

2.7. Реляционное исчисление

2.7.1. Отличие механизмов реляционного исчисления и реляционной алгебры

РИ как механизм манипулирования реляционными БД базируется основном на классической математической логике. РИ оперирует с понятиями формул и замкнуто относительно понятия отношения. Это означает, что формулы РИ определяются над отношениями и результатом вычисления также являются отношения. В результате любая формула может интерпретироваться как отношение, что позволяет использовать ее в других формулах.

Механизмы РА и РИ эквивалентны. Следовательно, для любого допустимого выражения РА можно построить эквивалентную формулу РИ, производящую такой же результат, и наоборот.

Как и в случае РА при работе с РБД формулы РИ определяются для РТ, результатом вычисления также является РТ.

Однако механизмы выражений РА и формул РИ различаются уровнем процедурности. В отличие от выражения РА, которое задает правила выполнения и имеет процедурную интерпретацию, формулировка в терминах РИ носит описательный характер. Если сформулировать указанный выше запрос с использованием РИ, то можно получить формулу следующим содержанием:

Выдать Ст_ФИО и Ст_код для таких студентов, что существует группа с таким же значением Гр_стар и значением Гр_год, равным 2000.

В этой формулировке мы указали лишь характеристики результирующей таблицы, не определяя способ ее формирования. В этом случае система должна сама решить, какие операции нужно выполнить над исходными таблицами и в каком порядке.

Таким образом, формула в декларативной форме всего лишь описывает свойства желаемого результата. Она только ставит условия, которым должны удовлетворять записи таблицы-результата, не описывая саму процедуру ее получения. Для такой формулы однозначная интерпретация отсутствует. Поэтому формулы РИ считают непроцедурными или декларативными.

2.7.2. Основные понятия реляционного исчисления

Перейдем к рассмотрению основных вопросов РИ.

РИ является прикладной ветвью формального механизма исчисления предикатов первого порядка. Базисными понятиями РИ являются понятие переменной с определенной для нее областью допустимых значений и понятие правильно построенной формулы Wff (Well-formed formula, далее в данном разделе просто формулы), опирающейся на переменные, предикаты и кванторы. Формулы служат для выражения условий, накладываемых на

кортежные переменные.

В зависимости от области определения переменной различают два вида РИ:

- исчисление кортежей (ИсК), где областями определения переменных являются таблицы БД, т.е. допустимым значением каждой переменной является запись некоторой таблицы;
- исчисление доменов (ИсД), где областями определения переменных являются домены, на которых определены поля таблиц БД, т.е. допустимым значением каждой переменной является значение некоторого домена.

Далее по мере необходимости будем использовать необходимые синтаксические конструкции.

2.7.3. Исчисление кортежей

Для определения кортежной переменной используется оператор:

RANGE OF T IS X₁,..., X_n,

где T – определяемая кортежная переменная;

X_i – имя таблицы или выражение ИсК.

Например, для того, чтобы определить переменную Студент, областью определения которой является тСтуденты, нужно применить конструкцию:

RANGE OF Студент IS тСтуденты.

Из данного определения следует, что в любой момент времени переменная Студент представляет некоторую запись тСтуденты. При использовании кортежных переменных в формулах можно ссылаться на значение поля. Например, для того, чтобы сослаться на значение поля Ст_ФИО переменной Студент, нужно употребить конструкцию Студент.Ст_ФИО.

Основой формул является простое условие (condition или сокращенно <усл>), которое представляет простое скалярное сравнение (comparison) вида:

значение θ значение,

где θ – любой скалярный оператор сравнения (=, #, >, <...);

значение – скалярная величина (значение поля или заданная константа).

Например, конструкция "Студент.Ст_код = 100" является простым сравнением. По определению формула, заключенная в круглые скобки, также является простым сравнением. Следовательно, можно записать формулы:

Wff ::= <усл>

Wff ::= (Wff).

В дальнейшем при изложении синтаксиса и при возможности лишние скобки при написании формул будем опускать.

Более сложные варианты формул строятся с помощью логических связок **NOT**, **AND**, **OR** и **IF ... THEN**.

Например, нижеперечисленные выражения являются формулами:

NOT Wff

<усл> AND Wff

<усл> OR Wff

IF <усл> THEN Wff

Допускается построение формул с помощью кванторов. При этом для переменной можно составить два вида кванторов, которые различаются своим значением. Эти кванторы представляют собой формулы и имеют следующие конструкции:

1) квантор существования **EXISTS T (Wff)** – существует по крайней мере одно такое значение переменной T, что вычисление Wff дает значение "истина";

2) квантор общности **FORALL T (Wff)** – для всех значений переменной T вычисление Wff дает значение "истина".

Если R – таблица с записями T1,...,Tm, T – переменная записи, определенная на этой таблице, то Wff(T) – формула, в которой T используется как свободная переменная. Тогда можно записать равносильные формулы

EXISTS T (Wff(T)) = False OR (Wff(T1) OR ...OR (Wff));

FORALL T (Wff(T)) = True AND (Wff(T1) AND ...AND (Wff)).

Пусть Студ1 и Студ2 – две кортежные переменные, определенные для тСтуденты. Тогда формула

EXISTS Студ2 (Студ1.Ст_стип > Студ2.Ст_стип)

для текущей записи переменной Студ1 принимает значение истины только в том случае, если во всей тСтуденты найдется запись, связанная с переменной Студ2 и такая, что значение ее поля Ст_стип удовлетворяет внутреннему условию сравнения стипендий.

