Adatbázisok elmélete 6. előadás

Katona Gyula Y.
Budapesti Műszaki és Gazdaságtudományi Egyetem
Számítástudományi Tsz.
I. B. 137/b
kiskat@cs.bme.hu
http://www.cs.bme.hu/˜kiskat

2005

Sorkalkulus ereje

Mivel lehet több mindent kifejezni?
Relációs algebrával, vagy sorkalkulussal?

Tétel. *A sorkalkulus relációsan teljes.*

Bizonyítás: Be kell látni, hogy minden reláció, ami relációs algebrával megadható, megadható sorkalkulussal is. Ehhez azt elég megmutatni, hogy
1. az alaprelációk megadhatók
2. a relációs algebrai alapműveletek (unió, különbség, szorzat, vetítés, szelekció) alaprelációkra alkalmazva megvalósíthatók
3. ha R és S nem alapreláció és ezekre alkalmazunk valami relációs alapműveletet, akkor az eredmény kifejezhető sorkalkulussal

Bizonyítás

- **alapreláció:** Tegyük fel, hogy R k oszlopos alapreláció
  $R = \left\{t^{(k)} \mid R(t)\right\}$
- **Unió:** *Tfh. S is k oszlopos*
  $R \cup S = \left\{t^{(k)} \mid R(t) \vee S(t)\right\}$
- **különbség:**
  $R \setminus S = \left\{t^{(k)} \mid R(t) \wedge \neg S(t)\right\}$
- *metszet:*
  $R \cap S = \left\{t^{(k)} \mid R(t) \wedge S(t)\right\}$
- **szorzat:** R legyen k oszlopos, S pedig l oszlopos

$$R \times S = \left\{t^{(k+l)} \mid \exists r^{(k)} \exists s^{(l)} \left(R(r) \wedge S(s) \wedge r[1] = t[1] \wedge \ldots \right.\right.$$
$$\left.\left. \wedge r[k] = t[k] \wedge s[1] = t[k+1] \wedge \ldots \wedge s[l] = t[k+l]\right)\right\}$$

- **vetület:** Legyen $R(A_1, \ldots, A_d, A_{d+1}, \ldots, A_k)$ reláció, vetítsük az első d-re
  $\pi_{A_1,\ldots,A_d}(R) = \left\{t^{(d)} \mid \exists r^{(k)} \left(R(r) \wedge r[1] = t[1] \wedge \ldots \wedge r[d] = t[d]\right)\right\}$

- **kiválasztás:**
  $\sigma_F(R) = \left\{t^{(k)} \mid R(t) \wedge F'\right\}$, ahol F' átfordítása sorkalkulusra $\Longrightarrow$ az i-edik attribútum helyett $t^{(n)}[i]$-t írunk.
  Pl. (evidenciával történő meggyőzés)
  $\sigma_{\text{ÁR>'150'} \wedge \text{TERMÉK='hamburger'}}(\text{TERMEL}) =$
  $\left\{t^{(4)} \mid \text{TERMEL}(t) \wedge t[4] > \text{'150'} \wedge t[3] = \text{'hamburger'}\right\}$
- Nem lényeges, hogy R, S alaprelációk.
  Ha $R = \left\{t^{(k)} \mid \phi(t)\right\}$ és $S = \left\{t^{(k)} \mid \psi(t)\right\}$, azaz R és S már valahogy ki van fejezve sorkalkulussal
  $\Longrightarrow R \cup S = \left\{t^{(k)} \mid \phi(t) \vee \psi(t)\right\}$
  többinél ugyanígy ($R(t)$ és $S(t)$ helyett $\phi(t)$-t és $\psi(t)$-t írunk). √

További példák

TERMÉK(GYÁRTÓ, MODELL, TÍPUS)
PC(MODELL, SEBESSÉG, MEMÓRIA, MEREVLEMEZ, CD, ÁR)
LAPTOP(MODELL, SEBESSÉG, MEMÓRIA, MEREVLEMEZ, KÉPERNYŐ, ÁR)
NYOMTATÓ(MODELL, SZÍNES, TÍPUS, ÁR)

A relációk jelentése:
TERMÉK: az adott nevű gyártó gyártja az adott modellszámú és adott típusú (PC, Laptop vagy nyomtató) terméket
PC: modellszám, sebesség megaHz-ben, memória megabájtban, merevlemez gigabájtban, a CD sebessége (pl. 4x), az ár
Laptop: mint PC-nél, plusz a képernyő mérete hüvelykben
Nyomtató: modellszám, színes-e (i/n), típusa (tintasugaras, lézer, mátrix), ára

A modellszámokról feltesszük, hogy egyediek.

