

⑨ Всър от данни. Релационал модел на данните УВОД??. Чака с БД, СУБД
→ Релационал модел на данните Видове модели??

Същото: Основата на релационната модел на данните е тои математическото понятие п-степенна релация. Висока релация е ико-от имената, които се съставят от п и имената и се наричат п-терки. Всичко тук една релация е моделът даден чак от данни, а всяка п-терка от релацията представява конкретен обект от този чак: действуващата в учебника Групировка на релационните модели: се на фиг ???

① Групировка на модела - възможност за опирале с релации като с члената, като с използват опиралия на така наречената релационна алгебра

② Висок степен на издаваност на данните

③ Съответства на начин човешкото мислене

④ Енергичен начин за представяне на данните и връзките между тях без вид на таблици

→ Даници

Нека P_1, \dots, P_n са ико-ва, не неправилно различни. Всър от тези ико-ва е ~~изписано~~, разглежда също ико-ва от допустими стойности на ~~нека~~ величина и се нарича този обект имена данни. Данните могат да бъдат ~~изброени~~ или с други други елементи

→ Релации

Високо подобожество на алгоритмово пръвредение $D_1 \times D_2 \times \dots \times D_n$ за п данни (данни) се нарича релация. В контекста на БД релацията може да е разглеждана като двумерна таблици, в която се съдържат данни, т.е. таблици ~~от~~ елементи, в която разбране са п-терки, а отътвърдете съдържат елементи само от една и съща обикон.

→ Атрибути

Всички именувани ико-ва от ~~табличното~~ представяне на релацията се нарича атрибут, т.е. атрибутите са означенията (имена) на колоните на релацията (напр. title, year, length, FilmType). С всички атрибут е асоциирана граница (напр. Movies(title:string, year:integer, length:integer, FilmType:string))

→ Кортекси

Редовене на релациите (без запазване) и паричат кортекси.
Всички от тях съдържаат един именник за всички атрибути на
релациите. (напр. (Star Wars, 1977, 124, color))

→ Схема на релациите

Структурата на дадена релация се задава с нейната схема
и тя е уравнена пред цялото време на тиято съществуване. ???
Схемата на релациите е определена от тиято име на релациите
и място от атрибутиите ѝ. Например: Movies (title, year, length, film_type)

→ Схема на релационна БД

В релационния модел представят се структури от една или повече
релационни схеми. Съвкупността от всички релационни схеми,
с които се отдават качествата от единици и връзките между тях
се нарича релационна схема на БД (направено същата на БД)

• Схема на релация = име на релация + атрибути

• Тази от единици = калкуляция от релации

• Схема на БД = място от съществите на релациите в БД

Място от кортекси за дадена релация се нарича ~~изразяване~~
на релациите. Експликация се упражнява с времето.
Кардиналността на даден изразяване на дадена релация назава се
тежестта от кортекси в релациите. Всичко място от конкретни
експликации на релации се нарича тежестта съставяне на релационна БД.

→ Реализация на релационна БД

Процесът на превъртане на една релационна БД се състои от
следните стъпки:

① Определение на данните (обикновите и тектните характеристики), които
ще се съхраняват

② Определение на връзките между единиците

③ Определение на именовани и на свързванието на колони

④ Определение на ограниченията върху единиците и връзките между тях

⑤ Определяне на евиденчески поддържатели (излишества и др.)

⑥ Реализация на БД

Дефинирането на отделните релационни схеми е първо свободно, но най-често са старти сценарите по правилата:

① Всички клас от обекти са представени с релация, чиято схема включва всички нови атрибути. Всички п-тажки от релационата представя контролирани обекти от класа. Контролнат атрибут или от欠缺 от класовите атрибути на класа, т.е. тези, които еднозначно идентифицират отделните му обекти, се приемат за ключ на релацията.

② Всички и дълги пъти обекти класове от обекти са представени чрез релации, чиято релационна схема включва класовите атрибути на всички от тези класове от обекти.

→ Видове операции върху релационната БД

Видовете операции са:

① DML (Data Manipulation Language) - операции за промяна, добавяне, премахване и съпътствие на данните, използват се най-често. DML операциите са: SELECT, INSERT, UPDATE и DELETE. След изпълнение на тези операции е необходимо да се извърши commit, за да станат данните постоянни.

② DDL (Data Definition Language) - операции за промяна в схемата, използват се не много често. Служат за издаване и промяна на структурата на таблици и др. обекти в БД (редове, колони, индекси). DDL операции са: CREATE, DROP, ALTER. След изпълнение промяните се прилагат нездадено.

