

2005-11-01

Coddova pravidla pro relační model:

- Ü **Pravidlo nezávislosti dat na distribuci** – výsledky operací nesmí být ovlivněny konkrétním umístěním dat v distribuovaných databázích.
- Ü **Pravidlo nenarušitelnosti SŘBD** – žádný uživatel nesmí obcházet nebo narušovat rozhraní SŘBD.

Nezávislost dat:

- Ü **Fyzická** – aplikace musí být izolována od změn fyzické datové struktury, změna aplikace nesmí způsobit nutnou změnu paměťového uložení struktur a opačně.
- Ü **Logická** – aplikační pohled je izolován od změn ve schématu databáze – změna struktury nevynucuje změnu programu.

Třístupňová architektura báze dat – 3 úrovně abstrakce při pohledu na evidovaná data:

- Ü **Fyzická úroveň** – nejnižší úroveň, popisuje jak jsou data fyzicky uložena, pracuje s ní programátor SŘBD.
- Ü **Konceptuální úroveň** – popisuje strukturu dat DB a jejich vzájemné vztah, tzv. schéma – modeluje reálný svět a vytváří jeho interní model pomocí logické a fyzické struktury dat, pracuje s ní tvůrce SŘBD, správce DB.
- Ü **Uživatelská úroveň** – popisuje pro konkrétního uživatele pouze tu část DB, na kterou má pravomoc, tzv. subschéma, jejich počet = počet uživatelů.

Datový model = souhrn pravidel pro reprezentaci logické orientace dat v databázi.

DATOVÉ MODELOVÁNÍ:**Vícestupňová architektura DBS:**

- Ü **Externí schéma** – popisuje datové struktury pro koncového uživatele (pohledy) = podmnožina konceptuálního schéma.
- Ü **Konceptuální schéma** – popisuje jednotlivé datové struktury a jejich vzájemné vazby – model.
- Ü **Interní schéma** – vlastní popis implementace datových struktur v implementačním prostředí.

Konceptuální datové modelování – Chenův E-R model:

- Ü Konceptuální datový model – struktura dat obecně bez ohledu na DBS
- Ü Logický datový model – konkretizuje model na daný DBS.
- Ü Datový model v definičním jazyku – výsledný pro rutinní zpracování.

Principy E-R modelování:

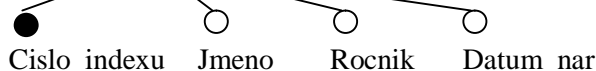
- Ü Orientace na objekty – pracuje s objekty RIDIC, VOZIDLO, nikoliv s identifikacemi
- Ü Funkcionální podstata vztahů – vztahy mezi objekty jsou definovány jako funkce, např. MA_PRIDELEN
- Ü ISA-hierarchie – pro práci s nadtypy a podtypy
- Ü Hierarchický mechanismus pro konstrukci objektů

Úrovně objektů v E-R modelování:

- Ü Entitní množina



- Ü Atribut



- Ü Vztahová množina

**ZPŮSOBY KONSTRUOVÁNÍ DATOVÉHO MODELU:**

„Zdola nahoru“ – východiskem je univerzální relace (U) obsahující všechny atributy řešené problémové domény, následuje specifikování funkčních závislostí mezi jednotlivými atributy. Např. U(Cislo_indexu, Jmeno, Rocnik, Datum_nar, Znamka, Katedra, Cislo_ped, Jmeno_ped, Datum_zkousky, Cislo_predmetu, Nazev_predmetu, Fakulta...)

- Ü Cislo_indexu -> Jmeno, Rocnik, Datum_nar

Ü Císlo_predmetu -> Nazev_predmetu

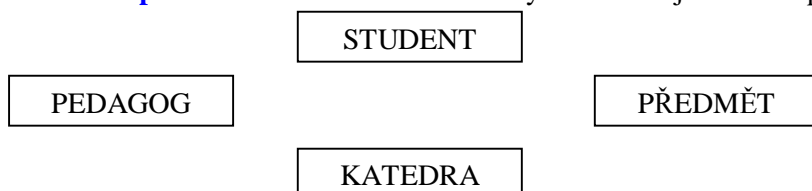
Ü Císlo_indexu, Císlo_predmetu -> Datum_zkousky, Znamka, Císlo_pedagoga

Ü Císlo_ped -> Jmeno_ped

Tento postup přímo přechází na logickou úroveň => pouze pro vytváření modelů s maximálně 25 atributy.

„Shora dolů“ – nejprve jsou specifikovány struktury entitních množin a následně funkční závislosti mezi nimi. Jednotlivé kroky modelování „shora dolů“:

Ü **1. krok – Specifikace entitních množin** – východiskem je verbální popis.

