tabulkami

- Metoda č. 1: Sestrojíme tabulku pro nadřazenou entitu *účet*. Sestrojíme tabulku pro každou entitní množinu, která odvozena (na nižší úrovni) – zahrneme do těchto tabulek i primární klíč generalizované množiny.

tabulka	atributy tabulky
<i>účet</i>	<i>číslo účtu, zůstatek, typ účtu</i>
<i>termínovaný účet</i>	<i>číslo účtu, úroková sazba</i>
<i>běžný účet</i>	<i>číslo účtu, překročení</i>

- Metoda č. 2: Sestrojíme tabulku jen pro každou entitní množinu na nižší úrovni.

tabulka	atributy tabulky
<i>termínovaný účet</i>	<i>číslo účtu, zůstatek, úroková sazba</i>
<i>běžný účet</i>	<i>číslo účtu, zůstatek, překročení</i>

Metoda č. 2 netvoří žádnou tabulku pro generalizovanou entitu *účet*.

Vztahy odpovídající agregaci

zákazník

<u><i>jméno</i></u>	<u><i>rodné číslo</i></u>	<i>ulice</i>	<i>město</i>
---------------------	---------------------------	--------------	--------------

půjčka

<u><i>číslo půjčky</i></u>	<i>částka</i>
----------------------------	---------------

vypůjčovatel

<u><i>rodné číslo</i></u>	<i>číslo půjčky</i>
---------------------------	---------------------

zaměstnanec

<u><i>rodné číslo</i></u>	<i>jméno</i>	<i>tel. číslo</i>
---------------------------	--------------	-------------------

úředník

<u><i>rodné číslo (zaměstnanec)</i></u>	<u><i>rodné číslo (zákazník)</i></u>	<u><i>číslo půjčky</i></u>
---	--------------------------------------	----------------------------