

## 58. Реляционный подход. Реляционная структура данных. Реляционная база данных. Манипулирование реляционными данными.

ОСНОВЫ ПРОЕКТИРОВАНИЯ РЕЛЯЦИОННЫХ БАЗ ДАННЫХ:

<http://citforum.ru/database/dbguide/>

SQL:

<http://ru.wikipedia.org/wiki/SQL>

### Реляционная структура данных

В конце 60-х годов появились работы, в которых обсуждались возможности применения различных табличных даталогических моделей данных, т.е. возможности использования привычных и естественных способов представления данных. Наиболее значительной из них была статья сотрудника фирмы IBM д-ра Э.Кодда (Codd E.F., A Relational Model of Data for Large Shared Data Banks. CACM 13: 6, June 1970), где, вероятно, впервые был применен термин "реляционная модель данных".

Будучи математиком по образованию Э.Кодд предложил использовать для обработки данных аппарат теории множеств (объединение, пересечение, разность, декартово произведение). Он показал, что любое представление данных сводится к совокупности двумерных таблиц особого вида, известного в математике как отношение – relation (англ.) [3, 7, 9].

Наименьшая единица данных реляционной модели – это отдельное атомарное (неразложимое) для данной модели значение данных. Так, в одной предметной области фамилия, имя и отчество могут рассматриваться как единое значение, а в другой – как три различных значения.

Доменом называется множество атомарных значений одного и того же типа. Так, на рис. 1 домен пунктов отправления (назначения) – множество названий населенных пунктов, а домен номеров рейса – множество целых положительных чисел.

Смысл доменов состоит в следующем. Если значения двух атрибутов берутся из одного и того же домена, то, вероятно, имеют смысл сравнения, использующие эти два атрибута (например, для организации транзитного рейса можно дать запрос "Выдать рейсы, в которых время вылета из Москвы в Сочи больше времени прибытия из Архангельска в Москву"). Если же значения двух атрибутов берутся из различных доменов, то их сравнение, вероятно, лишено смысла: стоит ли сравнивать номер рейса со стоимостью билета?

Отношение на доменах  $D_1, D_2, \dots, D_n$  (не обязательно, чтобы все они были различны) состоит из заголовка и тела. На рис. 1 приведен пример отношения для расписания движения самолетов.

Заголовок состоит из такого фиксированного множества атрибутов  $A_1, A_2, \dots, A_n$ , что существует взаимно однозначное соответствие между этими атрибутами  $A_i$  и определяющими их доменами  $D_i$  ( $i=1,2,\dots,n$ ).

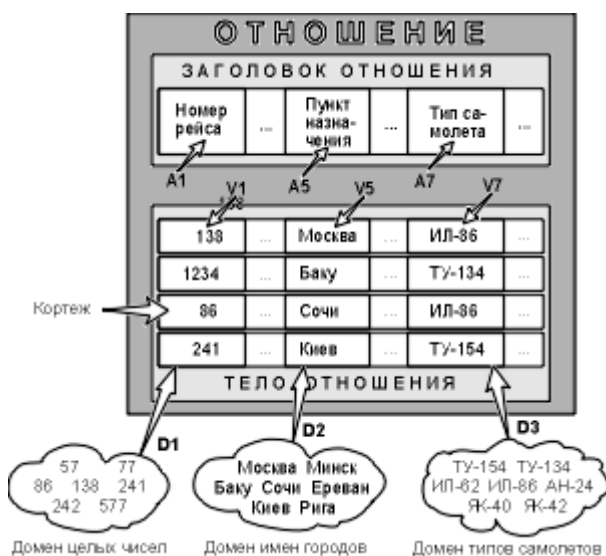


Рис. 1. Отношение с математической точки зрения ( $A_i$  - атрибуты,  $V_i$  - значения атрибутов)

Тело состоит из меняющегося во времени множества кортежей, где каждый кортеж состоит в свою очередь из множества пар атрибут-значение ( $A_i:V_i$ ), ( $i=1,2,\dots,n$ ), по одной такой паре для каждого атрибута  $A_i$  в заголовке. Для любой заданной пары атрибут-значение ( $A_i:V_i$ )  $V_i$  является значением из единственного домена  $D_i$ , который связан с атрибутом  $A_i$ .

Степень отношения – это число его атрибутов. Отношение степени один называют унарным, степени два – бинарным, степени три – тернарным, ..., а степени  $n$  –  $n$ -арным.