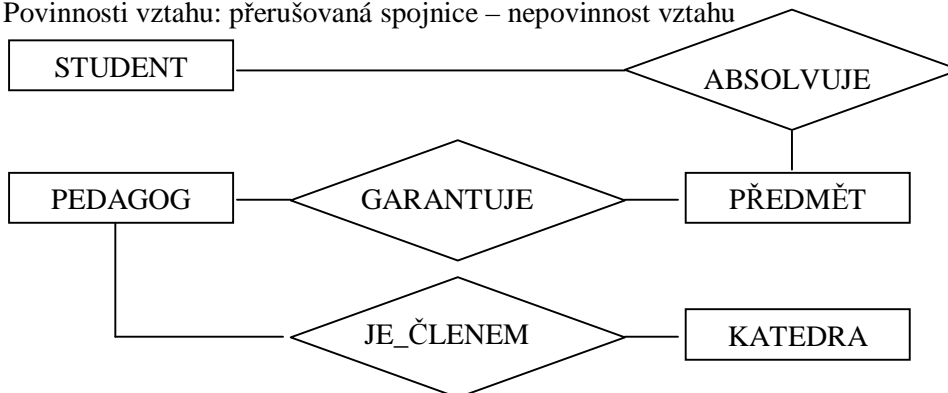


Specifikovaným objektům (entitním množinám) se přidělí vhodné jméno.

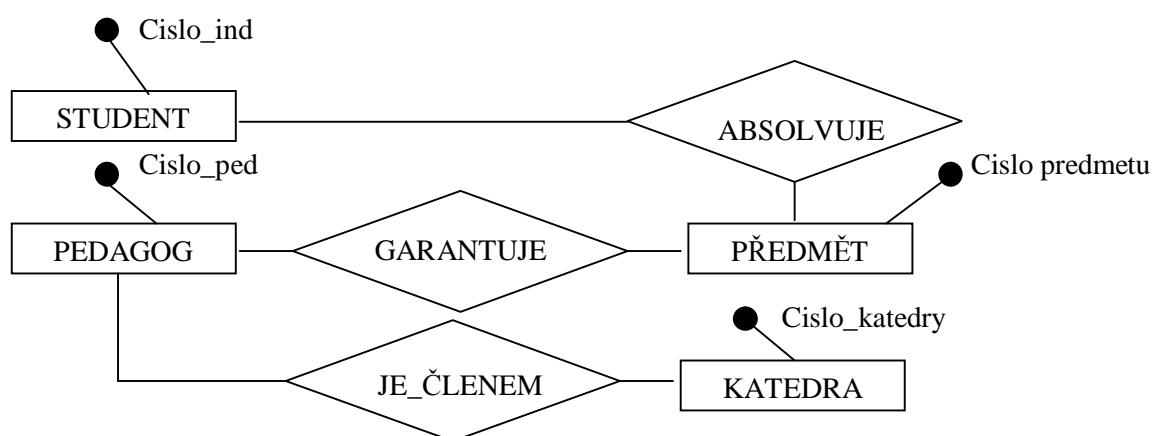
Ü **2. krok – Specifikace vztahů** – důležitá kardinalita vztahu:

- 1:1 + ———— +
- 1:N + ———— >
- M:N > ———— >

Povinnosti vztahu: přerušovaná spojnice – nepovinnost vztahu



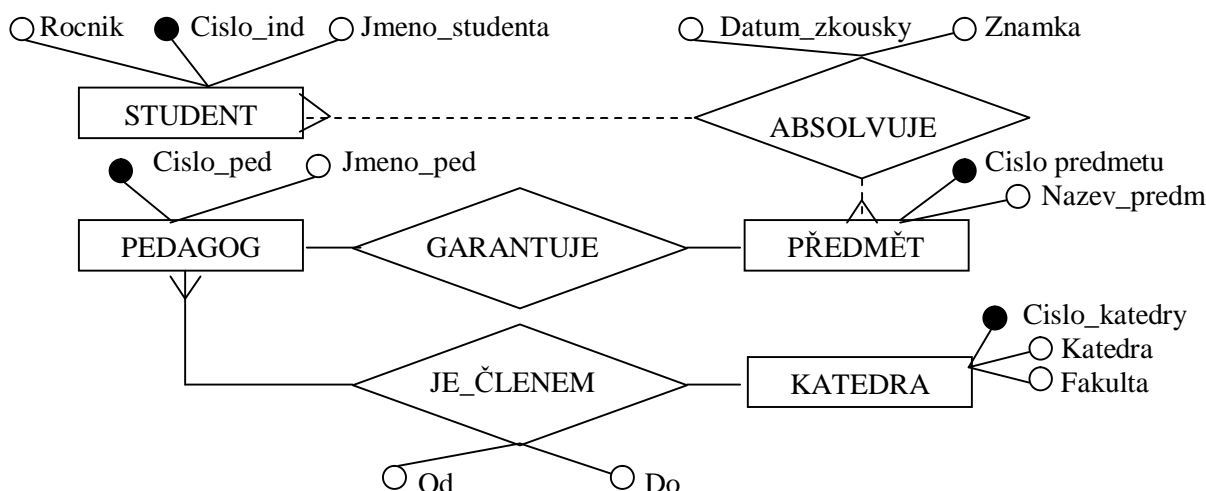
Ü **3. krok – Přiřazení primárních klíčů:**



