



# Realizacija ograničenja šeme RBP putem SUBP

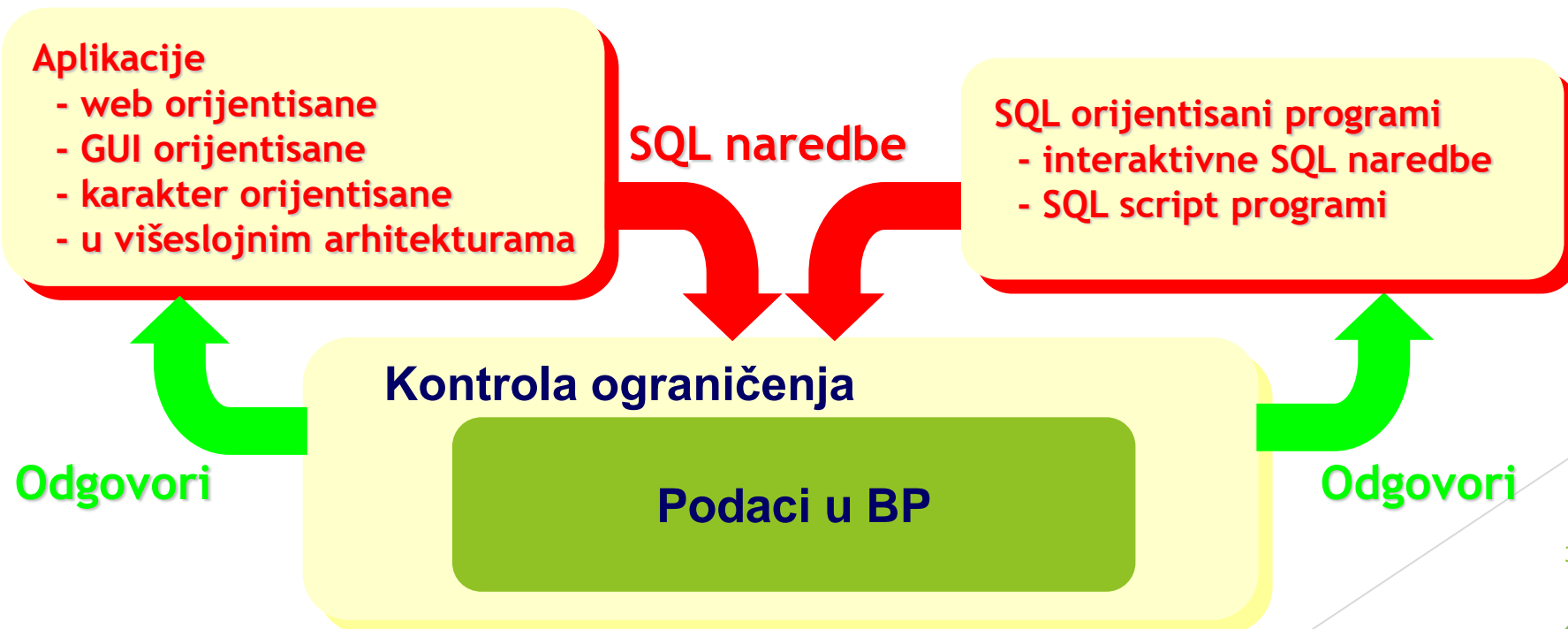
# Sadržaj

- ▶ Uvodne napomene
- ▶ Tipovi ograničenja u RMP
- ▶ Mehanizmi RSUBP
- ▶ Realizacija ograničenja pomoću RSUBP

# Realizacija integritetne komponente šeme BP

## ► Cilj

- Sva ograničenja realizovati na nivou servera BP
  - putem mehanizama koje poseduje SUBP



# Realizacija integritetne komponente šeme BP

- ▶ Kontrola ograničenja, implementiranih na nivou SUBP je centralna
  - ▶ ne može je zaobići ni jedan program ili korisnik
  - ▶ korisnici nisu svesni postojanja ograničenja, dok ne dođe do njegovog narušavanja
  - ▶ u slučaju pokušaja narušavanja ograničenja nekom operacijom ažuriranja, SUBP
    - ▶ aktivnim mehanizmom dovodi stanje BP u konzistentno, ili
    - ▶ izaziva grešku i prekida operaciju
      - ▶ prosleđuje korisničkom programu poruku o grešci
      - ▶ program obrađuje tu poruku i prosleđuje je korisniku

# Realizacija integritetne komponente šeme BP

- ▶ Implementacija ograničenja šeme BP
  - ▶ zadaju se parametri ograničenja
    - ▶ definiše se ograničenje, datog tipa
    - ▶ definišu se operacije (događaji) nad BP koje mogu dovesti do narušavanja ograničenja
    - ▶ za svaku operaciju, definiše se aktivnost (akcija) očuvanja konzistentnosti BP, u slučaju pokušaja narušavanja ograničenja
  - ▶ SUBP obezbeđuje automatsku proveru važenja ograničenja, saglasno definisanim parametrima

# Realizacija integritetne komponente šeme BP

## ► Parametri ograničenja

- svako ograničenje je vezano za određena obeležja i određene šeme relacija šeme BP
- operacije (događaji) koji, načelno, mogu dovesti do narušavanja ograničenja
  - upis nove torke u relaciju,
  - brisanje postojeće torke iz relacije i
  - modifikacija vrednosti postojeće torke u relaciji, nad čijom šemom relacije je ograničenje definisano

# Realizacija integritetne komponente šeme BP

## ► Parametri ograničenja

- akcije očuvanja konzistentnosti BP u slučaju pokušaja narušavanja ograničenja
  - vezuju se za svaku operaciju koja može narušiti ograničenje
  - dele se na
    - pasivne
    - aktivne
    - kombinovane
      - pod određenim uslovima aktivne, a pod drugim uslovima pasivne

# Realizacija integritetne komponente šeme BP

## ► Parametri ograničenja

### ► Pasivne akcije

- sprečavanje operacije koja bi narušila ograničenje

### ► Aktivne akcije

- obezbeđuju automatsko sprovođenje daljih operacija ažuriranja nad BP, koje će obezbediti očuvanje konzistentnosti BP
  - očuvanjem propisanih odnosa između podataka



# Realizacija integritetne komponente šeme BP

- ▶ Implementacija ograničenja šeme BP
  - ▶ ograničenja se implementiraju pomoću mehanizma SUBP
  - ▶ mehanizmu se, direktno ili posredno, pridružuju svi parametri ograničenja
    - ▶ definicija
    - ▶ kritične operacije - koje mogu narušiti ograničenje
    - ▶ za svaku kritičnu operaciju, akcija očuvanja konzistentnosti BP

# Realizacija integritetne komponente šeme BP

- ▶ Kontrola ograničenja šeme BP
  - ▶ SUBP pokreće mehanizam kontrole važenja ograničenja
    - ▶ automatski
    - ▶ nakon izvođenja kritične operacije za ograničenje

# Realizacija integritetne komponente šeme BP

## ► Prednosti

- Automatska kontrola implementiranih ograničenja na nivou SUBP
- Obezbeđena konzistentnost BP u svakom trenutku
  - ne može se narušiti upotrebom neistestiranih aplikacija, ili interaktivnog SQL-a
- Standardizacija načina za implementaciju ograničenja
  - ANSI SQL-92 (SQL2)
  - ANSI SQL:1999 (SQL3)
  - ANSI SQL:2003
  - ANSI SQL:2006 (ISO/IEC 9075-14:2006)

