

Relacioni model podataka

Motivacija, koncepcija, strukturalna, operacijska i integritetna komponenta

Sadržaj

- ▶ Motivacija i koncepcija relacionog modela podataka
- ▶ Strukturalna komponenta
- ▶ Integritetna komponenta
- ▶ Operacijska komponenta

Model podataka

- ▶ Strukturalna komponenta
 - ▶ primitivni i složeni koncepti
 - ▶ “gradivni” elementi modela podataka
 - ▶ pravila za kreiranje složenih koncepata
 - ▶ služi za modeliranje LSO, kao statičke strukture sistema - šeme BP
- ▶ Operacijska komponenta
 - ▶ upitni jezik (QL)
 - ▶ jezik za manipulisanje podacima (DML)
 - ▶ jezik za definiciju podataka (DDL)
 - ▶ služi za modeliranje dinamike izmene stanja

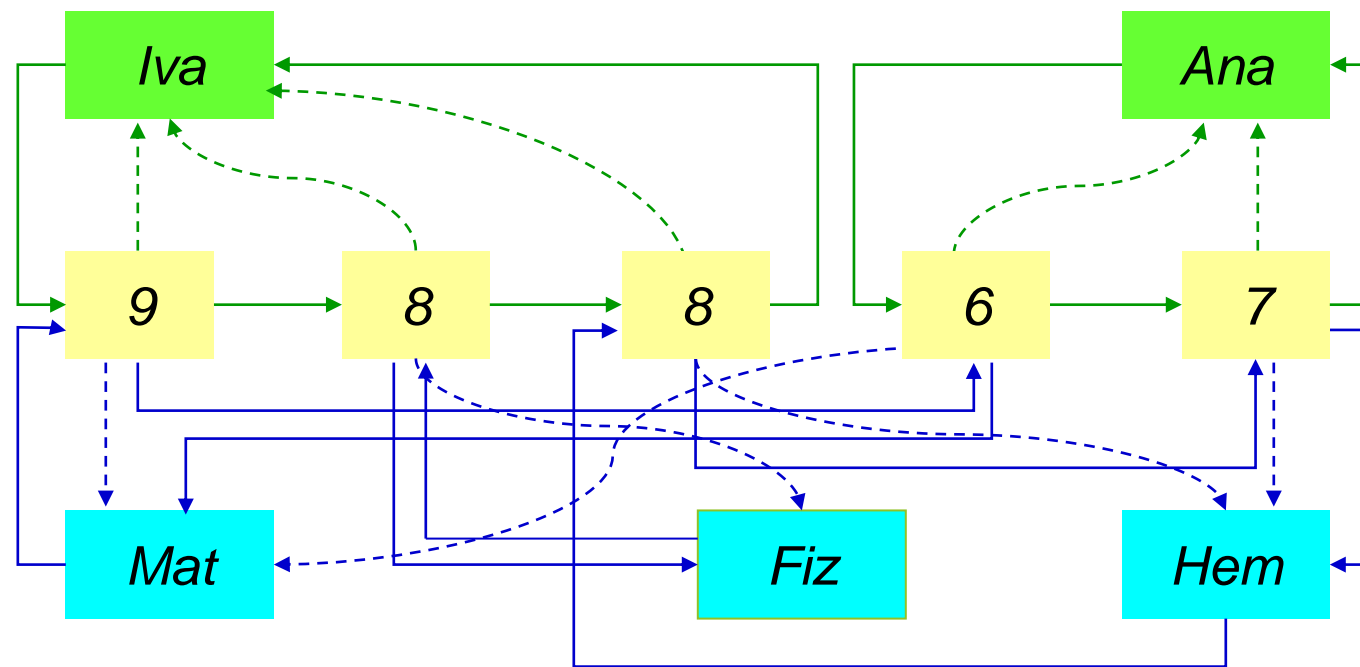
Model podataka

- ▶ Integritetna komponenta
 - ▶ skup tipova ograničenja (uslova integriteta)
 - ▶ služi za modeliranje ograničenja nad podacima u BP
- ▶ Nivoi apstrakcije
 - ▶ određeni modelom podataka
 - ▶ nivo intenzije (konteksta)
 - ▶ nivo tipa
 - ▶ opisuje npr. nivo logičke strukture obeležja - šeme
 - ▶ nivo ekstenzije (konkretizacije)
 - ▶ nivo pojave tipa
 - ▶ opisuje npr. nivo logičke strukture podataka

Mrežni i hijerarhijski model

- ▶ U korišćenju 1970-ih godina
- ▶ Nedostaci
 - ▶ čvrsta povezanost programa i fizičke strukture podataka
 - ▶ strukturalna kompleksnost
 - ▶ proceduralno orijentisani jezici za manipulaciju podacima
 - ▶ "ad hoc" razvijeni modeli
 - ▶ bez značajnije upotrebe matematičkih formalizama

Mrežni i hijerarhijski model



Relacioni model podataka

- ▶ U razvoju od 1970-ih godina
 - ▶ Codd, E. F. (1970). "A Relational Model of Data for Large Shared Data Banks", *Communications of the ACM* 13(6): 377-387.
 - ▶ Komercijalna upotreba od 199x-ih
- ▶ Motiv razvoja
 - ▶ otklanjanje nedostataka klasičnih modela podataka
 - ▶ čvrsta povezanost logičkih i fizičkih aspekata
 - ▶ strukturalna kompleksnost
 - ▶ navigacioni jezik
 - ▶ insistiranje na matematičkim osnovama u razvoju MP
 - ▶ matematička osnova RMP
 - ▶ teorija skupova i relacija
 - ▶ jako oslanjanje na predikatski račun I reda

Relacioni model podataka

► Zahtev

- nezavisnost programa od podataka, tj. fizičke strukture podataka

► Kod ranijih MP

- fizički aspekti BP ugrađeni u programe
 - raspodela slogova po zonama
 - fizički redosled \Rightarrow logičko grupisanje slogova
 - transformacija vrednosti ključa u adresu
 - lanci slogova sa pokazivačima
 - hijerarhijski redosled slogova
 - postupci zaštite podataka

Relacioni model podataka

▶ **Nezavisnost programa od podataka** - rešenje

- ▶ potpuno odvajanje prezentacionog od formata memorisanja
- ▶ relacija kao skup n-torki (torki)
- ▶ apstraktni opis relacije: šema relacije

$$N(R, C)$$

- ▶ R - skup obeležja
- ▶ C - skup ograničenja, pri čemu je
 - ▶ $K \subseteq C$ - obavezno zadat skup ključeva, koji je neprazan
- ▶ često, u početnim fazama projektovanja, šema relacije se posmatra kao struktura

$$N(R, K)$$

Relacioni model podataka

► Primer

- *Fakultet* ($\{SFK, NAZ, BIP\}, \{SFK\}$)
- $r(Fakultet) = \{$
 - $(PMF, Matematički, 7),$
 - $(EKF, Ekonomski, 4),$
 - $(ETF, Elektrotehnički, 9),$
 - $(MAF, Mašinski, 7)$ $\}$
- Upis nove torke
 - $(EKF, Elektronski, 8)$
- narušio bi ograničenje ključa (uslov integriteta)

Relacioni model podataka

► **Strukturalna jednostavnost**

- koncept relacije
 - osnova reprezentacije logičkih struktura podataka u RMP
 - ne sadrži nikakve informacije o fizičkoj organizaciji podataka
 - "prirodna" upotreba jednog od fundamentalnih matematičkih koncepata
- relacija - "homogena" i "uniformna" struktura
 - lako razumljiva korisnicima podataka
 - uobičajena je tabelarna reprezentacija
 - šema relacije \Rightarrow opis tabele (definicija tabele)
 - relacija \Rightarrow kompletan sadržaj tabele (tabela)

Relacioni model podataka

► **Strukturalna jednostavnost**

- selekcija podataka u operacijama nad BP
 - kod ranijih MP
 - upotreba fizičkih (relativnih ili apsolutnih) adresa
 - pozicioniranje upotrebom indikatora aktuelnosti ili
 - pozicioniranje putem odnosa između podataka
 - kod relacionog MP
 - asocijativno adresiranje
 - isključiva upotreba simboličkih adresa - vrlo često vrednosti ključa
 - svaki podatak u BP pronalazi se na osnovu naziva relacije, zadatih obeležja i vrednosti ključa
 - skup n-torki (torki) sa zajedničkom osobinom selektira se na uniforman način - zadavanjem istog logičkog uslova
 - SUBP vodi računa o transformaciji simboličke u relativnu adresu

Relacioni model podataka

► Primer

Fakultet

SFK	NAZ	BIP
<i>FIL</i>	<i>Filozofski</i>	<i>1</i>
<i>PMF</i>	<i>Matematički</i>	<i>7</i>
<i>ETF</i>	<i>Elektrotehnički</i>	<i>9</i>
<i>EKF</i>	<i>Ekonomski</i>	<i>4</i>
<i>MAF</i>	<i>Mašinski</i>	<i>7</i>