Кардинальное число или мощность отношения – это число его кортежей. Кардинальное число отношения изменяется во времени в отличие от его степени.

Поскольку отношение – это множество, а множества по определению не содержат совпадающих элементов, то никакие два кортежа отношения не могут быть дубликатами друг друга в любой произвольно-заданный момент времени. Пусть  $R$  – отношение с атрибутами  $A_1, A_2, \dots, A_n$ . Говорят, что множество атрибутов  $K=(A_i, A_j, \dots, A_k)$  отношения  $R$  является возможным ключом  $R$  тогда и только тогда, когда удовлетворяются два независимых от времени условия:

1. Уникальность: в произвольный заданный момент времени никакие два различных кортежа  $R$  не имеют одного и того же значения для  $A_i, A_j, \dots, A_k$ .
2. Минимальность: ни один из атрибутов  $A_i, A_j, \dots, A_k$  не может быть исключен из  $K$  без нарушения уникальности.

Каждое отношение обладает хотя бы одним возможным ключом, поскольку по меньшей мере комбинация всех его атрибутов удовлетворяет условию уникальности. Один из возможных ключей (выбранный произвольным образом) принимается за его первичный ключ. Остальные возможные ключи, если они есть, называются альтернативными ключами.

Вышеупомянутые и некоторые другие математические понятия явились теоретической базой для создания реляционных СУБД, разработки соответствующих языковых средств и программных систем, обеспечивающих их высокую производительность, и создания основ теории проектирования баз данных. Однако для массового пользователя реляционных СУБД можно с успехом использовать неформальные эквиваленты этих понятий:

- Отношение – Таблица (иногда Файл),
- Кортеж – Строка (иногда Запись),
- Атрибут – Столбец, Поле.

При этом принимается, что "запись" означает "экземпляр записи", а "поле" означает "имя и тип поля".

## Реляционная база данных

**Реляционная база данных** – это совокупность отношений, содержащих всю информацию, которая должна храниться в БД. Однако пользователи могут воспринимать такую базу данных как совокупность таблиц. Так на рис. 2 показаны таблицы базы данных, построенные по инфологической модели базы данных "Питание".

Блюда			Продукты			Состав		
БЛ	Блюдо	Вид	ПР	Продукт	Калор.	Б.Л	ПР	Вес (г)
1	Лобио	Закуска	1	Фасоль	3070	1	1	200
2	Харчо	Суп	2	Лук	450	1	2	40
3	Шашлык	Горячее	3	Масло	7420	1	3	30
4	Кофе	Десерт	4	Зелень	180	1	4	10
			5	Мясо	1660	2	5	80
			6	Томаты	240	2	2	30
			7	Рис	3340	2	6	40
			8	Кофе	2750	2	7	50
						2	3	15
						2	4	15
						3	5	180
						3	6	100
						3	2	40
						3	4	20
						4	8	8

Расход			Рецепты	
БЛ	Порций	Дата_Р	Б.Л	Рецепт
1	158	1/9/94	1	Ломаную очип
2	144	1/9/94		...
3	207	1/9/94		
4	235	1/9/94		
...	...	...		

Поставщики			Поставки				
ПОС	Поставщик	Город	ПОС	ПР	Вес (кг)	Цена	Дата_П
1	"Полесье"	Киев	1	6	120	0.45	27/8/94
2	"Наталка"	Киев	1	3	50	1.82	27/8/94
3	"Хуанхэ"	Пекин	1	2	50	0.61	27/8/94
4	"Лайма"	Рига	2	2	100	0.52	27/8/94
5	"Юрмала"	Рига	2	5	100	2.18	27/8/94
6	"Даугава"	Рига	2	4	10	0.88	27/8/94
			3	1	250	0.37	24/8/94
			3	7	75	0.44	24/8/94
			3	8	40	2.87	24/8/94
			4	3	70	1.56	30/8/94
			5	5	200	2.05	30/8/94
			6	6	15	0.99	30/8/94

Города	
Город	Страна
Киев	Украина
Пекин	Китай
Рига	Латвия

Рис. 2. База данных "Питание"