# Realizacija integritetne komponente šeme BP

## ► Prednosti

- Ne postoje funkcionalni razlozi za realizaciju ograničenja unutar programa
  - ograničenja se implementiraju i kontrolišu jedanput, na nivou SUBP
  - ograničenja se ne implementiraju i ne kontrolišu na nivou aplikativnih programa

# Realizacija integritetne komponente šeme BP

## ► Nedostaci

- Povišen stepen zavisnosti šeme BP od proizvođača, tipa i verzije SUBP
  - ne podržavaju svi SUBP, u istoj meri, postojeće standarde
  - standardi, generalno, nisu "idealni" - ne pokrivaju uvek sve neophodne detalje
- Ostaje potreba za realizacijom nekih ograničenja unutar programa
  - zbog obezbeđenja pogodnosti programa za upotrebu
  - dvostruka implementacija i kontrola ograničenja

# Sadržaj

- ▶ Uvodne napomene
- ▶ Tipovi ograničenja u RMP
- ▶ Mehanizmi RSUBP
- ▶ Realizacija ograničenja pomoću RSUBP

# Karakteristike tipa ograničenja

- ▶ **Karakteristike tipa ograničenja u modelu podataka (MP)**
  - ▶ oblast definisanosti
    - ▶ tip logičke strukture obeležja nad kojom se ograničenje definiše
  - ▶ oblast interpretacije
    - ▶ tip logičke strukture podataka nad kojom se ograničenje interpretira
  - ▶ formalizam za zapisivanje (definicija)
  - ▶ pravilo za interpretaciju (validaciju)

# Karakteristike tipa ograničenja

- ▶ **Karakteristike tipa ograničenja u modelu podataka (MP)**
  - ▶ skup kritičnih operacija nad bazom podataka
    - ▶ koje mogu dovesti do narušavanja ograničenja datog tipa
  - ▶ skup mogućih akcija kojima se obezbeđuje očuvanje validnosti baze podataka
    - ▶ pri pokušaju narušavanja ograničenja datog tipa
    - ▶ definiše se za svaku kritičnu operaciju



# Tipovi ograničenja u RMP

- ▶ **Mogući tipovi ograničenja u RMP**
  - ▶ ograničenje domena
  - ▶ ograničenje vrednosti obeležja
  - ▶ ograničenje torke
  - ▶ ograničenje ključa
  - ▶ ograničenje jedinstvenosti
  - ▶ zavisnost sadržavanja
  - ▶ ograničenje referencijalnog integriteta
  - ▶ ...

# Tipovi akcija

## ► Tipovi akcija očuvanja konzistentnosti

- pri pokušaju narušavanja ograničenja BP

### ► Pasivne akcije

#### ► *NoAction (Restrict)*

- zabrana sprovođenja operacije koja bi izazvala narušavanje kontrolisanog ograničenja

### ► Aktivne akcije

#### ► *Cascade*

- kaskadna propagacija operacije
  - na podatke, povezane s podacima koji se ažuriraju i kontrolišu putem ograničenja

#### ► *SetNull*

- svođenje na nula vrednosti
  - podataka, povezanih s podacima koji se ažuriraju i kontrolišu putem ograničenja

# Tipovi akcija

## ► Tipovi akcija očuvanja konzistentnosti

- pri pokušaju narušavanja ograničenja BP

### ► Aktivne akcije

#### ► *SetDefault*

- svođenje na predefinisane (inicijalne) vrednosti
  - podataka, povezanih s podacima koji se ažuriraju i kontrolišu putem ograničenja

#### ► *<<UserDef>>*

- specifikacija korisnički definisane akcije
  - posebno isprojektovane i isprogramirane
  - specificirane putem unapred određene sintakse

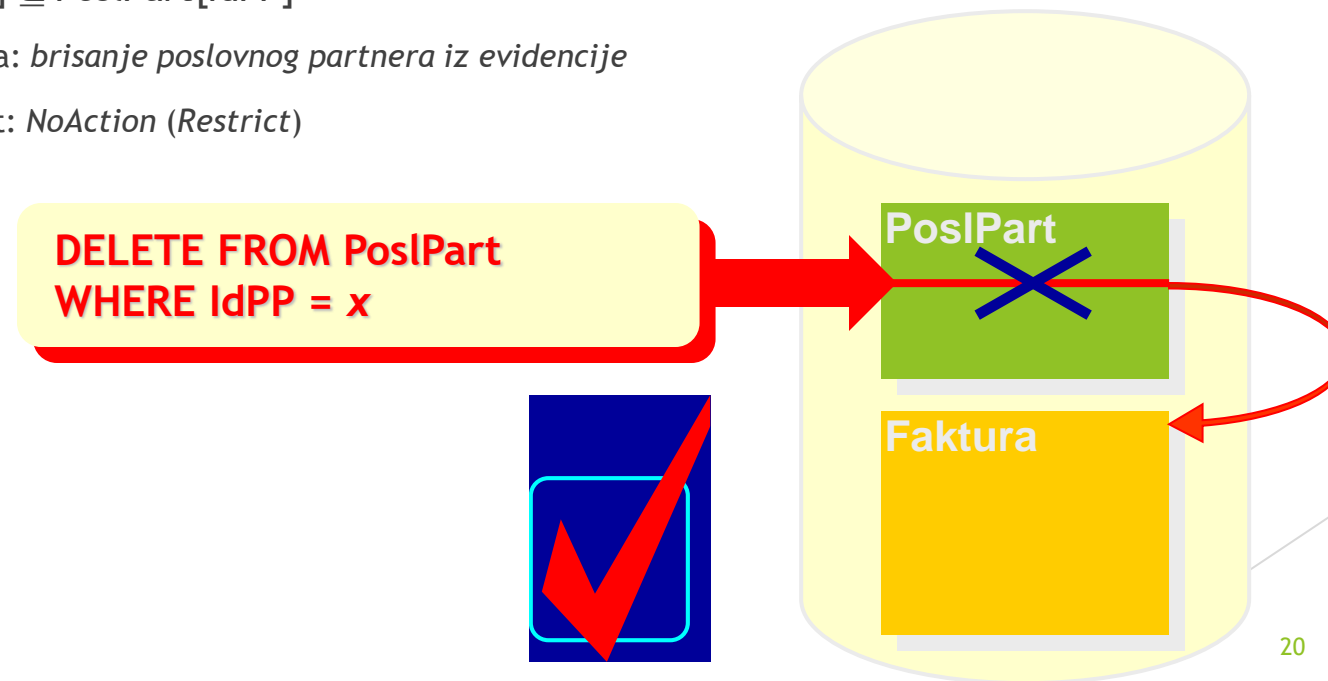
# Primer tipova akcija

## ► Primer

### ► jedno ograničenje i akcija očuvanja konzistentnosti

►  $Faktura[IdPP] \subseteq PoslPart[IdPP]$

- operacija: *brisanje poslovnog partnera iz evidencije*
- aktivnost: *NoAction (Restrict)*



