**FUNKCIONALNE ZAVISNOSTI**

Neka je R relacija i neka su X i Y proizvoljni podskupovi atributa iz R. Tada **Y funkcionalno zavisi od X**, u oznaci X🡪Y akko je svakoj važećoj vrednosti torke X u R pridružena tačno jedna vrednost Y iz R.

Koristi se još i termin **X funkcionalno određuje Y.**

Funkcionalna zavisnost: Ef : f(X) = Y

**Intuitivno**, funkcionalna zavisnost u relaciji R je tvrđenje oblika „Ako su dve torke od R identične na svim atributima A1,A2,...An, tada moraju da imaju iste vrednosti i u ostalim atributima B1,B2,...Bm“.

Dve torke su **identične** ako su im identične vrednosti respektivnih komponenata.

Oznaka A1,...An 🡪 B1,...Bm.

**Primeri**:

* Relacija DOSIJE

{Indeks} 🡪 {Ime}

{Indeks} 🡪 {Prezime}

{Indeks} 🡪 {God\_rodjenja}

{Indeks} 🡪 {Mesto\_rodjenja}

* Relacija PREDMET

{Id\_predmeta} 🡪 {Sifra}

{Id\_predmeta} 🡪 {Naziv}

{Id\_predmeta} 🡪 {Bodovi}

* Relacija ISPIT

{Indeks, Id\_predmeta, Godina\_roka, Oznaka\_roka} 🡪 {Ocena}

{Indeks, Id\_predmeta, Godina\_roka, Oznaka\_roka} 🡪 {Datum\_ispita}

* Relacija ISPITNI\_ROK

{Godina\_roka, Oznaka\_roka} 🡪 {Naziv}

**Ključevi relacije i funkcionalne zavisnosti**

Redefinicija ključeva: Podskup atributa X ⊆ R relacije R je **kandidat za ključ** relacije R ako važi:

1. ∀Y : Y = R\X => X 🡪 Y
2. !E Z, W : (Z e X) ^ (W = R\Z) ^ (Z 🡪 W)

**Nadključ (superključ**) relacije R je skup atributa koji uključuje kao podskup bar jedan kandidat za ključ relacije R.

**Napomene**:

* Svaka FZ predstavlja **ograničenje integriteta**. **Posledica**: svaki atribut relacije funkcionalno zavisi od nekog kandidata za ključ.
* FZ NE ZAVISI od trenutne vrednosti relvara i zbog toga nisu FZ {Oznaka\_roka} 🡪 {Naziv} i {Naziv} 🡪 {Oznaka\_roka}
* Dva skupa funkcionalnih zavisnosti S i T nad relacijom R su **ekvivalentni** ako je skup instanci relacjie R koji zadovoljava S jednak skupu instanci relacije R koji zadovoljava T.
* U opštem slučaju, skup S FZ se izvodi iz skupa T FZ ako svaka instanca relacije koja zadovoljava sve FZ iz T takođe zadovoljava sve FZ iz S. **Posledica**: dva skupa FZ S i T su **ekvivalentna** akko se svaka FZ u S izvodi iz FZ T i svaka FZ u T se izvodi iz FZ u S.

**Pravila i zatvorenja**

**Dekompozicija** **FZ**

A1, A2,...An 🡪 B1, B2,…Bm

je ekvivalentna sa skupom FZ

A1,…An 🡪 B1

A1,…An 🡪 B2

…

A1,…An 🡪 Bm

**Kompozicija FZ**

A1,…An 🡪 B1

A1,…An 🡪 B2

…

A1,…An 🡪 Bm

je ekvivalentna sa FZ

A1,…An 🡪 B1,…Bm

**Trivijalne I netrivijalne FZ**

**Trivijalna FZ** je FZ koja ne može a da ne bude zadovoljena za bilo koji skup vrednosti u relaciji.

* Y ⊆ X => FZ X 🡪 Y je trivijalna

FZ koja nije trivijalna je **netrivijalna.**

**Primeri :**

{Indeks} 🡪 {Indeks}

{Godina\_roka, Oznaka\_roka} 🡪 {Godina\_roka, Oznaka\_roka}

{Godina\_roka, Oznaka\_roka} 🡪 {Godina\_roka}

{Godina\_roka, Oznaka\_roka} 🡪 {Oznaka\_roka}

{id\_predmeta} 🡪 {Id\_predmeta}

…

**Zatvorenje skupa FZ**

**Definicija**: Neka je S skup FZ nad relacijom R. Skup svih FZ koje mogu da se izvedu iz skupa S se naziva **zatvorenje od S** i označava sa S+.

**Posledica**: dva skupa funkcionalnih zavisnosti S i T nad relacijom R su **ekvivalentna** ako važi S+ = T+.

**Određivanje zatvorenja skupa FZ**

Zatvorenje S+ skupa funkcionalnih zavisnosti S može da se odredi primenom pravila – **Armstrongovim aksiomama** kojima se nove FZ izvode iz postojećih.

