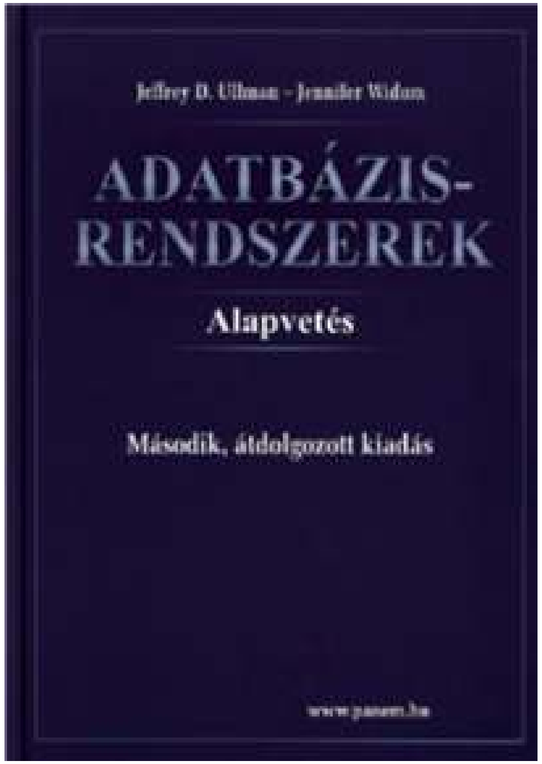
Relációs adatbázisok tervezése ---1

Tankönyv: Ullman-Widom:
Adatbázisrendszerek Alapvetés
Második, átdolgozott kiadás,
Panem, 2009



3.3.1. Bevezetés: anomáliák
3.3.2. Relációk felbontása
3.1. Funkcionális függőségek
3.2. Funkcionális függőségekre vonatkozó szabályok, Armstrong-axiómák, attribútumhalmazok lezártjának kiszámítására algoritmus

Relációs adatmodell története

- E.F. Codd 1970-ban publikált egy cikket
  A Relational Model of Data for Large Shared Data Banks
  Link: http://www.seas.upenn.edu/~zives/03f/cis550/codd.pdf

  amelyben azt javasolta, hogy az adatokat táblázatokban, **relációkban** tárolják. Az elméletére alapozva jött létre a relációs adatmodell, és erre épülve jöttek létre a relációs adatmodellen alapuló (piaci) relációs adatbázis-kezelők.
- Relációs sématervezés: függőségeken alapuló felbontás, normalizálás: Ezen a kurzuson a funkcionális függőségen alapuló Boyce_Codd normálformát (BCNF) és a 3NF-t, illetve a többértékű függőségen alapuló 4normálformát tanuljuk, és megvizsgáljuk a felbontások tulajdonságait (veszteségmentesség, függőségőrzés).

Tankönyv 3.fejezet: Bevezető példa

- Több tábla helyett -> vegyük egy táblában lenne Melyik séma jobb?

Sörivó(<u>név</u>, cím, <u>kedveltSörök</u>, gyártó, kedvencSör)

név	cím	kedveltSörök	gyártó	kedvencS
Janeway	Voyager	Bud	A.B.	WickedAle
Janeway	???	WickedAle	Pete's	???
Spock	Enterprise	Bud	???	Bud

- Redundáns adat, a ??? helyén mi szerepel, ha mindenkinek csak egy lakcíme és kedvencSöre lehet, vagyis a név meghatározza a címet és a kedvencSör-t és a Söröknek is egy gyártója lehet

Hibás tervezés problémái

A rosszul tervezettség anomáliákat is eredményez

Sörivó(<u>név</u>, cím, <u>kedveltSörök</u>, gyártó, kedvencSör)

név	cím	kedveltSörök	gyártó	kedvencS
Janeway	Voyager	Bud	A.B.	WickedAle
Janeway	???	WickedAle	Pete's	???
Spock	Enterprise	Bud	???	Bud

- Módosítási anomália: ha Janeway-t Jane-re módosítjuk, megtesszük-e ezt minden sornál?
- Törlési anomália: Ha senki sem szereti a Bud sört, azzal töröljük azt az infót is, hogy ki gyártotta.
- Beillesztési anomália: és felvinni ilyen gyártót?