Projektant

MBR	IME	PRZ	SFK
<i>M3</i>	<i>Iva</i>	<i>Ban</i>	<i>PMF</i>
<i>M1</i>	<i>Ana</i>	<i>Tot</i>	<i>MAF</i>
<i>M4</i>	<i>Ana</i>	<i>Ras</i>	<i>FIL</i>
<i>M8</i>	<i>Aca</i>	<i>Pap</i>	<i>ETF</i>
<i>M6</i>	<i>Iva</i>	<i>Ban</i>	<i>EKF</i>
<i>M5</i>	<i>Eva</i>	<i>Tot</i>	<i>ETF</i>

Relacioni model podataka

▶ **Strukturalna jednostavnost**

- ▶ povezivanje podataka
 - ▶ kod ranijih MP
 - ▶ upotreba fizičkih (relativnih ili apsolutnih) adresa u funkciji pokazivača
 - ▶ fizičko pozicioniranje logički susednih podataka
 - ▶ o čemu je svaki transakcioni program morao voditi računa
 - ▶ kod relacionog MP
 - ▶ upotreba simboličkih adresa - prenetih vrednosti ključa
 - ▶ rešenje putem prostiranja ključa
 - ▶ uvođenje pojma stranog ključa i ograničenja referencijalnog integriteta
 - ▶ rešenje putem kreiranja posebne tabele sa prostiranjem ključeva
 - ▶ u oba slučaja, transakcioni program ne vodi računa o pretvaranju simboličke u relativnu adresu

Relacioni model podataka

► Deklarativni jezik

- temelji se na primenjenim tehnikama povezivanja podataka sa prostiranjem ključa
- dva alata za upitni jezik
 - relacionala algebra
 - definisana na osnovama teorije skupova i skupovnih operacija
 - skupovni operatori: unija, presek, razlika
 - specijalizovani skupovni operatori: spoj (join), projekcija, selekcija, itd.
 - relacioni račun
 - relacioni račun nad torkama
 - relacioni račun nad domenima
 - definisani na osnovama predikatskog računa I reda

Relacioni model podataka

► SQL - Structured Query Language

- zasnovan na relacionom računu nad torkama
- deklarativan
- rad sa skupovima podataka (torki)
- osnovni oblik naredbe za upite SQL-a je

```
SELECT <lista obeležja>  
FROM <lista relacija>  
WHERE <logički izraz>
```


Relacioni model podataka

► Primer

► *SELECT IME, PRZ, BIP*
FROM Fakultet, Projektant
WHERE BIP > 5 AND
Fakultet.SFK = Projektant.SFK

<i>IME</i>	<i>PRZ</i>	<i>BIP</i>
<i>Iva</i>	<i>Ban</i>	<i>7</i>
<i>Ana</i>	<i>Tot</i>	<i>7</i>
<i>Aca</i>	<i>Pap</i>	<i>9</i>
<i>Eva</i>	<i>Tot</i>	<i>9</i>

Relacioni model podataka

► Primer

► *SELECT IME, PRZ, BIP*
FROM Fakultet NATURAL JOIN Projektant
WHERE BIP > 5

<i>IME</i>	<i>PRZ</i>	<i>BIP</i>
<i>Iva</i>	<i>Ban</i>	<i>7</i>
<i>Ana</i>	<i>Tot</i>	<i>7</i>
<i>Aca</i>	<i>Pap</i>	<i>9</i>
<i>Eva</i>	<i>Tot</i>	<i>9</i>

12 principa relacionog modela podataka

12 principa RMP

- ▶ Codd, E. F. (1990).
 - ▶ "The Relational Model for Data Base Management: Version 2", Addison Wesley.
 - ▶ **Rule 0:** The system must qualify as relational, as a database, and as a management system
 - ▶ For a system to qualify as a relational database management system (RDBMS), that system must use its *relational* facilities (exclusively) to *manage* the database.
 - ▶ **Rule 1:** The *information rule*
 - ▶ All information in the database is to be represented in one and only one way, namely by values in column positions within rows of tables.

12 principa RMP

► Rule 2: *The guaranteed access rule*

- All data must be accessible with no ambiguity. This rule is essentially a restatement of the fundamental requirement for primary keys. It says that every individual scalar value in the database must be logically addressable by specifying the name of the containing table, the name of the containing column and the primary key value of the containing row.

► Rule 3: *Systematic treatment of null values*

- The DBMS must allow each field to remain null (or empty). Specifically, it must support a representation of "missing information and inapplicable information" that is systematic, distinct from all regular values (for example, "distinct from zero or any other number", in the case of numeric values), and independent of data type. It is also implied that such representations must be manipulated by the DBMS in a systematic way.

12 principa RMP

- ▶ **Rule 4:** *Active online catalog based on the relational model*
 - ▶ The system must support an online, inline, relational catalog that is accessible to authorized users by means of their regular query language. That is, **users must be able to access the database's structure (catalog) using the same query language that they use to access the database's data.**

12 principa RMP

- ▶ **Rule 5:** The *comprehensive data sublanguage rule*

- ▶ The system must support at least one relational language that

- ▶ Has a linear syntax (whose statements are expressible, per some well-defined syntax, as character strings)
 - ▶ Can be used both interactively and within application programs,
 - ▶ Supports data definition operations (including view definitions), data manipulation operations (update as well as retrieval), security and integrity constraints, and transaction management operations (begin, commit, and rollback).

- ▶ **Rule 6:** The *view updating rule*

- ▶ All views that are theoretically updatable must be updatable by the system.

12 principa RMP

- ▶ **Rule 7:** *High-level insert, update, and delete*

- ▶ The system must support set-at-a-time *insert*, *update*, and *delete* operators. This means that data can be retrieved from a relational database in sets constructed of data from multiple rows and/or multiple tables. **This rule states that insert, update, and delete operations should be supported for any retrievable set rather than just for a single row in a single table.**

- ▶ **Rule 8:** *Physical data independence*

- ▶ **Changes to the physical level** (how the data is stored, whether in arrays or linked lists etc.) **must not require a change to an application based on the structure.**

12 principa RMP

- ▶ **Rule 9: *Logical data independence***
 - ▶ **Changes to the logical level** (tables, columns, rows, and so on) **must not require a change to an application based on the structure**. Logical data independence is more difficult to achieve than physical data independence.
- ▶ **Rule 10: *Integrity independence***
 - ▶ **Integrity constraints** must be specified separately from application programs and stored in the **catalog**. It must be possible to change such constraints as and when appropriate without unnecessarily affecting existing applications.

12 principa RMP

▶ Rule 11: *Distribution independence*

- ▶ The distribution of portions of the database to various locations should be invisible to users of the database. Existing applications should continue to operate successfully:
 - ▶ when a distributed version of the DBMS is first introduced; and
 - ▶ when existing distributed data are redistributed around the system.

▶ Rule 12: The *nonsubversion rule*

- ▶ If a relational system has a low-level (single-record-at-a-time) language, that low level cannot be used to subvert or bypass the integrity rules and constraints expressed in the higher-level relational language (multiple-records-at-a-time).

Sadržaj

- ▶ Motivacija i koncepcija relacionog modela podataka
- ▶ Strukturalna komponenta
- ▶ Integritetna komponenta
- ▶ Operacijska komponenta

Strukturalna komponenta

- ▶ Primitivni koncepti u RMP
 - ▶ **Obeležje (Atribut)**
 - ▶ reprezentuje osobinu (svojstvo) klase entiteta ili poveznika u realnom sistemu (RS)
 - ▶ **Domen**
 - ▶ specifikacija skupa mogućih vrednosti koje neka obeležja mogu da dobiju
 - ▶ Pravilo pridruživanja domena obeležjima
 - ▶ svakom obeležju obavezno se pridružuje tačno jedan domen

Strukturalna komponenta

- ▶ **Primitivni koncepti nivoa intenzije**
 - ▶ domen
 - ▶ obeležje
- ▶ **Primitivni koncept nivoa ekstenzije**
 - ▶ vrednost
- ▶ kreiranje svih ostalih (složenih) koncepata strukturalne komponente RMP
 - ▶ kombinovanjem (strukturiranjem) primitivnih koncepata
 - ▶ korišćenjem definisanih pravila u RMP

Strukturalna komponenta

- ▶ Skup primitivnih i složenih koncepata RMP
 - ▶ za opis LSO (nivo intenzije) i LSP (nivo ekstenzije)

Nivo intenzije

- Domen
- Obeležje
- Skup obeležja
- Šema relacije
- Šema BP

Nivo ekstenzije

- Vrednost
- Podatak
- Torka (N-torka)
- Relacija
- Baza podataka

Strukturalna komponenta

► Torka

- reprezentuje jednu pojavu entiteta ili poveznika
- pomoću torke se svakom obeležju, iz nekog skupa obeležja, dodeljuje konkretna vrednost
 - iz skupa mogućih vrednosti definisanog domenom
- formalno, za:
 - $U = \{A_1, \dots, A_n\}$
 - $DOM = \cup_{i=1}^n (dom(A_i))$
 - skup svih mogućih vrednosti
 - torka predstavlja preslikavanje

$$t : U \rightarrow DOM,$$
$$(\forall A_i \in U)(t(A_i) \in dom(A_i))$$

Strukturalna komponenta

► Primer

► $U = \{MBR, IME, POL, SPR, NAP\}$

► Torka t_1 definisana je na sledeći način

► $t_1(MBR) = 101$

► $t_1(IME) = Ana$

► $t_1(SPR) = 1100$

$t_1(POL) = \check{z}$

► $t_1(NAP) = Univerzitetski IS$

Strukturalna komponenta

► Primer

- Torka t_1 može se prikazati kao skup podataka

$$t_1 = \{(MBR, 101), (IME, Ana), (POL, \text{ž}), (SPR, 1100), (NAP, Univerzitetski IS)\}$$

