

Adatbázisok elmélete 10. előadás

Csima Judit — Katona Gyula Y.
Budapesti Műszaki és Gazdaságtudományi Egyetem
Számítástudományi Tsz.

I. B. 137/b
{csima,kiskat}@cs.bme.hu

2002. Október 15.

ADATBÁZISOK ELMÉLETE 10. ELŐADÁS

```
interface Színész {  
    attribute String név;  
    attribute Struct Cím  
        { string város, string utca } lakcím;  
};  
⇒ Színész(név, város, utca)
```

De pl. halmaz csak rosszul oldható meg: halmaz minden eleméhez új sor.

```
interface Színész {  
    attribute String név;  
    attribute Set <  
        Struct Cím{ string város, string utca }  
        > lakcím;  
};  
⇒ Színész(név, város, utca)
```

2

ADATBÁZISOK ELMÉLETE 10. ELŐADÁS

ODL séma átírása relációsémává

Legegyszerűbb eset ⇒ az osztályoknak csak attribútumai vannak, amik atomi típusúak:

```
interface Film {  
    attribute cím;  
    attribute év;  
    attribute hossz;  
    attribute szalagFajta;  
};  
⇒ Film(cím, év, hossz, szalagFajta)
```

Összetett típusú attribútumok: pl. rekordszerkezet OK ⇒

1

ADATBÁZISOK ELMÉLETE 10. ELŐADÁS

név	város	utca
Kabos Gyula	Budapest	Nyereg u.
Kabos Gyula	Budapest	Kápa u.
:		

3

Kapcsolatok átírása

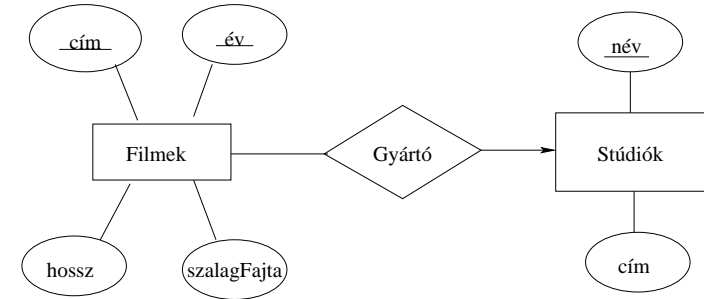
Egyértékű kapcsolat: kapcsolat \Rightarrow attribútumok: kulcs a másik osztályból
Csak egyik irányból csináljuk

```
interface Film {
  attribute cím;
  attribute év;
  attribute hossz;
  attribute szalagFajta;
  relationship Stúdió gyártó
  inverse Stúdió::gyárt; };
```

\Rightarrow **Film(cím, év, hossz, szalagFajta, stúdióNév)**

Többértékű kapcsolat: Ugyanaz a probléma, mint a halmaz típusú attribútum. Nem lehet jól megoldani. Ha több ilyen is van \Rightarrow katasztrófa

4



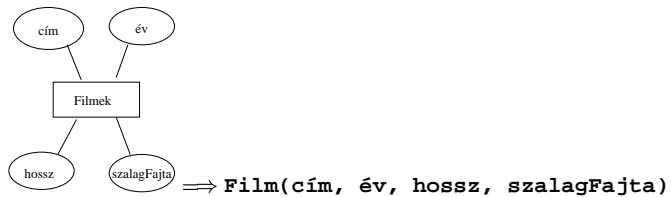
\Rightarrow Gyártó(cím, év, stúdióNév)
Film(cím, év, hossz, szalagFajta)
Stúdiók(stúdióNév, cím)

E/K-ból jobban lehet relációsat csinálni.

6

E/K modell átírása

Egyedhalmaz attribútumokkal:



Egyértékű és többértékű kapcsolat:

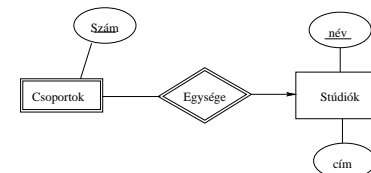
\Rightarrow Külön reláció, attribútumai: a kapcsolatban résztvevő egyedhalmaz kulcsainak uniója + kapcsolat attribútumai (esetleg átnevezés)

5

Gyenge egyedhalmazok kezelése

Ha W gyenge egyedhalmaz:

- Nem csak W attribútumait kell tartalmaznia, hanem azokat is, amiktől kulcs lesz. (Dupla keretes kapcsolat.)
- Ez minden olyan kapcsolatra is igaz, melyben W részt vesz.
- A dupla keretes kapcsolatokhoz nem kell külön reláció.



