

# Реляционное исчисление

Реляционное исчисление основано на разделе математической логики, который называется исчислением предикатов. Реляционное исчисление существует в двух формах: исчисление кортежей и исчисление доменов. Основное различие между ними состоит в том, что переменные исчисления кортежей являются переменными кортежей (они изменяются на отношении, а их значения являются кортежами), в то время как переменные исчисления доменов являются переменными доменов (они изменяются на доменах, а их значения являются скалярами).

## Исчисление кортежей

Реляционное исчисление является альтернативой реляционной алгебре. Внешне два подхода очень отличаются – исчисление описательное, а алгебра предписывающая, но на более низком уровне они представляют собой одно и то же, поскольку любые выражения исчисления могут быть преобразованы в семантически эквивалентные выражения в алгебре и наоборот.

Исчисление существует в двух формах: исчисление кортежей и исчисление доменов. Основное различие между ними состоит в том, что переменные исчисления кортежей являются переменными кортежей (они изменяются на отношении, а их значения являются кортежами), в то время как переменные исчисления доменов являются переменными доменов (они изменяются на доменах, а их значения являются скалярами; в этом смысле, действительно, "переменная домена" - не очень точный термин).

Выражение исчисления кортежей содержит заключенный в скобки список целевых элементов и выражение WHERE, содержащее формулу WFF ("правильно построенную формулу"). Такая формула WFF составляется из кванторов (EXISTS и FORALL), свободных и связанных переменных, литералов, операторов сравнения, логических (булевых) операторов и скобок. Каждая свободная переменная, которая встречается в формуле WFF, должна быть также перечислена в списке целевых элементов.

Упрощенный синтаксис выражений исчисления кортежей в форме БНФ имеет вид:

```
объявление-кортежной-переменной ::= RANGE OF переменная IS список-областей
область ::= отношение | реляционное-выражение
реляционное-выражение ::= (список-целевых-элементов)[WHERE wff]
целевой-элемент ::= переменная | переменная.атрибут [AS атрибут]
wff ::= условие | NOT wff | условие AND wff | условие OR wff | IF условие THEN wff |
      EXISTS переменная (wff) | FORALL переменная (wff) | (wff)
условие ::= (wff) | компаранд операция-отношения компаранд
```

По приведенной грамматике можно сделать следующие замечания.

- Квадратные скобки здесь указывают на компоненты, которые по умолчанию могут быть опущены.
- Категории отношение, атрибут и переменная – это идентификаторы (т. е. имена).
- Реляционное выражение содержит заключенный в скобки список целевых элементов и выражение WHERE, содержащее формулу wff («правильно построенную формулу»). Такая формула wff составляется из кванторов (EXISTS и FORALL), свободных и связанных переменных, констант, операторов сравнения, логических (булевых)

операторов и скобок. Каждая свободная переменная, которая встречается в формуле wff, должна быть также перечислена в списке целевых элементов.

- Категория условие представляет или формулу wff, заключенную в скобки, или простое скалярное сравнение, где каждый компаранд оператора сравнения – это либо скалярная константа, либо значение атрибута в форме переменная.атрибут.

Пусть кортежная переменная T определяются следующим образом:

RANGE OF T IS R1, R2, ..., Rn

Тогда отношения R1, R2, ..., Rn должны быть совместимы по типу т. е. они должны иметь идентичные заголовки, и кортежная переменная T изменяется на объединении этих отношений, т. е. её значение в любое заданное время будет некоторым текущим кортежем, по крайней мере одного из этих отношений.

Примеры объявлений кортежных переменных.

RANGE OF SX IS S RANGE OF SPX IS SP RANGE OF SY IS

(SX) WHERE SX.City = 'Смоленск',

(SX) WHERE EXISTS SPX (SPX.Sno = SX.Sno AND SPX.Pno = 1)

Здесь переменная кортежа SY может принимать значения из множества кортежей S для поставщиков, которые или размещены в Смоленске, или поставляют деталь под номером 1, или и то и другое.

Для сравнения с реляционной алгеброй рассмотрим некоторые запросы на языке исчисления кортежей, которые соответствуют рассмотренным ранее.

Для сравнения с реляционной алгеброй некоторые примеры соответствуют рассмотренным ранее. Любые примеры можно расширить, включив в них завершающий шаг присвоения, присвоив значение выражения некоторому именованному отношению; этот шаг для краткости опускается.