- Zadata je i torka t_2

$$t_2 = \{(MBR, 210), (IME, Aca), (POL, m), (SPR, 0105), (NAP, Polaris)\}$$

Strukturalna komponenta

► Restrikcija (“skraćenje”) torke t

- na skup obeležja $X \subseteq U$
- oznaka: $t[X]$
- svakom obeležju iz skupa X pridružuje se ona vrednost koju je imala polazna torka t
- formalno
 - $X \subseteq U$, $t: U \rightarrow DOM$,
 - $t[X]: X \rightarrow DOM$

$$(\forall A \in X)(t[X](A) = t(A))$$

Strukturalna komponenta

► Primer

► $t_2 = \{(MBR, 210), (IME, Aca), (POL, m), (SPR, 0105), (NAP, Polaris)\}$

► Neka je $X = MBR+IME$

► $t_2[X] = \{(MBR, 210), (IME, Aca)\}$

Strukturalna komponenta

► Relacija

- nad skupom obeležja U
- predstavlja konačan skup torki
- reprezentuje skup realnih entiteta ili poveznika
- Formalno

$$r(U) \subseteq \{t \mid t: U \rightarrow DOM\}, \quad |r| \in \mathbb{N}_0$$

Skup svih mogućih torki nad skupom obeležja U - $Tuple(U)$

Strukturalna komponenta

► Primer

► $U = \{MBR, IME, POL, SPR, NAP\}$

► $r_1(U) = \{t_1, t_2\}$

► $t_1 = \{(MBR, 101), (IME, Ana), (POL, \check{z}), (SPR, 1100), (NAP, Univerziteti IS)\}$

► $t_2 = \{(MBR, 210), (IME, Aca), (POL, m), (SPR, 0105), (NAP, Polaris)\}$

Strukturalna komponenta

► Primer

► $R = \{A, B, C\}$

► $\text{dom}(A) = \{a_1, a_2\}$

► $\text{dom}(B) = \{b_1, b_2\}$

► $\text{dom}(C) = \{c_1, c_2\}$

► $t_1 = \{(A, a_1), (B, b_1), (C, c_1)\}$

► $t_2 = \{(A, a_2), (B, b_2), (C, c_2)\}$

► $t_3 = \{(A, a_1), (B, b_1), (C, c_2)\}$

► $r(R) = \{t_1, t_2, t_3\}$

Strukturalna komponenta

- ▶ U relaciji se ne mogu pojaviti dve identične torke
 - ▶ to je onda ista torka, samo dva puta prikazana
- ▶ Uobičajena reprezentacija relacije
 - ▶ pomoću tabele
 - ▶ relaciju predstavlja kompletan sadržaj tabele
 - ▶ kratko, tabela
 - ▶ poredak obeležja (kolona tabele) ne utiče na informacije koje sa sobom nosi relacija - nebitan
 - ▶ poredak torki u relaciji ne utiče na informacije koje sa sobom nosi relacija - nebitan

Strukturalna komponenta

► Primeri

<i>Radnik</i>	<i>MBR</i>	<i>IME</i>	<i>POL</i>	<i>SPR</i>	<i>NAP</i>
t_1	101	Ana	ž	1100	Univerzitetški IS
t_2	210	Aca	m	0105	Polaris

$r(R)$	<i>A</i>	<i>B</i>	<i>C</i>
t_1	a_1	b_1	c_1
t_2	a_2	b_2	c_2
t_3	a_1	b_1	c_2

Strukturalna komponenta

► Šema relacije

- imenovani par

$$N(R, O)$$

- N - naziv šeme relacije (može biti izostavljen)
- R - skup obeležja šeme relacije
- O - skup ograničenja šeme relacije

► Pojava nad šemom relacije

- (R, O)
- bilo koja relacija $r(R)$, takva da zadovoljava sva ograničenja iz skupa O

Strukturalna komponenta

► Primer

- Data je šema relacije

Letovi({P, A, L}, O)

- $O = \{\text{"Pilot može da leti samo na jednom tipu aviona"}\}$

<i>Let1</i>	<i>P</i>	<i>A</i>	<i>L</i>	<i>Let2</i>	<i>P</i>	<i>A</i>	<i>L</i>
	<i>Pop</i>	<i>747</i>	<i>101</i>		<i>Pop</i>	<i>747</i>	<i>101</i>
	<i>Pop</i>	<i>747</i>	<i>102</i>		<i>Pop</i>	<i>737</i>	<i>102</i>
	<i>Ana</i>	<i>737</i>	<i>103</i>		<i>Ana</i>	<i>737</i>	<i>103</i>

- Da li prikazane relacije predstavljaju pojave nad datom šemom relacije?

Strukturalna komponenta

- ▶ **Relaciona šema baze podataka**

- ▶ (imenovani) par

(S, I)

- ▶ S - skup šema relacija

$$S = \{(R_i, O_i) \mid i \in \{1, \dots, n\}\}$$

- ▶ I - skup međurelacionih ograničenja

Strukturalna komponenta

► Primer

► Zadane su šeme relacija

- *Radnik({MBR, IME, PRZ, DATR},
{“Ne postoje dva radnika sa istom vrednošću za MBR. Svaki radnik poseduje vrednost za MBR.”})*
- *Projekat({SPR, NAP},
{“Ne postoje dva projekta sa istom vrednošću za SPR. Svaki projekat poseduje vrednost za SPR.”})*
- *Angažovanje({SPR, MBR, BRC},
{“Ne može se isti radnik na istom projektu angažovati više od jedanput. Pri angažovanju, vrednosti za MBR i SPR su uvek poznate.”})*

Strukturalna komponenta

► Primer

► $S = \{\text{Radnik}, \text{Projekat}, \text{Angažovanje}\}$

► $I = \{$

“radnik ne može biti angažovan na projektu, ako nije zaposlen”;

“na projektu ne može biti angažovan ni jedan radnik, dok projekat ne bude registrovan”

$\}$

► (S, I) predstavlja jednu relacionu šemu BP

Strukturalna komponenta

► Relaciona baza podataka

- jedna pojava nad zadatom relacionom šemom baze podataka (S, I)

$$s: S \rightarrow \{r_i \mid i \in \{1, \dots, n\}\}, (\forall i)s(R_i, O_i) = r_i$$

- svakoj šemi relacije iz skupa S odgovara jedna njena pojava
- skup relacija s mora da zadovoljava sva međurelaciona ograničenja iz skupa I

Strukturalna komponenta

- ▶ Baza podataka
 - ▶ reprezentuje jedno stanje realnog sistema
 - ▶ ažurira se, jer promene stanja realnog sistema treba da prate odgovarajuće promene podataka u BP
- ▶ Odnos šema BP - BP

Nivo intenzije

$(\{(R_1, O_1), \dots, (R_n, O_n)\}, I)$

Šema BP
statička (sporo promenljiva kategorija)
sistema BP

Nivo ekstenzije

$\{r_1(R_1), \dots, r_n(R_n)\}$

relaciona BP
dinamička (stalno promenljiva kategorija)
sistema BP

Strukturalna komponenta

► Primer

- $S = \{\text{Radnik}, \text{Projekat}, \text{Angažovanje}\}$
- $RBP = \{\text{radnik}, \text{projekat}, \text{angažovanje}\}$

Radnik

<i>MBR</i>	<i>IME</i>	<i>PRZ</i>	<i>DATR</i>
101	Ana	Pap	12.12.65.
102	Aca	Tot	13.11.48.
110	Ivo	Ban	01.01.49.
111	Olja	Kun	06.05.71.