Формула

FORALL Студ2 (Студ1.Ст_стип > Студ2.Ст_стип)

для текущей записи переменной Студ1 принимает значение истина в том и только в том случае, если для всех записей тСтуденты, связанных с переменной Студ2, значения поля Ст_стип удовлетворяют условию сравнения.

Переменные, входящие в формулу, могут быть свободными или связанными. Все переменные, входящие в формулу, при построении которой не использовались кванторы, являются свободными. Фактически, это означает, что если для какого-то набора значений свободных кортежных переменных при вычислении формулы получено истинное значение, то эти значения кортежных переменных могут входить в результирующую таблицу.

Если имя переменной использовано при построении формулы с помощью кванторов, то в этой формуле и во всех формулах, построенных с ее участием, эта переменная считается связанной переменной. Это означает, что такая переменная не видна за пределами минимальной Wff, связавшей эту переменную. При вычислении значения такой Wff используется не одно значение связанной переменной, а вся ее область определения.

На самом деле, правильнее говорить не о свободных и связанных переменных, а о свободных и связанных вхождениях переменных. Легко видеть, что если переменная T является связанной в какую-либо Wff1, то во всех Wff, включающих данную, может использоваться имя переменной T, которая может быть свободной или связанной, но в любом случае не имеет никакого отношения к вхождению переменной T в Wff1.

• Например:

- **EXISTS Студ2 (Студ1.Ст_код = Студ2.Ст_код) AND
FORALL Студ2 (Студ1.Ст_стип > Студ2.Ст_стип).**

Здесь мы имеем два связанных вхождения переменной Студ2 с совершенно разным смыслом.

Итак, Wff обеспечивают средства формулировки условия выборки из таблиц БД. Различают открытые и закрытые Wff. Открытая Wff содержит хотя бы одну свободную переменную.

Выражением РИ кортежей называется конструкция вида:
<целевой список> WHERE Wff.

Значением выражения является отношение, тело которого определяется wff, а набор атрибутов и их имена – целевым списком. Этот список строится из целевых элементов списка, каждый из которых является выражением вида

T.A [AS X],

где T – переменная кортежа;

A – имя атрибута сопоставляемого отношения, на котором определена переменная T;

X – новое имя соответствующего атрибута результирующего отношения.

Целевой элемент может быть именем простой переменной T, что эквивалентно наличию подписка T.A1, T.A2, ..., T.An, где Ai – атрибуты отношения, сопоставляемого с T.

Выражение РИ для запроса: нужно узнать коды и ФИО студентов, являющихся старостами групп, набранных в 2000 году.

**(тСтуденты.Ст_код, тСтуденты.Ст_ФИО)
WHERE EXISTS тГруппы
(тГруппы.Гр_стар = тСтуденты.Ст_код AND**

тГруппы.Гр_год = 2000).

Рассмотрим процесс вычисления РИ для РТ в общем виде.

Пусть переменные кортежей, определенные в указанном списке целевых элементов, будут T, U, \dots, V . Пусть таблицы, на которых изменяются переменные кортежа, будут tT, tU, \dots, tV соответственно. Пусть результирующие поля имеют имена P_1, P_2, \dots, P_r . Тогда вычисление выражения РИ пройдет в три этапа:

- 1) строится расширенное декартово произведение $tT \text{ TIMES } tU \dots \text{ TIMES } tV$;
- 2) записи, которые не удовлетворяют формуле в фразе WHERE, исключаются из результата п.1;
- 3) результат п.2 проецируется по полям P_1, P_2, \dots, P_r .

2.7.4. Исчисление доменов

В ИсД областью определения переменных являются не отношения, а домены. Применительно к БД с тГруппы и тСтуденты можно говорить, например, о доменных переменных ФИО (значения – допустимые сочетания фамилий, имен и отчеств) или Ст_код (значения – допустимые коды студентов).

Основным формальным отличием ИсД от ИсК является наличие дополнительного набора предикатов, позволяющих выражать так называемые условия принадлежности.

Если T – это n -арное отношение с атрибутами A_1, A_2, \dots, A_n , то условие принадлежности имеет вид

$$R(A_1:V_1, A_2:V_2, \dots, A_n:V_m) \ (m \leq n),$$

где V_i – это либо литерально задаваемая константа, либо переменная домена, либо имя кортежной переменной.

Условие принадлежности принимает значение истины только в том случае, если в отношении T существует кортеж, содержащий указанные значения указанных атрибутов. Если V_i – константа, то на атрибут A_i задается жесткое условие, не зависящее от текущих значений доменных переменных; если же V_i – имя доменной переменной, то условие принадлежности может принимать разные значения при разных значениях этой переменной.

Во всех остальных отношениях формулы и выражения ИсД выглядят похожими на формулы и выражения ИсК. В частности, различаются свободные и связанные вхождения доменных переменных.

Для примера сформулируем с использованием ИсД запрос. Для простоты будем считать, что определены доменные переменные, имена которых совпадают с именами полей тСтуденты, а в случае, когда требуется несколько доменных переменных, определенных на одном домене, мы будем добавлять в конце имени цифры или буквы.

Запрос "Выдать коды и ФИО студентов, не получающих минимальную

стипендию" будет иметь вид:

```
Ст_код, Ст_ФИО WHERE EXISTS  
Ст_стипМ (тСтуденты (Ст_стипМ) AND  
тСтуденты (Ст_код, Ст_ФИО, Ст_стип) AND  
Ст_стип > Ст_стипМ),  
где Ст_стипМ – минимальная стипендия.
```

Реляционное ИсД является основой большинства языков запросов, основанных на использовании форм. В частности, на ИсД базируется язык Query-by-Example, который был первым и наиболее удобным языком в семействе языков, основанных на табличных формах.