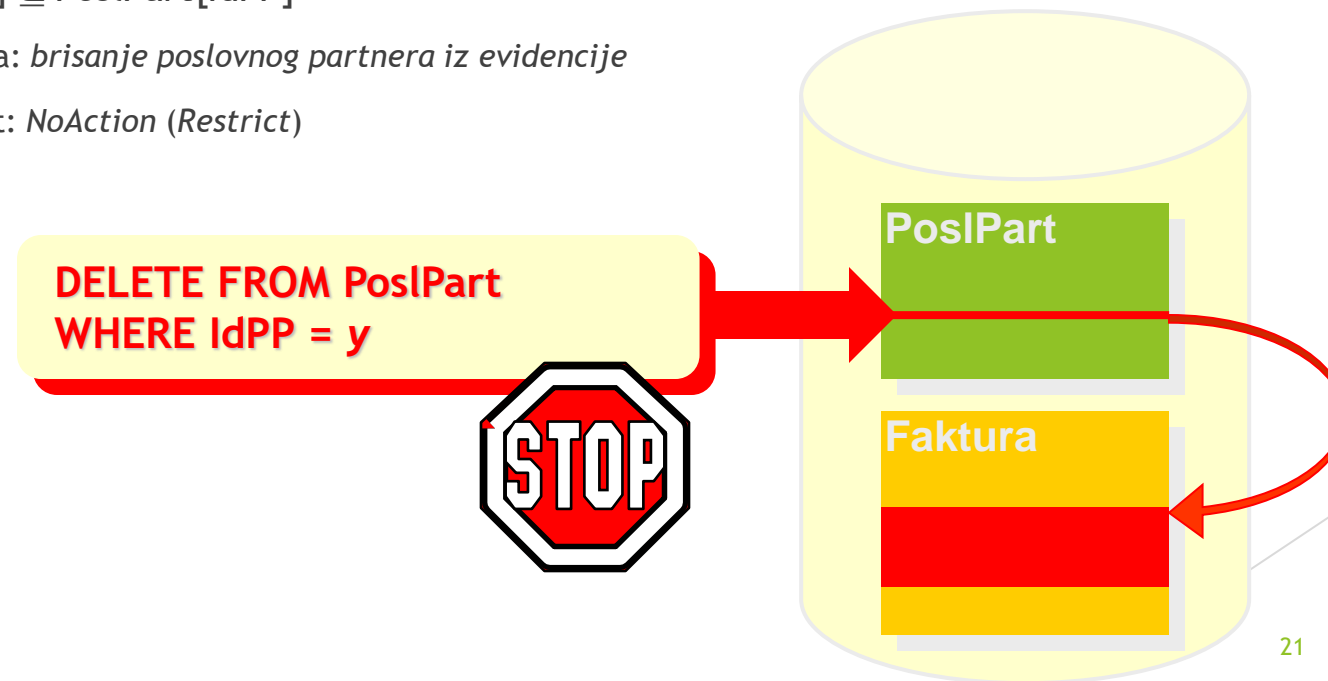
# Primer tipova akcija

## ► Primer

### ► jedno ograničenje i akcija očuvanja konzistentnosti

►  $Faktura[IdPP] \subseteq PoslPart[IdPP]$

- operacija: *brisanje poslovnog partnera iz evidencije*
- aktivnost: *NoAction (Restrict)*