Stúdió(név, cím)
Csoportok(szám, stúdióNév)
Egység(szám, stúdióNév, név)

7

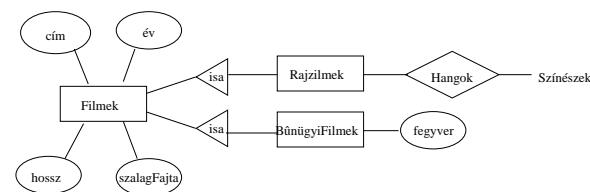
Egység(szám, stúdióNév, név) \Rightarrow Egység(szám, név) hiszen ugyanaz kétszer.

\Rightarrow Egység el is hagyható, hiszen összes attribútuma szerepel a Csoport-ban is.

Ez általában is igaz, hiszen a gyenge egyedhalmaz attribútumai között ott lesz a dupla keretes rombusz minden attribútuma.

Alosztályok kezelése E/K modellben

Minden alosztályhoz csak a főosztály kulcsát és saját attribútumait ill. kapcsolatait rendeljük. Az "isa" kapcsolathoz nem rendelünk relációt.



\Rightarrow

Film(cím, év, hossz, szalagFajta)

Rajzfilm(cím, év, hang)

BűnügyiFilm(cím, év, fegyver)

Hangok(cím, év, név)

Hátránya: egy film információi több helyre vannak szórva (BűnügyiRajzfilm).

Alosztályok kezelése ODL-ben

Film, Rajzfilm, BűnügyiFilm, BűnügyiRajzfilm esete. *(Itt minden egyed pont egy osztályban lehet benne.)*

Minden alosztályhoz egy reláció, minden attribútumával (öröklöttekkel is).

Film(cím, év, hossz, szalagFajta, stúdióNév, színésznév)

Rajzfilm(cím, év, hossz, szalagFajta, stúdióNév, színésznév, hang)

BűnügyiFilm(cím, év, hossz, szalagFajta, stúdióNév, színésznév, fegyver)

BűnügyiRajzfilm(cím, év, hossz, szalagFajta, stúdióNév, színésznév, hang, fegyver)

Hátrány: egy film kereséséhez mindet végig kell nézni.

Másik megoldás NULL értékkel

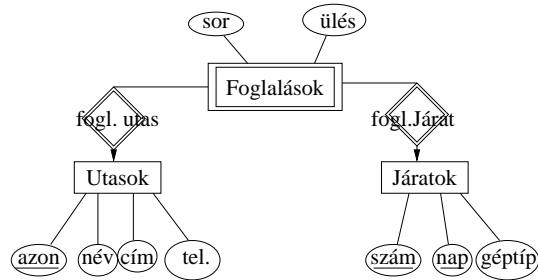
Film(cím, év, hossz, szalagFajta, stúdióNév, színésznév, hang, fegyver)

A hiányzó helyeket NULL-al töltjük ki.

Hátrány: elveszítetünk információt. Pl. egy néma rajzfilmről nem tudjuk, hogy rajzfilm.

Példák

Alakítsuk relációssá:



Utas(azon, név, cím, tel.)
 Járát(szám, nap, géptípus)
 Foglalások(azon, szám, nap, sor, ülés)

Sorkalkulus

Lekérdezőnyelvek típusai:

- relációs algebra (**LEAP**, letölthető, SIGMOD-ról link), **ISBL** nehezen emészthetőbb, *algebrai alapú; ez volt*
- sorkalkulus (**QUEL**, INGRES nyelve), *könnyebben emészthető, logikai alapú*
- oszlopkalkulus (**QBE, SQL**) nagyon hasonló

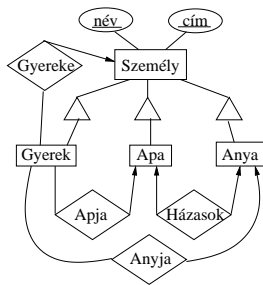
Sorkalkulus (Tuple calculus)

Formális modell, de már hasonlít az igazihoz.
Elsőrendű nyelv relációk kifejezésére.

Változók: t, r, s sorváltozók, a reláció sorait futják be.

$t^{(k)}$: k oszlopos reláción fut végig.