Projekat

<i>SPR</i>	<i>NAP</i>
11	X25
13	Polaris
14	Univ. IS

Angažovanje

<i>MBR</i>	<i>SPR</i>
101	11
101	14
102	14

Strukturalna komponenta

► Konzistentno stanje BP

- baza podataka $RBP = \{r_i \mid i \in \{1, \dots, n\}\}$ nad šemom (S, I) nalazi se u
 - **formalno konzistentnom stanju** ako
 - $(\forall r_i \in RBP)(r_i \text{ zadovoljava sva ograničenja odgovarajuće šeme } (R_i, O_i))$
 - RBP zadovoljava sva međurelaciona ograničenja iskazana putem I
 - **suštinski konzistentnom stanju** ako
 - se nalazi u formalno konzistentnom stanju i
 - predstavlja vernu sliku stanja realnog sistema
 - u praksi, nivo pojave grešaka u BP sveden je na ispod 2-3%
- SUBP može da kontroliše formalnu konzistentnost

Sadržaj

- ▶ Motivacija i koncepcija relacionog modela podataka
- ▶ Strukturalna komponenta
- ▶ **Integritetna komponenta**
- ▶ Operacijska komponenta

Integritetna komponenta

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

Integritetna komponenta

- ▶ Karakteristike tipa ograničenja
 - ▶ skup 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 operaciju koja može dovesti do narušavanja ograničenja

Implementacione šeme BP i pravila poslovanja

- ▶ 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
 - ▶ izaziva grešku i prekida operaciju
 - ▶ prosleđuje korisničkom programu poruku o grešci
 - ▶ program obrađuje tu poruku i prosleđuje je korisniku
 - ▶ Ili aktivnim mehanizmom dovodi stanje BP u konzistentno
 - ▶ automatski, nakon izvođenja kritične operacije za ograničenje

Integritetna komponenta

- ▶ Oblasti definisanosti u relacionom MP
 - ▶ **vanrelaciono** ograničenje
 - ▶ definiše se izvan konteksta šeme relacije
 - ▶ **jednorelaciono** (unutarrelaciono, lokalno) ograničenje
 - ▶ definiše se nad tačno jednom šemom relacije
 - ▶ **višerelaciono** ograničenje
 - ▶ definiše se nad skupom ili nizom šema relacija, koji sadrži bar dva člana

Integritetna komponenta

- ▶ Oblasti interpretacije u relacionom MP
 - ▶ **ograničenje vrednosti**
 - ▶ interpretira se nad tačno jednom vrednošću nekog obeležja
 - ▶ **ograničenje torke**
 - ▶ interpretira se nad jednom torkom bilo koje relacije
 - ▶ **relaciono ograničenje**
 - ▶ interpretira se nad skupom torki bilo koje relacije
 - ▶ **međurelaciono ograničenje**
 - ▶ interpretira se nad barem dve, bilo koje relacije

Integritetna komponenta

- ▶ Oblasti interpretacije u relacionom MP
 - ▶ ograničenje vrednosti
 - ▶ ograničenje torke
 - ▶ relaciono ograničenje
 - ▶ **međurelaciono ograničenje**
 - ▶ Napomena "bilo koja relacija":
 - ▶ jedna relacija iz baze podataka, ili
 - ▶ relacija koja je nastala primenom izraza relacione algebre nad jednom ili više drugih relacija - pogled
 - ▶ moguća i primena operatora spajanja

Integritetna komponenta

- ▶ Tipovi ograničenja u relacionom modelu podataka
 - ▶ ograničenje domena
 - ▶ ograničenje vrednosti obeležja
 - ▶ ograničenje torke
 - ▶ integritet entiteta (ograničenje ključa)
 - ▶ ograničenje jedinstvenosti vrednosti obeležja
 - ▶ zavisnost sadržavanja
 - ▶ ograničenje referencijalnog integriteta
 - ▶ funkcionalna zavisnost

Integritetna komponenta

► Specifikacija domena

$D(id(D), Predef)$

- D - naziv domena
- $id(D)$ - ograničenje (integritet) domena
- $Predef$ - predefinisana vrednost domena

► Ograničenje domena

$id(D) = (Tip, Dužina, Uslov)$

- Tip - tip podatka (primitivni domen), ili
oznaka prethodno definisanog domena
- $Dužina$ - dužina tipa podatka
- $Uslov$ - logički uslov

Integritetna komponenta

► Specifikacija domena

► *Tip*

- predstavlja jedinu obaveznu komponentu specifikacije ograničenja domena

► *Dužina*

- navodi se samo za tipove podataka (primitivne domene) koji to zahtevaju
- ne navodi za domene čiji *tip* ne predstavlja primitivni domen

► *Uslov*

- mora da ga zadovoljava svaka vrednost iz skupa mogućih vrednosti domena

► *Predef*

- mora da zadovolji ograničenja *tipa*, *dužine* i *uslova*

Integritetna komponenta

► Ograničenje domena

- interpretacija ograničenja
 - moguća za bilo koju vrednost - konstantu d
 - oznaka $id(D)(d)$

► Primeri

- $DPrezime((String, 30, \Delta), \Delta)$
- $DDatum((Date, \Delta, d \geq '01.01.1900'), \Delta)$
- $DOcena((Number, 2, d \geq 5 \wedge d \leq 10), \Delta)$
- $DPozOcena((DOcena, \Delta, d \geq 6), 6)$
 - Δ - komponenta u specifikaciji nije zadata

Integritetna komponenta

► Nula (nedostajuća, izostavljena) vrednost

- specijalna vrednost
- označava se posebnim simbolom
 - ω , ili ? (u literaturi) ili
 - NULL (u literaturi i SQL-u)
- moguća značenja
 - nepoznata - postojeća vrednost obeležja
 - nepostojeća vrednost obeležja
 - neinformativna vrednost obeležja
- skup mogućih vrednosti svih domena proširuje se nula vrednošću

$$DOM \cup \{\omega\}$$

- nula vrednost a priori zadovoljava svako ograničenje domena

Integritetna komponenta

► Specifikacija obeležja šeme relacije

► $A \in R, N(R, O)$

► zadaje se za svako obeležje šeme relacije

$(id(N, A), Predef)$

► $id(A)$ - ograničenje vrednosti obeležja

► $Predef$ - predefinisana vrednost obeležja

► Ograničenje vrednosti obeležja

$id(N, A) = (Domen, Null)$

► $Domen$ - oznaka (naziv) domena obeležja

► $Null \in \{T, \perp\}$ - ograničenje nula vrednosti obeležja

► T - dozvola dodele nula vrednosti obeležju u $r(N)$

► \perp - zabrana dodele nula vrednosti obeležju u $r(N)$

Integritetna komponenta

► Specifikacija obeležja šeme relacije

► *Domen* i *Null*

- obavezne komponente specifikacije

► *Predef*

- ako se navede, onda je on važeći
- u protivnom, važeći je *Predef* odgovarajućeg *Domena*, ili prvog sledećeg nasleđenog domena, za koji je *Predef* definisan

► Interpretacija ograničenja

- moguća za bilo koju vrednost obeležja d
- oznaka $id(N, A)(d)$

Integritetna komponenta

► Ograničenje torke

- izražava ograničenja na moguće vrednosti unutar jedne torke
- predstavlja skup ograničenja vrednosti obeležja, kojem je pridodat logički uslov
- formalno, za šemu relacije $N(R, O)$

$$id(N) = id(R) = (\{id(N, A) \mid A \in R\}, Uslov)$$

► *Uslov*

- logički uslov koji svaka torka mora da zadovolji
 - može, u ulozi operanda, da sadrži bilo koje obeležje date šeme relacije
- ## ► interpretacija ograničenja
- moguća za bilo koju torku nad skupom obeležja R , $id(N)(t)$

Integritetna komponenta

► Primer

► $Radnik(\{MBR, PRZ, IME, ZAN, BPJZ\}, O)$

<i>Radnik</i>	<i>Domen</i>	<i>Null</i>	<i>Predef</i>	
<i>MBR</i>	<i>MBRD</i>	\perp	Δ	
<i>PRZ</i>	<i>PRZD</i>	\perp	Δ	
<i>IME</i>	<i>IMED</i>	\perp	Δ	
<i>ZAN</i>	<i>ZAND</i>	\perp	Δ	
<i>BPJZ</i>	<i>BPJZD</i>	\top	Δ	

Uslov: $ZAN = \text{'prg'} \Leftrightarrow BPJZ \neq \omega$

Integritetna komponenta

► Primer

► *Radnik*({*MBR*, *PRZ*, *IME*, *ZAN*, *BPJZ*}, 0)

<i>Domen</i>	<i>Tip</i>	<i>Dužina</i>	<i>Uslov</i>	<i>Predef</i>
<i>MBRD</i>	<i>Number</i>	4	$d \geq 0$	Δ
<i>PRZD</i>	<i>String</i>	30	Δ	Δ
<i>IMED</i>	<i>String</i>	15	Δ	Δ
<i>ZAND</i>	<i>String</i>	3	Δ	Δ
<i>BPJZD</i>	<i>Number</i>	2	$d \geq 0$	0

Integritetna komponenta

► Ključ šeme relacije

- minimalni podskup skupa obeležja šeme relacije, na osnovu kojeg se jedinstveno može identifikovati svaka torka relacije nad datom šemom
- formalno, X je ključ ako
 - $1^0 (\forall u, v \in r(R))(u[X] = v[X] \Rightarrow u = v)$
 - $2^0 (\forall Y \subset X)(\neg 1^0)$
- oblast interpretacije
 - skup torki (relacija) nad datom šemom relacije

Integritetna komponenta

► Ključ šeme relacije

- u određenim situacijama (u procesu projektovanja šeme BP) skup ograničenja šeme relacije zadaje se samo kao skup ključeva

$$N(R, K)$$

► Primer

- šema relacije *Radnik*(R, K)
 - $R = \{MBR, IME, PRZ, DATR, POL, MESR, RBRE\}$
 - $K = \{MBR, DATR+MESR+POL+RBRE\}$

