

Objektové metody modelování

Základní pojmy objektového modelování
třídy a objekty, diagram tříd

Přednáška 3



**SLEZSKÁ
UNIVERZITA**

OBCHODNĚ PODNIKATELSKÁ
FAKULTA V KARVINĚ

RNDr. Zdeněk Franěk, Ph.D.

Ing. Tomáš Barčák, Ph.D.

Základní pojmy – objekty

➤ Co je to objekt?

Definice: Objekt je seskupení dat a funkcionality, které jsou spolu spojeny za účelem plnění soudržné množiny zodpovědností

➤ Objekt má:

- identitu, vlastnosti (atributy), chování (je realizováno metodami) a jedinečnou zodpovědnost (dovednost)



Základní pojmy – objekty

- Objekt poskytuje služby pomocí operací
 - Rozhraní objektu je množinou operací, které nabízí pro jiné objekty (nebo externí agenty)
- Objekt je černá skříňka, která nabízí služby svým klientům
- Objekty spolu komunikují předáváním zpráv:
 - Eliminace datových duplicit
 - Zprávy mohou být vykonány formou vykonání funkcí, znalost identity
 - Komunikace objektů pomocí operací jen definovaných v rozhraní !

Co je to třída

- Základní předpoklad – návrh modelu tříd (Class model), který v podstatě nezobrazuje jednotlivé objekty, ale šablonu-předpis pro vytvoření objektů = třída objektů

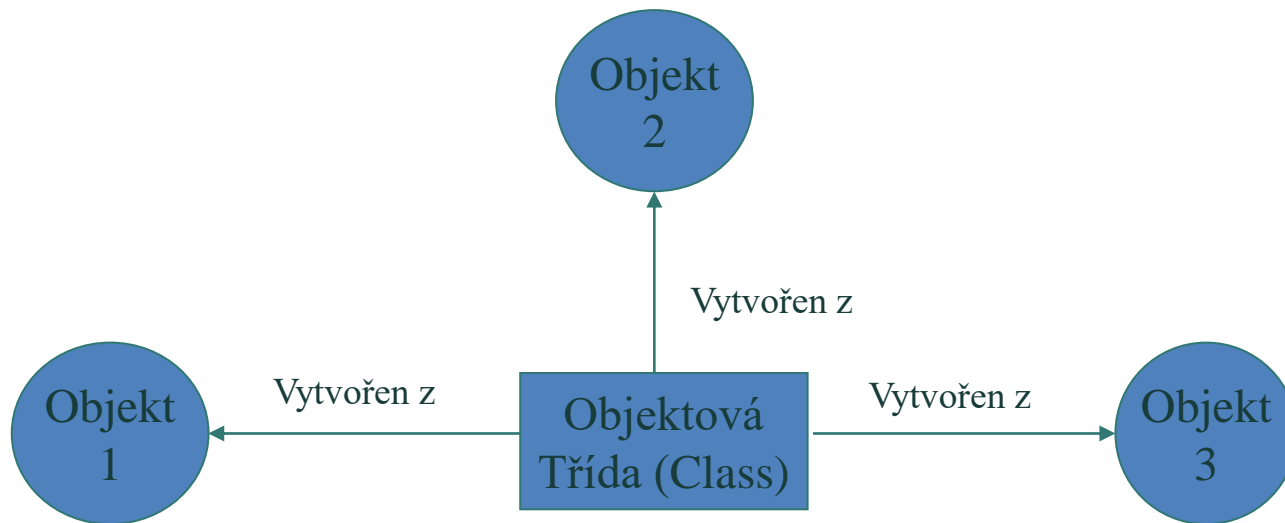
Vztah mezi třídou a objekty:

- Třída je to co navrhujeme a programujeme
- Objekty jsou to, co vytváříme (ze třídy) při běhu aplikace
- Každý objekt má jiný identifikátor a jiný stav v čase, což znamená jiné hodnoty v jeho proměnných.