Получить номера поставщиков из Смоленска со статусом больше 20	( SX.Sno ) WHERE SX.City = 'Смоленск' AND SX.Status > 20
Получить все такие пары номеров поставщиков, что два поставщика размещаются в одном городе	<p>( SX.Sno AS FirstSno, SY.Sno AS SecondSno ) WHERE SX.City = SY.City AND SX.Sno &lt; SY.Sno</p> <p><u>Замечание.</u> Спецификации «AS FirstSno» и «AS SecondSno» дают имена атрибутам результата; следовательно, такие имена недоступны для использования во фразе WHERE и потому второе сравнение во фразе WHERE «SX.Sno &lt; SY.Sno», а не «FirstSno &lt; SecondSno».</p>
Получить имена поставщиков, которые поставляют деталь с номером 2	SX.Sname WHERE EXISTS SPX ( SPX.Sno = SX.Sno AND SPX.Pno = 2 )

Получить имена поставщиков, которые поставляют по крайней мере одну красную деталь	<p>SX.Sname WHERE EXISTS SPX (SX.Sno = SPX.Sno AND EXISTS PX ( PX.Pno = SPX.Pno AND PX.Color = 'Красный' ) )</p> <p>Или эквивалентная формула (но в предваренной нормальной форме, в которой все кванторы записываются в начале формулы WFF):</p> <p>SX.Sname WHERE EXISTS SPX (EXISTS PX ( SX.Sno = SPX.Sno AND SPX. Pno = PX.Pno AND PX.Color = 'Красный' ) )</p>
Получить имена поставщиков, которые поставляют по крайней мере одну деталь, поставляемую поставщиком под номером 2	<p>SX.Sname WHERE EXISTS SPX ( EXISTS SPY (SX.Sno = SPX.Sno AND SPX.Pno = SPX.Pno AND SPY.sno = 2 ) )</p>
Получить имена поставщиков, которые поставляют все детали	<p>SX.SNAME WHERE FORALL PX ( EXISTS SPX ( SPX.Sno = SX.Sno AND SPX.Pno = PX.Pno ) )</p> <p>Или равносильное выражение без использования квантора FORALL:</p> <p>SX. SNAME WHERE NOT EXISTS PX ( NOT EXISTS SPX ( SPX.Sno = SX.Sno AND SPX.Pno = PX.Pno ) )</p> <p>Отметим, что содержание этого запроса – «имена поставщиков, которые поставляют все детали» - является точным на 100% (в отличие от случая с алгебраическим аналогом при использовании оператора DIVIDEBY).</p>
Получить имена поставщиков, которые не поставляют деталь под номером 2	<p>SX.Sname WHERE NOT EXISTS SPX ( SPX.Sno = SX.Sno AND SPX.Pno = 2 )</p>
Получить номера поставщиков, которые поставляют по крайней мере все детали, поставляемые поставщиком с номером 2	<p>SX.Sno WHERE FORALL SPY ( SPY.Sno &lt;&gt; 2 OR EXISTS SPZ ( SPZ.Sno = SX.Sno AND SPZ.Pno = SPY.Pno ) )</p>

Предваренная нормальная форма не является более или менее правильной по сравнению с другими формами, но немного попрактиковавшись можно убедиться, что в большинстве случаев это наиболее естественная формулировка запросов. Кроме того, эта форма позволяет уменьшить количество скобок, как показано ниже.

Формулу WFF ::= квантор\_1 переменная\_1 ( квантор\_2 переменная\_2 ( wff )), где каждый из кванторов квантор\_1 и квантор\_2 представляет или квантор EXISTS, или квантор FORALL, переменная\_1 и переменная\_2 - имена переменных) по умолчанию можно однозначно сократить к виду: квантор\_1 переменная\_1 квантор\_2 переменная\_2 ( wff )

Таким образом, приводимое в четвертом примере выражение исчисления можно переписать (при желании) следующим образом:

```
SX.Sname WHERE EXISTS SPX EXISTS PX ( SX.Sno = SPX.Sno AND SPX.Pno = PX.Pno  
AND PX.Color = 'Красный' )
```

### Вычислительные возможности исчисления кортежей

Добавить вычислительные возможности в исчисление довольно просто: необходимо расширить определение компарандов и целевых элементов так, чтобы они включали новую категорию - скалярные выражения, в которых операнды, в свою очередь, могут включать литералы, ссылки на атрибуты и (или) ссылки на итоговые функции. Для целевых элементов также требуется использовать спецификацию вида "AS attribute" для того, чтобы дать подходящее имя результирующему атрибуту, если нет очевидного наследуемого имени.