Integritetna komponenta

► Primer

- *Radnik*({*MBR*, *IME*, *PRZ*, *DATR*}, {*MBR*})
- *Projekat*({*SPR*, *NAP*}, {*SPR*})
- *Angažovanje*({*SPR*, *MBR*, *BRC*}, {*SPR*+*MBR*})

Integritetna komponenta

► Ograničenje ključa (integritet entiteta)

- šeme relacije $N(R, K)$
 - ključ $X \in K, X \subseteq R$
 - oznaka

Key(N, X)

- za sva obeležja ključa nula vrednosti su zabranjene

$$(\forall K_i \in K)(\forall A \in K_i)(Null(N, A) = \perp)$$

Integritetna komponenta

► Ograničenje ključa (integritet entiteta)

- svaka šema relacije mora posedovati najmanje jedan ključ ($K \neq \emptyset$)
 - proizilazi iz definicije pojma relacije

► ekvivalentni ključevi

- svi ključevi skupa ključeva K

► primarni ključ

- jedan izabrani ključ, od svih ekvivalentnih ključeva
- oznaka $K_p(N)$
- svaka šema relacije treba da poseduje tačno jedan primarni ključ
- koristi se u ulozi asocijativne (simboličke) adrese za povezivanje podataka u relacijama

Integritetna komponenta

► Ograničenje jedinstvenosti

- vrednosti obeležja šeme relacije $N(R, O)$
- Uniqueness Constraint

Unique(N, X)

- X - skup obeležja, $X \subseteq R$
- zahteva da ne-nula kombinacija vrednosti obeležja bude jedinstvena u relaciji nad $N(R, O)$
- formalno
 - $(\forall u, v \in r(R))((\forall A \in X)(u[A] \neq \omega \wedge v[A] \neq \omega) \Rightarrow (u[X] = v[X] \Rightarrow u = v))$

Integritetna komponenta

► Ograničenje jedinstvenosti

- oblast interpretacije
 - skup torki - relacija nad datom šemom $N(R, O)$
- skup svih ograničenja jedinstvenosti u šemi $N(R, O)$

$$Uniq = \{Unique(N, X) \mid X \subseteq R\}$$

Integritetna komponenta

► Primer

Radnik({*MBR*, *IME*, *PRZ*, *DATR*, *JMBG*}, *O*)

- $Uniq \subseteq O$
- $Uniq = \{Unique(Radnik, JMBG)\}$
- $Unique(Radnik, JMBG)$
 - zahteva da ako radnik poseduje ne-nula vrednost za *JMBG*, onda je ta vrednost jedinstvena u relaciji nad šemom *Radnik*

Integritetna komponenta

► Skup svih ograničenja šeme relacije

- praktično, kada šemu relacije treba implementirati u datom SUBP, zadaje se kao unija
 - skupa ključeva,
 - ograničenja jedinstvenosti i
 - ograničenja torke

$$N(R, K \cup \text{Uniq} \cup \{id(R)\})$$

Integritetna komponenta

► Primer

$Radnik(\{MBR, PRZ, IME, ZAN, BPJZ, JMBG\}, K \cup Uniq \cup \{id(R)\})$

- $K = \{MBR\}$
- $Uniq = \{Unique(Radnik, JMBG)\}$
- $id(R)$ - prethodno zadat, u tabelarnom obliku

Integritetna komponenta

► Zavisnost sadržavanja

- date su šeme relacije $N_i(R_i, O_i)$ i $N_j(R_j, O_j)$
- dati su domenski kompatibilni nizovi obeležja

$$X = (A_1, \dots, A_n), (\forall l \in \{1, \dots, n\})(A_l \in R_i),$$

$$Y = (B_1, \dots, B_n), (\forall l \in \{1, \dots, n\})(B_l \in R_j),$$

$$(\forall l \in \{1, \dots, n\})(\text{dom}(A_l) \subseteq \text{dom}(B_l))$$

- oznaka (pravilo zapisivanja)

$$N_i[X] \subseteq N_j[Y]$$

Integritetna komponenta

► Zavisnost sadržavanja

$$N_i[X] \subseteq N_j[Y]$$

- važi ako je za bilo koje dve relacije $r(R_i, O_i)$ i $s(R_j, O_j)$ zadovoljeno

$$(\forall u \in r)(\exists v \in s)(\forall l \in \{1, \dots, n\})(u[A_l] = \omega \vee u[A_l] = v[B_l])$$

- oblast definisanosti
 - niz od dve šeme relacije
- oblast interpretacije
 - relacije nad šemama N_i i N_j

Integritetna komponenta

► Primer

- date su relacije $r(N_i)$ i $s(N_j)$
- važi zavisnost sadržavanja $N_i[B] \subseteq N_j[B]$

r	A	B
	a_1	b_1
	a_2	b_2

s	B	C
	b_1	c_1
	b_2	c_1
	b_3	c_2

Integritetna komponenta

► Primer

- date su relacije $r(N_i)$ i $s(N_j)$
- važi zavisnost sadržavanja $N_i[(A, B)] \subseteq N_j[(C, D)]$

r	A	B
	a_1	b_1
	a_2	ω

s	C	D
	a_1	b_1
	a_2	b_2
	a_3	b_2

Integritetna komponenta

► Ograničenje referencijalnog integriteta

- zavisnost sadržavanja $N_i[X] \subseteq N_j[Y]$, kada je Y ključ šeme relacije $N_j(R_j, K_j)$
- N_i - **referencirajuća šema relacije**
- N_j - **referencirana šema relacije**

Integritetna komponenta

► Primer

- $Projekat[RUK] \subseteq Radnik[MBR]$
- $Angažovanje[MBR] \subseteq Radnik[MBR]$
- $Angažovanje[SPR] \subseteq Projekat[SPR]$

Radnik

<i>MBR</i>	<i>IME</i>	<i>PRZ</i>	<i>DATR</i>
101	Ana	Pap	12.12.85.
102	Aca	Tot	13.11.88.
110	Ivo	Ban	01.01.79.
111	Olja	Kun	06.05.81.

Angažovanje

<i>MBR</i>	<i>SPR</i>
101	11
101	14
111	14

Projekat

<i>SPR</i>	<i>NAP</i>	<i>RUK</i>
11	X25	101
13	Polaris	101
14	Univ.IS	111

Integritetna komponenta - $N_i[X] \subseteq N_j[Y]$

► Primer - unos novog projekta

X

Referencirana šema relacije

► $Projekat[RUK] \subseteq Radnik[MBR]$

Referencirajuća šema relacije

Y - ključ

Projekat

SPR	RUK	NAP
11	101	X25
13	101	Polaris
14	111	Univ.IS
15	110	P7
15	210	P7



Radnik

MBR	IME	PRZ	DATR
101	Ana	Pap	12.12.85.
102	Ana	Tot	13.11.88.
110	to	Ban	01.01.79.
111	Olja	Kun	06.05.81.

Integritetna komponenta - $N_i[X] \subseteq N_j[Y]$

- ▶ Primer-angažovanje radnika na novom projektu
 - ▶ $Angažovanje[MBR] \subseteq Radnik[MBR]$
 - ▶ $Angažovanje[SPR] \subseteq Projekat[SPR]$

Angažovanje

MBR	SPR
101	11
101	14
111	14
110	15
110	15

Projekat

SPR	NAP	RUK	DATR
101	X25	101	12.12.85.
102	Polaris	102	13.11.88.
110	Polaris	111	01.01.79.
115	Polaris	110	06.05.81.

Integritetna komponenta - $N_i[X] \subseteq N_j[Y]$

- Primer - brisanje radnika
 - $Projekat[RUK] \subseteq Radnik[MBR]$

DELETE FROM Radnik
WHERE MBR = 102

Projekat

MBR	SPR	NAP
101	101	X25
101	101	Polaris
111	111	Univ. IS
110	110	P7

Radnik

MBR	IME	PRZ	DATR
101	Ana	Pap	12.12.85.
102	Aca	Tot	13.11.88.
110	Ivo	Ban	11.79.
111	Olja	Kun	06.05.81.