Neka su A, B i C proizvoljni podskupovi atributa relacije R. Tada važe pravila:

* Refleksivnost: B ⊆ A => A 🡪 B
* Proširenje: А 🡪 B => AC 🡪 BC
* Tranzitivnost: A 🡪 B ^ B 🡪 C => A 🡪 C

**Dodatna pravila (izvode se iz prethodna tri):**

* Samo-određenje: A 🡪 A
* Dekompozicija: A 🡪 BC => A 🡪 B ^ A 🡪 C
* Unija: A 🡪 B ^ A 🡪 C => A 🡪 BC
* Kompozicija: A 🡪 B ^ C 🡪 D => AC 🡪 BD
* Opšta teorema unifikacije: A 🡪 B ^ C 🡪 D => A U (C - B) 🡪BD

**Primer algoritma za računanje zatvorenja skupa funkcionalnih zavisnosti F:**

**Inicijalno** F+ = F

**repeat**

**za svaku** FZ f e F+:

primeniti refleksivnost I prosirenje na f

dobijene FZ smestiti u F+

**za svaki par** FZ (f1, f2) e F+:

ako f1 i f2 mogu da se kombinuju pomocu tranzitivnosti

tada dodati dobijenu FZ u F+

**until** vise ne bude promena u F+

Primer : vidi slajdove

**Zatvorenje skupa atributa**

Često treba izračunati da li data FZ pripada zatvorenju skupa FZ.

U praksi se određivanje skupa FZ retko radi (algoritam je neefikasan)

Da bismo odredili da li je K **nadključ** treba odrediti skup atributa relacije R koji funkcionalno zavisi od K. Tačnije, **ako** je skup atributa K **nadključ** **tada** K funkcionalno određuje sve atributa u R.

K je **nadključ** **akko** je K+ od K (u odnosu na dati skup FZ) **jednako** skupu svih atributa relacije R.

**Definicija**: Neka je A = {A1,…An} skup atributa relacije R I S skup FZ nad R. **Zatvorenje skupa atributa A**  **u odnosu na skup FZ S** je skup atributa B takvih da svaka relacija koja zadovoljava sve FZ u skupu S zadovoljava i FZ A1,...An 🡪 B.

**Zatvorenje skupa** atributa A se označava sa A+.

Uvek važi da su **pojedinačni atributi iz A u A+.**

**Određivanje zatvorenja skupa atributa - algoritam**

Neka je A = {A1,…An} skup atributa i S skup FZ. Algoritam za određivanje zatvorenja A+ je:

1. Izvršiti dekompoziciju svih FZ tako da imaju samo jedan atribut na desnoj strani.
2. Neka je X skup atributa koji predstavljaju zatvorenje. Inicijalno X = {A1,…An}
3. Tražiti FZ oblika B1,...Bm 🡪 C takve da ∀i : Bi ⊂ X ^ C !⊂ X. Dodati C u X. Ponavljati sve dok ima promena u X.
4. Ukoliko ne postoji atribut koji bi mogao da se doda skupu X tada je X = {A1, A2,...An}+.

Primer: Videti slajdove

**Pokrivač skupa FZ**

Ako su S1 i S2 dva skupa FZ i ako je svaka FZ iz S1 uključena u S2, tj. ako je S1+ podskup od S2+ tada je S2 **pokrivač** za S1.

Ako je S1 pokrivač od S2 i S2 pokrivač od S1 tada su S1 i S2 **ekvivalentni**.

Ako RSUPB obezbedi FZ u S2 tada će automatski biti obezbeđene i u S1.

**Nereducibilni skup FZ**

**Definicija**: FZ X 🡪 Y u skupu S funkcionalnih zavisnosti je **levo-nereducibilna** ako iz X ne može da se ukloni nijedan atribut bez promene zatvorenja S+.

Zašto nereducibilni skup FZ?

Nadključ K je **kandidat za ključ** relacije R **akko** je njen **nereducibilni nadključ**

**Definicija**: Skup FZ je **nereducibilan** ako:

* Desna strana svake FZ u S sadrži tačno jedan atribut
* Leva strana svake FZ je levo-nereducibilna
* Nijedna FZ ne može da se ukloni iz S bez promene S+

Skup FZ koji zadovoljava prethodna pravila naziva se i **minimalni ili kanonički.**

**Nereducibilni skup FZ – algoritam**

1. Koristeći pravilo dekompozicije prepisati sve FZ u S tako da desna strana sadrži tačno jedan atribut.
2. Eliminisati sve redundantne FZ iz skupa funkcionalnih zavisnosti dobijenih prethodnim postupkom.

Primer: videti slajdove

**Projekcija FZ**

Neka se relacija R sa skupom funkcionalnih zavisnosti S projektuje na relaciju R1 = pi{R}. Koje FZ su važeće u R1?

Određuje se projekcija funkcionalnih zavisnosti S koja sadrži sve FZ takve da:

* Mogu da se izvedu iz S
* Uključuju jedino atrbute iz R1

**Projekcija FZ – algoritam**

1. Neka je T skup FZ u R1. Inicijalno je T = ∅
2. ∀X⊆R1 odrediti X+ u odnosu na FZ u S. Atributi koji su u R ali ne i u R1 mogu da se koriste u izračunavanju X+. Dodati u T sve netrivijalne FZ X 🡪 A takve da A⊂X+ ∧ A⊂R1
3. Određuje se minimalni skup T na sledeći način:

* Ako postoji F⊂T koja može da se izvede iz drugih FZ iz T, ukloniti FZ F iz T
* Neka je Y 🡪 B FZ u T sa najmanje dva atributa u Y i neka je Z dobijeno iz Y uklanjanjem jednog od atributa. Ako Z 🡪 B može da se izvede iz funkcionalnih zavisnosti u T (uključujući Y 🡪 B), tada se Y 🡪 B zamenjuje sa Z 🡪 B.
* Ponoviti prethodne korake sve dok ima promena u T

Primer: videti slajdove