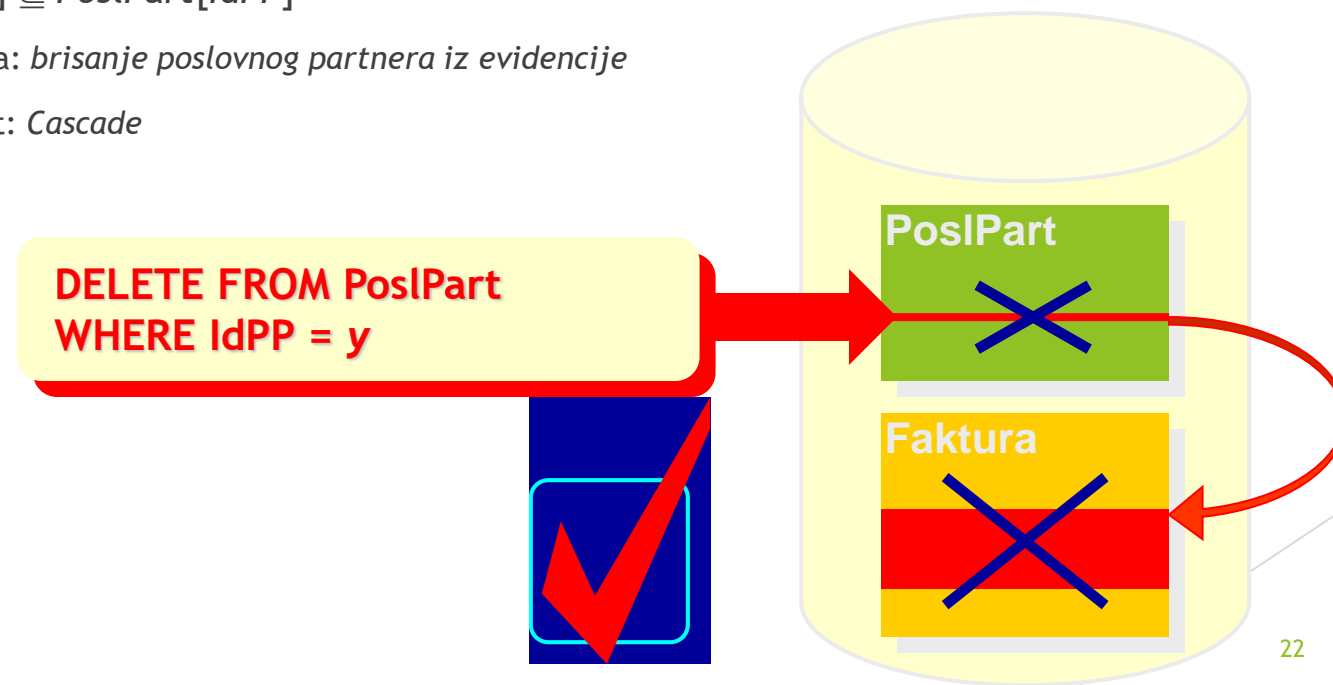
# Primer tipova akcija

## ► Primer

### ► jedno ograničenje i akcija očuvanja konzistentnosti

►  $Faktura[IdPP] \subseteq PoslPart[IdPP]$

- operacija: *brisanje poslovnog partnera iz evidencije*
- aktivnost: *Cascade*



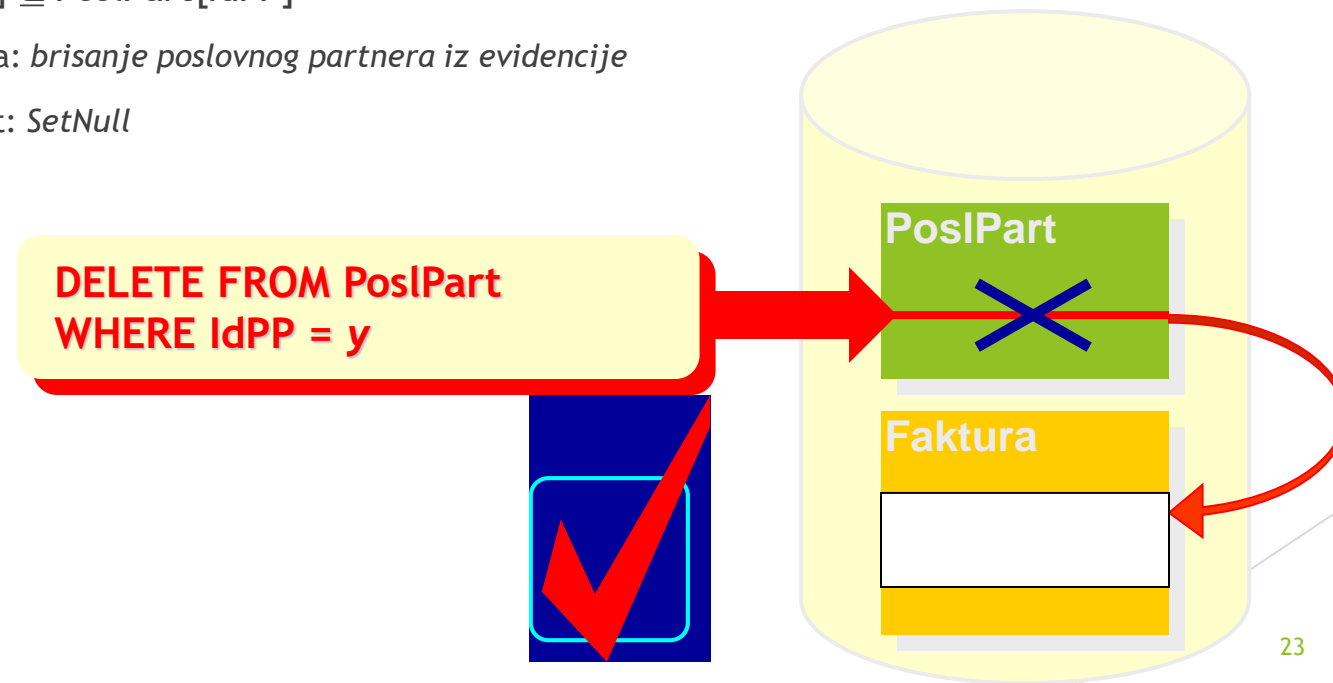
# Primer tipova akcija

## ► Primer

### ► jedno ograničenje i akcija očuvanja konzistentnosti

►  $Faktura[IdPP] \subseteq PoslPart[IdPP]$

- operacija: *brisanje poslovnog partnera iz evidencije*
- aktivnost: *SetNull*



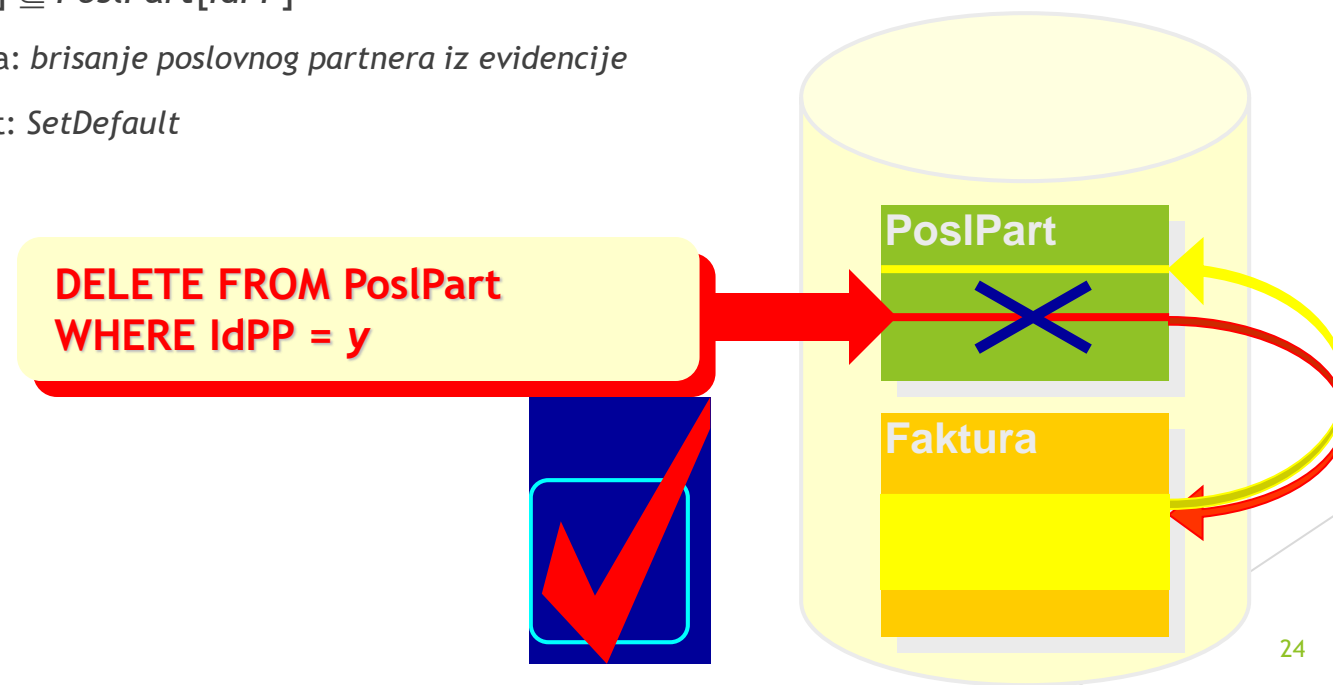
# Primer tipova akcija

## ► Primer

### ► jedno ograničenje i akcija očuvanja konzistentnosti

►  $Faktura[IdPP] \subseteq PoslPart[IdPP]$

- operacija: *brisanje poslovnog partnera iz evidencije*
- aktivnost: *SetDefault*





# Sadržaj

- ▶ Uvodne napomene
- ▶ Tipovi ograničenja u RMP
- ▶ **Mehanizmi RSUBP**
- ▶ Realizacija ograničenja pomoću RSUBP

# Mehanizmi RSUBP

- ▶ Namenjeni za implementaciju
  - ▶ skupa šema relacija šeme BP
  - ▶ ograničenja (integritetne komponente) šeme BP
  - ▶ ostalih pravila poslovanja, koja
    - ▶ ne rezultuju u ograničenjima šeme BP
    - ▶ odnose se, obično, na
      - ▶ unapred definisani redosled, obaveze i uslovljenosti izvođenja operacija nad BP, ili
      - ▶ obavezu izvođenja nekih operacija nad BP, pod određenim uslovima, definisanim putem odnosa vrednosti obeležja u BP

# Mehanizmi RSUBP

## ► Implementacija skupa šema relacija

- kreiranje, modifikovanje i brisanje korisnički definisanog domena
  - CREATE DOMAIN, ALTER DOMAIN, DROP DOMAIN
- kreiranje, modifikovanje i brisanje složenog tipa podatka
  - CREATE TYPE, DROP TYPE
- kreiranje, modifikovanje i brisanje tabele (šeme relacije)
  - CREATE TABLE, ALTER TABLE, DROP TABLE
- dodavanje, modifikovanje i brisanje kolone tabele (obeležja šeme relacije)
  - ALTER TABLE / ADD, MODIFY, DROP