Integritetna komponenta

► Funkcionalna zavisnost (FZ)

- izraz oblika $f: X \rightarrow Y$
 - gde su X i Y skupovi obeležja
 - f je oznaka FZ
 - X i Y su podskupovi skupa U
 - oznaka f se, u notaciji, često izostavlja
- semantika
 - ako je poznata X vrednost, poznata je i Y vrednost
 - svakoj X vrednosti odgovara samo jedna Y vrednost
- relacija r zadovoljava FZ $X \rightarrow Y$ ako važi
$$(\forall u, v \in r)(u[X] = v[X] \Rightarrow u[Y] = v[Y])$$
- oblast interpretacije
 - relacija $r(N)$ ili $r(U)$

Integritetna komponenta

► Funkcionalna zavisnost (FZ)

► Primer

► $MBR \rightarrow IME$

► ako dve torke imaju istu vrednost za MBR , moraju imati istu vrednost i za IME

► skup FZ se označava sa F

► $F = \{MBR \rightarrow IME, MBR + MES + GOD \rightarrow BRC, \dots\}$

► Trivijalna FZ

► svaka FZ koja je zadovoljena u bilo kojoj relaciji

► svaka FZ $X \rightarrow Y$, za koju važi $Y \subseteq X$

► Primer

► $MBR \rightarrow MBR, MBR \rightarrow \emptyset, AB \rightarrow A, \dots$

Integritetna komponenta

► Primer

► semantika uvedenih obeležja skupa U

- BRI - broj indeksa
- IME - ime studenta
- PRZ - prezime studenta
- BPI - broj položenih ispita
- OZP - oznaka predmeta
- NAP - naziv predmeta
- NAS - prezime nastavnika
- OCE - ocena na ispitu

Integritetna komponenta

► Primer

Student

<i>BRI</i>	<i>IME</i>	<i>PRZ</i>	<i>BPI</i>	<i>OZP</i>	<i>NAP</i>	<i>NAS</i>	<i>OCE</i>
159	Ivo	Ban	13	P1	Mat	Han	09
159	Ivo	Ban	13	P2	Fiz	Kun	08
013	Ana	Tot	09	P1	Mat	Pap	06
119	Eva	Kon	15	P3	Hem	Kiš	07
159	Ivo	Ban	13	P3	Hem	Kiš	10
119	Eva	Kon	15	P1	Mat	Han	09
159	Ivo	Ban	13	P4	Mat	Car	10
037	Eva	Tot	01	P4	Mat	Car	10

Integritetna komponenta

► Primer

- Relacija *Student* zadovoljava sledeće FZ

$$F = \{BRI \rightarrow IME + PRZ + BPI, IME + PRZ \rightarrow BRI, OZP \rightarrow NAP, NAS \rightarrow OZP + NAP, BRI + OZP \rightarrow OCE + NAS\}$$

- Relacija *Student* ne zadovoljava sledeće FZ

$$BRI \rightarrow OCE, OZP \rightarrow NAS, \dots$$

► Način identifikacije važećih FZ

- na osnovu odnosa i pravila poslovanja koji postoje u realnom sistemu

Integritetna komponenta

► Funkcionalna zavisnost

► logička posledica

- FZ f je logička posledica od skupa FZ F
- oznaka: $F \models f$
 - ako svaka relacija r koja zadovoljava F zadovoljava i f
 - $(\forall r \in SAT(U))(r \models F \Rightarrow r \models f)$
- skup FZ F_2 je logička posledica od skupa FZ F_1
- oznaka: $F_1 \models F_2$
 - ako $(\forall f \in F_2)(F_1 \models f)$

► implikacioni problem

- rešiti implikacioni problem, znači utvrditi da li važi $F \models f$

► ekvivalentnost skupova FZ

- oznaka: $F_1 \equiv F_2$
- ako $F_1 \models F_2 \wedge F_2 \models F_1$

Integritetna komponenta

▶ Funkcionalna zavisnost

▶ zatvarač (zatvorenje) skupa FZ

- ▶ oznaka F^+

- ▶ skup koji sadrži sve logičke posledice od F

- ▶ $F^+ = \{f \mid F \models f\}$

- ▶ važi za svaki F da $F \subseteq F^+$

- ▶ $F_1 \models F_2$ akko $F_2^+ \subseteq F_1^+$

▶ ekvivalentnost skupova FZ

- ▶ $F_1 \equiv F_2$ akko $F_1^+ = F_2^+$

Integritetna komponenta

► Funkcionalna zavisnost

► Armstrongova pravila izvođenja

► refleksivnost

$$\triangleright Y \subseteq X \mid\!-\! X \rightarrow Y$$

► proširenje

$$\triangleright X \rightarrow Y, W \subseteq V \mid\!-\! XV \rightarrow YW$$

► pseudotranzitivnost

$$\triangleright X \rightarrow Y, YV \rightarrow Z \mid\!-\! XV \rightarrow Z$$

► Izvedena pravila izvođenja

► uniranje desnih strana

$$\triangleright X \rightarrow Y, X \rightarrow Z \mid\!-\! X \rightarrow YZ$$

► dekompozicija desnih strana

$$\triangleright X \rightarrow Y, V \subseteq Y \mid\!-\! X \rightarrow V$$

► tranzitivnost

$$\triangleright X \rightarrow Y, Y \rightarrow Z \mid\!-\! X \rightarrow Z$$

Integritetna komponenta

► Primer

► varijante u označavanju

- primena pravila dekompozicije i uniranja desnih strana

$$\{BRI \rightarrow IME, BRI \rightarrow PRZ\} \equiv \{BRI \rightarrow IME + PRZ\}$$

- proizvoljno dekomponovanje levih strana nije dozvoljeno

$$\{BRI + OZP \rightarrow OCE\} \not\equiv \{BRI \rightarrow OCE, OZP \rightarrow OCE\}$$

Integritetna komponenta

► Primer

$U = \{A, B, C, D, E, F, G, H\}$

$F = \{BG \rightarrow H, DF \rightarrow GH, CH \rightarrow DA, B \rightarrow E, D \rightarrow BA, AG \rightarrow CF\}$

Da li važi:

1. $DA \rightarrow E?$
2. $DG \rightarrow FH?$

Integritetna komponenta

► Funkcionalna zavisnost

► Sistem armstrongovih pravila izvođenja je

- refleksivnost, proširenje i pseudotranzitivnost
- **korektan (neprotivrečan)**
 - svaka FZ koja se izvede primenom AP iz nekog skupa FZ predstavlja logičku posledicu tog skupa FZ
- **kompletan**
 - svaka logička posledica nekog skupa FZ može se izvesti primenom AP iz tog skupa
- **neredundantan (minimalan)**
 - ne može se eliminisati kao suvišno ni jedno od tri pravila izvođenja, a da prethodna dva svojstva ostanu očuvana

Integritetna komponenta

► Nepotpuna FZ

- $X \rightarrow Y \in F$ je nepotpuna
 - ako sadrži logički suvišno obeležje na levoj strani
 - $(\exists X' \subset X)(X' \rightarrow Y \in F^+)$
- Primer
 - $BRI + IME \rightarrow PRZ$, zbog $BRI \rightarrow IME$
 - redukuje se u $BRI \rightarrow PRZ$

► Tranzitivna FZ

- $X \rightarrow Z$ tranzitivna
 - ako važi $X \rightarrow Y \in F^+$ i $Y \rightarrow Z \in F^+$, a ne važi da je $Y \rightarrow X \in F^+$
- Primer
 - $NAS \rightarrow OZP$, $OZP \rightarrow NAP$, $\neg(OZP \rightarrow NAS)$
 - $NAS \rightarrow NAP$ je tranzitivna i logički suvišna

Integritetna komponenta

► Ključ šeme relacije i FZ

- X je ključ šeme relacije (R, F) , ako važi
 - 1^0 iz F sledi $X \rightarrow R$ ($X \rightarrow R \in F^+$)
 - 2^0 X je minimalni skup obeležja s osobinom 1^0
 - $\neg(\exists X' \subset X)(X' \rightarrow R \in F^+)$

► Zatvarač (zatvorenje) skupa obeležja

- skup svih obeležja koja funkcionalno zavise od X
- $X_F^+ = \{A \in U \mid X \rightarrow A \in F^+\}$

Integritetna komponenta

▶ Algoritam za izračunavanje zatvarača X_F^+

- ▶ $X_0 \leftarrow X$
- ▶ $(\text{Za } i \geq 0)(X_{i+1} \leftarrow X_i \cup \{A \in U \mid (\exists V \rightarrow W \in F)(V \subseteq X_i \wedge A \in W)\})$
- ▶ $(\text{Za } n \geq 0)(X_{n+1} = X_n \Rightarrow X_F^+ = X_n)$

▶ Generisanje jednog ključa šeme relacije

- ▶ polazi se od R i vrši se redukcija
 - ▶ izbacivanjem obeležja i izračunavanjem zatvarača ostatka
 - ▶ $X \leftarrow R$
 - ▶ Redukcija $Red(X)$: $(\forall A \in X)(A \in (X \setminus \{A\})_F^+ \Rightarrow X \leftarrow X \setminus \{A\})$

▶ Generisanje svih alternativnih ključeva

- ▶ polazi se od prvog generisanog ključa X , $K \leftarrow \{X\}$
 - ▶ $(\forall X \in K)(\forall V \rightarrow W \in F)(X \cap W \neq \emptyset \Rightarrow X_{newk} \leftarrow (X \setminus W) \cup V)$
 - ▶ Redukcija: $Red(X_{newk})$: $K \leftarrow K \cup \{Red(X_{newk})\}$

Integritetna komponenta

► Primer

$F = \{BRI \rightarrow IME+PRZ+BPI, IME+PRZ \rightarrow BRI, OZP \rightarrow NAP, NAS \rightarrow OZP+NAP, BRI+OZP \rightarrow OCE+NAS\}$

► šema relacije *Student* ima četiri ključa

- $K_1 = BRI+NAS, K_2 = IME+PRZ+NAS,$
- $K_3 = BRI+OZP, K_4 = IME+PRZ+OZP$

► Pojam ključa

- fundamentalan za teoriju i praksu relacionog MP
 - ne projektuju se ostala ograničenja šeme BP, dok se ne preciziraju ključevi svih šema relacija

Integritetna komponenta

► Primer

$U = \{A, B, C, D, E, F, G, H\}$

$F = \{BG \rightarrow H, DF \rightarrow GH, CH \rightarrow DA, B \rightarrow E, D \rightarrow BA, AG \rightarrow CF\}$

Izračunajte ključeve!