Úvod do objektového modelování a jazyka UML



Vztah mezi třídou a objekty – grafické znázornění:



Struktura tříd

- Struktury tříd jsou založeny na dvou principech:
 - a) **Zodpovědnost třídy** je jedním z klíčových faktorů objektově orientované analýzy a návrhu a znamená, že námi definovaný objekt nese zodpovědnost za danou problematiku (tj. „to o čem uchovává informace a chování“) a žádný jiný objekt se nesmí „plést do této zodpovědnosti“.
 - b) **Zapouzdření třídy** chápeme jako skrývání interních stavů a jejich nabízení prostřednictvím metod.
 - Atribut tříd je nositel informací o objektu
 - Název atributu (např. jméno)
 - Formát atributu (např. string)
 - Viditelnost (Public, Private, Protected)
-



Struktura tříd

- Operace tříd
 - chování objektu je definováno operacemi
 - aktualizací operace vykonávají operace s daty
 - operace typu interface poskytují rozhraní k jiným objektům
 - charakteristika operací je dána názvem, seznamem parametrů a návratovými hodnotami, tzv. signaturou
 - signatura musí být jednoznačná a unikátní
 - z analytického pohledu vystihuje co daná operace vykonává (např. najdi jméno, aktualizuj hodnoty, přidej nový záznam, vypočítej stipendium ...)

Základní pojmy – objekty a třídy

- Atributy objektu vyjadřují statické datové vlastnosti
 - atributy jsou zapouzdřeny uvnitř objektu, jsou skryty jiným objektům
 - přístup k atributům je možný jen zasláním zprávy, která vyvolá operaci
 - jinak vyjádřeno: s atributy mohou manipulovat jenom metody daného objektu
- Objekty jsou organizovány ve **třídách** sdružující jejich vlastnosti

Základní pojmy – objekty a třídy

- **Třída** představuje šablonu (stupeň řízení) pro skupinu instancí (příslušnost), které nazýváme objekty
- Šablona popisuje vnitřní strukturu objektu
- Objekty stejné třídy mají stejné operace, atributy a metody
- Třídy jsou využívány pro vytváření objektů
- Model tříd dává základ pro funkci jednotlivých objektů
- Modelování tříd je klíčovým prvkem objektově orientovaného vývoje

Vztahy mezi třídami

- **Agregace**
 - jedna třída je částí druhé
- **Kompozice**
 - agregace, kdy podřízený objekt nemůže existovat samostatně
- **Asociace**
 - znázorňuje vztahy mezi jednou či více třídami (1 ku 1, 1 k mnoha, ...)
- **Generalizace (dědění)**
 - vztah mezi obecnou třídou (super class resp. parent) a jejími potomky (subclass resp. child)
 - dědí se všechny vlastnosti tj. atributy, relace, operace a omezení)

(PŘÍKLADY JSOU UVEDENY NA DALŠÍCH SNÍMCÍCH) ➡

Vysvětlení pojmů na případové studii

Modelová situace

SW firma získala zakázku na analýzu, návrh a vývoj IS, který by funkčně pokrýval potřeby sběrný oprav elektrospotřebičů. Sběrna oprav je „zákazník“ a předmětem podnikání je zprostředkování oprav ve značkových i neznačkových servisech podle druhu el.spotřebičů.

Pro naše potřeby modelujeme jeden Modul IS pro zprostředkování oprav.