$t^{(k)}[i]$: A t sorváltozó i -edik komponense.



Személy(név, cím); Gyerek(név, cím); Apa(név, cím)
 Anyja(név, cím)
 Apja(apanév, apacím, gyereknev, gyerekcím)
 Anyja(anyanev, anyacim, gyereknev, gyerekcim)
 Házast(fnév, fcím, nőnev, nőcím)
 Gyereke(gyereknev, gyerekcim, szülőnev, szülőcim)

Pl. egy sor \implies (R. M., Budapest, hamburger, 180), akkor
 $t[3]$ = 'hamburger' és $t[\text{ÁR}] = 180$

A sorkalkulus által kifejezett reláció:

$\{t^{(k)} \mid \phi(t)\} \implies$ azon t -k, amikre $\phi(t)$ igaz, ahol ϕ egy megengedett formula+ valami még.

Megengedett formulák:

atomok :

- $R^{(k)}(t^{(k)})$: akkor igaz, ha $t \in R$, azaz a sor benne van a relációban.
 - $t^{(k)}[i] \theta s^{(l)}[j]$
 - $t^{(k)}[i] \theta c$
 - $c \theta t^{(k)}[i]$
- ahol $\theta \in \{<, >, =, \neq, \leq, \geq\}$, t, s sorváltozók, c konstans érték.
 Világos, mikor igaz.

építkezési szabályok :

- ϕ, ψ formulák, akkor $\phi \vee \psi, \phi \wedge \psi, \neg \phi$ is formulák.
 Világos, hogy mikor igaz.

- ϕ formula, s sorváltozó, akkor $\forall s\phi, \exists s\phi$ is formula.

Világos, hogy mikor igaz.

Kötött változó: ha vonatkozik rá kvantor,

Szabad változó: ha nem,

Sorkalkulus által kifejezett reláció (pontosan) : $\{t^{(k)} \mid \phi(t)\} \implies$ azon t -k, amikre $\phi(t)$ igaz, ahol ϕ egy megengedett formula és ϕ -ben t az *egyetlen szabad* változó.

$$\{s^{(3)} \mid \exists u\exists v \left(\text{MENNYISÉG}(u) \wedge \text{ÁRU}(v) \wedge u[1] = 2002-01-15 \wedge u[2] = 'A123' \wedge v[1] = 'A123' \wedge s[1] = u[3] \wedge s[2] = v[2] \wedge s[3] = v[3] \right)\}$$

Mely áruk azok, amelyekkel van azonos egységárú másik áru?

$$\{s^{(1)} \mid \exists u\exists v \left(\text{ÁRU}(v) \wedge \text{ÁRU}(u) \wedge s[1] = v[2] \wedge v[3] = u[3] \wedge \neg(v[1] = u[1]) \right)\}$$

Példák sorkalkulus alkalmazására

ÁRU(ÁRUKÓD, ÁRUNÉV, EGYSÉGÁR)
 MENNYISÉG(DÁTUM, ÁRUKÓD, DB)
 BEVÉTEL(DÁTUM, ÖSSZEG)
 BEFIZ(ÖSSZEG, BEFIZ) BEFIZ=ÖSSZEG-4000

Az 2002. jan. 1. utáni napok bevételei a dátummal együtt:

$$\left(\sigma_{\text{DÁTUM} > '2002-01-01'}(\text{BEVÉTEL}) \right)$$

$$\{t^{(2)} \mid \text{BEVÉTEL}(t) \wedge t[1] \geq 2002-01-01\}$$

Az 2002. jan. 15-i bevétel és a befizetett összeg:

$$\pi_{\text{ÖSSZEG, BEFIZ}} \left(\sigma_{\text{DÁTUM} = '2002-01-15'}(\text{BEVÉTEL}) \bowtie \text{BEFIZ} \right)$$

$$\{u^{(2)} \mid \text{BEFIZ}(u) \wedge \exists v(\text{BEVÉTEL}(v) \wedge v[1] = 2002-01-15 \wedge v[2] = u[1])\}$$

Hány darabot adtak el 2002. jan. 15-én az A123 kódú áruból, mi a neve és az ára?

$$\pi_{\text{DB, ÁRUNÉV, EGYSÉGÁR}} \left(\sigma_{\text{ÁRUKÓD} = 'A123' \wedge \text{DÁTUM} = '2002-01-15'}(\text{MENNYISÉG} \bowtie \text{ÁRU}) \right)$$