# Mehanizmi RSUBP

## ► Implementacija ograničenja šeme BP

### ► deklarativni mehanizmi

- aktivnosti provere važenja ograničenja i očuvanja konzistentnosti se, većim delom, podrazumevaju
  - SQL klauzula CONSTRAINT
  - CREATE DOMAIN, CREATE ASSERTION

### ► proceduralni mehanizmi

- aktivnosti provere važenja ograničenja i očuvanja konzistentnosti se, većim delom, programiraju
  - putem proceduralnog jezika
  - CREATE TRIGGER
  - CREATE PROCEDURE, CREATE FUNCTION
  - CREATE PACKAGE, CREATE PACKAGE BODY

# Mehanizmi RSUBP

- ▶ **Implementacija pravila poslovanja koja ne rezultuju u ograničenjima šeme BP**
  - ▶ **proceduralni mehanizmi**
    - ▶ uslovi i redosled izvođenja operacija, koji definišu pravilo poslovanja se, većim delom, programiraju
      - ▶ putem proceduralnog jezika
      - ▶ CREATE TRIGGER
      - ▶ CREATE PROCEDURE, CREATE FUNCTION
      - ▶ CREATE PACKAGE, CREATE PACKAGE BODY

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi

- za implementaciju ograničenja na, najvećim delom, podrazumevani način
  - bez programiranja procedure kontrole ograničenja

## ► Proceduralni mehanizmi

- za implementaciju ograničenja koja se ne mogu u potpunosti opisati putem deklarativnih mehanizama
- za implementaciju pravila poslovanja koja ne rezultuju u ograničenjima šeme BP

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► CREATE DOMAIN

- kreiranje domena
- opšti oblik sintakse

```
CREATE DOMAIN Naziv_domena  
[AS] Tip_podatka[(Dužina)]  
[DEFAULT {Konstanta | Funkcija | NULL}]  
[CHECK (Logičkilzraz)]
```

- *Dužina* se navodi kada je to dozvoljeno i potrebno
  - *Logičkilzraz* mora biti izračunljiv
  - Naznaka za bilu koju vrednost u *Logičkomlrazu* je VALUE
- 
- konkretni SUBP ne podržavaju uvek ovaj mehanizam

# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ deklarativno definisanje ograničenja, različitih tipova
    - ▶ predstavlja sastavni deo naredbe CREATE TABLE, ili ALTER TABLE
      - ▶ CREATE TABLE (... , CONSTRAINT ...)
      - ▶ ALTER TABLE ADD CONSTRAINT ...
      - ▶ ALTER TABLE DROP CONSTRAINT ...
  - ▶ većina SUBP podržava ovaj mehanizam u određenoj meri



# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

- opšti oblik sintakse

[CONSTRAINT *NazivOgr*] *SpecifikacijaTipaOgraničenja*

{ CHECK | NOCHECK } CONSTRAINT

{ ALL | *NazivOgraničenja* }

### ► *SpecifikacijaTipaOgraničenja*

- NOT NULL - ograničenje nula vrednosti
- PRIMARY KEY ... - ograničenje primarnog ključa
- UNIQUE ... - ograničenje jedinstvenosti
- CHECK ... - ograničenje torke
- FOREIGN KEY ... - ograničenje stranog ključa

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► NOT NULL

- uvek se zadaje na nivou obeležja šeme relacije (kolone tabele)

#### ► CREATE TABLE

```
(...,  
    Kolona Tip(Dužina) NOT NULL,  
    ...)
```

#### ► CREATE TABLE

```
(...,  
    Kolona Tip(Dužina) CONSTRAINT Naziv NOT NULL,  
    ...)
```

# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ NOT NULL
      - ▶ uvek se zadaje na nivou obeležja šeme relacije (kolone tabele)
      - ▶ proverava se prilikom svakog pokušaja
        - ▶ upisa nove vrednosti obeležja, ili
        - ▶ modifikacije postojeće vrednosti obeležja
      - ▶ u slučaju pokušaja narušavanja ograničenja, jedina moguća aktivnost je sprečavanje operacije (NO ACTION)
    - ▶ Svi savremeni SUBP podržavaju klauzulu NOT NULL

# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ PRIMARY KEY [(Lista\_obeležja)]
      - ▶ zadaje se na nivou obeležja šeme relacije koje jedino predstavlja primarni ključ
        - ▶ bez navođenja liste obeležja, ili
      - ▶ na nivou celokupne šeme relacije (tabele)
        - ▶ sa navođenjem liste obeležja
        - ▶ uobičajeno i opštije rešenje

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► PRIMARY KEY [(Lista\_obeležja)]

- zadavanje na nivou obeležja

#### ► CREATE TABLE

```
(...,  
    Kolona Tip(Dužina) CONSTRAINT Naziv PRIMARY KEY,  
    ...)
```

#### ► CREATE TABLE

```
(...,  
    Kolona Tip(Dužina) PRIMARY KEY,  
    ...)
```

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► PRIMARY KEY [(Lista\_obeležja)]

- zadavanje na nivou šeme relacije

#### ► CREATE TABLE

(lista specifikacija kolona tabele,

...

CONSTRAINT Naziv PRIMARY KEY (Lista\_obeležja),

...)

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► PRIMARY KEY [(Lista\_obeležja)]

- podrazumeva se, bez posebnog deklarisanja, da je svako obeležje u *Lista\_obeležja* deklarirano kao NOT NULL
- proverava se prilikom svakog pokušaja
  - upisa nove vrednosti obeležja ključa, ili
  - modifikacije postojeće vrednosti obeležja ključa
- u slučaju pokušaja narušavanja ograničenja, jedina moguća aktivnost je sprečavanje operacije (NO ACTION)

#### ► Svi savremeni SUBP podržavaju klauzulu PRIMARY KEY

- kod nekih SUBP, pokretanje ove klauzule automatski izaziva kreiranje "unique" indeksa (B<sup>+</sup> stabla) nad *Lista\_obeležja*

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► UNIQUE [*(Lista\_obeležja)*]

- zadaje se na nivou obeležja šeme relacije koje jedino zadovoljava ograničenje jedinstvenosti
  - bez navođenja liste obeležja, ili
- na nivou celokupne šeme relacije (tabele)
  - sa navođenjem liste obeležja koja zadovoljava svojstvo jedinstvenosti
  - uobičajeno i opštije rešenje



# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► UNIQUE [(Lista\_obeležja)]

- zadavanje na nivou obeležja

#### ► CREATE TABLE

```
(...,  
    Kolona Tip(Dužina) CONSTRAINT Naziv UNIQUE,  
    ...)
```

#### ► CREATE TABLE

```
(...,  
    Kolona Tip(Dužina) UNIQUE,  
    ...)
```

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► UNIQUE [(Lista\_obeležja)]

- zadavanje na nivou šeme relacije

#### ► CREATE TABLE

*(lista specifikacija kolona tabele,*

...

CONSTRAINT Naziv UNIQUE (*Lista\_obeležja*),

...)

