

Релационен модел

Таблицы
Схеми
Преобразуване от E/R модел в релационен

1

Релационен модел

- Днешните разработки на бази данни почти винаги се базират на *релационния модел*
- Основен елемент в модела е *релацията* - двумерна таблица, в която се съхраняват данните
 - Основава се на математическото понятие релация (mathematical relation)
- Атрибутите служат за имена на колони
- Името на релацията и множеството от атрибутите ѝ формират схемата на релацията
- Множеството от всички схеми на релации в една БД - схема на БД - *relational database schema (database schema)*

2

Защо релации?

- Простота на модела
- Съответства на начина на човешкото мислене
- Еднообразен начин за представяне на данните и връзките във вид на таблица

3

Основни понятия

- Атрибути
- Схеми
- Кортежи
- Домейни

4

Атрибути

- Атрибути - имената на колоните на релацията.
 - *title, year, length* и *filmType*.
- Атрибутите на една релация са практически атрибутите от съответната същност в E/R-модела

5

Схеми

- Името на релацията и множеството от атрибутите ѝ - схема на релацията.
- Изписваме схемата на релация като оказваме името, следвано от атрибутите в скоби.
 - *Movies(title, year, length, filmType)*
 - Атрибутите в релационната схема са множество, а не списък. Независимо от това, ние трябва да укажем стандартна подредба на атрибутите един спрямо друг.
- В релационния модел, проектът се състои една или няколко релационни схеми. Това множество се нарича релационна схема на базата данни (или накратко схема на базата от данни).

6

Schema

- *Relation schema* =
relation name + attributes
- *Database* = collection of relations.
- *Database schema* =
set of all relation schemas in the database

7

Пример : релация Movies

Columns (attributes): arity of relation

Movies	title	year	length	filmType
	Star Wars	1977	124	color
	Mighty Ducks	1991	104	color
	Wayne's World	1992	95	color

Relation Name

Rows (tuples)

Schema: **Movies(title, year, length, filmType)**

Релацията е м-во от уникални кортежи
Cardinality of a relation is the number of tuples in it

8

Еквивалентно представяне

- Начинът на подреждане на атрибутите не е съществен
- Пример - кортежите
(Star Wars, 1977, 124, color) и
(1977, Star Wars, color, 124)
представят един и същ елемент
- Съвет: избира се един начин на подреждане на атрибутите и се спазва през цялото време
- Същата релация:

Movies	title	year	length	filmType
	Mighty Ducks	1991	104	color
	Star Wars	1977	124	color
	Wayne's World	1992	95	color

9

Кортежи

- Редовете в релацията (без заглавната част - атрибутите) се наричат кортежи (tuples).
 - един компонент за всеки от атрибутите
(Star Wars, 1977, 124, color)
 - Следва се вече определената последователност за атрибутите от релационната схема
- Релацията е м-во от кортежи (**set of tuples** \Rightarrow no tuple may appear more than once)
- Релационният модел изисква всеки компонент на кортеж да бъде атомарен (неделим)
 - Прости типове - integer, string
 - Сложни типове като lists, arrays не са разрешени

10

Домейни

- Асоциират се с всеки атрибут
 - elementary type
- Компонентите на всеки кортеж принадлежат на домейна за съответния атрибут
- Пример: Movies
Movies(
title:string,
year:integer,
length:integer,
filmType:string)

11

Екземпляри на релации

- Една реална релация не може да бъде статична, а се променя с времето
- Два типа промени:
 - insert, update, deletion of tuples (много често)
 - промени в схемата (по-рядко)
- Множеството от кортежи за дадена релация ще наричаме екземпляр на релацията
 - Екземплярите на релациите се променят с времето
 - Релациите съхраняват current instance
 - Cardinality of relation instance = num. of tuples in relation
- Схема \neq Екземпляр (Schema \neq Instance)
- Relation instance, database instance

12

От E/R диаграми към релации

Основни правила:

- Всяка същност се преобразува в отделна релация със същите атрибути
- Всяка връзка се преобразува в релация, чиито атрибути са:
 - Ключовете на свързаните същности
 - Атрибутите на връзката (ако има такива)

13

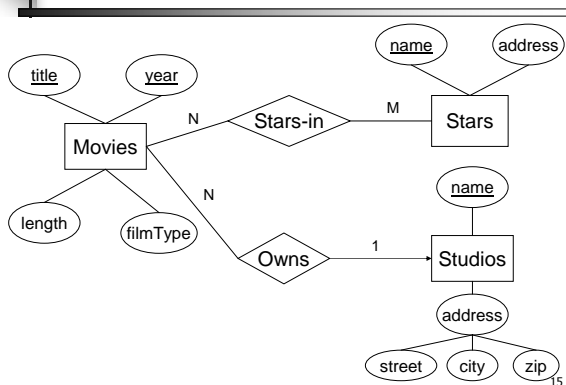
От E/R диаграми към релации (2)

