

Реляционные базы данных. Проектирование баз данных

Проектирование реляционной базы данных на основе декомпозиции универсального отношения

Проектирование реляционной базы данных фактически сводится к **устранению избыточных функциональных зависимостей** (а при необходимости избыточных многозначных зависимостей и зависимостей по соединению) из предварительного набора отношений, полученного каким-либо способом.

В том случае, когда проектируемая база данных сравнительно невелика (общее число атрибутов не превышает 20-30), предварительный набор отношений можно представить в виде одного отношения, называемого **универсальным**. В него включаются все представляющие интерес атрибуты.

Проектирование реляционной базы данных на основе декомпозиции универсального отношения

В качестве примера построим **универсальное отношение** для базы данных **publications**:

PUBLICATIONS(AUTHOR, TITLE, YEARPUB, PUBLISHER, PUBL_URL, SITE, SITE_URL)

здесь

- AUTHOR - имя автора
- TITLE - название книги
- YEARPUB - год издания книги
- PUBLISHER - наименование издательства
- PUBL_URL - ссылка на веб-сервер издательства
- SITE - наименование Internet-ресурса
- SITE_URL - указатель на Internet-ресурс

Функциональные зависимости, имеющиеся в полученном отношении, представлены на следующей схеме:

(1) TITLE --> YEARPUB

(2) -----> PUBLISHER --> PUBL_URL

(3) SITE ---> SITE_URL

Проектирование реляционной базы данных на основе декомпозиции универсального отношения

Разработать БД для учета ежемесячной оплаты за услуги Internet - провайдера. БД должна содержать следующие сведения:

наименование абонента, его IP-адрес и E-mail, время начала и конца сеанса, продолжительность и стоимость сеанса, тариф и его название, сведения об уплате с указанием дат и сумм, сумма долга, а также задолженность абонентов за отчетный квартал.

Проектирование реляционной базы данных на основе декомпозиции универсального отношения

| Наименование поля | Индекс | Тип | Размер | Формат |
|--------------------------|--------|-----|--------|--------|
| Наименование_абонента | V | T | 50 | |
| ИНН | | T | 14 | |
| IP-Адрес | | T | 16 | |
| E-mail | | G | 25 | |
| Начало_сеанса | V | D | 8 | ПФД |
| Конец_сеанса | | D | 8 | ПФД |
| Длительность | | Ч | ДЦ | |
| Стоимость_сеанса | | Ч | ОПТ | Ф2 |
| Тариф | | T | 20 | |
| Цена за 1с | | Ч | ОПТ | Ф2 |
| Дата_оплаты | | D | КФД | |
| Сумма_оплаты | | Ч | ОПТ | Ф2 |
| Сумма_долга | | Ч | ОПТ | Ф2 |
| Год | | Ч | Ц | |
| Месяц | | Ч | Ц | |
| Задолженность_за_месяц | | Ч | ОПТ | Ф2 |
| Квартал | | Ч | Б | |
| Задолженность_за_квартал | | Ч | ОПТ | Ф2 |

Проектирование реляционной базы данных на основе декомпозиции универсального отношения

Присутствуют следующие ФЗ:

- IP-Адрес -> Наименование_абонента
- ИНН-> Наименование_абонента
- Код_абонента-> Наименование_абонента
- Длительность -> Начало_сеанса, Конец_сеанса
- Стоимость_сеанса -> Длительность, Цена, Тариф
- Цена -> Тариф
- Сумма_оплаты -> Дата_оплаты
- Сумма_долга -> Сумма_оплаты, Стоимость_сеанса
- Квартал -> Начало_сеанса
- Год -> Начало_сеанса
- Задолженность_за_квартал -> Год, Квартал, Сумма_долга

Реляционные базы данных. Нормализация отношений

Нормализация – это метод создания набора отношений с заданными свойствами на основе требований к данным

Процесс нормализации был впервые предложен Э. Ф. Коддом (1972). Нормализация часто выполняется в виде последовательности тестов для некоторого отношения с целью проверки его соответствия (или несоответствия) требованиям заданной нормальной формы. Таким образом, процесс нормализации основывается на концепции нормальных форм. Говорят, что отношение находится в некоторой нормальной форме, если удовлетворяет заданному набору условий.

