

2.5. Теоретические языки запросов

Операции, выполняемые над отношениями, можно разделить на две группы. Первую группу составляют операции над множествами, к которым относятся операции: объединения, пересечения, разности, деления и декартова произведения. Вторую группу составляют специальные операции над отношениями, к которым, в частности, относятся операции: проекции, соединения, выбора.

В различных СУБД реализована некоторая часть операций над отношениями, определяющая в какой-то мере возможности данной СУБД и сложность реализации запросов к БД.

- В реляционных СУБД для выполнения операций над отношениями используются две группы языков, имеющие в качестве своей математической основы теоретические языки запросов, предложенные Э.Коддом:

- реляционная алгебра;
- реляционное исчисление.

Эти языки представляют минимальные возможности реальных языков манипулирования данными в соответствии с реляционной моделью и эквивалентны друг другу по своим выразительным возможностям. Существуют не очень сложные правила преобразования запросов между ними.

В реляционной алгебре операнды и результаты всех действий являются отношениями. Языки реляционной алгебры являются процедурными, так как отношение, являющееся результатом запроса к реляционной БД, вычисляется при выполнении последовательности реляционных операторов, применяемых к отношениям. Операторы состоят из операндов, в роли которых выступают отношения, и реляционных операций. Результатом реляционной операции является отношение.

Языки исчислений, в отличие от реляционной алгебры, являются непроцедурными (описательными, или декларативными) и позволяют выражать запросы с помощью предиката первого порядка (высказывания в виде функции), которому должны удовлетворять кортежи или домены отношений.

Запрос к БД, выполненный с использованием подобного языка, содержит лишь информацию о желаемом результате. Для этих языков характерно наличие наборов правил для записи запросов.

2.6. Реляционная алгебра

Предложив реляционную модель данных, 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 получает в результате новую таблицу, т.е. позволяет "разрезать" или "склеивать" таблицы (рис. 1).

Реляционные операторы обладают одним важным свойством: они замкнуты относительно понятия отношения. Это означает, что выражения реляционной алгебры определяются над отношениями реляционных БД и результатом вычисления также являются отношения.

Поскольку результатом любой реляционной операции является некоторое отношение, запрос, представленный на языке реляционной алгебры, может быть вычислен на основе вычисления элементарных алгебраических операций с учетом их старшинства и возможного наличия скобок.

Набор основных алгебраических операций состоит из восьми операций, которые делятся на два класса - теоретико-множественные операции и специальные реляционные операции, дополненные некоторыми специальными операциями, специфичными для баз данных.

В состав теоретико-множественных операций входят традиционные операции над множествами:

- объединение;
- пересечение;
- разность;
- декартово произведение.

Специальные реляционные операции включают:

- выборку;
- проекцию;
- естественное соединение;
- деление.



Рис. 1. Некоторые операции реляционной алгебры

Операции объединения, пересечения и разности требуют от операндов совместимости по типу. Два отношения совместимы по типу, если:

1. Каждое из них имеет одно и то же множество имен атрибутов (одна и та же степень),
2. Соответствующие атрибуты (с одинаковыми именами) определены на одном и том же домене.

Отношение В			
ID_NUM	NAME	CITY	AGE
1809	Иванов	Москва	45
1896	Галкин	Иваново	40

Отношение А			
ID_NUM	NAME	CITY	AGE
1809	Иванов	Москва	45
1996	Петров	Нижний Новгород	39
1777	Сидоров	Рязань	21

Объединением двух совместимых по типу отношений **A** и **B** ($A \cup B$) называется отношение с тем же заголовком, как в отношениях **A** и **B**, и с телом, состоящим из множества кортежей **t**, принадлежащих **A** или **B** или обоим отношениям.

ID_NUM	NAME	CITY	AGE
1809	Иванов	Москва	45
1996	Петров	Нижний Новгород	39
1777	Сидоров	Рязань	21
1896	Галкин	Иваново	40

При выполнении операции объединения двух отношений создается отношение, включающее кортежи, входящие хотя бы в одно из отношений-операндов. Обратите внимание, что повторяющиеся кортежи удаляются по определению отношения.

Пересечением двух совместимых по типу отношений **A** и **B** ($A \cap B$) называется отношение с тем же заголовком, как в отношениях **A** и **B**, и с телом, состоящим из множества кортежей **t**, принадлежащих одновременно обоим отношениям **A** и **B**.