1. Каждая таблица состоит из однотипных строк и имеет уникальное имя.
2. Строки имеют фиксированное число полей (столбцов) и значений (множественные поля и повторяющиеся группы недопустимы). Иначе говоря, в каждой позиции таблицы на пересечении строки и столбца всегда имеется в точности одно значение или ничего.
3. Строки таблицы обязательно отличаются друг от друга хотя бы единственным значением, что позволяет однозначно идентифицировать любую строку такой таблицы.
4. Столбцам таблицы однозначно присваиваются имена, и в каждом из них размещаются однородные значения данных (даты, фамилии, целые числа или денежные суммы).
5. Полное информационное содержание базы данных представляется в виде явных значений данных и такой метод представления является единственным. В частности, не существует каких-либо специальных "связей" или указателей, соединяющих одну таблицу с другой. Так, связи между строкой с БЛ = 2 таблицы "Блюда" на рис. 3.2 и строкой с ПР = 7 таблицы продукты (для приготовления Харчо нужен Рис), представляется не с помощью указателей, а благодаря существованию в таблице "Состав" строки, в которой номер блюда равен 2, а номер продукта – 7.
6. При выполнении операций с таблицей ее строки и столбцы можно обрабатывать в любом порядке безотносительно к их информационному содержанию. Этому способствует наличие имен таблиц и их столбцов, а также возможность выделения любой их строки или любого набора строк с указанными признаками (например, рейсов с пунктом назначения "Париж" и временем прибытия до 12 часов).

## Манипулирование реляционными данными

Предложив реляционную модель данных, Э.Ф.Кодд создал и инструмент для удобной работы с отношениями – реляционную алгебру. Каждая операция этой алгебры использует одну или несколько таблиц (отношений) в качестве ее операндов и продуцирует в результате новую таблицу, т.е. позволяет "разрезать" или "склеивать" таблицы (рис. 3).

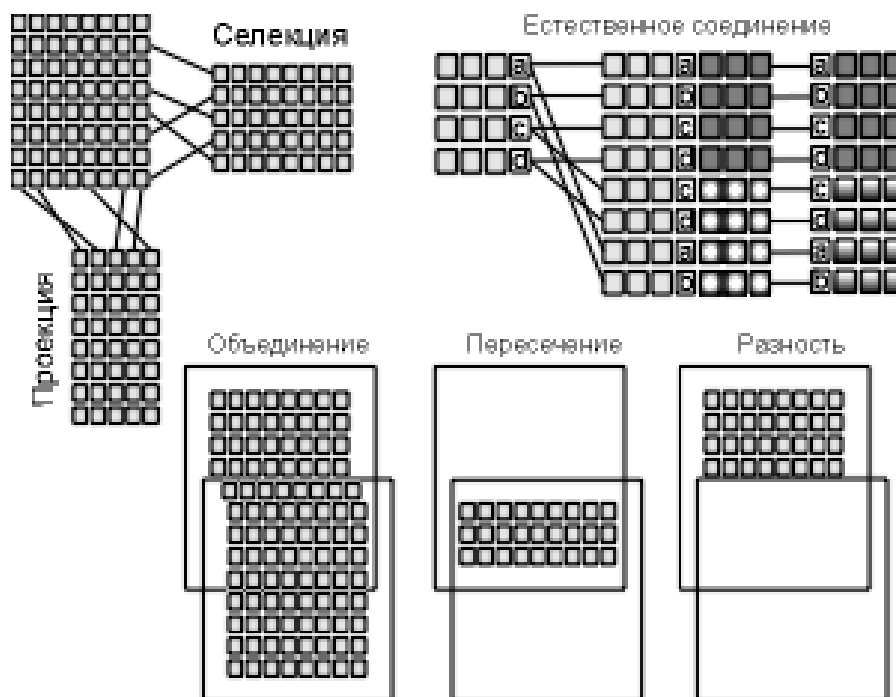


Рис. 3. Некоторые операции реляционной алгебры

Созданы языки манипулирования данными, позволяющие реализовать все операции реляционной алгебры и практически любые их сочетания. Среди них наиболее распространены SQL (Structured Query Language – структуризованный язык запросов) и QBE (Query-By-Example – запросы по образцу). Оба относятся к языкам очень высокого уровня, с помощью которых пользователь указывает, какие данные необходимо получить, не уточняя процедуру их получения.

С помощью единственного запроса на любом из этих языков можно соединить несколько таблиц во временную таблицу и вырезать из нее требуемые строки и столбцы (селекция и проекция).