Ненормализованная форма – это таблица, содержащая одну или несколько повторяющихся групп данных.

Первая нормальная форма – это отношение, в котором все используемые домены содержат только скалярные значения.

Реляционные базы данных. Нормализация отношений

Процесс нормализации начинается с преобразования данных из формата источника (например, из формата стандартной формы ввода данных) в формат таблицы со строками и столбцами.

На исходном этапе таблица находится в ненормализованной форме (ННФ) и часто называется **ненормализованной таблицей**.

Для преобразования ненормализованной таблицы в первую нормальную форму (1НФ) в исходной таблице следует найти и устранить все повторяющиеся группы данных. Повторяющейся группой называется группа, состоящая из одного и более атрибутов таблицы, в которой возможно наличие нескольких значений для единственного значения ключевого атрибута таблицы.

Существует два подхода..

Реляционные базы данных. Нормализация отношений

Проектирование реляционной базы данных на основе декомпозиции универсального отношения

В первом подходе повторяющиеся группы устраняются путем ввода соответствующих данных в пустые столбцы строк с повторяющимися данными. Иными словами, пустые места при этом заполняются дубликатами неповторяющихся данных. Этот подход часто называют «выравниванием» (flattening) таблицы. Полученные в результате этих действий таблица, которая теперь будет называться отношением, содержит атомарные (или единственные) значения на пересечении каждой строки с каждым столбцом, а потому находится в первой нормальной форме. В результате такого подхода в полученное отношение вносится некоторая избыточность данных, которая в ходе дальнейшей нормализации будет устранена.

Реляционные базы данных. Нормализация отношений

Проектирование реляционной базы данных на основе декомпозиции универсального отношения

Во втором подходе один атрибут или группа атрибутов назначаются ключом ненормализованной таблицы, а затем повторяющиеся группы изымаются и помещаются в отдельные отношения вместе с копиями ключа исходной таблицы. Далее в новых отношениях устанавливаются первичные ключи. Иногда ненормализованные отношения могут содержать одну или несколько повторяющихся групп внутри повторяющихся групп первого порядка. В таких случаях данный прием применяется до тех пор, пока повторяющихся групп совсем не останется. Полученный набор отношений будет находиться в первой нормальной форме только тогда, когда ни в одном из них не будет повторяющихся групп атрибутов.

Реляционные базы данных. Нормализация отношений

В теории реляционных баз данных обычно выделяется следующая последовательность нормальных форм:

- первая нормальная форма (1NF);
- вторая нормальная форма (2NF);
- третья нормальная форма (3NF);
- нормальная форма Бойса-Кодда (BCNF);
- четвертая нормальная форма (4NF);
- пятая нормальная форма, или нормальная форма проекции-соединения (5NF или PJ/NF).

Реляционные базы данных. Нормализация отношений

Простой атрибут – это атрибут, значения которого атомарные (т. е. неделимы).

Сложный атрибут может иметь значение, представляющее собой конкатенацию нескольких значений одного или разных доменов. Аналогом сложного атрибута может быть агрегат или повторяющийся агрегат данных.

Первая нормальная форма (1NF)

Основные критерии:

□ Все строки должны быть различными.

□ Все элементы внутри ячеек должны быть атомарными (не списками или агрегатами).

Другими словами, элемент является атомарным, если его нельзя разделить на части, которые могут использовать в таблице независимо друг от друга.

Реляционные базы данных. Нормализация отношений

Пример не 1NF

таблицы:

| Категория | Товары |
|-----------|---------------------|
| Книги | Война и Мир, Азбука |
| Игрушки | Юла |

В этом примере в одной из ячеек содержится список из двух элементов: Война и Мир, Азбука, т.е. он является не атомарным.

Исправить можно так:

| Категория | Товары |
|-----------|-------------|
| Книги | Война и Мир |
| Книги | Азбука |
| Игрушки | Юла |

Реляционные базы данных. Нормализация отношений

Методы приведения к 1NF:

- Устраните повторяющиеся группы в отдельных таблицах (одинаковые строки).
- Создайте отдельную таблицу для каждого набора связанных данных.
- Идентифицируйте каждый набор связанных данных с помощью первичного ключа (защита от дублирования строк):

Id – идентификатор (первичный ключ),

Code – шифр рубрики, Theme – название рубрики,

Title – название книги,

Author – автор, Editor – редактор,

Type – тип издания (учебник, учебное пособие, сборник и.т.п.),

Year – год издания, Pg – количество страниц.