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► UNIQUE [(Lista\_obeležja)]

- obeležja u *Lista\_obeležja* mogu biti deklarirana kao NOT NULL, a ne moraju
- NOT NULL obeležja se eksplicitno deklariraju
- proverava se prilikom svakog pokušaja
  - upisa nove vrednosti obeležja iz liste, ili
  - modifikacije postojeće vrednosti obeležja iz liste
- u slučaju pokušaja narušavanja ograničenja, jedina moguća aktivnost je sprečavanje operacije (NO ACTION)

#### ► Svi savremeni SUBP podržavaju klauzulu UNIQUE

- Kod nekih SUBP, kontrola ovog ograničenja nije u skladu s teoretski definisanim pravilom interpretacije

# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ CHECK (*Logičkilzraz*)
      - ▶ zadaje se na nivou obeležja šeme relacije koje je jedino upotrebljeno u *Logičkomlrazu*
      - ▶ na nivou celokupne šeme relacije (tabele)
        - ▶ obavezno, kada *Logičkilzraz* obuhvata više od jednog obeležja šeme relacije
        - ▶ uobičajeno i opštije rešenje

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► CHECK (*Logičkilzraz*)

- zadavanje na nivou obeležja

#### ► CREATE TABLE

(...,

*Kolona Tip(Dužina) CONSTRAINT Naziv CHECK (Logičkilzraz),*

...)

#### ► CREATE TABLE

(...,

*Kolona Tip(Dužina) CHECK (Logičkilzraz),*

...)

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► CHECK (*Logičkilzraz*)

- zadavanje na nivou šeme relacije

#### ► CREATE TABLE

(*lista specifikacija kolona tabele,*

...

CONSTRAINT Naziv CHECK (*Logičkilzraz*),

...)

# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ CHECK (*Logičkilzraz*)
      - ▶ obeležja upotrebljena u *Logičkomlrazu*
        - ▶ mogu biti deklarirana kao NOT NULL, a ne moraju
        - ▶ mogu pripadati skupu obeležja date šeme relacije (uobičajeno), a ne moraju
      - ▶ dozvoljeno je da se u *Logičkomlrazu* vrše pozivi prethodno isprogramiranih funkcija
        - ▶ u tim funkcijama se mogu koristiti obeležja drugih šema relacija
    - ▶ konkretni SUBP, uobičajeno, ne dozvoljavaju da *Logičkilzraz* sadrži
      - ▶ obeležja iz drugih šema relacija
      - ▶ podupite (SELECT)
      - ▶ funkcije koje bi narušavale prethodna dva pravila

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► CHECK (*Logičkilzraz*)

- *Logičkilzraz* mora biti izračunljiv za svaku torku relacije nad datom šemom
- Moguće vrednosti izračunatog *Logičkoglzraza*
  - TRUE,
  - FALSE, ili
  - NULL



# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ CHECK (*Logičkilzraz*)
      - ▶ proverava se prilikom svakog pokušaja
        - ▶ upisa nove torke u relaciju, ili
        - ▶ modifikacije postojeće vrednosti obeležja, obuhvaćenog zadatim logičkim izrazom
    - ▶ ograničenje
      - ▶ je narušeno, kada je rezultat logičkog izraza FALSE
      - ▶ nije narušeno, kada je rezultat izraza TRUE, ili NULL
    - ▶ u slučaju pokušaja narušavanja ograničenja, jedina moguća aktivnost je sprečavanje operacije
      - ▶ NO ACTION

# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**

- ▶ SQL klauzula CONSTRAINT

- ▶ FOREIGN KEY [(ListaObeležja)]

- REFERENCES *NazivRefŠR* [(ListaRefObeležja)]

- [MATCH { FULL | PARTIAL }]

- [ON DELETE {NO ACTION | CASCADE | SET DEFAULT | SET NULL} ]

- [ON UPDATE {NO ACTION | CASCADE | SET DEFAULT | SET NULL} ]

# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ FOREIGN KEY ...
      - ▶ deklarirše se u referencirajućoj šemi relacije
      - ▶ *NazivRefŠR*
        - ▶ naziv referencirane šeme relacije
      - ▶ *ListaObeležja*
        - ▶ lista obeležja stranog ključa u referencirajućoj šemi relacije
      - ▶ *ListaRefObeležja*
        - ▶ lista obeležja u referenciranoj šemi relacije
        - ▶ može se izostaviti kada se navodi primarni ključ

# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ FOREIGN KEY ...
      - ▶ zadaje se na nivou obeležja šeme relacije koje jedino predstavlja strani ključ
        - ▶ bez navođenja liste obeležja, ili
      - ▶ na nivou celokupne šeme relacije (tabele)
        - ▶ sa navođenjem liste obeležja
        - ▶ uobičajeno i opštije rešenje

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► FOREIGN KEY ...

► zadavanje na nivou obeležja

#### ► CREATE TABLE

(...,

*Kolona Tip(Dužina)*

CONSTRAINT Naziv FOREIGN KEYREFERENCES *ŠemaRel(ListaOb)*,

...)

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► FOREIGN KEY ...

- zadavanje na nivou šeme relacije

#### ► CREATE TABLE

*(lista specifikacija kolona tabele,*

*...*

CONSTRAINT Naziv FOREIGN KEY (*Lista\_obeležja*) REFERENCES NazivRefŠR (*ListaRefObeležja*),

*...)*

# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ FOREIGN KEY ...
      - ▶ Obeležja u *ListaObeležja* mogu biti deklarirana kao NOT NULL, a ne moraju
      - ▶ Obeležja u *ListaRefObeležja* mogu biti deklarirana kao NOT NULL, a ne moraju
      - ▶ *ListaObeležja* definiše strani ključ u referencirajućoj šemi relacije

# Mehanizmi RSUBP

## ► Deklarativni mehanizmi (SQL:2006)

### ► SQL klauzula CONSTRAINT

#### ► FOREIGN KEY ...

► *ListaRefObeležja* definiše niz obeležja  $Y$  koji u referenciranoj šemi relacije može predstavljati:

- primarni ključ ( $Key(N_j, Y), Y = K_p(R_j)$ ),
- alternativni ključ ( $Key(N_j, Y), Y \neq K_p(R_j)$ ),
- skup obeležja s definisanim ograničenjem jedinstvenosti ( $Unique(N_j, Y)$ ), ili
- bilo koji niz obeležja, domenski kompatibilan sa nizom obeležja *ListaObeležja*

► većina proizvođača SUBP zahteva da *ListaRefObeležja* definiše niz  $Y$ , takav da važi  $Key(N_j, Y)$ , ili  $Unique(N_j, Y)$



# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ FOREIGN KEY ...
      - ▶ ograničenje se proverava prilikom svakog pokušaja
        - ▶ upisa nove torke u referencirajuću relaciju
        - ▶ modifikacije vrednosti stranog ključa, datog putem *ListaObeležja*
        - ▶ brisanja postojeće torke iz referencirane relacije
        - ▶ modifikacije vrednosti obeležja, sadržanih u *ListaRefObeležja*

# Mehanizmi RSUBP

- ▶ **Deklarativni mehanizmi (SQL:2006)**
  - ▶ SQL klauzula CONSTRAINT
    - ▶ FOREIGN KEY ...
      - ▶ pokušaj upisa nove torke u referencirajuću relaciju
        - ▶ jedina moguća aktivnost očuvanja konzistentosti je NO ACTION
      - ▶ pokušaj modifikacije vrednosti stranog ključa, datog putem *ListaObeležja*
        - ▶ jedina moguća aktivnost očuvanja konzistentosti je NO ACTION