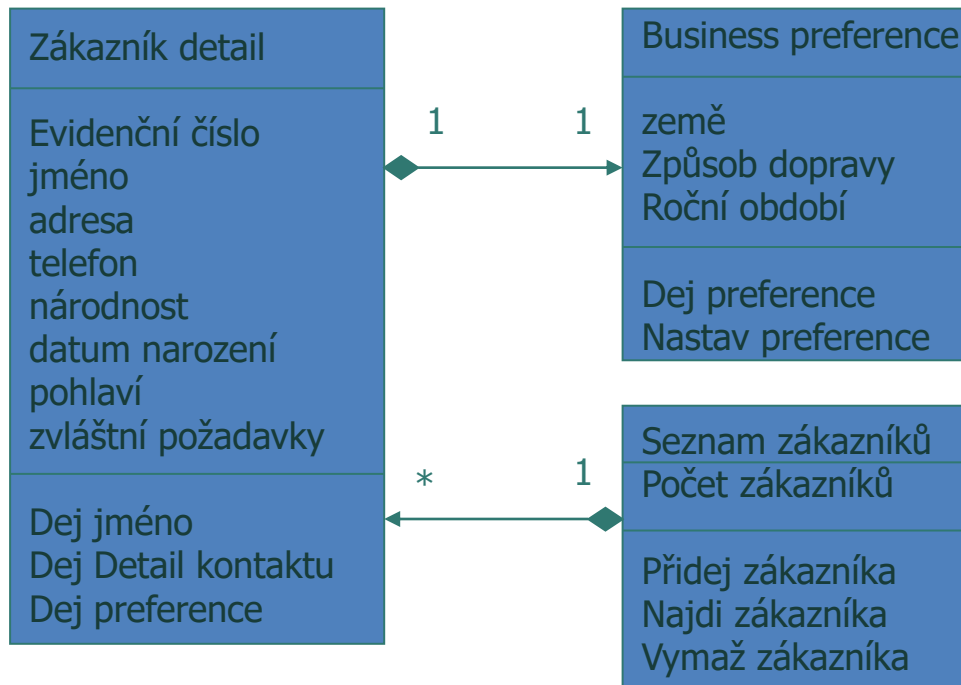
Vysvětlení pojmů na případové studii II

Požadavky zákazníka (requirements) na modul IS oprava elektrospotřebičů:

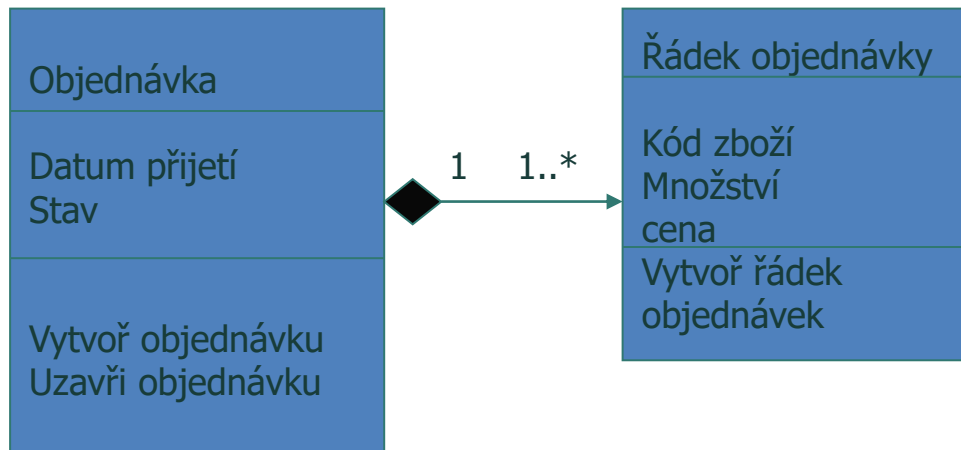
01. Příjem zakázky na opravu elektrospotřebiče
02. Výdej zakázky majiteli
03. Správa číselníků
04. Monitoring oprav
05. Evidence zákazníků
06. Vyhodnocení oprav
07. Vyřízení reklamace opravy
08. Oprava spotřebiče v servisu

Ke každému požadavku zpracován detailní popis.