Integritetna komponenta

► Projekcija skupa funkcionalnih zavisnosti na skup obeležja

- dati su skup fz F i skup obeležja $X \subseteq U$
- projekcija $F|_X$ predstavlja skup svih funkcionalnih zavisnosti koje logički slede iz F , a definisane su u skupu obeležja X
- formalno

$$F|_X = \{V \rightarrow W \mid F \models V \rightarrow W \wedge VW \subseteq X\}$$

► Primer

- $F = \{A \rightarrow B, B \rightarrow C, BE \rightarrow F, A \rightarrow D\}$
- $F|_{ACDEF} = \{A \rightarrow C, AE \rightarrow F, A \rightarrow D, \text{sve trivijalne fz}\}$

Integritetna komponenta - zadaci za vežbu

- ▶ $\text{radnik}(\{\text{Mbr}, \text{Ime}, \text{Prz}, \text{Sef}, \text{Plt}, \text{God}, \text{Pre}\}, \{\text{Mbr}\}),$
- ▶ $\text{projekat}(\{\text{Spr}, \text{Ruk}, \text{Nap}, \text{Nar}\}, \{\text{Spr}\}),$
- ▶ $\text{radproj}(\{\text{Spr}, \text{Mbr}, \text{Brc}\}, \{\text{Spr} + \text{Mbr}\}),$

- ▶ $\text{radnik}[\text{Sef}] \subseteq \text{radnik}[\text{Mbr}],$
- ▶ $\text{projekat}[\text{Ruk}] \subseteq \text{radnik}[\text{Mbr}],$
- ▶ $\text{radproj}[\text{Mbr}] \subseteq \text{radnik}[\text{Mbr}],$
- ▶ $\text{radproj}[\text{Spr}] \subseteq \text{projekat}[\text{Spr}].$

Integritetna komponenta - zadaci za vežbu

► Izračunati zatvarače

1. $U = \{A, B, C, D, E, F\}$, $F = \{AB \rightarrow AC, CD \rightarrow E, A \rightarrow B, AE \rightarrow F\}$
 - a. $(AD)^+$
2. $U = \{A, B, C, D, E, F, G\}$, $F = \{AB \rightarrow C, C \rightarrow A, BC \rightarrow D, ACD \rightarrow B, D \rightarrow EG, BE \rightarrow C, CG \rightarrow BD, CE \rightarrow AG\}$
 - a. $(BD)^+$
3. $U = \{A, B, C, D, E, I, J\}$, $F = \{A \rightarrow B, A \rightarrow C, A \rightarrow E, D \rightarrow C, E \rightarrow I, BI \rightarrow J\}$
 - a. $(AI)^+$
 - b. $(DJ)^+$
 - c. $(BE)^+$

Integritetna komponenta - zadaci za vežbu

► Odrediti ključeve

1. $U = \{A, B, C, D, E\}, \quad F = \{AB \rightarrow CDE, E \rightarrow A, CD \rightarrow B\}$
2. $U = \{A, B, C, D, E, F\}, \quad F = \{AB \rightarrow C, C \rightarrow A, C \rightarrow D, AB \rightarrow E, AB \rightarrow F, E \rightarrow F\}$
3. $U = \{A, B, C, D, E, F, G, H\}, \quad F = \{AB \rightarrow CE, C \rightarrow B, ED \rightarrow F, F \rightarrow G\}$

Integritetna komponenta - zadaci za vežbu

► Dokazati da je FZ logička posledica skupa FZ

1. $U = \{A, B, C, D, E\}$, $F = \{A \rightarrow B, B \rightarrow C, AC \rightarrow D, BD \rightarrow E, C \rightarrow E\}$
 - a. $F \models A \rightarrow D$
 - b. $F \models AD \rightarrow E$
2. $U = \{A, B, C, D, E, F\}$, $F = \{A \rightarrow F, AB \rightarrow CE, AC \rightarrow D, EB \rightarrow D, D \rightarrow A, F \rightarrow AE\}$
 - a. $F \models AB \rightarrow D$
3. $U = \{A, B, C, D, E, F, G\}$, $F = \{AB \rightarrow C, C \rightarrow A, BC \rightarrow D, ACD \rightarrow B, D \rightarrow EG, BE \rightarrow C\}$
 - a. $F \models CE \rightarrow B$
 - b. $F \models BD \rightarrow C$
 - c. $F \models CE \rightarrow D$
 - d. $F \models ABG \rightarrow E$
 - e. $F \models CD \rightarrow B$
4. $U = \{A, B, C, D, E, F\}$, $F = \{AB \rightarrow AC, CD \rightarrow E, A \rightarrow B, AE \rightarrow F\}$
 - a. $F \models AD \rightarrow F$

Sadržaj

- ▶ Motivacija i koncepcija relacionog modela podataka
- ▶ Strukturalna komponenta
- ▶ Integritetna komponenta
- ▶ Operacijska komponenta

Operacijska komponenta

- ▶ Jezik za manipulaciju podacima u RMP
 - ▶ operacije za ažuriranje relacija
 - ▶ dodavanje nove torke (Add)
 - ▶ brisanje postojeće torke (Delete)
 - ▶ modifikacija podataka postojeće torke (Update)
- ▶ Jezik za definiciju podataka u RMP
 - ▶ operacije za upravljanje šemom BP
 - ▶ kreiranje, brisanje i modifikovanje delova šeme BP
- ▶ Upitni jezik u RMP
 - ▶ operacije za izražavanje upita nad jednom relacijom, ili skupom relacija
 - ▶ pružanje podataka na uvid korisniku

Operacijska komponenta

- ▶ Upitni jezik sačinjavaju
 - ▶ operatori za izražavanje upita
 - ▶ pravila za formiranje operandu upita - izraza
 - ▶ pravila za primenu tih operatora
- ▶ Vrste teoretskih upitnih jezika u RMP
 - ▶ relacionala algebra
 - ▶ zasnovana na teoriji skupova i skupovnih operacija
 - ▶ relacioni račun
 - ▶ nad torkama
 - ▶ nad domenima
 - ▶ zasnovani na predikatskom računu I reda

Operacijska komponenta

► Osnovne skupovne operacije nad relacijama

► Unija

$$r(R) \cup s(R) = \{t \mid t \in r \vee t \in s\}$$

► Presek

$$r(R) \cap s(R) = \{t \mid t \in r \wedge t \in s\}$$

► Razlika

$$r(R) - s(R) = \{t \mid t \in r \wedge t \notin s\}$$

Operacijska komponenta

► Primer

r	A	B
	a_1	b_1
	a_2	b_2

s	A	B
	a_1	b_1
	a_3	b_3

$r \cup s$	A	B
	a_1	b_1
	a_2	b_2
	a_3	b_3

$r \cap s$	A	B
	a_1	b_1

$r - s$	A	B
	a_2	b_2

Operacijska komponenta

► **Selekcija**

- torki iz relacije
- omogućava izbor (selektovanje) torki relacije po nekom kriterijumu

$$\sigma_F(r(R)) = \{t \in r \mid F(t)\}$$

- logičkom formulom F izražava se kriterijum po kojem se torke relacije r selektuju
- biće selektovane samo one torke, za koje je formula F tačna
 - zahteva se formalno definisanje sintakse za zapisivanje selekcionih formula tipa F

Operacijska komponenta

► Primer

► $\sigma_F(r(R)), F ::= PLT > 5000$

<i>r</i>	<i>MBR</i>	<i>IME</i>	<i>POL</i>	<i>SPR</i>	<i>PLT</i>
	101	Ana	ž	11	3400
	102	Aca	m	14	4200
	110	Ivo	m	11	7000
	111	Olja	ž	11	7200

σ_F

Operacijska komponenta

► Upit

- prikazati radnike čija je plata veća od 4000 i rade na projektu sa šifrom 11
- $\sigma_{PLT > 4000 \wedge SPR = 11}(r)$

<i>MBR</i>	<i>IME</i>	<i>POL</i>	<i>SPR</i>	<i>PLT</i>
110	Ivo	m	11	7000
111	Olja	ž	11	7200

Operacijska komponenta

- ▶ **Projekcija (restrikcija) relacije**
 - ▶ izdvajanje vrednosti pojedinih kolona iz relacije
 - ▶ projektovanje relacije na podskup skupa obeležja
 - ▶ $X \subseteq R$

$$\pi_X(r(R)) = \{t[X] \mid t \in r(R)\}$$

Operacijska komponenta

► Primer

- P - pilot
- A - tip aviona
- L - broj leta

► Upit:

- prikazati pilote i tipove aviona na kojima lete:
- $\pi_{PA}(r(PAL))$

r	P	A	L
	<i>Aca</i>	<i>747</i>	<i>101</i>
	<i>Ivo</i>	<i>737</i>	<i>101</i>
	<i>Aca</i>	<i>747</i>	<i>102</i>
	<i>Ana</i>	<i>DC9</i>	<i>110</i>