$A \cap B$

ID NU	NAME	CITY	AGE
1809	Иванов	Москва	45

Операция пересечения двух отношений создает отношение, включающее все кортежи, входящие в оба отношения-операнда.

Разностью двух совместимых по типу отношений **A** и **B** ($A - B$) называется отношение с тем же заголовком, как в отношениях **A** и **B**, и с телом, состоящим из множества кортежей **t**, принадлежащих отношению **A** и не принадлежащих отношению **B**.

$A - B$

ID NU	NAME	CITY	AGE
1996	Петров	Нижний	39
1777	Сидоров	Рязань	21

Отношение, являющееся разностью двух отношений, включает все кортежи, входящие в первое отношение, такие, что ни один из них не входит во второе отношение.

Декартово произведение двух отношений A и B ($A \times B$), где A и B не имеют общих имен атрибутов, определяется как отношение с заголовком, представляющим собой сцепление двух заголовков исходных отношений A и B , и телом, состоящим из множества кортежей t таких что первым, является любой кортеж отношения A , а вторым – любой кортеж, принадлежащий отношению B . Кардинальное число результирующего отношения равно произведению кардинальных чисел исходных отношений, а степень равняется сумме степеней.

Пусть отношение A – содержит имена всех текущих поставщиков, а отношение B – номера всех текущих деталей. Тогда $A \times B$ – это все текущие пары поставщик – деталь и деталь – поставщик.

Отношение A	Отношение B	
S1	P1	
S2	P2	
S3	P3	
	P4	

$A \times B$

S1	P1	S2	P1	S3	P1
S1	P2	S2	P2	S3	P2
S1	P3	S2	P3	S3	P3
S1	P4	S2	P4	S3	P4

На практике явное использование операции декартово произведение требуется только для очень сложных запросов. Эта операция включена в реляционную алгебру по концептуальным соображениям: (декартово произведение требуется как промежуточный шаг при определении операции θ - соединения, которая используется довольно часто).

Выборка – это сокращенное название θ - выборки, где θ означает любой скалярный оператор сравнения ($=, \neq, \leq, \geq$).

θ - **выборкой**, из отношения A по атрибутам X и Y (A where X и Y) называется отношение, имеющее тот же заголовок, что и отношение A , и тело, содержащее множества кортежей t отношения A , для которых проверка условия X и Y дает значение истина. Атрибуты X и Y должны быть определены на одном и том же домене, а оператор должен иметь смысл для этого домена.

Операция выборка (или операция ограничение отношения) - создает новое отношение, содержащее только те строки отношения – операнда, которые удовлетворяют некоторому условию ограничения.

Пример операций выборки.

Отношение A

ID_NUM	NAME	CITY	AGE
1809	Иванов	Москва	45
1996	Петров	Нижний	39
1777	Сидоров	Рязань	21
1896	Галкин	Москва	30

A where CITY = 'Москва'

ID_NUM	NAME	CITY	AGE
1809	Иванов	Москва	45
1896	Галкин	Москва	30

A where CITY = 'Москва' and AGE < 40

ID_NUM	NAME	CITY	AGE
1896	Галкин	Москва	30

Проекцией отношения A по атрибутам X, Y, ..., Z (A[X, Y, ..., Z]), где каждый из атрибутов принадлежит отношению A, называется отношение с заголовком {X, Y, ..., Z} и с телом, содержащим множество всех кортежей вида <X:x, Y:y, ..., Z:z> таких, что в отношении A имеется кортеж, атрибут X которого имеет значение x, атрибут Y имеет значение y, ..., атрибут Z имеет значение z.

Тем самым, при выполнении операции проекции получается «вертикальное» подмножество данного отношения, то есть подмножество, получаемое исключением всех атрибутов, отношения-операнда с естественным уничтожением потенциально возникающих кортежей-дубликатов.

Пример операций проекции.

A [NAME, CITY]

NAME	CITY
Иванов	Москва
Петров	Нижний
Сидоров	Рязань
Галкин	Москва

A [CITY]

CITY
Москва
Нижний
Рязань

Соединение отношений - создает новое отношение, каждый кортеж которого является результатом сцепления кортежей операндов (исходных отношений). Соединение имеет две разновидности: *естественное соединение* и *соединение по условию* (θ - соединение).

Пусть $X=\{X_1, X_2, \dots, X_m\}$, $Y=\{Y_1, Y_2, \dots, Y_n\}$, $Z=\{Z_1, Z_2, \dots, Z_k\}$.