# Mehanizmi RSUBP

## ▶ Deklarativni mehanizmi (SQL:2006)

### ▶ SQL klauzula CONSTRAINT

#### ▶ FOREIGN KEY ...

- ▶ pokušaj brisanja postojeće torke iz referencirane relacije
  - ▶ specifikacija aktivnosti očuvanja konzistentosti baze podataka putem klauzule ON DELETE
  - ▶ ON DELETE {NO ACTION | CASCADE | SET DEFAULT | SET NULL}
  - ▶ podrazumevana aktivnost je NO ACTION

# Mehanizmi RSUBP

## ▶ Deklarativni mehanizmi (SQL:2006)

### ▶ SQL klauzula CONSTRAINT

#### ▶ FOREIGN KEY ...

- ▶ pokušaj modifikacije vrednosti obeležja, sadržanih u *ListaRefObeležja*
  - ▶ specifikacija aktivnosti očuvanja konzistentosti baze podataka putem klauzule ON UPDATE
  - ▶ ON UPDATE {NO ACTION | CASCADE | SET DEFAULT | SET NULL}
  - ▶ podrazumevana aktivnost je NO ACTION

# Mehanizmi RSUBP

## ▶ **Proceduralni mehanizmi**

- ▶ okidači (trigeri)
  - ▶ CREATE TRIGGER
- ▶ procedure i funkcije baze podataka
  - ▶ CREATE PROCEDURE
  - ▶ CREATE FUNCTION
- ▶ paketi baze podataka
  - ▶ CREATE PACKAGE
  - ▶ CREATE PACKAGE BODY

# Mehanizmi RSUBP

## ► Proceduralni mehanizmi

- okidači (trigери)
- procedure i funkcije baze podataka
  - CREATE PROCEDURE
  - CREATE FUNCTION
  - ne postoji standardna sintaksa za njihovo kompletno specificiranje
    - SQL:2006 propisuje standard za deklarisanje trigera, procedura i funkcija, ali ne postoji standard za specificaciju proceduralnog dela
    - postojeći SUBP podržavaju sopstvene jezike i koncepte
      - Oracle: PL/SQL
      - Microsoft SQL Server: Transact-SQL (T-SQL)
- paketi baze podataka
  - ne postoji standardna sintaksa za njihovo specificiranje

# Mehanizmi RSUBP

## ► **Specifikacija trigera**

- oblast aktiviranja
  - tabela (ili pogled) nad kojom se definiše
- specifikacija operacija koje ga pokreću
- uslovi pod kojima se triger aktivira
- vreme aktiviranja
  - neposredno pre ili posle same operacije
- frekvencija aktiviranja
  - jednom za celu operaciju, ili
  - za svaku torku, koja je predmet operacije, pojedinačno
- aktivnost (procedura) koju triger treba da realizuje

# Mehanizmi RSUBP

- ▶ Aktiviranje trigeru
  - ▶ automatski
  - ▶ prilikom izvođenja operacije koja može da pokrene triger
  - ▶ ako su ostvareni specificirani uslovi za aktiviranje trigeru
  - ▶ izvodi se specificirana aktivnost trigeru
  - ▶ aktiviranje trigeru se ne može izbeći voljom korisnika



# DML trigeri - komponente

- ▶ Oblast definisanosti
  - ▶ jedna tabela, ili
  - ▶ jedan pogled
- ▶ Vreme okidanja
  - ▶ BEFORE      neposredno pre akcije naredbe
  - ▶ AFTER        neposredno nakon akcije naredbe
  - ▶ INSTEAD OF    umesto same akcije naredbe (samo za poglede)

# Trigeri - komponente

- ▶ Pokretači trigeru
  - ▶ INSERT
  - ▶ UPDATE [OF lista\_kolona]
  - ▶ DELETE
- ▶ Frekvencija aktiviranja (tip) trigeru
  - ▶ Svaka torka, koja je predmet DML naredbe - FOR EACH ROW - Row Level Trigger
    - ▶ Dodatno, logički uslov pokretanja Row Level trigeru
  - ▶ DML naredba u celini - Statement Level Trigger
- ▶ Aktivnost (procedura - PL/SQL blok), koju triger realizuje, kada je pokrenut

# Oblikovanje trigeru baze podataka

```
CREATE [OR REPLACE] TRIGGER Naziv_Trigera
BEFORE | AFTER | INSTEAD OF
    INSERT | DELETE | UPDATE [OF ListaObeležja]
    [OR INSERT | DELETE | UPDATE [ OF ListaObeležja ] ... ]
ON Naziv_Tabele
[FOR EACH ROW [WHEN (LogičkiUslovPokretanjaTrigera)]]
[REFERENCING OLD NazivOld AS NEW AS NazivNew]
[DECLARE
    Deklarativni deo - lokalne deklaracije
]
BEGIN
    Izvršni_deo -- proceduralni deo, specifikacija aktivnosti
[EXCEPTION Deo_za_obradu_izuzetaka -- naredbe oblika WHEN...THEN]
END [Naziv_Trigera];
```

# Referenciranje predmetnih podataka u Row Level triggerima

- ▶ Podaci koji su predmet pokretačke DML naredbe, mogu biti referencirani unutar tela trigeri:
  - ▶ :OLD.naziv\_kolone
    - ▶ "stara" vrednost kolone - before image
    - ▶ ovakvo referenciranje ima smisla u slučaju pokretačke UPDATE ili DELETE naredbe
  - ▶ :NEW.naziv\_kolone
    - ▶ "nova" vrednost kolone - after image
    - ▶ ovakvo referenciranje ima smisla u slučaju pokretačke UPDATE ili INSERT naredbe

# Mehanizmi RSUBP

## ► Procedure i funkcije

- proceduralno specificirani programi
- definisani na nivou SUBP
- pozivaju se po potrebi
- ne aktiviraju se automatski
- čuvaju se u izvornom i kompajliranom, optimizovanom obliku
- pozivaju se iz trigera, ili direktno iz korisničkih programa

# Mehanizmi RSUBP

## ► Paketi

- biblioteke deklaracija i programa
- definisani na nivou SUBP
- čuvaju se u izvornom i kompajliranom, optimizovanom obliku
- sadrže javni i privatni deo - koncept učenja
- koriste se za "tematsko" organizovanje softvera na nivou SUBP
- podržavaju perzistenciju podataka na nivou sesije
- podržavaju preklapanje (overloading) procedura i funkcija

# Sadržaj

- ▶ Uvodne napomene
- ▶ Tipovi ograničenja u RMP
- ▶ Mehanizmi RSUBP
- ▶ Realizacija ograničenja pomoću RSUBP

# Realizacija ograničenja pomoću RSUBP

## ► Ograničenje domena

- Mogući načini realizacije
  - CREATE DOMAIN (SQL:2006)
  - CREATE / ALTER TABLE, CONSTRAINT CHECK
  - CREATE TRIGGER



# Realizacija ograničenja pomoću RSUBP

## ► Realizacija ograničenja

```
CREATE TABLE radnik(  
    Mbr integer NOT NULL,  
    Ime varchar(20) NOT NULL,  
    Prz varchar(25) NOT NULL,  
    Sef integer,  
    Plt decimal(10, 2),  
    Pre decimal(6, 2),  
    God date NOT NULL,  
    CONSTRAINT radnik_PK PRIMARY KEY (Mbr),  
    CONSTRAINT radnik_FK FOREIGN KEY (Sef) REFERENCES Radnik (Mbr),  
    CONSTRAINT radnik_CH CHECK (Plt>500)  
);
```