- Melyek azok a PC modellek, amelynek sebessége legalább 150?

$$\pi_{\text{MODELL}}\left(\sigma_{\text{SEBESSÉG} >= 150}(\text{PC})\right)$$

- Mely gyártók készítenek legalább egy gigás merevlemezű laptopot?

$$\pi_{\text{GYÁRTÓ}}\left(\text{TERMÉK} \bowtie \sigma_{\text{MEREVLEMEZ}>= 1}(\text{LAPTOP})\right)$$

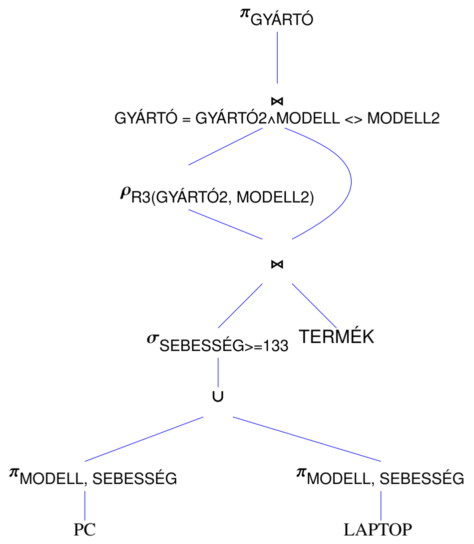
TERMÉK(GYÁRTÓ, MODELL, TÍPUS)
PC(MODELL, SEBESSÉG, MEMÓRIA, MEREVLEMEZ, CD, ÁR)
LAPTOP(MODELL, SEBESSÉG, MEMÓRIA, MEREVLEMEZ, KÉPERNYŐ, ÁR)
NYOMTATÓ(MODELL, SZÍNES, TÍPUS, ÁR)

- Adjuk meg a B gyártó által gyártott összes termék modellszámát és árát típustól függetlenül!
  $\pi_{\text{MODELL, ÁR}}\left(\sigma_{\text{GYÁRTÓ='B'} \wedge \text{TÍPUS = 'PC'}}(\text{TERMÉK}) \bowtie \text{PC}\right) \cup$
  $\pi_{\text{MODELL, ÁR}}\left(\sigma_{\text{GYÁRTÓ='B'} \wedge \text{TÍPUS = 'LAPTOP'}}(\text{TERMÉK}) \bowtie \text{LAPTOP}\right) \cup$
  $\pi_{\text{MODELL, ÁR}}\left(\sigma_{\text{GYÁRTÓ='B'} \wedge \text{TÍPUS = 'NYOMTATÓ'}}(\text{TERMÉK}) \bowtie \text{NYOMTATÓ}\right)$
- Melyek azok a gyártók, akik laptopot gyártanak, de PC-t nem?
  $\text{TERMÉK1} = \rho_{\text{TERMÉK1}}\left(\pi_{\text{GYÁRTÓ, TÍPUS}}(\text{TERMÉK})\right)$
  $\pi_{\text{GYÁRTÓ}}\left(\sigma_{\text{TÍPUS='LAPTOP'}}(\text{TERMÉK1})\right) \setminus \pi_{\text{GYÁRTÓ}}\left(\sigma_{\text{TÍPUS='PC'}}(\text{TERMÉK1})\right)$

TERMÉK(GYÁRTÓ, MODELL, TÍPUS)
PC(MODELL, SEBESSÉG, MEMÓRIA, MEREVLEMEZ, CD, ÁR)
LAPTOP(MODELL, SEBESSÉG, MEMÓRIA, MEREVLEMEZ, KÉPERNYŐ, ÁR)
NYOMTATÓ(MODELL, SZÍNES, TÍPUS, ÁR)