Příklad 1: Vazba typu agregace



Příklad 2: Vazba typu kompozice

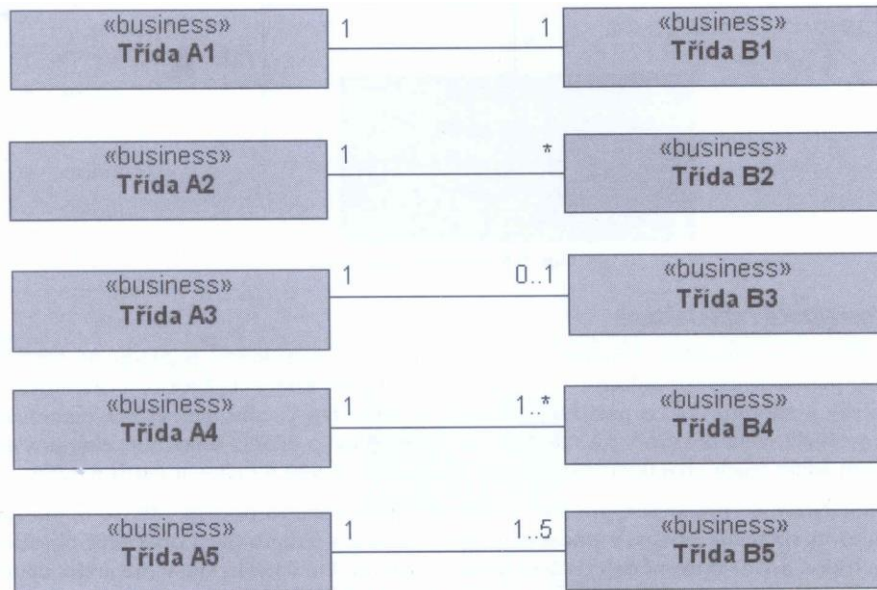


Příklad 3: Vazba typu asociace



Instance třídy FIRMA má asociaci na žádnou nebo více instancí třídy Osoba
Instance třídy = konkrétní výskyt objektu, resp. samostatná existence objektu

Příklad 4: Vazba typu asociace – typy vazeb

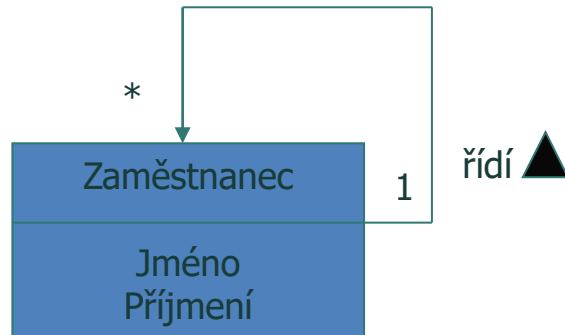


- ← Instance třídy A1 má asociaci na právě jednu instanci třídy B1.
- ← Instance třídy A2 má asociaci na žádnou nebo více instancí třídy B2.
- ← Instance třídy A3 má asociaci na nula nebo jednu instanci třídy B3.
- ← Instance třídy A4 má asociaci na jednu nebo více instancí třídy B4.
- ← Instance třídy A5 má asociaci na jednu až pět instancí třídy B5.

Příklad 5: Reflexivní asociace



- Objekt zaměstnanec může mít nula nebo více podřízených.
- Řada zaměstnanců nebude mít podřízené, ale každý zaměstnanec bude mít svého nadřízeného.