Реляционные базы данных. Нормализация отношений

| ID | Code | Theme | Author | Title | Editor | Type | Year | Pg |
|-----|-------|-------------|-------------|----------------------|--------------|-----------------|------|-----|
| 200 | 681.3 | ПО ВТ | Бочков С. | Язык СИ | Садчиков П. | учебник | 1990 | 384 |
| | | | Субботин Д. | | | | | |
| 100 | 681.3 | ПО ВТ | Джехани Н. | Язык АДА | | учебник | 1960 | 552 |
| 300 | 621.5 | МО | Крон Г. | Диакоптика | Баранов А. | учебник | 1972 | 544 |
| 876 | 007 | ИИ | Гик Е.Я. | Шахматы и математика | Кикоин И. | учебное пособие | 1983 | 176 |
| | | | | | Капица С. | | | |
| 440 | 32.97 | ВТ | | ПУ для ПЭВМ | Витенберг А. | справочник | 1992 | 208 |
| 385 | 001.8 | Информатика | Фролов Г. | Элементы информатики | Храмов А. | учебное пособие | 1989 | 304 |
| | | | Кузнецов Э. | | Рожков П. | | | |

Реляционные базы данных. Нормализация отношений

| ID | Code | Theme | Author | Title | Editor | Type | Year | Pg |
|-----|-------|-------------|-------------|----------------------|--------------|-----------------|------|-----|
| 200 | 681.3 | ПО ВТ | Бочков С. | Язык СИ | Садчиков П. | учебник | 1990 | 384 |
| 200 | 681.3 | ПО ВТ | Субботин Д. | Язык СИ | Садчиков П. | учебник | 1990 | 384 |
| 100 | 681.3 | ПО ВТ | Джехани Н. | Язык АДА | | учебник | 1960 | 552 |
| 300 | 621.5 | МО | Крон Г. | Диакоптика | Баранов А. | учебник | 1972 | 544 |
| 876 | 007 | ИИ | Гик Е.Я. | Шахматы и математика | Кикоин И. | учебное пособие | 1983 | 176 |
| 876 | 007 | ИИ | Гик Е.Я. | Шахматы и математика | Капица С. | учебное пособие | 1983 | 176 |
| 440 | 32.97 | ВТ | | ПУ для ПЭВМ | Витенберг А. | справочник | 1992 | 208 |
| 385 | 001.8 | Информатика | Фролов Г. | Элементы информатики | Храмов А. | учебное пособие | 1989 | 304 |
| 385 | 001.8 | Информатика | Кузнецов Э. | Элементы информатики | Рожков П. | учебное пособие | 1989 | 304 |

Реляционные базы данных. Нормализация отношений

Вторая нормальная форма (2NF)

Пусть R является переменной отношения, а X и Y — произвольными подмножествами множества атрибутов переменной отношения R . Тогда Y функционально зависимо от X : $X \rightarrow Y$

Для **любого допустимого значения переменной отношения R** каждое значение множества X отношения R связано точно с одним значением множества Y отношения R .

Иначе говоря, если два кортежа переменной отношения R совпадают по значению X , они также совпадают и по значению Y .

$X \rightarrow Y$

детерминант

зависимая часть

Реляционные базы данных. Нормализация отношений

Вторая нормальная форма (2NF)

Вторая нормальная форма (2НФ) основана на понятии полной функциональной зависимости. Наиболее важные на практике нормальные формы отношений основываются на фундаментальном в теории реляционных баз данных понятии функциональной зависимости.

Определение 1. Функциональная зависимость

В отношении R атрибут Y функционально зависит от атрибута X (X и Y могут быть составными) в том и только в том случае, если каждому значению X соответствует в точности одно значение Y : $R.X (r) R.Y$.

Определение 2. Полная функциональная зависимость

Функциональная зависимость $R.X (r) R.Y$ называется полной, если атрибут Y не зависит функционально от любого точного подмножества X .