Естественным соединением отношений A(X,Y) и B(Y,Z) (A JOIN B) называется отношение с заголовком {X, Y, Z} и с телом, содержащим множество всех кортежей вида <X:x, Y:y, Z:z> таких, для которых в отношении

А значение атрибута X равно x , а значение атрибута Y равно y , и в отношении B значение атрибута Y равно y , а атрибута Z равно z . При естественном соединении производится сцепление строк операндов соединения по общим атрибутам.

Замечание 1. Соединения не всегда выполняются по внешнему ключу и соответствующему потенциальному ключу, хотя такие соединения очень распространены и являются важным частным случаем.

Замечание 2. Если отношения A и B не имеют общих атрибутов, то выражение $A \text{ JOIN } B$ эквивалентно $A \times B$.

Отношение A (поставщики)

ID_NUM	NAME	CITY	STATUS
1809	Иванов	Москва	20
1996	Петров	Нижний Новгород	15
1777	Сидоро	Рязань	10

Отношение B (детали)

IP_NUM	NAIMEN	CITY	WEIGHT
P123	Болт	Москва	12
P896	Гайка	Нижний Новгород	14
P432	Шарнир	Москва	15

$A \text{ JOIN } B$

ID_NUM	NAME	STATUS	CITY	IP_NUM	NAIMEN	CITY	WEIGHT
1809	Иванов	20	Москва	P123	Болт	Москва	12
1809	Иванов	20	Москва	P432	Шарнир	Москва	15
1996	Петров	15	Нижний Новгород	P896	Гайка	Нижний Новгород	14

Тета – соединение. Пусть отношения A и B не имеют общих имен атрибутов и θ определяется так же, как в операции выборки.

θ -соединением отношения A по атрибуту X с отношением B по атрибуту Y называется результат вычисления выражения $(A \times B) \text{ WHERE } X \theta Y$.

θ - соединение – это отношение с тем же заголовком, что и при декартовом произведении отношений A и B , и с телом, содержащим множество кортежей $t \in A \times B$ таких что вычисление условия X и Y дает значение истина для данного кортежа. Атрибуты X и Y должны быть определены на одном и том же домене, а оператор должен иметь смысл для этого домена.

Таким образом, операция θ -соединение эквивалентна двум операциям: нахождению расширенного декартова произведения двух отношений (при необходимости с переименованием соответствующих атрибутов) и последующему выполнению указанной выборки из полученного результата. Если условие выполнено, полученная строка включается в отношение – результат.

Пример операции θ -соединения.

Отношение А (поставщики)

ID_NUM	NAME	CITY	STATUS
1809	Иванов	Москва	20
1996	Петров	Нижний	15
1777	Сидоров	Рязань	10

Отношение В (поставки)

ID_NUM	IP_NUM	QTY
1809	P123	100
1809	P896	200
1777	P432	150
1996	P432	150
1996	P123	250

(A×B (RENAME ID_NUM AS AID_NUM) WHERE QTY <200

ID_NUM	NAME	CITY	STATUS	AID_NUM	IP_NUM	QTY
1809	Иванов	Москва	20	1809	P123	100
1996	Петров	Нижний	15	1996	P432	150
1777	Сидоров	Рязань	10	1777	P432	150

Операция деления

У операции реляционного деления два операнда - бинарное и унарное отношения. Пусть $X=\{X_1, X_2, \dots, X_m\}$, $Y=\{Y_1, Y_2, \dots, Y_n\}$.

Делением отношений $A(X,Y)$ на $B(Y)$ (A/B) называется отношение с заголовком $\{X\}$ и телом, содержащим множество всех кортежей $\{X:x\}$, таких что существует кортеж $\{X:x, Y:y\}$, который принадлежит отношению A для всех кортежей $\{Y:y\}$, принадлежащих отношению B .

Деление отношений создает новое отношение, содержащее атрибуты первого отношения, отсутствующие во втором отношении и кортежи первого отношения, которые совпали кортежами второго. Для выполнения этой операции второе отношения должно содержать лишь атрибуты, совпадающие с атрибутами первого.

Замечание: Операция деления полезна тогда, когда запрос содержит слово «все».