Т. к. что смысл скалярного выражения легко воспринимается, подробности опускаются. Однако синтаксис для ссылок на итоговые функции будет показан:

aggregate\_function ( expression [ , attribute ] ), где

- aggregate\_function - это COUNT, SUM, AVG, MAX или MIN (возможны, конечно, и некоторые другие функции),
- expression - это выражение исчисления кортежей (вычисляющее отношение),
- attribute - это такой атрибут результирующего отношения, по которому подсчитывается итог.

Для функции COUNT аргумент attribute не нужен и опускается; для других итоговых функций его можно опустить по умолчанию, если и только если вычисление аргумента expression дает отношение степени один и в таком случае единственный атрибут результата вычисления выражения expression подразумевается по умолчанию. Обратите внимание, что ссылка на итоговую функцию возвращает скалярное значение и поэтому допустима в качестве операнда скалярного выражения.

Из сказанного можно сделать следующие выводы:

- Итоговая функция действует в некоторых отношениях как новый тип квантора. В частности, если аргумент expression в данной ссылке на итоговую функцию представляется в виде "( tic ) WHERE f", где tic- целевой элемент списка (target\_item\_commlist), а f - это формула WFF, и если экземпляр переменной кортежа T свободен в формуле f, то такой экземпляр переменной T связан в ссылке на итоговую функцию aggregate\_function ( ( tic ) WHERE f [ , attribute ] )
- Пользователи, владеющие языком SQL, могут заметить, что с помощью двух следующих аргументов в ссылке на итоговую функцию, expression и attribute, можно избежать необходимых для SQL специальных приемов использования оператора DISTINCT для исключения дублирующих кортежей, если это требуется, перед выполнением итоговой операции. Вычисление аргумента expression дает отношение, из которого повторяющиеся кортежи всегда исключаются по определению. Аргумент attribute обозначает атрибут такого отношения, по которому выполняются итоговые вычисления, а дублирующие значения перед подсчетом итога из такого атрибута не удаляются. Конечно, атрибут может не содержать никаких дублирующих значений в любом случае, в частности, если такой атрибут является первичным ключом.

Получить номера деталей и их вес всех типов деталей, вес которых превышает 10 ед.	( PX.Pno, PX.Weight AS GMWT ) WHERE PX.Weight > 10
Получить всех поставщиков, добавив для каждого литеральное значение "Поставщик"	( SX, 'Поставщик' AS TAG )
Получить каждую поставку с полными данными о входящих в нее деталях и общим весом поставки	( SPX.Sno, SPX.Qty, PX, PX.Weight * SPX.Qty AS SHIPWT ) WHERE PX.Pno = SPX.Pno
Для каждой детали получить ее номер и общее поставляемое количество	( PX.Pno, SUM ( SPX WHERE SPX.Pno = PX.Pno, Qty ) AS TOTQTY )
Получить общее количество поставляемых деталей	( SUM (SPX, Qty ) AS GRANDTOTAL ) )
Для каждого поставщика получить его номер и общее количество поставляемых им деталей	( SX.Sno, COUNT ( SPX WHERE SPX.Sno = SX.Sno ) AS NUMBER_OF_PARTS )
Получить города, в которых хранится а) более пяти красных деталей; б) не более пяти красных деталей	PX.City WHERE COUNT ( PY WHERE PY.City = PX.City AND PY.Color = 'Красный' ) > 5

## Исчисление доменов

Реляционное исчисление, ориентированное на домены (или исчисление доменов), отличается от исчисления кортежей тем, что в нем используются переменные доменов вместо переменных кортежей, т. е. переменные, принимающие свои значения в пределах домена, а не отношения. (Замечание. "Переменную домена" было бы лучше назвать "скалярной переменной", так как ее значения - это элементы домена, т.е. скаляры, а не сами домены.)