Определение 3. Транзитивная функциональная зависимость

Функциональная зависимость $R.X \rightarrow R.Y$ называется транзитивной, если существует такой атрибут Z , что имеются функциональные зависимости $R.X \rightarrow R.Z$ и $R.Z \rightarrow R.Y$ и отсутствует функциональная зависимость $R.Z \rightarrow R.X$. (При отсутствии последнего требования мы имели бы "неинтересные" транзитивные зависимости в любом отношении, обладающем несколькими ключами.)

Реляционные базы данных. Нормализация отношений

Определение 4. *Неключевой атрибут*

Неключевым атрибутом называется любой атрибут отношения, не входящий в состав первичного ключа (в частности, первичного).

Определение 5. *Взаимно независимые атрибуты*

Два или более атрибута взаимно независимы, если ни один из этих атрибутов не является функционально зависимым от других.

В нормализованном отношении все неключевые атрибуты функционально зависят от ключа отношения.

Говорят, что неключевой атрибут функционально полно зависит от составного ключа, если он функционально зависит от ключа, но не находится в функциональной зависимости ни от какой части составного ключа.

В некотором отношении атрибут B называется **полностью функционально зависимым** от атрибута A , если атрибут B функционально зависит от полного значения атрибута A и не зависит ни от какого подмножества полного значения атрибута A .

Реляционные базы данных. Нормализация отношений

Функциональная зависимость $A \rightarrow B$ является *полной* функциональной зависимостью, если удаление какого-либо атрибута из A приводит к утрате этой зависимости.

Частичной функциональной зависимостью называется такая зависимость $A \rightarrow B$, если в A есть некий атрибут, при удалении которого эта зависимость сохраняется.

Основные критерии 2НФ:

- Таблица должна находиться в первой нормальной форме.
- Любое её поле, не входящее в состав первичного ключа, функционально полно зависит от первичного ключа.
- Каждый неключевой атрибут отношения находится в нетранзитивной (т. е. прямой) в зависимости от потенциального ключа.

| Категория | Дата | Скидка | Товар |
|-----------|------------|--------|-----------------|
| Книги | 10.10.2008 | 10% | PHP for dummies |
| Ноутбуки | 11.10.2008 | 20% | Acer |
| Книги | 10.10.2008 | 10% | Windows XP |

Реляционные базы данных. Нормализация отношений

В этой таблице первичный ключ составляют первые два столбца (**Категория и Дата**).

Скидка функционально полно зависит от них обоих, так как определяется одновременно категорией и датой. А вот с товаром проблемы. Она зависит только от категории и не зависит от даты (в любой день будет лежать там, пока не купят). Поэтому, говорят, что **он зависит функционально не полно от всего первичного ключа и функционально полно от его части (Категория)**. Это нарушает требования второй нормальной формы.

Методы приведения к 2NF:

- Создайте отдельные таблицы для наборов значений, относящихся к нескольким записям.
- Свяжите эти таблицы с помощью внешнего ключа (В нашем случае – это поле Категория).

Реляционные базы данных. Нормализация отношений

КНИГИ

| ID | Code | Theme | Title | Type | Year | Pg |
|-----|-------|-------------|----------------------|-----------------|------|-----|
| 200 | 681.3 | ПО ВТ | Язык СИ для ПК | учебник | 1990 | 384 |
| 100 | 681.3 | ПО ВТ | Язык АДА | учебник | 1960 | 552 |
| 300 | 621.5 | МО | Диакоптика | учебник | 1972 | 544 |
| 876 | 007 | ИИ | Шахматы и математика | учебное пособие | 1983 | 176 |
| 440 | 32.97 | ВТ | ПУ для ПЭВМ | справочник | 1992 | 208 |
| 385 | 001.8 | Информатика | Элементы информатики | учебное пособие | 1989 | 304 |

Ключом отношения КНИГИ является комбинация полей (ID, Author, Editor). Все поля, не входящие в состав ключа, зависят только от идентификатора книги. Поэтому отношение должно быть разбито на два: КНИГИ и КНИГИ–АВТОРЫ–РЕДАКТОРЫ

Эти отношения связаны по внешнему ключу, которым является поле ID.