Relációs sémák tervezése

- Cél: az anomáliák és a redundancia megszüntetése.
  - Módosítási anomália : egy adat egy előfordulását megváltoztatjuk, más előfordulásait azonban nem.
  - Törlési anomália : törléskor olyan adatot is elveszítünk, amit nem szeretnénk.
  - Beillesztési anomália : megszorítás, trigger kell, hogy ellenőrizni tudjuk (pl. a kulcsfüggőséget)
  - Redundancia (többszörös tárolás feleslegesen)

Dekomponálás (felbontás)

A fenti problémáktól dekomponálással (felbontással) tudunk megszabadulni:

- Definíció:
  $d=\{R_1,...,R_k\}$ az (R,F) dekompozíciója, ha nem marad ki attribútum, azaz $R_1\cup...\cup R_k=R$.
  Az adattábla felbontását projekcióval végezzük.
- Példa:
  $R=ABCDE$, $d=\{AD,BCE,ABE\}$
  3 tagú dekompozíció, ahol
  $R_1=AD$, $R_2=BCE$, $R_3=ABE$,

Felbontásra vonatkozó elvárások

- Elvárások

(1) A vetületek legyenek jó tulajdonságúak, és a vetületi függőségi rendszere egyszerű legyen (normálformák: BCNF, 3NF, 4NF, később)

(2) A felbontás is jó tulajdonságú legyen, vagyis ne legyen információvesztés: Veszteségmentes legyen a felbontás (VM)

(3) Függőségek megőrzése a vetületekben (FŐ)

- Tételek (ezekre nézünk majd algoritmusokat)
  - Mindig van VM BCNF-ra való felbontás
  - Mindig van VM FŐ 3NF-ra való felbontás

Relációs sématervezés (vázlat)

- I. Függőségek: a sématervezésnél használjuk
  - Funkcionális függőség (10ea_RelTerv1FF)
  - Többértékű függőség (12ea_RelTerv3NF)
- II. Normalizálás: „jó” sémákra való felbontás
  - Funkcionális függ. -> BCNF (11ea_RelTerv2VM)
  - Funkcionális függ. -> 3NF (12ea_RelTerv3NF)
  - Többértékű függ. -> 4NF (12ea_RelTerv3NF)
- III. Felbontás tulajdonságai: „jó” tulajdonságok
  - Veszteségmentesség (11ea_RelTerv2VM)
  - Függőségőrző felbontás (12ea_RelTerv3NF)

Funkcionális függőségek

- X ->Y *az R* relációra vonatkozó megszorítás, miszerint ha két sor megegyezik *X* összes attribútumán, *Y* attribútumain is meg kell, hogy egyezzenek.
  - Jelölés: *X*, *Y*, *Z* ,… attribútum halmazokat; *A*, *B*, *C*,… attribútumokat jelöl.
  - Jelölés: {*A*,*B*,*C* } attribútum halmaz helyett *ABC*-t írunk.

Funkcionális függőségek (definíció)

Definíció. Legyen R(U) egy relációséma, továbbá X és Y az U attribútum-halmaz részhalmazai. X-től *funkcionálisan függ* Y (jelölésben $X \rightarrow Y$), ha bármely R feletti T tábla esetén valahányszor két sor megegyezik X-en, akkor megegyezik Y-on is $\forall$ t1,t2$\in$T esetén (t1[X]=t2[X]$\Rightarrow$ t1[Y]=t2[Y]).

Ez lényegében azt jelenti, hogy az X-beli attribútumok értéke egyértelműen meghatározza az Y-beli attribútumok értékét.

- Jelölés: R |= $X \rightarrow Y$ vagyis
  R kielégíti $X \rightarrow Y$ függőséget

Példa: Funkcionális függőség

Sörivók(név, cím, kedveltSörök, gyártó, kedvencSör)

Feltehetjük például, hogy az alábbi FF-ek teljesülnek:

név	cím	kedveltSörök	gyártó	kedvencSör
Janeway	Voyager	Bud	A.B.	WickedAle
Janeway	Voyager	WickedAle	Pete's	WickedAle
Spock	Enterprise	Bud	A.B.	Bud

Mert név -> cím

Mert név -> kedvencSör

Mert kedveltSörök -> gyártó

Jobboldalak szétvágása (FF)

- X->A_1A_2…A_n akkor és csak akkor teljesül R relációra, ha X->A_1, X->A_2,…, X->A_n is teljesül R-en.
- Példa: A->BC ekvivalens A->B és A->C függőségek kettősével.
- Baloldalak szétvágására nincs szabály!!!
- Ezért elég nézni, ha a FF-k jobboldalán egyetlen attribútum szerepel

