

Тема 19. Теория проектирования реляционных баз данных

Аномалии схем баз данных

При проектировании реляционных баз данных часто возникает проблема выбора схемы отношения из нескольких вариантов. При этом некоторые схемы оказываются предпочтительнее других, оцениваемых как худшие. Почему так происходит?

Рассмотрим схему **Поставщики (Название, Адрес, Товар, Цена)**. При работе с соответствующим отношением возможны следующие проблемы:

- Избыточность. Адрес поставщика повторяется для каждого товара.
- Потенциальная противоречивость (аномалии обновления).
Вследствие избыточности можно обновить адрес в одном кортеже и забыть сделать это в другом. В результате некоторые поставщики будут иметь два адреса.
- Аномалии включения. В базу данных не может быть записан адрес поставщика, если он не поставляет хотя бы один товар.
- Аномалии удаления. При удалении всех товаров, поставляемых данным поставщиком, утрачивается его адрес.

Названные проблемы исчезнут, если заменить отношение **Поставщики** двумя отношениями:

- **Поставщик_адрес (Название, Адрес),**
- **Поставщик_Товар_Цена (Название, Адрес, Цена).**

Однако при этом возникают другие проблемы. Всем этим и занимается теория проектирования реляционных баз данных. Центральное место в этой теории отводится идее зависимостей данных (функциональных и многозначных).

Функциональные зависимости

Существуют два основных вида ограничений на отношения:

- семантические ограничения на значения атрибутов,
- функциональные и многозначные зависимости.

Примеры семантических ограничений: рост человека не должен превышать 3-х метров, рабочий стаж не может быть больше возраста и т.д. Эти ограничения не вызывают больших проблем при работе с БД. Наоборот, функциональные зависимости, требующие равенства или неравенства компонентов разных кортежей, оказывают большое влияние на проектирование БД.

Определение. Пусть $R(A_1, A_2, \dots, A_n)$ – схема отношения, а X и Y – подмножества множества атрибутов схемы. Говорят, что X функционально определяет Y (Y функционально зависит от X) (обозначается $X \rightarrow Y$), если все кортежи отношения R с одинаковыми значениями атрибутов из X имеют одинаковые значения атрибутов из Y .

Примером функциональной зависимости является зависимость $X \rightarrow Y$, где X – ключевые атрибуты отношения, а Y – любое подмножество атрибутов отношения.

Следует отметить, что функциональные зависимости являются утверждениями обо всех отношениях с данной схемой. Из исследования конкретного отношения нельзя сделать вывод, что в схеме имеются те или иные зависимости. Исследование конкретного отношения, однако, позволяет сделать вывод, что некоторые зависимости не имеют места.

Решение о наличии той или иной зависимости в БД принимается проектировщиком и в дальнейшем поддерживается СУБД.

Примеры функциональных зависимостей.

В отношении Покупатели имеется зависимость Фамилия \rightarrow Адрес, Баланс; в отношении Заказы – зависимость Фамилия, Товар \rightarrow Количество; в отношении Поставщики – зависимости Название \rightarrow Адрес и Название, Товар \rightarrow Цена.

Логические следствия зависимостей

Пусть R – схема, а A, B, C – некоторые её атрибуты. Пусть в R существуют зависимости $A \rightarrow B, B \rightarrow C$. Тогда можно утверждать, что зависимость $A \rightarrow C$ также имеет место в R .

В общем случае, пусть F – множество функциональных зависимостей для R и $X \rightarrow Y$ – некоторая функциональная зависимость. Говорят, что зависимость $X \rightarrow Y$ логически следует из F , если для каждого отношения r со схемой R , удовлетворяющего зависимостям из F , удовлетворяется также зависимость $X \rightarrow Y$.

Обозначим F^+ логическое замыкание F – множество функциональных зависимостей, логически следующих из F . Если $F = F^+$, то F – полное семейство зависимостей.

Ключи

Пусть $R = A_1, A_2, \dots, A_n$; F – функциональные зависимости R ; $X \subseteq A_1, A_2, \dots, A_n$. Тогда X – ключ R , если выполняются условия:

1. функциональная зависимость $X \rightarrow A_1, A_2, \dots, A_n$ принадлежит F^+ (однозначная идентификация);
2. $Y \subset X \Rightarrow Y \rightarrow A_1, A_2, \dots, A_n \notin F^+$ (минимальность ключа).

Условие 1 означает, что ключ однозначно идентифицирует каждый кортеж отношения; условие 2 утверждает, что никакая часть ключа сама ключом не является.

Отношение может иметь несколько ключей, один из них принимается за первичный ключ, другие считаются возможными ключами.

Пример.

В отношении Покупатели (Фамилия, Адрес, Баланс) ключом является атрибут Фамилия, т.к. зависимость Фамилия \rightarrow Фамилия, Адрес, Баланс $\in F^+$ и выполняется условие 2. Других ключей в отношении нет.

Аксиомы функциональных зависимостей

Аксиомы функциональных зависимостей позволяют выводить из известных зависимостей новые зависимости.