Реляционные базы данных. Нормализация отношений

| ID | Author | Editor |
|-----|-------------|--------------|
| 200 | Бочков С. | Садчиков П. |
| 200 | Субботин Д. | Садчиков П. |
| 100 | Джехани Н. | |
| 300 | Крон Г. | Баранов А. |
| 876 | Гик Е.Я. | Кикоин И. |
| 876 | Гик Е.Я. | Капица С. |
| 440 | | Витенберг А. |
| 385 | Фролов Г. | Храмов А. |
| 385 | Кузнецов Э. | Рожков П. |

КНИГИ–АВТОРЫ– РЕДАКТОРЫ

в 2NF форме нет ключевых атрибутов, зависящих от части сложного потенциального ключа.

Второе важное значение второй нормальной формы состоит в том, что она по определению запрещает наличие неключевых атрибутов, которые вообще не зависят от потенциального ключа.

Таким образом, 2NF запрещает создавать отношения как несвязанные (хаотические, случайные) наборы атрибутов.

Реляционные базы данных. Нормализация отношений

Третья нормальная форма (3NF)

Основные критерии:

- Таблица находится во второй нормальной форме.
- Любой её не ключевой атрибут функционально зависит только от первичного ключа.
- Каждый неключевой атрибут отношения находится в нетранзитивной (т.е. прямой) в зависимости от потенциального ключа.

Методы приведения к 3NF:

- Удаление полей, не зависящих от ключа
- Проще говоря, второе правило требует выносить все не ключевые поля, содержимое которых может относиться к нескольким записям таблицы в отдельные таблицы.

Реляционные базы данных. Нормализация отношений

Пусть X, Y, Z – атрибуты некоторого отношения. При этом $X \twoheadrightarrow Y$ и $Y \twoheadrightarrow Z$, но обратное соответствие отсутствует, т.е. Z не зависит от Y или Y не зависит от X . Тогда говорят, что Z транзитивно зависит от X ($X \twoheadrightarrow Z$).

Нормализация 2НФ-отношений с образованием 3НФ-отношений включает устранение транзитивных зависимостей. Если в отношении существует транзитивная зависимость между атрибутами, то в таком случае транзитивно-зависимые атрибуты удаляются из него и помещаются в новое отношение вместе с копией детерминанта.

Отношение находится в 3НФ, если оно находится во 2НФ и в нем отсутствуют транзитивные зависимости.

Для отношения КНИГИ атрибут Theme зависит от атрибута Code, а не от ключа (хотя название рубрики, естественно, соответствует её шифру). Поэтому для приведения отношения к 3НФ нужно выделить из него ещё одно отношение РУБРИКАТОР.

Реляционные базы данных. Нормализация отношений

КНИГИ

| ID | Code | Title | Type | Year | Pg |
|-----|-------|----------------------|-----------------|------|-----|
| 200 | 681.3 | Язык СИ для ПК | учебник | 1990 | 384 |
| 100 | 681.3 | Язык АДА | учебник | 1960 | 552 |
| 300 | 621.5 | Диакоптика | учебник | 1972 | 544 |
| 440 | 32.97 | ПУ для ПЭВМ | справочник | 1992 | 208 |
| 876 | 007 | Шахматы и математика | учебное пособие | 1983 | 176 |
| 385 | 001.8 | Элементы информатики | учебное пособие | 1989 | 304 |

| Code | Theme |
|-------|-------------|
| 681.3 | ПО ВТ |
| 621.5 | МО |
| 007 | ИИ |
| 32.97 | ВТ |
| 001.8 | Информатика |

РУБРИКАТОР

Реляционные базы данных. Нормализация отношений

Нормальная форма Бойса-Кодда (BCNF)

При проектировании БД третья нормальная форма схем отношений достаточна в большинстве случаев, и приведением к третьей нормальной форме процесс проектирования реляционной базы данных обычно заканчивается.

Основные критерии:

- Таблица находится в третьей нормальной форме
- В таблице должен быть только один потенциальный первичный ключ.

Начальное определение Э. Ф. Кодда для 3НФ не совсем подходит для отношений с перечисленными ниже условиями:

- а) отношение имеет два (или более) потенциальных ключа;
- б) два потенциальных ключа являются сложными;
- в) потенциальные ключи перекрываются (т. е. имеют, по крайней мере, один общий атрибут).

Реляционные базы данных. Нормализация отношений

Определение. Детерминант

Детерминант - любой атрибут, от которого полностью функционально зависит некоторый другой атрибут.