Kulcs, szuperkulcs

- Funkcionális függőség $X \rightarrow Y$ speciális esetben, ha $Y = U$, ez a kulcsfüggőség.
- R(U) relációséma esetén az U attribútum-halmaz egy K részhalmaza akkor és csak akkor szuperkulcs, ha a $K \rightarrow U$ FF teljesül.
- A kulcsot tehát a függőség fogalma alapján is lehet definiálni: olyan K attribútum-halmazt nevezünk kulcsnak, amelytől az összes többi attribútum függ (vagyis szuperkulcs), de K-ból bármely attribútumot elhagyva ez már nem teljesül (vagyis minimális szuperkulcs)

Példa: szuperkulcs, kulcs

Sörivók(név, cím, kedveltSörök, gyártó, kedvencSör)

- {név, kedveltSörök} szuperkulcs, ez a két attr. meghatározza funkcionálisan a maradék attr-kat.

  név -> cím kedvencSör

  kedveltSörök -> gyártó

- {név, kedveltSörök} kulcs, hiszen sem {név}, sem {kedveltSörök} nem szuperkulcs.

  név -> gyártól; kedveltSörök -> cím nem telj.

- Az előbbi kívül nincs több kulcs, de számos szuperkulcs megadható (ami ezt tartalmazza)

Másik példa (több kulcs is lehet)

- Legyen ABC sémán def.FF-ek: $AB \text{->} C$ és $C \text{->} B$.
  - Példa1: A = utca, B = város, C = irányítószám.
  - Példa2: A = oktató, B = időpont, C = kurzus.
- Itt két kulcs is van: $\{A,B\}$ és $\{A,C\}$.

Az implikációs probléma

- Legyenek X_1 -> A_1, X_2 -> A_2,…, X_n -> A_n adott FF-k, szeretnénk tudni, hogy Y -> B teljesül-e olyan relációkra, amire az előbbi FF-k teljesülnek.
  - Példa: A -> B és B -> C teljesülése esetén A -> C biztosan teljesül.
- Implikációs probléma eldöntése definíció alapján (minden előfordulásra ellenőrizni) túl nehéz, de van egyszerűbb lehetőség: levezetési szabályok segítségével, lásd Armstrong-axiómák.

Armstrong-axiómák

Legyen R(U) relációséma és $X,Y \subseteq U$, és jelölje XY az X és Y attribútum-halmazok egyesítését.

F legyen funkcionális függőségek tetsz. halmaza.

Armstrong axiómák:

- A1 (reflexivitás): $Y \subseteq X$ esetén $X \rightarrow Y$.
- A2 (bővíthetőség): $X \rightarrow Y$ és tetszőleges Z esetén $XZ \rightarrow YZ$.
- A3 (tranzitivitás): $X \rightarrow Y$ és $Y \rightarrow Z$ esetén $X \rightarrow Z$.

Levezetés fogalma

- F implikálja $X \rightarrow Y$-t (F-nek következménye $X \rightarrow Y$), ha minden olyan táblában, amelyben F összes függősége teljesül, azokra $X \rightarrow Y$ is teljesül.

  Jelölés: $F \models X \rightarrow Y$, ha F implikálja $X \rightarrow Y$ –et.

- $X \rightarrow Y$ levezethető F-ből, ha van olyan $X_1 \rightarrow Y_1, \ldots, X_k \rightarrow Y_k, \ldots, X \rightarrow Y$ véges levezetés, hogy $\forall k$-ra $X_k \rightarrow Y_k \in F$ vagy $X_k \rightarrow Y_k$ az FD1,FD2,FD3 axiómák alapján kapható a levezetésben előtte szereplő függőségekből.

  Jelölés: $F \vdash X \rightarrow Y$, ha $X \rightarrow Y$ levezethető F-ből

További levezethető szabályok:

4. Szétvághatósági (vagy felbontási) szabály
   $F \vdash X \rightarrow Y$ és $Z \subseteq Y$ esetén $F \vdash X \rightarrow Z$.
5. Összevonhatósági (vagy unió) szabály
   $F \vdash X \rightarrow Y$ és $F \vdash X \rightarrow Z$ esetén $F \vdash X \rightarrow YZ$.
6. Pszeudotranzitivitás
   $F \vdash X \rightarrow Y$ és $F \vdash WY \rightarrow Z$ esetén $F \vdash XW \rightarrow Z$.