- Melyek azok a gyártók, amelyek gyártanak legalább két, egymástól különböző, legalább 133 Mhz-en működő PC-t vagy Laptopot? (Nincs két azonos modellszám!)
  $R1 = \pi_{\text{MODELL, SEBESSÉG}}(\text{PC}) \cup \pi_{\text{MODELL, SEBESSÉG}}(\text{LAPTOP})$
  $R2 = \pi_{\text{GYÁRTÓ, MODELL}}\left(\sigma_{\text{SEBESSÉG}>=133}(\text{R1}) \bowtie \text{TERMÉK}\right)$
  $R3 = \rho_{R3(\text{GYÁRTÓ2, MODELL2})}(R2)$
  $R4 = R2 \bowtie_{\text{GYÁRTÓ = GYÁRTÓ2} \wedge \text{MODELL <> MODELL2}} R3$
  $R5 = \pi_{\text{GYÁRTÓ}}(\text{R4})$

Megjegyzés: kifejezésfával is meg lehet adni a relációs algebrai kifejezéseket:



További műveletek

Ezek nincsenek benne a relációs algebrában, de fontosak, nem túl műveletigényesek.

aggregátumok: MIN, MAX, AVG, SUM, CNT (szám)
Pl. leggyorsabb gép, átlagár, hányféle printer
eredmény mindig egy szám

aggregátum csoportosítva: Bizonyos feltételek szerinti partíciókban aggregátumok.
Pl. átlagos ár tintasugaras nyomtatók között, egy gyártónak hány terméke van
$\Longrightarrow$ eredmény egy reláció pl. (gyártó, szám) párokból.

rekurzív lezárás: (hagyományos adatkezelésben ritka, intelligensebb rendszerekben inkább)
Pl. reláció: ki főnöke kinek $\Longrightarrow$ lezárás: ki felettese kinek
reláció: melyik városból melyikbe van repülő járat $\Longrightarrow$ lezárás: átszállással el lehet-e jutni
Ezt a relációs algebra nem tudja, csak fix mélységre: pl. max 4 átszállás,
(SQL nem igazán támogatja, de azért kicsit igen: SQL3, majd lesz)

A NULL érték

Lehet, hogy vannak kitöltetlen mezők, ezt meg akarjuk engedni. Pl. ha csak a termelő anyjának neve hiányzik, attól még a termék ára kell. $\Longrightarrow$ NULL érték

2 alapvető értelmezés (majd SQL-nél lesz, hogy hogyan kell megmondani, hogy melyik van éppen, illetve, hogy lehet-e egyáltalán NULL valahol):
- $\nexists$
- $\exists$, de nem ismerjük.

Attól függően, hogy hogyan értelmezzük a NULL-t:
Mi legyen egy ilyen kérdéssel?:
Pl. $\pi_{\text{CÍM='BP'}}(\text{TERMELŐ})$
Ilyenkor belevegyük-e ha a cím NULL?

Külső illesztés (outer join)

Definíció. *R, S relációk $\Longrightarrow$ $R ⟕ S$ **bal külső illesztés**: $R \bowtie S$-hez azokat az R-beli sorokat is hozzáveszük, amihez nem illeszkedik S-beli. Hiányzó helyekre NULL kerül.*

Pl. SZEMÉLY(NÉV, KÓD), CÍM(KÓD, CÍM)
SZEMÉLY $\bowtie$ CÍM $\Longrightarrow$ akinek nincs címe nem lesz rajta
SZEMÉLY $⟕$ CÍM $\Longrightarrow$ kiderül, kinek nincs meg a címe

SQL-ben van, relációs algebrával elvileg nem fejezhető ki (NULL miatt), de elkerülhető.
Ha a relációs algebrát úgy definiáljuk, hogy kiindulhatuk konstans relációból is, akkor:
$(R \bowtie S) \cup (R \setminus (R \ltimes S)) \times \{(\text{NULL}, \ldots, \text{NULL})\}$

Van **jobb külső illesztés** is: $R ⟖ S$
Teljes külső illesztés: $R ⟗ S := (R ⟕ S) \cup (R ⟖ S)$

Példa:

R	A	B	C
	a	b	2
	a	c	3
	b	a	4

S	D	C
	a	2
	b	3
	x	2
	y	1

$R ⟕ S$	A	B	C	D
	a	b	2	a
	a	b	2	x
	a	c	3	b
	b	a	4	NULL

R	A	B	C
	a	b	2
	a	c	3
	b	a	4

S	D	C
	a	2
	b	3
	x	2
	y	1

$R ⟖ S$	A	B	C	D
	a	b	2	a
	a	b	2	x
	a	c	3	b
	NULL	NULL	1	y

$R ⟗ S$	A	B	C	D
	a	b	2	a
	a	b	2	x
	a	c	3	b
	b	a	4	NULL
	NULL	NULL	1	y

Külső unió

Részben kompatibilis relációk egyesítésére:
DIÁK(NÉV, TÉMAVEZ, TSZK)
TANÁR(NÉV, TSZK, BEOSZT)

DIÁK $\cup_k$ TANÁR				
	NÉV	TSZK	TÉMAVEZ	BEOSZT
diák				NULL
tanár			NULL	

Multihalmazos szemantika

A relációs algebrában ugyan minden reláció halmaz, ezért nincsenek többszörös sorok, de pl. SQL-nél lesznek. A multihalmazokkal kicsit máshogy vannak a halmazműveletek:

Ha a t sor $m_R(t)$ példányban van meg R-ben és $m_S(t)$ példányban van meg S-ben, akkor

- $m_{(R\cup S)}(t) := m_R(t) + m_S(t)$ példányban lesz meg R és S uniójában
- $m_{(R\cap S)}(t) :=$ min$\{m_R(t), m_S(t)\}$ példányban lesz meg R és S metszetében
- $m_{(R\setminus S)}(t) :=$ max$\{m_R(t) - m_S(t), 0\})$ példányban lesz meg $R \setminus S$-ben

- Kiválasztásnál, vetítésnél nincs változás.

ODL séma átírása relációsémává

Legegyszerűbb eset $\Longrightarrow$ az osztályoknak csak attribútumai vannak, amik atomi típusúak:

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

$\Longrightarrow$ Film(<u>cím</u>, <u>év</u>, hossz, szalagFajta)

A kulcs az ODL-es kulcs lesz, ha egy van csak belőle; ezt aláhúzással jelöljük. Ha több kulcs van: egyiket jelöljük, a többit írásban (de persze azok is kulcsok a relációsémában is).

Összetett típusú attribútumok: pl. rekordszerkezet OK $\Longrightarrow$

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

$\Longrightarrow$ Színész(<u>név</u>, város, utca)
Kulcs: ugyanúgy, mint a nem összetett attribútumnál

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

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

$\Longrightarrow$ Színész(név, város, utca)

név	város	utca
Gálvölgyi J.	Budapest	Nyereg u. 2.
Gálvölgyi J.	Budapest	Kantár u. 3.
⋮		

Kulcs: elromlik az ODL-es kulcs, lehet, hogy ami ott kulcs volt, itt már nem lesz az $\Longrightarrow$ baj

Kapcsolatok átírása

Ha valamelyik irányba egyértékű a kapcsolat: ha a ***C*** és ***D*** közti kapcsolat ***D*** felé egyirányú, akkor a ***C*** osztály átírásakor bevesszük a ***D*** osztály kulcsát is (ha több van, akkor egyet)
Csak egyik irányból csináljuk, akkor is, ha a másik irányba is "egy" a kapcsolat

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

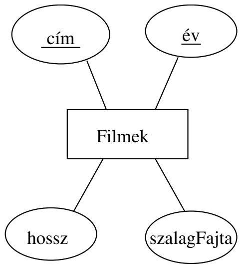
$\Longrightarrow$ Film(cím, év, hossz, szalagFajta, **stúdióNév**)
Feltéve, hogy a stúdiónév kulcs a Stúdió osztályban

Kulcs: mivel a kapcsolat „egy" jellegű volt, ezért az osztály kulcsa jó lesz kulcsnak a relációsémában is