Vztahy mezi třídami:

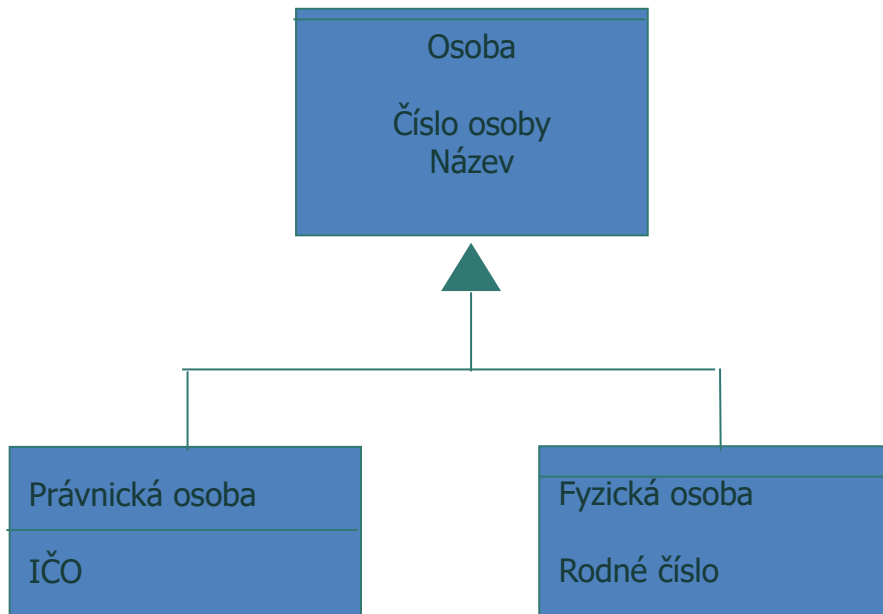
- **Abstraktní třída**
 - zvláštní třída bez konkrétní instance, zobecnění
- **Polymorfismus**
 - některé objekty mají totožná rozhraní realizovaná pomocí operací, ale metody, které se skrývají za těmito operacemi, jsou rozdílné
- **Asociační třídy**
 - typ vazeb n..m

- **PŘÍKLADY viz dále**

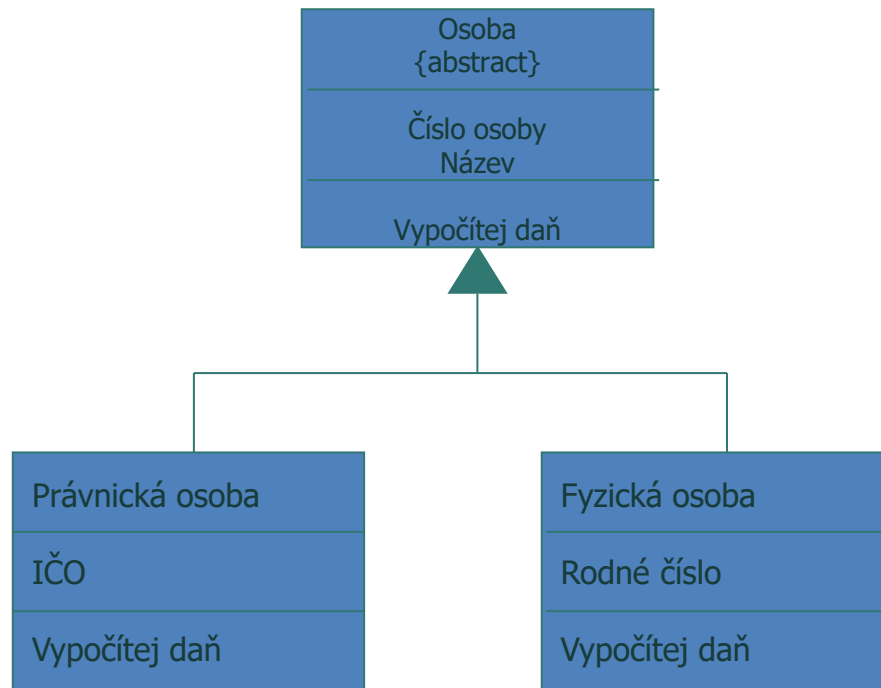
Příklad 6: Abstraktní třída



- Abstraktní třída je zvláštní třída – ve vývojovém prostředí nebude nikdy vytvářena její konkrétní instance



Příklad 7: Polymorfismus objektů



Některé objekty mají totožná rozhraní realizovaná pomocí operací, ale metody, které se skrývají za těmito operacemi jsou rozdílné = polymorfismus.