Отношение A	Отношение B	Отношение B1	Отношение B2																											
<table><tr><th>S#</th><th>P#</th></tr><tr><td>S1</td><td>P1</td></tr><tr><td>S1</td><td>P2</td></tr><tr><td>S1</td><td>P3</td></tr><tr><td>S1</td><td>P4</td></tr><tr><td>S2</td><td>P1</td></tr><tr><td>S2</td><td>P3</td></tr><tr><td>S3</td><td>P2</td></tr><tr><td>S3</td><td>P3</td></tr></table>	S#	P#	S1	P1	S1	P2	S1	P3	S1	P4	S2	P1	S2	P3	S3	P2	S3	P3	<table><tr><th>P#</th></tr><tr><td>P1</td></tr></table>	P#	P1	<table><tr><th>P#</th></tr><tr><td>P2</td></tr><tr><td>P3</td></tr></table>	P#	P2	P3	<table><tr><th>P#</th></tr><tr><td>P1</td></tr><tr><td>P2</td></tr><tr><td>P3</td></tr></table>	P#	P1	P2	P3
S#	P#																													
S1	P1																													
S1	P2																													
S1	P3																													
S1	P4																													
S2	P1																													
S2	P3																													
S3	P2																													
S3	P3																													
P#																														
P1																														
P#																														
P2																														
P3																														
P#																														
P1																														
P2																														
P3																														
	<div>A/B</div> <table><tr><td>S1</td></tr><tr><td>S2</td></tr></table>	S1	S2	<div>A/B1</div> <table><tr><td>S1</td></tr><tr><td>S3</td></tr></table>	S1	S3	<div>A/B2</div> <table><tr><td>S1</td></tr></table>	S1																						
S1																														
S2																														
S1																														
S3																														
S1																														

Кроме рассмотренных выше операций в состав алгебры включаются:

- операция присваивания, позволяющая сохранить в базе данных результаты вычисления алгебраических выражений (**A:=B**),
- операция переименования атрибутов, дающая возможность корректно сформировать заголовок (схему) результирующего отношения (**A RENAME X AS Y**),
- операция расширения, позволяющая создавать новое отношение, дополненное атрибутом, значения которого получены посредством некоторых скалярных вычислений (**EXTEND <имя отношения> AS <скалярное выражение>**),
- операция подведения итогов, дающая возможность разбивать множество кортежей отношения на группы в соответствии с содержимым одного или нескольких атрибутов, и внутри каждой группы применять определенный оператор агрегирования (аналог операции **GROUP BY** в языке SQL).

Операторы агрегирования предназначены для подведения итогов в определенном столбце таблицы – отношения, например, для нахождения суммарных, средних, минимальных и максимальных значений (**SUM, AVG, MIN, MAX, COUNT**).

SUMMARIZE <реляционное выражение> BY (атрибуты) ADD <функция агрегирования> AS <новое имя атрибута>

Пример: **SUMMARIZE SP BY (P#) ADD SUM(QTY) AS TOTAL_QTY**

Отношение SP

S#	P#	QTY
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S2	P1	300
S2	P3	200
S3	P2	100
S3	P3	200
S3	P4	100

Результат

P#	TOTAL_QTY
P1	600
P2	300
P3	800
P4	300

Примеры использования реляционных операторов.

Пример №1.

Ежемесячно из цехов поступают отчеты о выпуске продукции за прошедший месяц, содержащие номер цеха, код продукции, дату выпуска и количество выпущенной продукции в таблице Новая продукция.

Новая продукция (номер цеха, код продукции, дата выпуска, количество)

На заводе имеется сводная таблица, содержащая сведения о выпускаемой продукции в целом, с такой же структурой.

Выпуск продукции (номер цеха, код продукции, дата выпуска, количество)

Так как атрибуты отношений совпадают, то для обновления сведений об ассортименте и количестве выпускаемой продукции необходимо отношение Новая продукция объединить с исходным отношением Выпуск продукции.

Новая продукция \cup Выпуск продукции

Пример №2.

Имеется набор экзаменационных ведомостей - отношений с совпадающими атрибутами: Ведомость_i (группа_i, номер зачетной книжки_i, фамилия студента_i, дата_i, дисциплина_i, оценка_i)

Подготовить список студентов, получивших только отличные оценки, с атрибутами Номер зачетной книжки и Фамилия студента.

Для экзаменационных ведомостей нужной группы:

1. Выполняем ограничение исходных отношений, отбирая из каждого в новое отношение кортежи, удовлетворяющие условию оценка_i = 'отлично'. Получили списки отличников группы по дисциплинам.

Отличники_i := Ведомость_i WHERE оценка_i = 'отлично'

2. Выполняем проекцию полученных отношений, отбирая из каждого только атрибуты номер зачетной книжки и фамилия студента. Получили новые списки отличников, в которых остались только номера зачетных книжки фамилии студентов.