Специални случаи:

- слабите м-ва същности (weak entity sets) не могат да се преобразуват директно в релации
- "Isa"-връзките и подкласовете изискват специална обработка
- понякога комбинираме 2 релации

14

Movie Database



15

Example

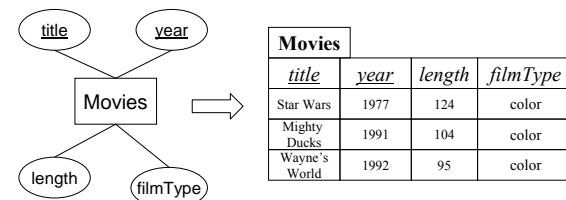
Movies	<i>title</i>	<i>year</i>	<i>length</i>	<i>filmType</i>
	Star Wars	1977	124	color
	Mighty Ducks	1991	97	color
	Wayne's World	1992	109	color

Owns	<i>title</i>	<i>year</i>	<i>studioName</i>
	Star Wars	1977	Fox
	Mighty Ducks	1991	Disney
	Wayne's World	1992	Paramount

Studios	<i>studioName</i>	<i>street</i>	<i>city</i>	<i>zip</i>
	Fox	Fox Blvd.	Hollywood	90037
	Disney	Main Str.	Buena Vista	90065
	Paramount	Av. of the Stars	Hollywood	90037

16

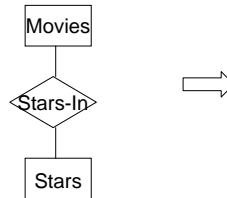
Преобразуване на същности



- 1 таблица за всяка същност (strong entity)
 - Ключът на същността става ключ на релацията

17

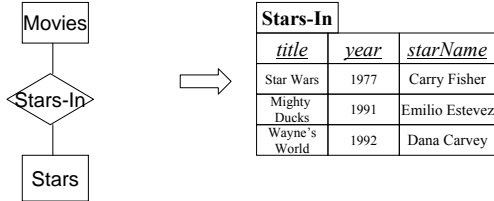
От E/R връзки към релации



- Използвайте ключа на I същност и ключа на II същност
- Name → starName

18

От E/R връзки към релации

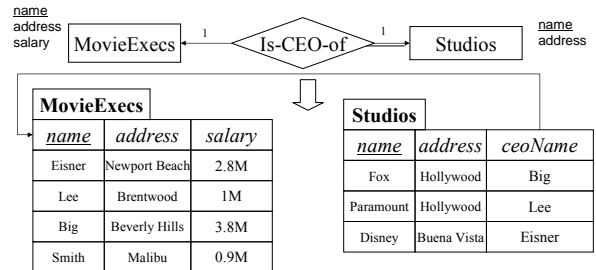


Stars-In		
<u>title</u>	<u>year</u>	<u>starName</u>
Star Wars	1977	Carry Fisher
Mighty Ducks	1991	Emilio Estevez
Wayne's World	1992	Dana Carvey

- Използвайте ключа на I същност и ключа на II същност
 - Name → starName

19

Relationship Types - 1:1



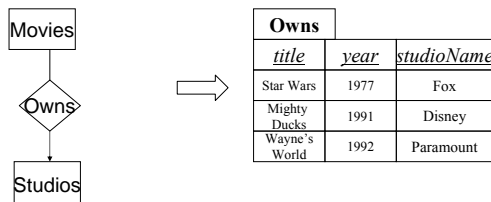
MovieExecs		
<u>name</u>	<u>address</u>	<u>salary</u>
Eisner	Newport Beach	2.8M
Lee	Brentwood	1M
Big	Beverly Hills	3.8M
Smith	Malibu	0.9M

Studios		
<u>name</u>	<u>address</u>	<u>ceoName</u>
Fox	Hollywood	Big
Paramount	Hollywood	Lee
Disney	Buena Vista	Eisner

- Не е необходимо връзката да се представя като отделна релация

20

От E/R връзки към релации



Owns		
<u>title</u>	<u>year</u>	<u>studioName</u>
Star Wars	1977	Fox
Mighty Ducks	1991	Disney
Wayne's World	1992	Paramount

- Използвайте ключа на I същност и ключа на II същност
 - Name → studioName

21

Комбиниране на релации (1)

- Пример:
- Movies(title,year,length,filmType) и Owns(title,year,studioName) се комбинират
- Movies1(title,year,length,filmType, studioName) .