P	A
<i>Aca</i>	<i>747</i>
<i>Ivo</i>	<i>737</i>
<i>Ana</i>	<i>DC9</i>

Operacijska komponenta

► Primer

► Posmatra se relacija r

r	MBR	IME	POL	SPR	PLT
	101	Ana	ž	11	3400
	102	Aca	m	14	4200
	110	Ivo	m	11	7000
	111	Olja	ž	11	7200

Operacijska komponenta

► Upit

- prikazati matične brojeve i imena radnika čija plata je veća od 4000, a rade na projektu sa šifrom 11
- $F ::= PLT > 4000 \wedge SPR = 11$
- $\pi_{MBR+IME}(\sigma_F(r))$

<i>MBR</i>	<i>IME</i>
<i>110</i>	<i>Ivo</i>
<i>111</i>	<i>Olja</i>

Operacijska komponenta

► Prirodni spoj relacija

- spajanje torki različitih relacija po osnovu istih vrednosti zajedničkih obeležja
- Date su relacije $r(R)$ i $s(S)$

$$r(R) \bowtie s(S) = \{t \in \text{Tuple}(RS) \mid t[R] \in r \wedge t[S] \in s\}$$

Operacijska komponenta



► Primer

<i>r</i>	<i>A</i>	<i>B</i>	<i>C</i>		<i>s</i>	<i>B</i>	<i>C</i>	<i>D</i>
	<i>a</i> ₁	<i>b</i> ₁	<i>c</i> ₁	↔		<i>b</i> ₁	<i>c</i> ₁	<i>d</i> ₁
	<i>a</i> ₁	<i>b</i> ₂	<i>c</i> ₂	↗		<i>b</i> ₁	<i>c</i> ₁	<i>d</i> ₂
	<i>a</i> ₁	<i>b</i> ₃	<i>c</i> ₃	↔		<i>b</i> ₃	<i>c</i> ₃	<i>d</i> ₃
						<i>b</i> ₄	<i>c</i> ₂	<i>d</i> ₂

<i>r</i> ▷ ◁ <i>s</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>
	<i>a</i> ₁	<i>b</i> ₁	<i>c</i> ₁	<i>d</i> ₁
	<i>a</i> ₁	<i>b</i> ₁	<i>c</i> ₁	<i>d</i> ₂
	<i>a</i> ₁	<i>b</i> ₃	<i>c</i> ₃	<i>d</i> ₃

Operacijska komponenta

► Primer

r	A	B		s	C	D
	a_1	b_1			c_1	d_1
	a_2	b_2			c_2	d_2

$r \bowtie s$	A	B	C	D
	a_1	b_1	c_1	d_1
	a_1	b_1	c_2	d_2
	a_2	b_2	c_1	d_1
	a_2	b_2	c_2	d_2

Operacijska komponenta

► Primer

Radnik

<i>MBR</i>	<i>IME</i>	<i>PLT</i>	<i>POL</i>
<i>101</i>	<i>Ana</i>	<i>3400</i>	<i>ž</i>
<i>102</i>	<i>Aca</i>	<i>4200</i>	<i>m</i>
<i>110</i>	<i>Ivo</i>	<i>7000</i>	<i>m</i>
<i>111</i>	<i>Olja</i>	<i>7200</i>	<i>ž</i>

Radproj

<i>MBR</i>	<i>SPR</i>
<i>101</i>	<i>11</i>
<i>101</i>	<i>14</i>
<i>102</i>	<i>14</i>
<i>110</i>	<i>13</i>
<i>110</i>	<i>11</i>

Projekat

<i>SPR</i>	<i>NAP</i>
<i>11</i>	<i>X25</i>
<i>13</i>	<i>Polaris</i>
<i>14</i>	<i>Univ. IS</i>

Operacijska komponenta

► Upit

- izlistati matične brojeve radnika, šifre i nazive projekata na kojima rade
- *Radproj* ▷◁ *Projekat*

<i>MBR</i>	<i>SPR</i>	<i>NAP</i>
<i>101</i>	<i>11</i>	<i>X25</i>
<i>101</i>	<i>14</i>	<i>Univ. IS</i>
<i>102</i>	<i>14</i>	<i>Univ. IS</i>
<i>110</i>	<i>13</i>	<i>Polaris</i>
<i>110</i>	<i>11</i>	<i>X25</i>

Operacijska komponenta

► Upit

- Izlistati matične brojeve i imena radnika, koji rade na projektu sa šifrom 11
- $\pi_{MBR+IME}(\sigma_{SPR=11}(Radproj) \triangleright \triangleleft Radnik)$, ili
- $\pi_{MBR+IME}(\sigma_{SPR=11}(Radproj \triangleright \triangleleft Radnik))$

<i>MBR</i>	<i>IME</i>
<i>101</i>	<i>Ana</i>
<i>110</i>	<i>Ivo</i>

Operacijska komponenta

► Dekartov proizvod relacija

- spajanje formiranjem svih mogućih kombinacija torki iz dve relacije
- $R \cap S = \emptyset$

$$r(R) \times s(S) = \{t \in Tuple(RS) \mid t[R] \in r \wedge t[S] \in s\}$$

► Theta spajanje relacija

- selektovanje torki po nekom kriterijumu iz dekartovog proizvoda relacija

$$r(R) \bowtie_F s(S) = \sigma_F(r \times s)$$

Operacijska komponenta

► Primer

► date su relacije

► *r* - red vožnje *Niš - Beograd*

► *s* - red vožnje *Beograd - Novi Sad*

<i>r</i>	<i>PNI</i>	<i>DBG</i>
	06:00	09:00
	08:00	10:30
	13:00	16:00

<i>s</i>	<i>PBG</i>	<i>DNS</i>
	10:00	11:15
	12:00	13:30

Operacijska komponenta

► Upit

- pregled svih mogućih varijanti za putovanje od *Niša* do *Novog Sada* s presedanjem u *Beogradu*
- $r \triangleright \triangleleft_{DBG < PBG} s = \sigma_{DBG < PBG}(r \times s)$

$r \triangleright \triangleleft_{DBG < PBG} s$	<i>PNI</i>	<i>DBG</i>	<i>PBG</i>	<i>DNS</i>
	06:00	09:00	10:00	11:15
	06:00	09:00	12:00	13:30
	08:00	10:30	12:00	13:30

Operacijska komponenta - zadaci za vežbu

- ▶ $\text{radnik}(\{\text{Mbr}, \text{Ime}, \text{Prz}, \text{Sef}, \text{Plt}, \text{God}, \text{Pre}\}, \{\text{Mbr}\}),$
- ▶ $\text{projekat}(\{\text{Spr}, \text{Ruk}, \text{Nap}, \text{Nar}\}, \{\text{Spr}\}),$
- ▶ $\text{radproj}(\{\text{Spr}, \text{Mbr}, \text{Brc}\}, \{\text{Spr} + \text{Mbr}\}),$

- ▶ $\text{radnik}[\text{Sef}] \subseteq \text{radnik}[\text{Mbr}],$
- ▶ $\text{projekat}[\text{Ruk}] \subseteq \text{radnik}[\text{Mbr}],$
- ▶ $\text{radproj}[\text{Mbr}] \subseteq \text{radnik}[\text{Mbr}],$
- ▶ $\text{radproj}[\text{Spr}] \subseteq \text{projekat}[\text{Spr}].$

Operacijska komponenta - zadaci za vežbu

- ▶ Koristeći operacije operacijske komponente RMP rešiti sledeće zadatke:
 1. Prikazati šifre i nazive projekata na kojima radi Ivan Tasić.
 2. Prikazati matične brojeve radnika koji rade na projektu sa šifrom 10 ili rade 5 sati.
 3. Prikazati projekte na kojima rade radnici čija je plata veća od 2500.
 4. Prikazati radnike (MBR, IME, PRZ) kojima je šef radnik sa MBR=10, ili zarađuju više od 2600.
 5. Prikazati podatke o projektima na kojima ne radi nijedan radnik.
 6. Prikazati podatke o radnicima koji ne rade na projektu sa šifrom P1.
 7. Prikazati radnike koji nisu rukovodioci projekata.
 8. Prikazati projekte na kojima radi šef radnika sa MBR=2001.
 9. Prikazati sve projekte na kojima rade oni radnici koji rade na projektu sa nazivom "Univerzitetski IS".
 10. Prikazati radnike koji rade na projektu sa nazivom "Nabavka".
 11. Prikazati radnike (IME, PRZ) koji na projektu sa nazivom "Prodaja" rade više od 10 sati nedeljno.
 12. Prikazati projekte (SPR, NAP) kojima rukovodi radnik Milan Perić.

Sadržaj

- ▶ Motivacija i koncepcija relacionog modela podataka
- ▶ Strukturalna komponenta
- ▶ Operacijska komponenta
- ▶ Integritetna komponenta

Pitanja i komentari



Kraj prezentacije

Relacioni model podataka

Motivacija, koncepcija, strukturalna, operacijska i integritetna komponenta