С практической точки зрения большинство очевидных различий между версиями исчисления для доменов и кортежей основано на том, что версия для доменов поддерживает дополнительную форму условия, которое мы будем называть условием принадлежности. В общем виде условие принадлежности можно записать так:

$R ( \text{pair}, \text{pair}, \dots ),$

где  $R$  - это имя отношения, а каждая пара  $\text{pair}$  имеет вид  $A : v$  (где  $A$  - атрибут отношения  $R$ , а  $v$  - или переменная домена, или литерал). Проверка условия дает значение истина, если и только если существует кортеж в отношении  $R$ , имеющий определенные значения для определенных атрибутов. Например, вычисление выражения

$SP ( \text{Sno} : 1, \text{Pno} : 1 )$  - дает значение истина, если и только если в отношении  $SP$  существует кортеж со значением  $\text{Sno}$ , равным 1, и значением  $\text{Pno}$ , равным 1. Аналогично, условие принадлежности

SP ( Sno : SX, Pno : PX ) - принимает значение истина, если и только если в отношении SP существует кортеж со значением Sno, эквивалентным текущему значению переменной домена SX (какому бы то ни было), и значением Pno, эквивалентным текущему значению переменной домена PX (опять же какому бы то ни было).

Далее будем подразумевать существование переменных доменов с именами: образуемыми добавлением букв X, Y, Z, ... к соответствующим именам доменов. Напомним, что в базе данных поставщиков и деталей каждый атрибут имеет такое же имя, как и соответствующий ему домен, за исключением атрибутов Sname и Pname, для которых соответствующий домен называется просто Name.

Множество всех номеров поставщиков.	( SX )
Множество всех номеров поставщиков отношения S.	( SX ) WHERE S ( Sno : SX )
Подмножество номеров поставщиков из города Смоленск.	( SX ) WHERE S ( Sno : SX, City : 'Смоленск' )
Получить номера и города поставщиков, поставляющих деталь под номером 2.  Обратите внимание, что для этого запроса, выраженного в терминах исчисления кортежей, требовался квантор существования; заметьте также, что это первый из приведенных здесь примеров, где действительно необходимы скобки для списка целевых элементов.	( SX, CityX ) WHERE S ( Sno : SX, City : CityX ) AND SP ( Sno : SX, Pno : 2 )
Получить такие пары 'номер поставщика' – 'номер детали', что поставщики и детали размещены в одном городе".	( SX, PX ) WHERE S (Sno : SX, City : CityX ) AND P ( Pno : PX, City : CityY ) AND CityX <> CityY

Для сравнения с реляционной алгеброй некоторые примеры соответствуют рассмотренным ранее.

Получить номера поставщиков из Смоленска со статусом, большим 20	SX WHERE EXISTS StatusX (StatusX > 20 AND S ( Sno : SX, Status : StatusX, City : 'Смоленск' ) )  Обратите внимание, что кванторы все еще требуются. Этот пример несколько неуклюжий по сравнению с его аналогом, выраженным в терминах исчисления кортежей. С другой стороны, конечно, есть случаи, когда верно обратное; смотрите, в частности, более сложные примеры, приведенные ниже.
Получить все такие пары номеров поставщиков, что два поставщика размещаются в одном городе	( SX AS FirstSno, SY AS SecondSno ) WHERE EXSIST CityZ ( S ( Sno : SX, City : CityZ ) AND S (Sno : SY, City : CityZ ) AND SX < SY )

Получить имена поставщиков, которые поставляют по крайней мере одну красную деталь	NAMEX WHERE EXISTS SX EXISTS PX ( S ( Sno : SX, Sname : NameX ) AND SP ( Sno : SX, Pno : PX ) AND P ( Pno : PX, Color : 'Красный' ) )
Получить имена поставщиков, которые поставляют все типы деталей	NameX WHERE EXISTS SX ( S ( Sno : SX, Sname : NameX ) AND FORALL PX ( IF P ( Pno : PX ) THEN SP (Sno : SX, Pno : PX ) ) )
Получить имена поставщиков, которые не поставляют деталь с номером 2	NameX WHERE EXISTS SX ( S ( Sno : SX, Sname : NameX ) AND NOT SP (Sno : SX, Pno : 2 ) )
Получить номера поставщиков, которые поставляют по крайней мере все типы деталей, поставляемых поставщиком с номером 2	SX WHERE FORALL PX ( IF SP ( Sno : 2, Pno : PX ) THEN SP (Sno : SX, Pno : PX ) )
Получить номера деталей, которые или весят более 16 фунтов, или поставляются поставщиком с номером 2, или и то, и другое	PX WHERE EXISTS WeightX ( P ( Pno : PX, Weight : WeightX ) AND WeightX > 16 ) OR SP ( Sno : 2, Pno : PX )

Исчисление доменов, как и исчисление кортежей, формально эквивалентно реляционной алгебре (т.е. оно реляционно полно). Для доказательства можно сослаться, например, на работы Ульмана (Ullman).