22

Комбиниране на релации (2)

Movies1(title,year,length,filmType, studioName)

Movies1	<u>title</u>	<u>year</u>	<u>length</u>	<u>filmType</u>	<u>studioName</u>
	Star Wars	1977	124	color	Fox
	Mighty Ducks	1991	97	color	Disney
	Wayne's World	1992	109	color	Paramount

23

Комбиниране на релации (3)

- Когато имаме същност E с връзка R "M:1" от E към F могат да се комбинират една релация със схемата, състояща се от:
 - Всички атрибути на E
 - Ключовите атрибути за F
 - Атрибутите на връзката R (ако има такива)



24

Рискове при връзки M:M

Комбинирането на Movies1 със Stars-In ще доведе до излишество

Movies1	title	year	length	filmType	studioName	starName
	Star Wars	1977	124	color	Fox	Carry Fisher
	Mighty Ducks	1991	97	color	Disney	Emilio Estevez
	Wayne's World	1992	109	color	Paramount	Dana Carvey
	Star Wars	1977	124	color	Fox	Mark Hamill
	Star Wars	1977	124	color	Fox	Harrison Ford
	Wayne's World	1992	109	color	Paramount	Mike Meyers

Redundancy

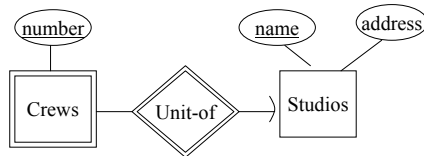
25

Слаби множества същности

- Релацията за слаба същност W трябва да включва не само неключовите атрибути на W , но също и ключовите атрибути на “поддържащата” същност,
- Поддържащата (double-diamond) връзка създава излишество и не се преобразува в релация

26

Пример



- Studios(name, address)
- Crews(number, studioName)
 - add non-key attributes
- ~~Unit-of(number, studioName, name)~~

27

Изводи

- Ако W е слаба същност, съставянето на релация за W ще бъде със схема от:
 - Всички атрибути на W ,
 - Всички атрибути на поддържащите връзки,
 - За всяка поддържаща връзка за W , която е “много към един” (many-one) от W към същност E , всички ключови атрибути на E .
- Не се съставя релация за никоя от поддържащите връзки към W !

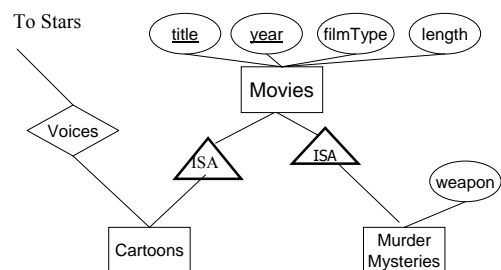
28

Конвертиране на иерархии от подкласове в релации

- При “ISA” иерархия от същности:
 - съществува същност, корен на иерархията;
 - тази същност има ключ, който служи да се идентифицира всеки елемент, представен от иерархията;
 - даден елемент може да има компоненти, които принадлежат на същностите на кое да е поддърво (subtree) на иерархията, стига това поддърво да включва корена.

29

Преобразуване на подкласове в релации



- The Movie hierarchy

30

М-ва същности с подкласове: 3 подхода

E/R подход: за всяко м-во същности E от йерархията създаваме по релация, която включва ключовите арибути на корена и атрибутите на E

Обектно-ориентиран подход : за всяко възможно поддърво, включително корена, се създава релация, която включва всички атрибути на всички м-ва същности от поддървото

Null стойности : създаваме 1! релация с всички атрибути на всички м-ва същности от йерархията

31

E/R подход

Създаване на релация за всяко м-во същности

- ако м-то същности не е корен, добавя с ключът на корена

- ◆ Movies(title, year, length, filmType)
- ◆ MurderMysteries(title, year, weapon)
- ◆ Cartoons(title, year)

- ◆ Voices(title, year, starName)

32

ОО подход

Има 4 възможни поддървета, вкл. корена:

Само Movies

Movies(title, year, length, filmType)

Movies и Cartoons

MoviesC(title, year, length, filmType)

Movies и Murder-Mysteries

MoviesMM(title, year, length, filmType, weapon)

Всички 3 м-ва същности

MoviesCMM(title, year, length, filmType, weapon)

Връзка Voices – M:M

Voices(title, year, starName)

33

Използване на Nulls

Ако ни е позволено да използваме null-стойности (the NULL value as in SQL) като стойности в кортежите, можем да пресъздадем йерархията на същностите с единствена релация

- всички атрибути от всички същности

Movies(title, year, length, filmType, weapon)

- Voices

Voices(title, year, starName)

34

Сравнение на подходите

- Кои филми от 1999 са по-дълги от 150 мин.?
- Какви оръжия са използвани в cartoons, по-дълги от 150 мин.?

35

Сравнение

- E/R подходът е добър при заявки от вида "What films of 1999 were longer than 150 minutes?"
 - само Movies relation.
- O-O подходът е добър при заявки от вида "What weapons were used in cartoons of over 150 minutes in length?"
 - MoviesCMM relation.
- Използването на nulls спестява място, освен в случаите на много атрибути, чиито стойности обичайно са *null*.

36