Mindkét irányban többértékű kapcsolat: Ugyanaz a probléma, mint a halmaz típusú attribútum. Nem lehet jól megoldani, sok sor lesz és a kulcs is elromlik. Ha több ilyen kapcsolat is van $\Longrightarrow$ katasztrófa

E/K modell átírása

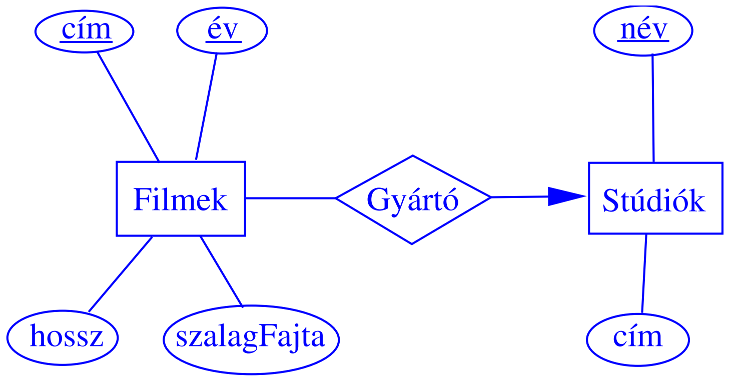
Egyedhalmaz attribútumokkal:



$\Longrightarrow$ Film(cím, év, hossz, szalagFajta)

A reláció kulcsa = az egyedhalmaz kulcsa

Egyértékű és többértékű kapcsolat:
$\Longrightarrow$ Külön reláció, attribútumai: a kapcsolatban résztvevő egyedhalmaz kulcsainak uniója + kapcsolat attribútumai (esetleg átnevezés)
Az így kapott reláció kulcsa: a kapcsolatban résztvevő egyedhalmaz kulcsainak uniója



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

E/K-ból jobban lehet relációsat csinálni.

Megjegyzés: ha bináris több-egy kapcsolatról van szó, akkor van jobb megoldás is, az amit ODL-nél csináltunk:
ha az ***E*** és ***F*** közti kapcsolat ***F*** felé egyirányú, akkor az ***E*** egyedhalmaz átírásakor bevesszük az ***F*** osztály kulcsát is. Ez ugyanazért lesz jó, miért az ODL-es és így eggyel kevesebb tábla lesz.

Így az előbbi E/K diagram esetén nem kell külön tábla a kapcsolatnak, hanem a

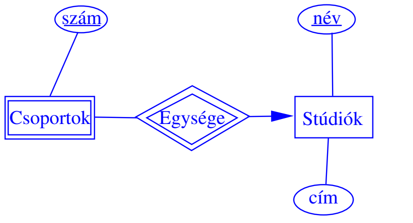
Film(cím, év, hossz, szalagFajta, **stúdióNév**)

lesz a Film tábla.

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 és amelyben így szerepel *W* kulcsa.
- A dupla keretes kapcsolatokhoz nem kell külön reláció (mert az az infó már egyszer szerepel a gyenge egyedhalmaz megadásánál).



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

`Egység(szám, stúdióNév, név)` ⟹ `Egység(szám, név)` hiszen ugyanaz kétszer.
⟹ `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 ODL-ben

`Film, Rajzfilm, Krimi, KrimiRajzfilm` esete. *(Itt minden egyed pont egy osztályban lehet benne, ezért kellett KrimiRajzfilm osztályt is megadni, pl. a Macskafogó miatt.)*