Определение. Нормальная форма Бойса-Кодда

Отношение R находится в нормальной форме Бойса-Кодда (BCNF) в том и только в том случае, если каждый детерминант является возможным ключом.

Методы приведения к BCNF:

Вынести в отдельные таблицы потенциальные первичные ключи

Реляционные базы данных. Нормализация отношений

Четвертая нормальная форма (4NF)

В этой форме дополнительное правило должно исключать многозначные зависимости. Другими словами, все строки таблицы должны быть независимыми друг от друга. В том смысле, что наличие какой-то строки X , не должно означать, что строка Y тоже где-то есть в этой таблице.

Многозначная зависимость существует, если заданным значениям атрибута X соответствует множество, состоящее из нуля (или более) значений атрибута Y ($X \twoheadrightarrow Y$). Если в отношении присутствуют многозначные зависимости, то схема отношения должна находиться в 4НФ.

Различают **тривиальные и нетривиальные** многозначные зависимости.

Тривиальной называется такая многозначная зависимость $X \twoheadrightarrow Y$, для которой Y является подмножеством X или $X \cup Y$ состоит из всех атрибутов R).

Реляционные БД. Нормализация отношений

По Риссанену: переменную отношения $R\{A, B, C\}$, удовлетворяющую функциональным зависимостям $A \rightarrow B$ и $B \rightarrow C$, желательно разбивать на проекции по атрибутам $\{A, B\}$ и $\{B, C\}$ (а не на проекции по атрибутам $\{A, B\}$ и $\{A, C\}$)

Для многозначных зависимостей также верно:

$$A \twoheadrightarrow B \text{ и } B \twoheadrightarrow C$$

Реляционные БД. Нормализация отношений

Зависимости соединения

Рассмотрим переменную отношения SPJ (поставщики, детали и проекты).

Эта переменная отношения состоит только из ключевых атрибутов, не содержит нетривиальных функциональных и многозначных зависимостей и поэтому находится в 4НФ.

| <u>S</u> | <u>P</u> | <u>J</u> |
|----------|----------|----------|
| S1 | P1 | J2 |
| S1 | P2 | J1 |
| S2 | P1 | J1 |
| S1 | P1 | J1 |

Реляционные БД. Нормализация отношений

SPJ

| <u>S</u> | <u>P</u> | <u>J</u> |
|----------|----------|----------|
| S1 | P1 | J2 |
| S1 | P2 | J1 |
| S2 | P1 | J1 |
| S1 | P1 | J1 |

SP

| <u>S</u> | <u>P</u> |
|----------|----------|
| S1 | P1 |
| S1 | P2 |
| S2 | P1 |

PJ

| <u>P</u> | <u>J</u> |
|----------|----------|
| P1 | J2 |
| P2 | J1 |
| P1 | J1 |

JS

| <u>J</u> | <u>S</u> |
|----------|----------|
| J2 | S1 |
| J1 | S1 |
| J1 | S2 |

соединение по P

| <u>S</u> | <u>P</u> | <u>J</u> |
|----------|----------|----------|
| S1 | P1 | J2 |
| S1 | P2 | J1 |
| S2 | P1 | J1 |
| S2 | P1 | J2 |
| S1 | P1 | J1 |

соединение по J,S

исходное отношение
SPJ

Реляционные БД. Нормализация отношений

Зависимости соединения

Пусть R — переменная отношения, а A, B, \dots, Z — произвольные подмножества множества ее атрибутов. Переменная отношения

R удовлетворяет следующей зависимости соединения

$$* \{A, B, \dots, Z\}$$

тогда и только тогда, когда любое допустимое значение переменной отношения R эквивалентно соединению ее проекций по подмножествам A, B, \dots, Z множества атрибутов

Зависимость соединения является таким же ограничением для данной переменной отношения, как многозначная и функциональная зависимости.

Реляционные базы данных. Нормализация отношений

➤ **Четвертая нормальная форма (4НФ)** - отношение в нормальной форме Бойса-Кодда, которое не содержит нетривиальных многозначных зависимостей.