③ DCL (Data Control Language) - операции, които контролират достъпа до обектите в базата от данни. (GRANT, REVOKE)

→ Задачи към релационната БД

▷ DML:

• SELECT * FROM books - извлича всички хоризонти от релационна

SELECT books.title, COUNT * AS Authors и връща всички атрибути

FROM books JOIN book_authors ON books.isbn = book-auth. isbn

WHERE book.price > 100.00

GROUP BY books.title;

22.

- INSERT INTO my-table (Field1, Field2, Field3)
VALUES ('test', 'N', null);
- UPDATE my-table SET field1 = 'updated value'
WHERE Field2 = 'N';
- DELETE FROM my-table WHERE Field2 = 'N';

② DDL:

- CREATE TABLE employees (

id	INTEGER	PRIMARY KEY,
first-name	CHAR(50)	NULL,
last-name	CHAR(75)	NOT NULL,
date-of-birth	DATE	NULL

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

③ DCL: ??? ???

→ Релационна алгебра

Алгебрата е математическа система, при която чрез използването на различни оператори и операнди могат да се получават нови такива.

Релационна алгебра (RA) е такава алгебра, чието операции са релации или пременливи, представявани релации. Операторите са проектирани да извършват такива действия, които ще са способни да извършват с релации в една БД. RA е основата на всички език за заявки или БД-възможни СУБД използват RA като основен език за изразени на заявките.

Ако релациите са ин-ва (т.е. всички кортежи са също точно единични), говори за ядро на RA, а ако релациите са мултиплексни - за разширение на RA. За по-съдържателна реализация на операциите, СУБД използва мултиплексни от кортежи.

2.2. Погрешка: PA не е ~~множество~~ (по дефиниция на Тигорски) - чувствувам отпора, когато не могат да дадат описание чрез представата на PA, но могат да се опишат на програмен език.

→ Основни операции: → нуждам се да открио

① Обединение (union) - бинарна, комутативна, асоциативна операция
- $R \cup S$

Израза са използвана дума UNION

$\text{SELECT movieTitle AS title, movieYear AS year}$
 FROM starsIn неба работи ли ??

$(\text{SELECT title, year FROM Movie})$ не ни хареса

първото израза не е от нормиран, когато са в starsIn Movie. Ако една норма приложва и в starsIn и в Movie, не се губят всички записи в обединението

R		S		R ∪ S	
title	year	starTitle	starYear	title	year
t1	1	t3	3	t1	1
t2	2	t4	4	t2	2

② Разлика (set difference) - две пъти разлика, но с комутативно
($m.e. R - S \neq S - R$)

~~($R \setminus S$)~~ Ключова дума EXCEPT

Израза са използвана дума INTERSECT

~~($\text{SELECT name, address FROM MovieStar WHERE gender = 'F'}$)~~

~~INTERSECT~~

~~($\text{SELECT name, address FROM MovieStar WHERE networth > 1000000}$)~~

- първото е множество от нормиран, когато са

$R - S$ е разлика на релациите $R \cup S$. Резултатът е множество от всички норми от R , когато те са в S .

~~($\text{SELECT Name, address FROM MovieStar}$)~~

EXCEPT

~~($\text{SELECT name, address from MovieExec}$);~~

R		S		$R - S$	
Name	Age	Name	Age	Name	Age
Pesho	24	Maria	23	Pesho	24
Maria	29	Stoyan	30		

3
5

③ Сече - $R \cap S$ е резултат на релациите R и S . Операцията е асоциативна и ханумативна. Резултатът е мн-во от ел. (кортежи), които са еднакви и в R и S . Даден е и се навърва само единични в резултата. Кодова група - INTERSECT.

(SELECT name, address FROM MovieStar WHERE gender = 'F')
INTERSECT

(SELECT name, address FROM MovieStar WHERE networth > 1000000)

R	S	$R \cap S$																		
<table border="1"> <tr> <td>name</td> <td>age</td> </tr> <tr> <td>Pesho</td> <td>34</td> </tr> <tr> <td>Maria</td> <td>29</td> </tr> <tr> <td>Maria</td> <td>29</td> </tr> </table>	name	age	Pesho	34	Maria	29	Maria	29	<table border="1"> <tr> <td>name</td> <td>age</td> </tr> <tr> <td>Maria</td> <td>40</td> </tr> <tr> <td>Stoyan</td> <td>27</td> </tr> </table>	name	age	Maria	40	Stoyan	27	<table border="1"> <tr> <td>name</td> <td>age</td> </tr> <tr> <td>Maria</td> <td>29</td> </tr> </table>	name	age	Maria	29
name	age																			
Pesho	34																			
Maria	29																			
Maria	29																			
name	age																			
Maria	40																			
Stoyan	27																			
name	age																			
Maria	29																			

Първа търсена да е първо?