Příklad 8: Generalizace – dědičnost mezi třídami



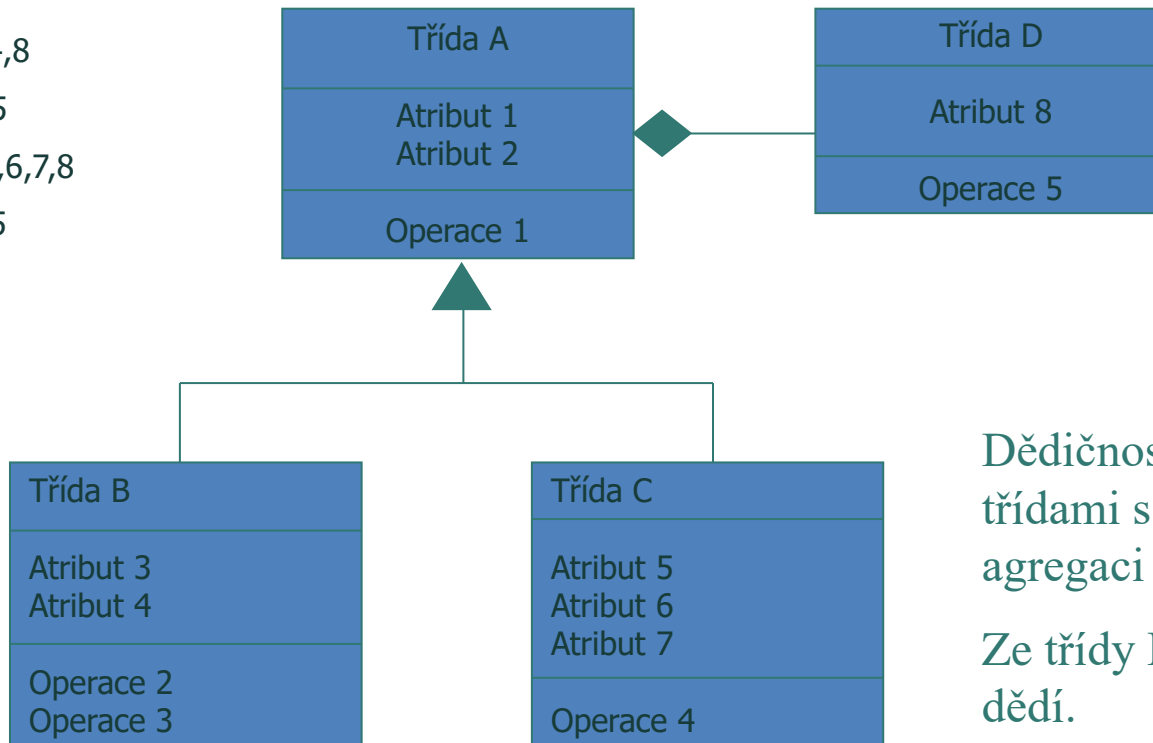
Objektová třída

B atributy 1,2,3,4,8

operace 1,2,3,5

C atributy 1,2,5,6,7,8

operace 1,2,4,5



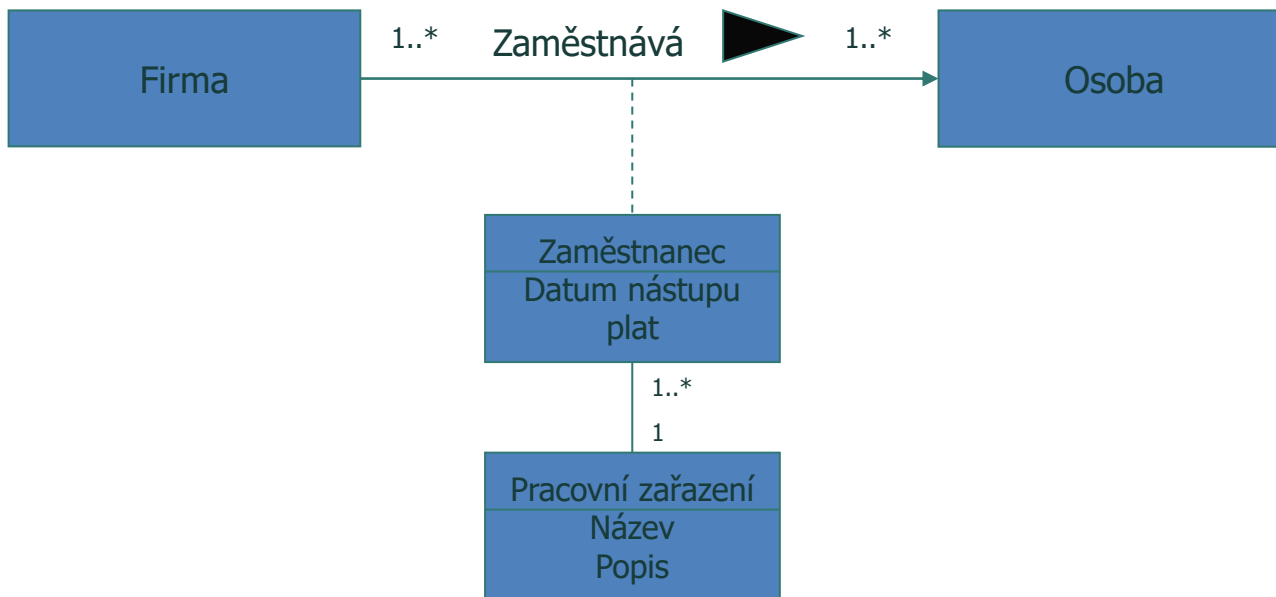
Dědičnost mezi třídami s rozšířením o agregaci na třídu D.

Ze třídy D se rovněž dědí.

Příklad 9: Asociační třída s relací na další třídu



- Asociační třídy dovolují přiřadit atributy, operace a další rysy k asociační vazbě, která řeší vztah mezi třídami mnoha ku mnoha.