Ü **4. krok – Transformace modelu do logické struktury** – konceptuální datový model je rozpracován do logické struktury „předběžných“ relací, tyto relace jsou ovlivněny kardinalitou vztahů:

- Vztah 1:1 – je-li účast obou entitních množin ve vztahu povinná, postačí pro realizaci pouze jedna relace, jinak NULL hodnoty v primárních klících.
- Vztah 1:N – nejčastější, řešení minimálně 2 relacemi.
- Vztah N:M – pro maximální

Ü **5. krok – Doplnění zbývajících atributů do předběžných relací:**



Ü 6. krok – Prověření modelu z hlediska normalizace – strukturální správnost a konzistence datového modelu.

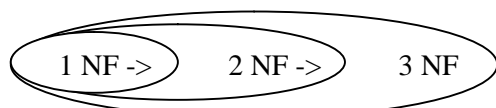
- Cíle datové normalizace:**
 - Umožnění reprezentace každé relace v DB
 - Optimalizace algoritmů vyhledávání
 - Zjednodušení aktualizace a rušení vět
- = postupný reverzibilní proces nahrazování dané množiny relací relacemi, které mají jednodušší strukturu
- Vlastnosti datové normalizace:**
 - Umožňuje ověření správnosti navrženého modelu
 - Slouží k dekompozičnímu návrhu tabulek s minimální redundancí dat
 - Není deterministická – více správných řešení
 - Lze ji kombinovat s ostatními technikami – dekompozice a syntéza

- Chybné návrhy** – evidence prospěchu:

Jméno_žáka	Datum_zkoušení	Známka
J. Dvořák	10.9.,15.10.,29.10.	1,3,3

Jméno_žáka	Datum_zk1	Známka1	Datum_zk n	Známka n
J. Dvořák	10.9.	1	29.10.	3

- Normální forma** – evidence prospěchu:



- Přínosy datové normalizace:**
 - Zabránění vzniku duplicitních dat (redundancí)
 - Šetří kapacitu média
 - Zjednodušuje aktualizaci a vyhledávání dat
- Dekomponované relace možno opět funkčně zvolit (operace JOIN)

- Normální forma:**

- 1. normální forma** – Relace nesmí obsahovat násobná data:

Prospěch (Cislo_stud, Jmeno, Adresa, Datum_zkousky, Znamka)

0001A	Dvořák	Kolín 22	10.9.02	4
0002A	Kalaš	Čáslav 5	10.9.02	1
0001A	Dvořák	Kolín 22	17.9.02	2
0002A	Kalaš	Čáslav 5	21.9.02	2
0001A	Dvořák	Kolín 22	1.10.02	2

<u>Cislo_stud</u>	Jmeno	Adresa	Datum_zkousky	Znamka
-------------------	-------	--------	---------------	--------

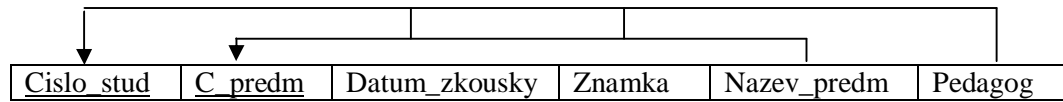
Po normalizaci:

STUDENTI	<u>Cislo_stud</u>	Jmeno	Adresa
----------	-------------------	-------	--------

	<u>Cislo_stud</u>	Datum_zkousky	Znamka
--	-------------------	---------------	--------

ZNÁMKY

- **2. normální forma** – Všechna neklíčová data relace musí funkčně záviset na celém primárním klíči



Po normalizaci:

ZKOUŠKY

<u>Cislo_stud</u>	<u>C_predm</u>	Datum_zkousky	Znamka	Pedagog
-------------------	----------------	---------------	--------	---------

PŘEDMĚTY

<u>C_predm</u>	Nazev_predm
----------------	-------------

Funkční závislost – datový prvek B záznamu je funkčně závislý -> ke každé hodnotě A náleží nejvýše jedna hodnota B -> A identifikuje B.