Для отношения КНИГИ–АВТОРЫ–РЕДАКТОРЫ атрибуты Author и Editor образуют две многозначные зависимости от первичного ключа, и при этом значения этих атрибутов не зависят друг от друга. Поэтому для приведения отношения к 4НФ нужно разбить его на два отношения КНИГИ–АВТОРЫ и КНИГИ–РЕДАКТОРЫ

| ID | Author |
|-----|-------------|
| 200 | Бочков С. |
| 200 | Субботин Д. |
| 100 | Джехани Н. |
| 300 | Крон Г. |
| 876 | Гик Е.Я. |
| 385 | Фролов Г. |
| 385 | Кузнецов Э. |

| ID | Editor |
|-----|--------------|
| 200 | Садчиков П. |
| 300 | Баранов А. |
| 876 | Кикоин И. |
| 876 | Капица С. |
| 440 | Витенберг А. |
| 385 | Храмов А. |
| 385 | Рожков П. |

Реляционные БД. Нормализация отношений

Пятая нормальная форма (5NF)

Рональд Фагин в 1981 году

Отношение находится в пятой нормальной форме (иначе - в проекционно-соединительной нормальной форме) тогда и только тогда, когда каждая нетривиальная зависимость соединения в нем определяется потенциальным ключом (ключами) этого отношения

Определение 12. Зависимость соединения

Отношение $R(X, Y, \dots, Z)$ удовлетворяет зависимости соединения $*$ (X, Y, \dots, Z) в том и только в том случае, когда R восстанавливается без потерь путем соединения своих проекций на X, Y, \dots, Z .

Определение 13. Пятая нормальная форма

Отношение R находится в пятой нормальной форме (нормальной форме проекции-соединения - PJ/NF) в том и только в том случае, когда любая зависимость соединения в R следует из существования некоторого возможного ключа в R .

Реляционные БД. Нормализация отношений

ДОМЕННО-КЛЮЧЕВАЯ НОРМАЛЬНАЯ ФОРМА (ДКНФ)

Ограничение домена – ограничение, предписывающее использовать для определённого атрибута значения только из некоторого заданного домена. Ограничение по своей сути является заданием перечня (или логического эквивалента перечня) допустимых значений типа и объявлением о том, что указанный атрибут имеет данный тип.

Переменная отношения находится в ДКНФ тогда и только тогда, когда каждое наложенное на неё ограничение является логическим следствием ограничений доменов и ограничений ключей, наложенных на данную переменную отношения.

Отношение в ДКНФ не имеет аномалий модификации. Иными словами, чтобы не менялось - ничего не потеряется, если соблюдены все ограничения по ключей и доменов. Если выполнять некоторые правила, то при любых действиях с таблицей ее целостность не пострадает и вся необходимая информация сохранится.

Реляционные БД. Нормализация отношений

Если рассматривать на примере, то правила действуют примерно так: нельзя просто удалить категорию из таблицы категорий, если с этой категорией связаны, например, продукты из таблицы продуктов.

Прежде чем удалять категорию, необходимо выполнить предыдущие действия в таблице продуктов (например, поле, отвечающее за id категории этого товара, нужно сделать NULL или удалить все записи данной категории).

Любая переменная отношения, находящаяся в ДКНФ, обязательно находится в 5НФ. Однако не любую переменную отношения можно привести к ДКНФ.

Реляционные БД. Нормализация отношений

Шестая нормальная форма (6NF)

Введена К. Дейтом как обобщение пятой нормальной формы для темпоральных баз данных.

Переменная отношения находится в шестой нормальной форме тогда и только тогда, когда она удовлетворяет всем нетривиальным зависимостям соединения.

Из определения следует, что переменная находится в 6НФ тогда и только тогда, когда она неприводима, то есть не может быть подвергнута дальнейшей декомпозиции без потерь. Каждая переменная отношения, которая находится в 6НФ, также находится и в 5НФ.

Идея «декомпозиции до конца» выдвигалась до начала исследований в области хронологических данных, но не нашла поддержки. Однако для хронологических баз данных максимально возможная декомпозиция позволяет бороться с избыточностью и упрощает поддержание целостности базы данных.

Реляционные БД. Нормализация отношений

Для хронологических баз данных определены U_операторы, которые распаковывают отношения по указанным атрибутам, выполняют соответствующую операцию и упаковывают полученный результат. В данном примере соединение проекций отношения должно производиться при помощи оператора U_JOIN.

Нормализация сокращает дублирование данных, но появление новых отношений усложняет поддержку логической целостности данных.

Может возникнуть так называемая излишняя нормализация.