Příklad užití vztahů mezi třídami – část účetního systému

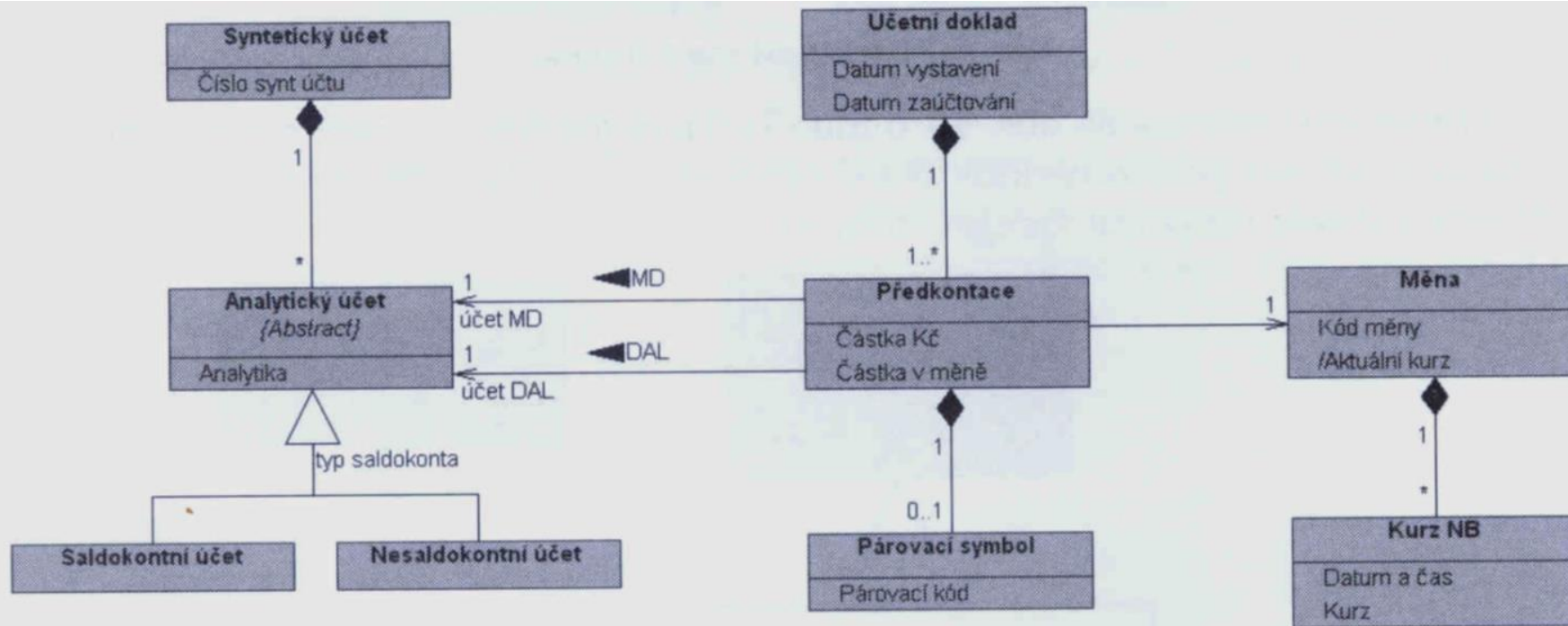


Diagram tříd



- Diagram tříd zobrazuje strukturu a vztahy mezi objektovými třídami navrhovaného IS.

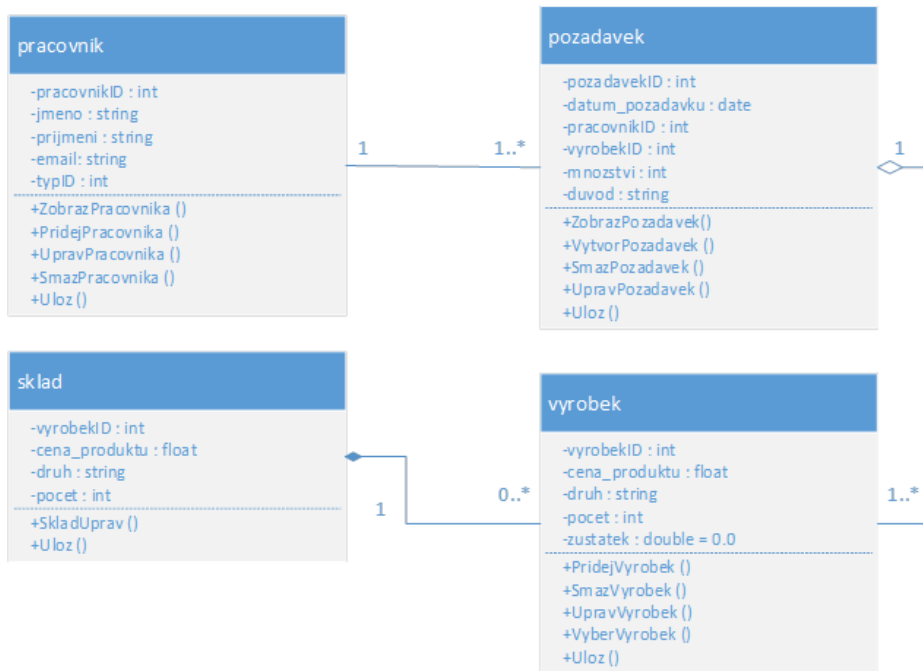


Diagram tříd
modulu skladu



Děkuji za pozornost

Otázky?
