

Лекция 2

Реляционная алгебра, операции реляционной алгебры

Цель

Изучить теоретические основы реляционной алгебры, освоить основные операции и их практическое применение для манипуляции данными в реляционных базах данных.

Основные вопросы

1. Понятие и назначение реляционной алгебры;
2. Базовые операции реляционной алгебры;
3. Дополнительные операции;
4. Примеры применения операций;
5. Связь реляционной алгебры с SQL;

Лекция

Введение в реляционную алгебру

Реляционная алгебра - это формальный язык запросов, основанный на теории множеств, который предоставляет набор операций для манипуляции отношениями. Разработана Эдгаром Коддом как теоретическая основа реляционных баз данных.

Основные характеристики:

- Закрытость относительно операций (результат операции - отношение);
- Процедурный характер (последовательность операций);
- Математическая строгость;
- Основа для оптимизации запросов;

Базовые операции реляционной алгебры

Кодд определил восемь базовых операций, которые можно разделить на две группы:

Операции теории множеств

Объединение (Union) - $R \cup S$

Объединяет два отношения с одинаковой схемой, удаляя дубликаты.

Условия выполнения:

- Отношения должны быть совместимы по объединению (одинаковые атрибуты);
- Атрибуты должны иметь одинаковые домены;

Пример:

Студенты_физики \cup Студенты_математики

Пересечение (Intersection) - $R \cap S$

Возвращает кортежи, присутствующие в обоих отношениях.

Пример:

Студенты_физики \cap Студенты_отличники

Разность (Difference) - $R - S$

Возвращает кортежи, присутствующие в R, но отсутствующие в S.

Пример:

Все_студенты - Студенты_сдавшие_экзамен

Декартово произведение (Cartesian Product) - $R \times S$

Объединяет каждый кортеж R с каждым кортежем S.

Результат:

- Схема: все атрибуты R и S;
- Мощность: $|R| \times |S|$ кортежей;

Специальные реляционные операции

Выбор (Selection) - $\sigma_{\theta}(R)$

Выбирает кортежи из отношения R, удовлетворяющие условию θ .

Условие θ может включать:

- Сравнения: $=, \neq, <, >, \leq, \geq$;
- Логические операции: AND, OR, NOT;

Пример:

$\sigma_{\{Зарплата > 50000\}}(Сотрудники)$

Проекция (Projection) - $\pi_{a_1, a_2, \dots, a_n}(R)$

Возвращает отношение с указанными атрибутами, удаляя дубликаты.

Пример:

$\pi_{\{Имя, Фамилия\}}(Сотрудники)$

Соединение (Join) - $R \bowtie S$

Комбинация декартова произведения и выборки.

Естественное соединение (Natural Join) - $R \bowtie S$

- Объединяет кортежи с одинаковыми значениями общих атрибутов;
- Общие атрибуты включаются в результат один раз;

Пример:

$Сотрудники \bowtie Отделы$

Тета-соединение (Theta Join) - $R \bowtie_{\theta} S$

Декартово произведение с последующей выборкой по условию θ .

Пример:

$R \bowtie_{\{R.A > S.B\}} S$

Деление (Division) - $R \div S$

Операция, обратная декартову произведению.

Условия:

- Атрибуты S должны быть подмножеством атрибутов R
- Результат содержит атрибуты R - S

Дополнительные операции

Внешнее соединение (Outer Join)

Левое внешнее соединение (Left Outer Join) - $R \bowtie S$

Сохраняет все кортежи левого отношения, даже если нет соответствия в правом.

Правое внешнее соединение (Right Outer Join) - $R \ltimes S$

Сохраняет все кортежи правого отношения.

Полное внешнее соединение (Full Outer Join) - $R \Join S$

Сохраняет все кортежи обоих отношений.

Полусоединение (Semijoin) - $R \ltimes S$

Возвращает кортежи R, которые имеют соответствие в S.

Пример:

Студенты \ltimes Студенты_сдавшие_экзамен

Антисоединение (Antijoin) - $R \not\bowtie S$

Возвращает кортежи R, которые не имеют соответствия в S.

Примеры комплексных запросов

Пример 1: Найти имена сотрудников, работающих в отделе 'ИТ'

$\pi_{\{Имя\}}(\sigma_{\{Название_отдела='ИТ'\}}(Сотрудники \bowtie Отделы))$

Пример 2: Найти сотрудников, не имеющих проектов

Сотрудники - $\pi_{\{Сотрудники.*\}}(Сотрудники \ltimes Назначения_проектов)$

Пример 3: Найти отделы, в которых работают все сотрудники с зарплатой > 100000

$\pi_{\{\text{Отдел_ид}\}}(\sigma_{\{\text{Зарплата} > 100000\}}(\text{Сотрудники})) \div \pi_{\{\text{Сотрудник_ид}\}}(\sigma_{\{\text{Зарплата} > 100000\}}(\text{Сотрудники}))$

Связь с SQL

Реляционная алгебра является теоретической основой SQL:

Реляционная алгебра	SQL
$\sigma_{\theta}(R)$	SELECT * FROM R WHERE θ
$\pi_a(R)$	SELECT a FROM R
$R \cup S$	SELECT * FROM R UNION SELECT * FROM S
$R \bowtie S$	SELECT * FROM R NATURAL JOIN S
$R \times S$	SELECT * FROM R CROSS JOIN S

Практическое значение реляционной алгебры

Для разработчиков БД:

- Понимание преобразования запросов;
- Основы для оптимизации;
- Проверка корректности запросов;

Для оптимизаторов:

- Алгебраические преобразования;
- Выбор порядка операций;
- Оценка стоимости запросов;

Контрольные вопросы

1. Что такое реляционная алгебра и каково ее назначение?
2. Перечислите и охарактеризуйте базовые операции реляционной алгебры.
3. В чем разница между выбором и проекцией?
4. Какие типы соединений вы знаете? В чем их различия?
5. Как операция деления связана с квантором всеобщности?
6. Приведите пример комплексного выражения реляционной алгебры.

Литература

1. Дейт К. Дж. Введение в системы баз данных. - Глава 6;
2. Коннолли Т., Бегг К. Базы данных: проектирование, реализация и сопровождение. - Глава 4;
3. Кузнецов С.Д. Основы баз данных. - Глава 4;
4. Ульман Дж. Основы систем баз данных. - Глава 2.