

Теме за държавен изпит

23. Бази от данни. Релационен модел на данните.

Основен елемент в модела е релацията - двумерна таблица, в която се съхраняват данните. [Атрибутите служат за имена на колонки. Името на релацията и мн-вото от атрибутите и формират схемата на релацията. Мн-вото от всички схеми на релации в БД е схема на БД.]

Предимства на релациите са, че са прости и съответстват на човешкото мислене. Също така са еднообразен начин за представяне на данните. Имаме четири основни понятия в модела:

def | Името на релацията и мн-вото от атрибутите и образуват схема на релацията. Атрибутите са множество, а не списък. Все пак, ние трябва да укажем тяхна стандартна подредба.

- релационна схема - име на релация + атрибут
- база от данни - колекция от релации
- схема на БД - мн-вото от всички релационни схеми в БД.

def | Атрибутите са означенията (имената) на колонките в релацията. Описват съдържанието на колонките. Атрибутите на една релация са практически атрибутите от съответната същност в ЕКР модел.

def | Картените са редове в релацията като изключим имената на колонките (заглавната част). ~~За~~ За всеки от атрибутите има по един компонент в картен. Релацията е множество от картени. Релационният модел изисква всеки компонент на картен да бъде атамуран.

def Домейните се асоциират с всеки атрибут. Компонентите на всеки картен принадлежат на домейна за съответния атрибут.

Една релация не може да бъде статична, а се променя с времето.

Релациите съхраняват instant instances. ~~Множеството~~ Множеството от картенни ~~за~~ за дадена релация наричаме екземпляр на релацията. Всяко множество от конкретни ~~за~~ екземпляри на релациите в БД наричаме текущо състояние на БД. Кардиналността на един екземпляр на дадена релация е равен на текущия брой картенни в релацията.

Реализация на релационна БД:

Процесът на проектиране на една рел. БД се състои от следните стъпки.

- определяне на данните (обектите и техните характеристики), които ще се съхраняват
- определяне на връзките между обектите
- определяне на ключовете и свързващите колонки
- определяне на ограниченията върху обектите
- отстраняване на евентуални недостатъци (напр. измислява "идр.")
- реализиране на БД.

~~Алгоритъм~~ Верифиране на схем на релациите:

- Всеки клас от обекти се представя с релация, която схема включва всички негови атрибути, които стават и атрибути на релацията. Всеки картен представя конкретен обект от класа. Списъкът от ключови атрибути (може и да е един), ~~т.е. тези, които~~ на класа, т.е. тези, които еднозначно ~~определят~~ идентифицират отделните обекти от класа, се приемат за ключ на релацията.

• Връзките между два или повече класове от обекти се представят чрез релации, като релационна схема включва класовите атрибути на всички от тези класове.

Видове операции в/у рел. БД. Заявки към рел. БД:

Като отбелязахме, една релация се променя с времето. Възможни са два вида операции в/у рел. БД:

- промени в схемата (не много често) - чрез Data Definition Language (DDL). Тези операции служат за дефиниране и промяна на структурата на таблиците и другите обекти в БД. DDL операциите са CREATE (създаване), DROP (спрехване) и ALTER (за промяна). Примери:

```
CREATE TABLE employees [  
    id                INTEGER      PRIMARY KEY,  
    first_name        CHAR(50)     NULL,  
    last_name         CHAR(50)     NOT NULL,  
    date-of-birth     DATE         NULL
```

];

```
ALTER TABLE employees ADD place-of-birth CHAR(50);  
DROP TABLE employees;
```

- селектиране, добавяне, промяна и изтриване на кортени (мат-често) - чрез Data Manipulation Language (DML). DML операциите са SELECT, INSERT, UPDATE, DELETE. Примери:

```
SELECT * FROM books;
```

```
INSERT INTO my-table ( field1, field2, field3) VALUES ('test', 'N', 21);
```

```
UPDATE my-table SET field1 = 'updated value' WHERE field2 = 'N';
```

```
DELETE FROM my-table WHERE field2 = 'N';
```


Релационна алгебра:

Алгебра, чито операнди са релации или променливи, представящи релации. Операторите отразяват най-разпространените манипулации върху релации в БД. В резултат, РА може да се използва като език за заявки за релации. Ако релациите са множества от кортени, то говорим за ядро на РА. Ако релациите са мултимножества от кортени - разширена РА.