Minden alosztályhoz egy reláció, minden attribútumával és kapcsolatával (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)
```

Kulcs: a főosztálynál úgy, ahogy eddig volt, az alosztály meg örökli a kulcsot, ha tudja

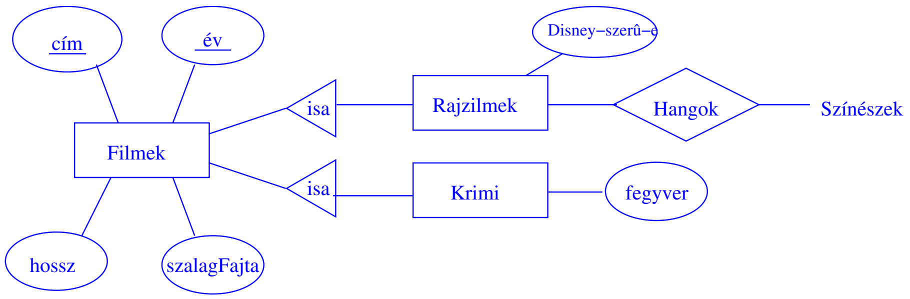
Hátrány: egy film kereséséhez mind a négy relációt végig kell nézni, ha nem tudjuk, hogy hova tartozik a keresett film.

Megjegyzés: ebben a példában a kulcsok elromlanak a többes kapcsolatok miatt

Alosztályok kezelése E/K modellben

E/K-ban nem kell egy egyednek egyetlen egyedhalmazban lennie (ezért nem lesz itt KrimiRajzfilm egyedhalmaz), előfordulhat, hogy egy filmre vonatkozó információk szét vannak szórva.
A relációs sémára való átíráskor gondoskodunk róla, hogy a részinfókból vissza tudjuk állítani az egészet.

Átírás: Minden alosztályhoz csak a főosztály kulcsát és saját attribútumait rendeljük.
Az alosztály kulcsa a főosztály kulcsa lesz, így a kapcsolatba is ezt viszi magával az alosztály.
Az “isa” kapcsolathoz nem rendelünk relációt.



⟹
```
Film(cím, év, hossz, szalagFajta)
Rajzfilm(cím, év, Disney-szerű-e)
Krimi(cím, év, fegyver)
Hangok(cím, év, Színésznév)
```

Hátránya: egy film információi több helyre vannak szórva (pl. Macskafogónál: a hossz és a szalagfajta a Film-ben, az, hogy nem Disney-is, az a Rajzfilmben, hangok a Hangokban. De ezeket az infókat össze lehet rakni, a (cím, év) kulcs menti természetes illesztéssel).

Másik megoldás NULL értékkel

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

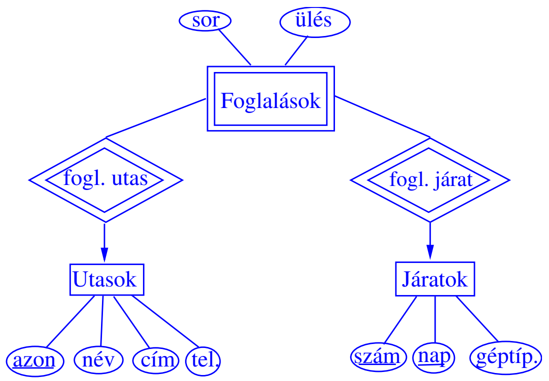
A hiányzó helyeket NULL-al töltjük ki.

Hátrány:

1. elveszíthetünk információt. Pl. egy olyan krimiről, amiben nincs fegyver, nem tudjuk, hogy krimi
2. a (cím, év) pár nem lesz kulcs, ugyanúgy, ahogy az ODL-es átírásnál sem lett

Példák

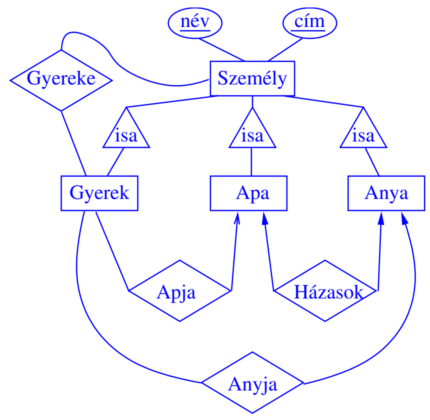
Alakítsuk relációssá:




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

Megjegyzés: ha a Foglalások nem gyenge egyedhalmaz lett volna, hanem kapcsolat a két egyedhalmaz között, akkor is ugyanez a séma relációs jött volna ki.

Példák




```
Személy(név, cím); Gyerek(név, cím); Apa(név, cím)
Anya(név, cím)
Apja(apanév, apacím, gyereknév, gyerekcím)
Anyja(anyanév, anyacím, gyereknév, gyerekcím)
Házast(fnév, fcím, nőnév, nőcím)
Gyereke(gyereknév, gyerekcím, szülőnév, szülőcím)
```