Операците обединение, сече и разлика се прилагат върху релации, които имат еднакви схеми и еднакви заместители за всички атрибути. Търси да се изучат резултатът от операците, наложени на две релации преди да бъдат подредени, така, че първата на резултата да е една и съща. Търсиха също оператор с много подходящ, когато искаме да приложим някак от направените обединение, сече и разлика върху две релации, които имат различни атрибути с еднакви заместители, но с различни имена.

④ Департманско произведение $R \times S$

$R \times S$ е департманско произведение на релациите R и S . Резултатът е релация, която съдържа кортежи с по-голяма дължина, които комбинират всички кортежи от R с всички кортежи от S . Търсиха също, атрибутите на релацията от място на операндата предишните атрибутите на тази от място в резултантната релация. Тако имена от атрибутите имена щели да са еднакви. Но за да се измине това, това са всички имена на атрибутите да са еднакви. Това е и във времето R и S , и са на първото означение $R.A$ и $R.S$ в схемата.

R	S	$R \times S$																								
<table border="1"> <tr> <td>A</td> <td>B</td> </tr> <tr> <td>1</td> <td>x</td> </tr> <tr> <td>2</td> <td>y</td> </tr> </table>	A	B	1	x	2	y	<table border="1"> <tr> <td>C</td> </tr> <tr> <td>z</td> </tr> <tr> <td>p</td> </tr> </table>	C	z	p	<table border="1"> <tr> <td>A</td> <td>B</td> <td>C</td> </tr> <tr> <td>1</td> <td>x</td> <td>z</td> </tr> <tr> <td>2</td> <td>y</td> <td>z</td> </tr> <tr> <td>1</td> <td>x</td> <td>p</td> </tr> <tr> <td>2</td> <td>y</td> <td>p</td> </tr> </table>	A	B	C	1	x	z	2	y	z	1	x	p	2	y	p
A	B																									
1	x																									
2	y																									
C																										
z																										
p																										
A	B	C																								
1	x	z																								
2	y	z																								
1	x	p																								
2	y	p																								

SELECT # FROM R, S

6

⑤ Проекция

Проектиращият оператор с използва чисто искаме да създадем нова релация от релацията R, като използва само лист от ненините атрибути.

R

A	B
1	X
2	Y

SELECT A FROM R

⑥ Селекция

Този оператор приложи иди релацията R връща резултат, който е подмножество от кортекси на R, които удовлетворяват начинът предварително зададен предикат

R

A	B
1	X
2	Y

A	B
2	Y

SELECT * FROM R
WHERE A > 1

→ Допълнителни: - ну дали да са от други

⑦ Сечења - в горните

② Частно-дихарна операция, при което:

Нека R ($A_1, A_2, \dots, A_n, B_1, \dots, B_k$) е ^(кофактор) релация и Q (B_1, \dots, B_k) е релация. Тогава частното $R \div Q = S(A_1, A_2, \dots, A_n)$ за всички $\forall q \in Q$ също R, ако $S \times Q = R$, то $S = R \div Q$ и $Q = R \div S$.

③ Естествено сечење - делене с R $\div S$ образува нова релация, в която всяка тази двойка кортекси от R и S, която съвпадат (или единствено идентични) за съвпадащите атрибути (такива съвпадащи имена иденти) т.е. ако A_1, \dots, A_n са съвпадащи атрибути в R и S, то тогава S се успешно съвпада, ако всички единствени идентици са A_1, \dots, A_n . Тогава резултатът кортекси автоматично се принасят.

A	B	C
1	2	
2	3	
3	4	
		5
		6
		7
		8
		9
		10

A	B	C
1	2	
2	3	
3	4	

$R \div S = \cap_L (\sigma_C(R \times S))$, където $\sigma_C =$ селекция по $R, A_i = S, A_i \in R$ и $R_A_i = S_A_i, \dots$ и \cap_L съвместно по всички атрибути от R и всички от L

④ Сечење - (Theta-сечење) Понякога бих искал да съвмеси кортекси на дада условие, и не само на дада едини атрибути. Нека K и S са релации. Операцията $R \div_C S$ на дада условие C е аналогична на 1) Дихарното произведение $R \div_A S = \sigma_C(R \times S)$, т.е. съдържа всички съвпади кортекси от S и R, които отг. на даданото условие.

Ако това повтори се инициалите на отговарящите търсещи ѝ са
преписани във всичко. Резултатната схема е обединение на схемите
Р и Г са 5, както при декартова.