# Realizacija ograničenja pomoću RSUBP

## ► Ograničenje vrednosti obeležja

### ► Mogući načini realizacije

- CREATE / ALTER TABLE, CONSTRAINT NOT NULL
- CREATE TRIGGER

# Realizacija ograničenja pomoću RSUBP

## ► Realizacija ograničenja

```
CREATE TABLE radnik(  
    Mbr integer NOT NULL,  
    Ime varchar(20) NOT NULL,  
    Prz varchar(25) NOT NULL,  
    Sef integer,  
    Plt decimal(10, 2),  
    Pre decimal(6, 2),  
    God date NOT NULL,  
    CONSTRAINT radnik_PK PRIMARY KEY (Mbr),  
    CONSTRAINT radnik_FK FOREIGN KEY (Sef) REFERENCES Radnik (Mbr),  
    CONSTRAINT radnik_CH CHECK (Plt>500)  
);
```

# Realizacija ograničenja pomoću RSUBP

## ► Ograničenje torke

- Mogući načini realizacije
  - CREATE / ALTER TABLE, CONSTRAINT CHECK
  - CREATE TRIGGER

# Realizacija ograničenja pomoću RSUBP

## ► Primer

### ► šema relacije

► *Radnik*({*MBR*, *PRZ*, *IME*, *POL*, *DATR*, *JMBG*}, *C*)

### ► Napomena:

Zahteva se da prvih 7 cifara vrednosti za *JMBG* odgovara datumu rođenja *DATR*, zadatom u naznačenom formatu *DDMMYYY*

# Realizacija ograničenja pomoću RSUBP

## ► Realizacija ograničenja

### ► CREATE TABLE, CONSTRAINT CHECK

```
CREATE TABLE RADNIK  
  ( MBR DMBR NOT NULL,  
    PRZ DPRZ NOT NULL,  
    IME DIME NOT NULL,  
    POL DPOL NOT NULL,  
    DATR DATUM NOT NULL,  
    JMBG DJMBG,  
    CONSTRAINT TupleCon_Radnik CHECK (Substr(JMBG, 1, 7) = To_Char(DATR, 'DDMMYYYY')  
  )  
)
```

# Realizacija ograničenja pomoću RSUBP

## ► Ograničenje ključa

### ► Mogući načini realizacije

#### ► CREATE / ALTER TABLE, CONSTRAINT

##### ► PRIMARY KEY

##### ► za primarni ključ

##### ► UNIQUE i NOT NULL

##### ► za ostale, ekvivalentne ključeve

#### ► CREATE TRIGGER

# Realizacija ograničenja pomoću RSUBP

## ► Realizacija ograničenja

```
CREATE TABLE radnik(  
    Mbr integer NOT NULL,  
    Ime varchar(20) NOT NULL,  
    Prz varchar(25) NOT NULL,  
    Sef integer,  
    Plt decimal(10, 2),  
    Pre decimal(6, 2),  
    God date NOT NULL,  
    CONSTRAINT radnik_PK PRIMARY KEY (Mbr),  
    CONSTRAINT radnik_FK FOREIGN KEY (Sef) REFERENCES Radnik (Mbr),  
    CONSTRAINT radnik_CH CHECK (Plt>500)  
);
```



# Realizacija ograničenja pomoću RSUBP

## ► Realizacija ograničenja

```
CREATE TABLE radproj(  
    Spr integer NOT NULL,  
    Mbr integer NOT NULL,  
    Brc integer NOT NULL,  
    CONSTRAINT radproj_PK PRIMARY KEY (Spr, Mbr),  
    CONSTRAINT radproj_rad_FK FOREIGN KEY (Mbr) REFERENCES radnik(Mbr),  
    CONSTRAINT radproj_prj_FK FOREIGN KEY (Spr) REFERENCES projekat(Spr)  
);
```

# Realizacija ograničenja pomoću RSUBP

## ► Ograničenje jedinstvenosti

- Mogući načini realizacije
  - CREATE / ALTER TABLE, CONSTRAINT UNIQUE
  - CREATE TRIGGER

# Realizacija ograničenja pomoću RSUBP

## ► Realizacija ograničenja

```
CREATE TABLE projekat    (  
    Spr integer not null,  
    Ruk integer not null,  
    Nap varchar(30),  
    Nar varchar(30),  
    CONSTRAINT projekat_PK PRIMARY KEY (Spr),  
    CONSTRAINT projekat_FK FOREIGN KEY (Ruk) REFERENCES Radnik (Mbr),  
    CONSTRAINT projekat_UK UNIQUE (Nap)  
);
```

# Realizacija ograničenja pomoću RSUBP

- ▶ **Ograničenje referencijalnog integriteta**
  - ▶ Mogući načini realizacije
    - ▶ CREATE / ALTER TABLE, CONSTRAINT FOREIGN KEY
    - ▶ CREATE TRIGGER

# Realizacija ograničenja pomoću RSUBP

## ► Primer

### ► šeme relacija

- $Radnik(\{MBR, \dots\}, C_1)$
- $Projekat(\{SPR, \dots\}, C_2)$
- $RadProj(\{SPR, MBR, BRC\}, C_3)$ 
  - $Key(RadProj, \{SPR, MBR\})$

### ► ograničenja referencijalnog integriteta

- $RadProj[SPR] \subseteq Projekat[SPR], Key(Projekat, SPR)$
- $RadProj[MBR] \subseteq Radnik[MBR], Key(Radnik, MBR)$

# Realizacija ograničenja pomoću RSUBP

## ► Realizacija ograničenja

- ALTER TABLE, CONSTRAINT FOREIGN KEY

```
ALTER TABLE RadProj
```

```
ADD CONSTRAINT RCon_Radproj_Proj  
FOREIGN KEY (SPR) REFERENCES PROJEKAT(SPR)
```

```
ON DELETE RESTRICT /* NO ACTION */;
```

```
ALTER TABLE RadProj
```

```
ADD CONSTRAINT RCon_Radproj_Radn  
FOREIGN KEY (MBR) REFERENCES RADNIK(MBR)
```

```
ON DELETE CASCADE;
```

# Realizacija ograničenja pomoću RSUBP

## ► Realizacija ograničenja

```
CREATE TABLE radproj(  
    Spr integer NOT NULL,  
    Mbr integer NOT NULL,  
    Brc integer NOT NULL,  
    CONSTRAINT radproj_PK PRIMARY KEY (Spr, Mbr),  
    CONSTRAINT radproj_rad_FK FOREIGN KEY (Mbr)  
    REFERENCES radnik(Mbr),  
    CONSTRAINT radproj_prj_FK FOREIGN KEY (Spr)  
    REFERENCES projekat(Spr)  
);
```

# Sadržaj

- ▶ Uvodne napomene
- ▶ Tipovi ograničenja u RMP
- ▶ Mehanizmi RSUBP
- ▶ Realizacija ograničenja pomoću RSUBP



# Pitanja i komentari



Kraj prezentacije



# Realizacija ograničenja šeme RBP putem SUBP