Отличники_i [номер зачетной книжки, фамилия студента]

3. Пересечение последних даст нам искомое отношение - Список отличников, содержащее номера зачетных книжки фамилии общие для всех списков отличников

Пример №3.

Используя ежемесячные отчеты цехов о выпуске продукции (смотри пример №1), подготовить сведения о выпуске новых видов продукции за последний квартал.

Для решения этой задачи выполняем:

1. Из отношения **Выпуск продукции** делаем выборку **дата выпуска больше последней даты прошлого квартала**. Результат ограничения помещаем во временную **таблицу_1**:

Таблица_1:=Выпуск продукции WHERE дата выпуска>'30/06/2014'

2. Затем над той же исходной таблицей выполняем ограничение **дата выпуска не больше последней даты прошлого квартала** и заносим результат во временную **таблицу_2**:

Таблица_2:=Выпуск продукции WHERE дата выпуска<='30/09/2014'

3. Разность отношений 1 и 2 даст искомые сведения.

Таблица_1 – Таблица_2

Замечание: Один и тот же запрос можно сформулировать несколькими способами. В приведенном ниже запросе используется оператор выборки по дате:

Выпуск продукции WHERE дата выпуска > '30/06/2014'

AND дата выпуска <= '30/09/2014'

Пример №4.

Список студентов факультета содержит для каждого студента **ФИО**, **Дату рождения**, **Шифр группы** и **Признак наличия стипендии** (да, нет).

Список (фамилия студента, группа, дата рождения, стипендия)

Необходимо сформировать список студентов заданной группы, получающих стипендию.

Для этого:

1. Создаем вспомогательное отношение с атрибутами **Шифр группы** и **Признак наличия стипендии**.

Стипендия (группа, стипендия)

2. Заполняем один кортеж этого отношения, поместив в него шифр заданной группы и отметку о получении стипендии (да).

группа стипендия	
	8109 да

3. Деление исходного списка на вспомогательное отношение создаст искомый список с атрибутами: **ФИО** и **Дата рождения**.

Вот другая формулировка того же запроса:

Сделаем θ - выборку из отношения **Список**. Найдем все кортежи, которые удовлетворяют двум условиям: шифр группы совпадает с заданным и **признак наличия стипендии** = 'да'.

Возьмем проекцию полученного отношения по атрибутам **ФИО** и **Дата рождения**. **Список WHERE группа = 8109 AND стипендия = 'да' [ФИО, дата рождения]**

Пример №5.

Информация о поставщиках, деталях и поставках содержится в трех отношениях. Отношение **Поставщики (S)** содержит номер поставщика, его имя, город и статус **S (S#, SNAME, CITY, STATUS)**

Отношение **Детали (P)** содержит информацию о коде детали, наименовании, весе, цвете и месте хранения.

P (P#, PNAME, WEIGHT, COLOR, CITY)

Отношение **Поставка (SP)** содержит сведения о номере поставщика, коде детали и количестве.

SP (S#, P#, QTY)

Необходимо получить имена поставщиков, которые не поставляют деталь с кодом **P2**.

Рассмотрим пошаговое решение этого запроса:

1. Возьмем проекцию отношения **S** по атрибуту **S#**. Получим отношение **T1** с одним атрибутом:

T1 := S[S#]

2. Из отношения **SP** выберем кортежи, в которых код детали равен **P2**. Получим отношение **T2** с той же самой структурой, что и исходное отношение **SP**:

T2 := SP WHERE P# = 'P2'

3. Возьмем проекцию отношения **T2** по атрибуту **S#**. Получим отношение **T3** с одним атрибутом:

T3 := T2 [S#]

4. Разность отношений **T1** и **T3** даст номера тех поставщиков, которые не поставляют деталь с кодом **P2**.

T4 := T1 - T3

Операция естественного соединения по атрибуту **S#** позволяет сформировать отношение **T5** с такой же структурой, что и отношение **S**, но кортежи этого отношения будут содержать информацию лишь о тех

поставщиках, которые не поставляют деталь P2:

$T5 := T4 \text{ JOIN } S$

5. Выполним проекцию отношения T5 по атрибуту SNAME. Получим искомое отношение, содержащее имена поставщиков:

$T6 := T5 [SNAME]$

Выразим данный запрос в виде одной формулы.

$((S [S\#] - (SP \text{ WHERE } P\# = 'P2') [S\#]) \text{ JOIN } S) [SNAME]$