Пусть U – множество атрибутов, F – множество функциональных зависимостей. Имеют место следующие правила вывода:

A1. Рефлексивность. Если $Y \subseteq X \subseteq U$, то $X \rightarrow Y$ логически следует из F. Это тривиальная зависимость: правая часть зависимости целиком содержится в ее левой части.

A2. Пополнение. Если $X \rightarrow Y$ и $Z \subseteq U$, то $XZ \rightarrow YZ$.

A3. Транзитивность. Если $X \rightarrow Y$ и $Y \rightarrow Z$, то $X \rightarrow Z$.

Указанные аксиомы полны и надежны, т.е. позволяют вывести все зависимости из F^+ и не позволяют вывести зависимости, не принадлежащие F^+ .

Производные правила вывода

Из аксиом A1, A2, A3 следуют производные правила вывода, которые как и основные позволяют выводить новые зависимости.

A4. Объединение. Если $X \rightarrow Y$ и $X \rightarrow Z$, то $X \rightarrow YZ$.

A5. Псевдотранзитивность. Если $X \rightarrow Y$ и $WY \rightarrow Z$, то $XW \rightarrow Z$.

A6. Декомпозиция. Если $X \rightarrow Y$ и $Z \subseteq Y$, то $X \rightarrow Z$.

Правила объединения и декомпозиции дают важное следствие: зависимость $X \rightarrow A_1, A_2, \dots, A_n$ справедлива, если и только если зависимость $X \rightarrow A_i$ справедлива для всех $i = [1..n]$.

В качестве примера использования аксиом приведем доказательство производных правил вывода.

A4. $X \rightarrow Y \Rightarrow X \rightarrow XY$; (пополнение по X) (1)

$X \rightarrow Z \Rightarrow XY \rightarrow YZ$. (пополнение по Y) (2)

Следовательно, $X \rightarrow YZ$. (транзитивность 1, 2)

A5. $X \rightarrow Y \Rightarrow WX \rightarrow WY$; (пополнение по W) (3)

$WY \rightarrow Z$. (дано) (4)

Следовательно, $WX \rightarrow Z$. (транзитивность 3, 4)

A6. $Z \subseteq Y \Rightarrow Y \rightarrow Z$; (рефлексивность) (5)

$X \rightarrow Y$. (дано) (6)

Следовательно, $X \rightarrow Z$. (транзитивность 6, 5)

Введем важное понятие замыкания множества атрибутов относительно множества функциональных зависимостей.

Пусть F – множество функциональных зависимостей на U , и пусть $X \subseteq U$. Тогда X^+ , замыкание X относительно F , есть множество атрибутов A таких, что зависимость $X \rightarrow A$ может быть выведена из F .

Можно показать, что зависимость $X \rightarrow Y$ выводится из F , если и только если $Y \in X^+$. Это обстоятельство позволяет заменить проверку $X \rightarrow Y$ проверкой утверждения $Y \in X^+$, которая выполняется намного быстрее.

Декомпозиция схем отношений

Декомпозицией схемы отношения $R = \{A_1, A_2, \dots, A_n\}$ называется замена ее совокупностью $\rho = \{R_1, R_2, \dots, R_k\}$, где каждое $R_i \subseteq R$ и $\cup R_i = R$. Схемы могут пересекаться.

Декомпозиция может снять часть проблем, связанных с избыточностью. С другой стороны, желательно, чтобы декомпозиция не приводила к потере информации, чтобы всегда можно было бы восстановить исходное отношение из его разложения. При этом естественно ожидать, что отношения из разложения будут проекциями исходного отношения на схемы R_i . В этом случае восстановление отношения потребует соединения отношений разложения.

Определение. Пусть R – схема отношения; $\{R_1, R_2, \dots, R_k\}$ – декомпозиция R , F – множество зависимостей. Говорят, что декомпозиция обладает **свойством соединения без потерь** (сохраняет данные) относительно F , если каждое отношение r для схемы R , удовлетворяющее F , может быть представлено в виде:

$$r = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie \dots \bowtie \pi_{R_k}(r).$$

Т.е. r является естественным соединением его проекций на все R_i . Свойство соединения без потерь желательно для декомпозиции.

Рассмотрим часто встречающуюся на практике декомпозицию вида $\rho = \{R_1, R_2\}$. Такая декомпозиция обладает свойством соединения без потерь относительно множества функциональных зависимостей F , если и только если зависимости $R_1 \cap R_2 \rightarrow R_1 - R_2$ или $R_1 \cap R_2 \rightarrow R_2 - R_1$ принадлежат F^+ .

Декомпозиции, сохраняющие зависимости

Еще одно важное свойство декомпозиций связано с сохранением зависимостей.

Определение. Проекцией множества зависимостей F на множество атрибутов Z (обозначается $\pi_Z(F)$) называется множество зависимостей $X \rightarrow Y$ в F^+ таких, что $XY \subseteq Z$. Говорят, что декомпозиция ρ сохраняет множество зависимостей F , если из объединения множеств зависимостей $\pi_{R_i}(F)$ ($i=1..k$), логически следуют все зависимости F .

Следует отметить, что декомпозиция может обладать свойством соединения без потерь относительно F , но не сохранять F . И наоборот, может сохранять F , но не обладать свойством соединения без потерь.