1. Операции върху множества - в \mathcal{D} мулти, т.е. R и S са ~~свързани~~ ^{свързани} релационна ~~схема~~ (т.е. еднакъв брой атрибути и $\forall i \text{ dom}(A_i) = \text{dom}(B_i)$ и редът на стр. съвпада).

- Обединение - $R \cup S$. Основна операция, която е бинарна, комутативна и асоциативна

- разлика - $R - S$. Основна операция, която е бинарна и не е комутативна и асоциативна

- сечение - $R \cap S$. Допълнителна операция, която е бинарна, комутативна и асоциативна

2. Операции, които отстраняват части от релацията:

- проекция - унарна основна операция. Нека R е релация с n атрибута със схема $R(A_1, \dots, A_n)$. Проекцията на R по отношение на атрибутите $\{A_1, \dots, A_k\}$ е k -членна релация ~~схема~~

$R(A_1, \dots, A_k)$, която се получава от R чрез премахване на атрибутите, които не фигурират в схемата на R и в която повтарящите се кортени присъстват веднъж. Записва се $\pi_{\langle \text{attr-list} \rangle}(R)$, където $\langle \text{attr-list} \rangle$ е списък от атрибутите, които ще са в резултата.

- селекция - унарна основна операция. Нека R е релация с n атрибута със схема $R(A_1, \dots, A_n)$. Мра Селекцията на R по отношение на дадено условие F е друга релация R , която е със същата релационна схема и всички кортени удовлетворява условието F . Записва се $\sigma_{\langle \text{predicate} \rangle}(R)$,

Кдето $\langle \text{predicate} \rangle$ е условие, а предикатите имат следния вид:
 $\langle \text{attr} \rangle \langle \text{op} \rangle \langle \text{attr} \rangle$ или $\langle \text{attr} \rangle \langle \text{op} \rangle \langle \text{const} \rangle$. $\langle \text{attr} \rangle$ е атрибут,
 $\langle \text{const} \rangle$ е константа, а $\langle \text{op} \rangle$ е оператор като $=, \neq, <, >, \leq, \geq, \text{AND}, \text{OR}$.

~~Задаване).~~

~~• Частно - допълнителна операция, която е бинарна, комутативна и асоциативна - R: S.~~

3. Комбиниране на кортежите на релации

• Декартово произведение - основна операция, която е бинарна, комутативна и асоциативна. Записваме $R \times S$. Нека са дадени две релации

$R(A_1, \dots, A_n)$ и $Q(B_1, \dots, B_m)$. $R \times Q$ е трета релация R със схема $R(A_1, \dots, A_n, B_1, \dots, B_m)$, съдържаща всички възможни concatenции на кортежи от R и Q . Ако в R и Q има съвпадащ атрибут A , то в резултата се уточняват с $R.A$ и $Q.A$.

• Θ -свързване - $R \bowtie S$. Свързване на релациите $R(A_1, \dots, A_n)$ и $Q(B_1, \dots, B_m)$ по отношение на условието C е трета релация R със схема $R(A_1, \dots, A_n, B_1, \dots, B_m)$, която е такова подмножество на $R \times Q$, в което всеки кортеж отговаря на условието C . При изпълнението на операцията, първо се прави $R \times Q$ и след това се прави селекция по C . (Тази операция е допълнителна.)

• ~~Естествен~~ еднi join - Θ -свързване, при което условието C включва само по свързване на атрибутите. ~~Тази операция е допълнителна.~~

• Естествено свързване - $R \Join S$. Разширение на еднi join, при което се избърива свързване по всички атрибути с еднакви имена. Автоматично се отстранява повторението се колона.

Представяне на допълнителните операции чрез основните:

• сечение: $R \cap S = R - (R - S)$

• съединение: $R \bowtie S = \sigma_c(R \times S)$

• естествено съединение: $R \Join S = \pi_L(\sigma_c(R \times S))$, където:

- L е списък от атрибутите на R и атрибутите на S , които не са в R .

- c е условие от вида $R.A_1 = S.A_1 \text{ AND } \dots \text{ AND } R.A_n = S.A_n$,

където A_1, \dots, A_n са всички общи атрибути за R и S .

Приоритет на ~~операциите~~ операторите:

1. унарни оператори - сечение, проекция (, преименуване)

2. декартово произведение и съединение

3. сечение

4. Обединение и разлика.