Bizonyítás (4): Reflexivitási axióma miatt $F \vdash Y \rightarrow Z$, és tranzitivitási axióma miatt $F \vdash X \rightarrow Z$.

Bizonyítás (5): Bővítési axióma miatt $F \vdash XX \rightarrow YX$ és $F \vdash YX \rightarrow YZ$, és $XX=X$, valamint a tranzitivitási axióma miatt $F \vdash X \rightarrow YZ$.

Bizonyítás (6): Bővítési axióma miatt $F \vdash XW \rightarrow YW$, és $YW=WY$, és a tranzitivitási axióma miatt $F \vdash XW \rightarrow Z$.

Szétvághatóság/összevonhatóság

- A szétvághatósági és összevonhatósági szabályok következménye:

  $F \vdash X \rightarrow Y \Leftrightarrow \forall A_i \in Y$ esetén $F \vdash X \rightarrow A_i$

- A következmény miatt feltehető, hogy a függőségek jobb oldalai 1 attribútumból állnak.

- Fontos! A függőségeknek csak a jobboldalát lehet szétbontani, a baloldalra ez természetesen nem igaz (például {filmcím, év} $\rightarrow$ stúdió)

Armstrong-axiómák (tétel)

- TÉTEL: Az Armstrong-axiómarendszer helyes és teljes, azaz minden levezethető függőség implikálódik is, illetve azok a függőségek, amelyeket F implikál azok levezethetők F-ből.

$$F \vdash X \rightarrow Y \Leftrightarrow F \models X \rightarrow Y$$

Implikáció eldöntése --- Lezárással

- Implikációs probléma:
  Legyenek X_1 -> A_1, X_2 -> A_2,…, X_n -> A_n adott FF-k, szeretnénk tudni, hogy Y -> B teljesül-e minden olyan relációkra, amire az előbbi FF-k teljesülnek. Hogyan tudjuk ellenőrizni, hogy egy előfordulás nem teljesíti Y -> B ?
- Mivel az Armstrong axiómarendszer helyes és teljes, elegendő a levezetési szabályokkal levezetni. Még a levezetési szabályoknál is van egyszerűbb út: kiszámítjuk *Y lezártját*: Y^+-t
- Attribútum-halmaz lezárására teszt:

Attribútumhalmaz lezártja (definíció)

- Adott *R* séma és *F* funkcionális függőségek halmaza mellett, X^+ az összes olyan *A* attribútum halmaza, amire *X->A* következik *F*-ből.
- (R,F) séma esetén legyen $X \subseteq R$.
- Definíció: $X^{+(F)} := \{A \mid F \vdash X \rightarrow A\}$ az X attribútum-halmaz lezárása F-re nézve.

Attribútumhalmaz lezártja (lemma)

- LEMMA: $F \vdash X \to Y \Leftrightarrow Y \subseteq X^+$.

  Bizonyítás:

  ($\Rightarrow$) $\forall A \in Y$ esetén a reflexivitás és tranzitivitás miatt $F \vdash X \to A$, azaz $Y \subseteq X^+$.

  ($\Leftarrow$) $\forall A \in Y \subseteq X^+$ esetén $F \vdash X \to A$, és az egyesítési szabály miatt $F \vdash X \to Y$.

- Lemma következménye: az implikációs probléma megoldásához elég az X^+-t hatékonyan kiszámolni.

Algoritmus X^+attr.halmaz lezártja

- Input: X attribútumhz., F funk.függőségek hz.
- Output: X^+ (zárás, típusa: attribútumhalmaz)
- Algoritmus X^+ kiszámítására:

  /* Iteráció, amíg X(n) változik */

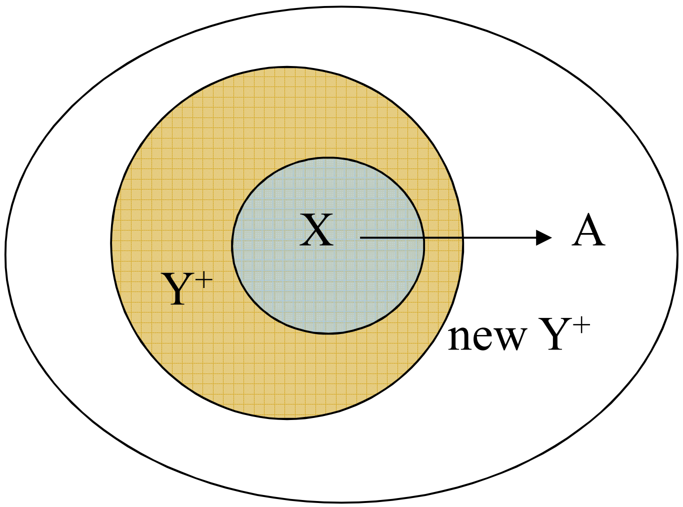
  $X(0):=X$

  $X(n+1):= X(n) \cup \{A \mid \underline{W \rightarrow Z \in F}, A \in Z, W \subseteq X(n)\}$

  Ha $X(v+1)=X(v)$, akkor Output: $X(v)=X^+$.
- Miért működik az X^+ lezárási algoritmus? (Tankönyv 3.2.5. szakasz, 81-83.oldal)

Lezárás (teszt)

- Kiindulás: $Y^+ = Y$
- Indukció: Olyan FF-ket keresünk, melyeknek a baloldala már benne van Y^+-ban. Ha $X \to A$ ilyen, A-t hozzáadjuk Y^+-hoz.



A lezárást kiszámító algoritmus „helyes”

- Az algoritmus „tényleg” X^+-t számítja ki. Vagyis:
  1. Ha az A attribútum valamely *j*-re belekerül X^j-be, akkor A valóban eleme X^+-nak.
  2. Másfelől, ha A $\in X^+$, akkor létezik olyan *j*, amire A belekerül X^j-be.
- Az első állítás: Miért csak az igaz funkcionális függőségeket fogadja el a lezárási algoritmus? Könnyen bizonyítható indukcióval [Tk.3.2.5.]

A lezárást kiszámító algoritmus „teljes”

- A második állítás: Miért talál meg minden igaz függőséget a lezárási algoritmus? [Tk.3.2.5.]
- Konstrukciós bizonyítás: Tegyük fel, hogy $A \in X^+$, és nem olyan *j*, amire A belekerül X^j-be.

	X^+ elemei	más attribútumok
t	111 … 111	000 … 000
s	111 … 111	111 … 111

- Ekkor I-re minden F^+-beli függőség teljesül
- I-re viszont nem teljesül $X \rightarrow A$

Példa: Attribútumhalmaz lezárása

R=ABCDEFG, {AB→C, B→G, CD→EG, BG →E}

X=ABF, X^+=?

X(0):=ABF

X(1):=ABF∪{C,G}=ABCFG

X(2):=ABCFG ∪{C,G,E}=ABCEFG

X(3):=ABCEFG

X^+= ABCEFG

Tankönyv 3.5.2. feladata (111.o.)

- Órarend adatbázis: Kurzus(K), Oktató(O), Időpont(I), Terem(T), Diák(D), Jegy(J)
- Feltételek:

  Egy kurzust csak egy oktató tarthat: $K \rightarrow O$.

  Egy helyen egy időben egy kurzus lehet: $IT \rightarrow K$.

  Egy időben egy tanár csak egy helyen lehet: $IO \rightarrow T$.

  Egy időben egy diák csak egy helyen lehet: $ID \rightarrow T$.

  Egy diák egy kurzust egy végső jeggyel zár: $KD \rightarrow J$.
- R=KOITDJ F= $\{K \rightarrow O,\ IT \rightarrow K,\ IO \rightarrow T,\ ID \rightarrow T,\ KD \rightarrow J\}$
- Feladat: Határozzuk meg a (R, F) kulcsait az X^+ kiszámítási algoritmusa segítségével.

Kérdés/Válasz

- Köszönöm a figyelmet! Kérdés/Válasz?

- A gyakorlaton lesz illetve HF: Oracle PL/SQL
- HF: Oracle Példatár 8.fejezetének feladatai: PL/SQL blokk felépítése (deklarációs rész, végrehajtható rész és kivételkezelő rész). Változók használata. Vezérlési szerkezetek, feltételes utasítások, ciklusutasítások, stb.

- Az Eljut-feladat (illetve változatai) megoldása PL/SQL programmal (insert utasítás 2.alakjával és ciklussal) [Segítség: lásd a gyakorlat oldalán]