

Лекция 2

Модель данных SQL и
реляционная алгебра

Модель данных SQL

- SQL-ориентированная БД представляет собой набор таблиц, каждая из которых в любой момент времени содержит некоторое *мультимножество* строк, соответствующих заголовку таблицы.
- Отличия от РМД:
 - Мультимножества строк
 - Для таблицы поддерживается порядок столбцов, соответствующих порядку их определения
- Т.е. таблица из SQL \neq отношению из РМД, хотя они похожи
- Имеется две основных разновидности таблиц, хранимых в БД: традиционная и типизированная таблицы

Традиционная таблица

- Последовательность столбцов с указанными типами данных
- Поддерживаются следующие типы:
 - Точные числовые (int, long int, decimal, numeric)
 - Приближенные числовые (float, double)
 - Типы символьных строк (char, varchar)
 - Типы битовых строк
 - Типы даты и времени, временных интервалов
 - Булевский тип
 - Типы коллекций
 - Пользовательские типы
 - Ссылочные типы

NULL

- Неопределенное значение, которое разрешается использовать вместо любого значения типа данных
- Для любой бинарной операции ob:
 - $A \text{ ob } \text{NULL} = \text{NULL}$
 - $\text{NULL ob } A = \text{NULL}$
 - $\text{NULL ob NULL} = \text{NULL}$
- Для любой операции сравнения comp (<, >, ==, ...):
 - $A \text{ comp } \text{NULL} = \text{unknown}$
 - $\text{NULL comp } A = \text{unknown}$
 - $\text{NULL comp NULL} = \text{unknown}$

Типы коллекций

Есть 2 вида: тип массива и тип мультимножества

- Тип массива требует указания длины, есть операции доступа и изменения элемента по номеру
- Элементы типа коллекции могут быть любого типа данных, определенного к моменту определения данного типа коллекции
- Анонимный строчный тип – безымянный структурный тип, значения которого – строки из элементов ранее определенных типов
- При объявлении типа мультимножества можно явно запретить наличие в его значениях элементов-дубликатов

Пользовательские типы (UDT)

Есть 2 вида: индивидуальный и структурный тип

- Индивидуальный тип – именованный тип данных, основанный на единственном предопределенном типе
- Структурный тип – включающий один и более атрибутов любого из допустимых типов данных, в том числе другого структурного типа. Можно использовать механизм наследования от ранее определенного структурного типа

Типизированная таблица

- Типизированные таблицы можно определять с использованием механизма наследования
- Добавляется self-referencing столбец типизированных уникальных идентификаторов строк
- Подтаблица наследует у супертаблицы способ генерации значения ссылочного типа и все ограничения целостности

Манипулирование данными

- Структура оператора SELECT:

SELECT [DISTINCT] (1)

FROM (2)

[WHERE (3)]

[GROUP BY (4)]

[HAVING (5)]

[ORDER BY (6)]

Ограничения целостности

- Целостность сущности – каждый кортеж любого отношения должен отличаться от любого другого кортежа этого отношения (т.е. любое отношение должно обладать первичным ключом).
- Ссылочная целостность – для каждого значения внешнего ключа, появляющегося в дочернем отношении, в родительском отношении должен найтись кортеж с таким же значением первичного ключа.

Ограничения целостности

- Т.к. таблицы в SQL могут содержать мультимножества строк, то в модели SQL отсутствует обязательное предписание об ограничении целостности сущности
- Ссылочная целостность в модели данных SQL поддерживается в обязательном порядке, но в трех разных вариантах, лишь один из которых полностью соответствует реляционной модели

Ограничения целостности

Есть три спецификации (matching) внешнего ключа и первичного ключа в таблице, на которую идёт ссылка:

- FULL (полная) требует, чтобы для любого внешнего ключа существовала строка с точно таким же первичным ключом
- SIMPLE (простая) требует для внешнего ключа либо полностью соответствовать некоторому первичному ключу, либо (в случае составного ключа) во всех столбцах быть NULL
- PARTIAL (частичная) требует либо полной неопределённости во всех столбцах, либо полного соответствия первичному ключу, либо когда лишь часть столбцов NULL, то найдётся такая строка, что указанные столбцы внешнего ключа совпадают со столбцами первичного ключа

Реляционная алгебра

Алгебра A

Действующие понятия:

- A - имя атрибута; T - имя типа (домена)
- Атрибут - это упорядоченная пара вида $\langle A, T \rangle$
- Заголовок Hr – это множество атрибутов, т.е. пар $\langle A, T \rangle$
- Кортеж tr – это множество упорядоченных триплетов вида $\langle A, T, v \rangle$,
 v – конкретное значение.
- Тело Br – это множество кортежей tr
- Элемент заголовка - это атрибут
- Элемент тела – это кортеж
- Элемент кортежа – это упорядоченный триплет вида $\langle A, T, v \rangle$

Реляционное дополнение

Пусть s обозначает результат операции $\langle \text{NOT} \rangle r$. Тогда:

- $Hs = Hr$
- $Bs = \{ts : \exists tr(tr \notin Br \wedge ts = tr)\}$
- Пример. В тип данных ДОПУСТИМЫЕ_ПРОЕКТЫ, на котором определен атрибут ПРО_НОМ отношения ПРОЕКТЫ, входит пять значений {1,2,3,4,5}. Тогда:

ПРОЕКТЫ	
ПРО_НОМ	
1	
2	

<NOT> ПРОЕКТЫ	
ПРО_НОМ	
3	
4	
5	

Удаление атрибута

Пусть s обозначает результат операции $r \text{ <REMOVE> } A$. Тогда:

- $Hs = Hr \setminus \{ \langle A, T \rangle \}$
- $Bs = \{ ts : \exists tr \exists v ((tr \in Br) \wedge (\langle A, T, v \rangle \in tr) \wedge (ts = tr \setminus \{ \langle A, T, v \rangle \})) \}$

СОТРУДНИКИ

СОТР_НОМЕР	СОТР_ИМЯ	СОТР_ЗАРП	ПРО_НОМ
2934	Иванов	21400	1
2935	Петров	24400	2
2936	Сидоров	19200	1
2937	Федоров	21000	1

⋮

СОТРУДНИКИ <REMOVE> ПРО_НОМ

СОТР_НОМЕР	СОТР_ИМЯ	СОТР_ЗАРП
2934	Иванов	21400
2935	Петров	24400
2936	Сидоров	19200
2937	Федоров	21000

⋮

Переименование атрибутов

Пусть s обозначает результат операции r $\langle \text{RENAME} \rangle (A, B)$. Для выполнения операции в схеме отношения r должен присутствовать атрибут A и не должен присутствовать атрибут B . Тогда:

- $Hs = (Hr \setminus \{ \langle A, T \rangle \}) \cup \{ \langle B, T \rangle \}$
- $Bs = \{ ts : \exists tr \exists v ((tr \in Br) \wedge (\langle A, T, v \rangle \in tr) \wedge (ts = (tr \setminus \{ \langle A, T, v \rangle \}) \cup \{ \langle B, T, v \rangle \})) \}$

Реляционная конъюнкция

Пусть s обозначает результат операции $r1 \text{ <AND> } r2$. Тогда:

- $Hs = Hr1 \cup Hr2$
- $Bs = \{ts : \exists tr1 \exists tr2 ((tr1 \in Br1) \wedge (tr2 \in Br2) \wedge (ts = tr1 \cup tr2))\}$

Реляционная дизъюнкция

Пусть s обозначает результат операции $r1 <OR> r2$. Тогда:

- $Hs = Hr1 \cup Hr2$
- $Bs = \{ts : \exists tr1 \exists tr2 (((tr1 \in Br1) \vee (tr2 \in Br2)) \wedge (ts = tr1 \cup tr2))\}$

Реляционная дизъюнкция

Пример:

ПРОЕКТЫ_1

НАЗВАНИЕ	РУКОВОДИТЕЛЬ
ПРОЕКТ1	Иванов
ПРОЕКТ2	Сидоров

НОМЕРА_ПРОЕКТОВ

НОМЕР
1
2

ПРОЕКТЫ_1 <OR> НОМЕРА_ПРОЕКТОВ

НАЗВАНИЕ	РУКОВОДИТЕЛЬ	НОМЕР
ПРОЕКТ1	Иванов	1
ПРОЕКТ2	Иванов	1
ПРОЕКТ3	Иванов	1
ПРОЕКТ1	Сидоров	1
ПРОЕКТ2	Сидоров	1
ПРОЕКТ3	Сидоров	1
ПРОЕКТ1	Иванов	2
ПРОЕКТ2	Иванов	2
ПРОЕКТ3	Иванов	2
ПРОЕКТ1	Сидоров	2
ПРОЕКТ2	Сидоров	2
ПРОЕКТ3	Сидоров	2
ПРОЕКТ1	Иванов	3
ПРОЕКТ2	Сидоров	3

Ограничения: НАЗВАНИЕ принимает значения ПРОЕКТ1, ПРОЕКТ2, ПРОЕКТ3; РУКОВОДИТЕЛЬ принимает значения Иванов, Сидоров; НОМЕР принимает значения 1, 2, 3.

Полнота Алгебры А

Операции алгебры Кодда:

- объединение (UNION)
- пересечение (INTERSECT)
- вычитание (MINUS)
- взятие расширенного декартова произведения (TIMES)
- переименование атрибутов (RENAME)
- ограничение (WHERE)
- проекция (PROJECT)
- соединение (Θ -JOIN)
- деление (DIVIDE BY)
- присваивание

WHERE {сравнение с константой}

WHERE (СОТР_ЗАРП=11000)

СОТРУДНИКИ_1

СОТР_НОМЕР	СОТР_ИМЯ	СОТР_ЗАРП	РУК_НОМЕР
2934	Иванов	11400	2934
2935	Петров	14400	2934
2936	Сидоров	9200	2934
2937	Федоров	11000	2934
2938	Иванова	11000	2941
2939	Сидоренко	9200	2941
2940	Федоренко	11000	2941
2941	Иваненко	11000	2941

ЗАРП_11000

СОТР_ЗАРП
11000

СОТРУДНИКИ_1 <AND> ЗАРП_11000

СОТР_НОМЕР	СОТР_ИМЯ	СОТР_ЗАРП	РУК_НОМЕР
2937	Федоров	11000	2934
2938	Иванова	11000	2941
2940	Федоренко	11000	2941
2941	Иваненко	11000	2941

WHERE {сравнение атрибутов}

WHERE (СОТР_НОМЕР = РУК_НОМЕР)

РУК_НОМЕР
2934
2941

СОТР_НОМЕР
2934
2941

СОТРУДНИКИ_1 <AND> ((СОТРУДНИКИ_1 <REMOVE> СОТР_НОМЕР <REMOVE>
СОТР_ИМЯ <REMOVE> СОТР_ЗАРП) <RENAME> (РУК_НОМЕР, СОТР_НОМЕР))

СОТР_НОМЕР	СОТР_ИМЯ	СОТР_ЗАРП	РУК_НОМЕР
2934	Иванов	11400	2934
2941	Иваненко	11000	2941

MINUS и DIVIDE BY

$$r1 \text{ MINUS } r2 = r1 \text{ <AND> (<NOT> } r2)$$

Пусть имеются отношения $r1\{A,B\}$ и $r2\{B\}$. Тогда

$$r1 \text{ DIVIDE BY } r2 =$$

$$(r1 \text{ PROJECT } A) \text{ MINUS } (((r2 \text{ TIMES } (r1 \text{ PROJECT } A)) \text{ MINUS } r1) \text{ PROJECT } A)$$

DIVIDE BY

Разберём последовательно почему это так:

- Результатом выполнения операции $r1 \text{ PROJECT } A$ является унарное отношение со схемой $\{A\}$, кортежи которого содержат все значения атрибута A из тела отношения $r1$
- Результат выражения $r2 \text{ TIMES } (r1 \text{ PROJECT } A)$ – это отношение со схемой $\{A, B\}$, в тело которого входят все возможные комбинации значений B в теле отношения $r2$ и атрибута A в теле $r1$
- В теле результата выражения $(r2 \text{ TIMES } (r1 \text{ PROJECT } A)) \text{ MINUS } r1$ останутся только кортежи с таким значением атрибута A , которые не должны попасть в результат операции $r1 \text{ DIVIDE BY } r2$
- Проецируем результат предыдущего выражения на A и получаем множество тех A , которые не принимают всех значений B из второго операнда
- Выполнение завершающей операции MINUS дает желаемый результат
- Таким образом, приведённые выражения эквивалентны. Но нам уже известно, что операции PROJECT , MINUS , TIMES выражаются через операции Алгебры A , а значит и DIVIDE BY тоже выражается через операции Алгебры A
- $r1 \text{ DIVIDE BY } r2 =$
 $(r1 \text{ <REMOVE> } B) \text{ <AND> <NOT> } (((r2 \text{ <AND> } (r1 \text{ <REMOVE> } B)) \text{ <AND> <NOT> } r1) \text{ <REMOVE> } B)$

Избыточность Алгебры A

Стандартным полным базисом является $\{\neg, \wedge, \vee\}$ (отрицание, конъюнкция, дизъюнкция). Этот набор избыточен, поскольку верны тождества де Моргана:

- $A \wedge B = \neg(\neg A \vee \neg B)$
- $A \vee B = \neg(\neg A \wedge \neg B)$

В математической логике существуют полные базисы из одной функции:

- $sh(A, B) = \neg A \vee \neg B$ - «штрих Шеффера»
- $pi(A, B) = \neg A \wedge \neg B$ - «стрелка Пирса»

Полнота этих базисов доказывается выводом стандартного полного базиса $\{\neg, \wedge, \vee\}$:

- $sh(A, A) = \neg A$
- $sh(\neg A, \neg B) = A \vee B$
- $\neg sh(A, B) = A \wedge B$

Таким образом, базис из одного штриха Шеффера является полным

Реляционный аналог штриха Шеффера

Пусть s обозначает результат операции $\langle sh \rangle(r_1, r_2)$. Тогда:

- $Hs = Hr_1 \cup Hr_2$
- $Bs = \{ts : \exists tr_1 \exists tr_2 ((tr_1 \notin Br_1 \vee tr_2 \notin Br_2) \wedge (ts = tr_1 \cup tr_2))\}$

Избыточность <RENAME>

СОТРУДНИКИ

СОТР_НОМЕР	СОТР_ИМЯ	СОТР_ЗАРП	ПРО_НОМ
2934	Иванов	11400	1
2935	Петров	14400	1
2936	Сидоров	9200	1
2937	Федоров	11000	1
2938	Иванова	11000	2
2939	Сидоренко	9200	2
2940	Федоренко	11000	2
2941	Иваненко	11000	2

СОТРУДНИКИ <AND> ПРО_НОМ_НОМЕР_ПРОЕКТА

СОТР_НОМЕР	СОТР_ИМЯ	СОТР_ЗАРП	ПРО_НОМ	НОМЕР_ПРОЕКТА
2934	Иванов	11400	1	1
2935	Петров	14400	1	1
2936	Сидоров	9200	1	1
2937	Федоров	11000	1	1
2938	Иванова	11000	2	2
2939	Сидоренко	9200	2	2
2940	Федоренко	11000	2	2
2941	Иваненко	11000	2	2

ПРО_НОМ_НОМЕР_ПРОЕКТА

ПРО_НОМ	НОМЕР_ПРОЕКТА
1	1
2	2
3	3
4	4
5	5