

## Нормализация отношений

Нормализация отношений – одна из основополагающих частей теории реляционных баз данных. Метод проектирования базы данных, который позволяет привести базу данных к минимальной избыточности.

Избыточность данных создает предпосылки для появления различных аномалий, снижает производительность, делает управление данными не гибким и не очень удобным. Таким образом, можно сделать вывод, что нормализация нужна для

- устранения аномалий,
- повышения производительности,
- повышения удобства управления данными.

Она имеет своей целью избавиться от избыточности в отношениях и модифицировать их структуру таким образом, чтобы процесс работы с ними не был обременен различными посторонними сложностями.

Выделяют последовательность нормальных форм:

- первая нормальная форма 1НФ;
- вторая нормальная форма 2НФ;
- третья нормальная форма 3НФ;
- усиленная нормальная третья форма или нормальная форма Бойса-Кодда (БКНФ);
- четвертая нормальная форма 4НФ;
- пятая нормальная форма 5НФ.

*Наиболее часто применяемые формы 1НФ, 2НФ, 3НФ. Они далее и будут рассмотрены*

**Первая нормальная форма.** Отношение находится в первой нормальной форме, (1НФ) если все его атрибуты простые (атомарные или неделимые), то есть если ни один из его атрибутов нельзя разделить на более простые атрибуты, которые соответствуют каким-то другим свойствам описываемой сущности.

*Примеры*

1) Отношение *Товар* = (Код товара, Название, Цвет, Вес, Цена\_за\_единицу) находится в 1НФ, так как все атрибуты в данном отношении являются простыми;

2) Отношение *Сотрудник* = (Таб\_номер, ФИОсотрудника, Дата\_рождения; Дети) не находится в 1НФ, так как атрибуты «ФИО сотрудника», «Дата\_рождения», «Дети» могут быть сложными:

- атрибут «ФИО\_сотрудника» содержит три части — фамилию, имя и отчество сотрудника фирмы, поэтому должен быть разбит на три атомарных атрибута, каждый из которых может использоваться самостоятельно;

- атрибут «Дата\_рождения», как и атрибут «ФИОсотрудника», не является атомарным, поскольку включает в себя три понятия — «Год», «Месяц» и «День». Однако в БД существуют типы данных — *Date* и *Date Time*, которые позволяют выполнять над датами специальные операции, например, <дата\_1> — <дата\_2> = <количество\_дней\_между\_ними>. В связи с этим атрибуты дат в базах данных считаются атомарными;

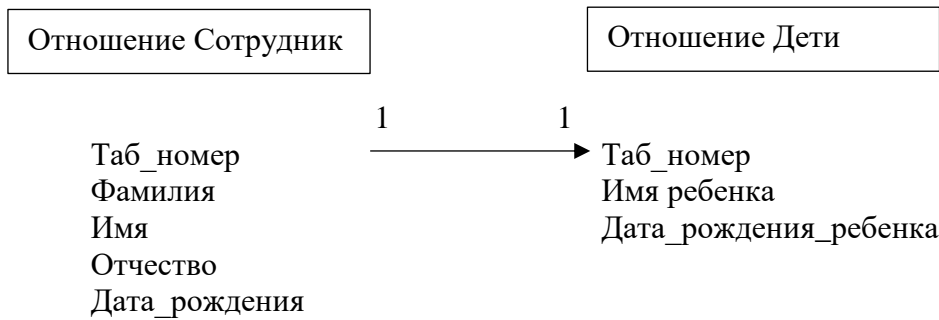
- атрибут «Дети» также является сложным, так как может содержать сведения об имени ребенка и его дату рождения.

Для приведения отношения *Сотрудник* к 1НФ необходимо декомпозировать его на следующие два отношения:

Сотрудник = Таб\_номер, Фамилия, Имя, Отчество, Дата\_рождения)

Дети = Таб\_номер, Имя ребенка, Дата\_рождения\_ребенка)

При этом связь между этими отношениями осуществляется через ключевое поле «Таб\_номер», тип связи «один к одному».



Основная операция метода – операция проекции

Вторая нормальная форма. Отношение находится во второй нормальной форме (2НФ), если оно находится в НФ и каждый неключевой атрибут функционально полно зависит от первичного ключа. Если в отношении нет частичных функциональных зависимостей, то оно находится во 2НФ.

*Если в отношении не ключевой атрибут функционально зависит от части составного ключа, то между ключом и не ключевым атрибутом имеет место **частичная функциональная зависимость**.*

*Если в отношении не ключевой атрибут функционально зависит от составного ключа и не находится в функциональной зависимости от любых его частей, то не ключевой атрибут **функционально полно зависит** от ключа.*

Рассмотрим отношение, в котором присутствуют частичные и полные функциональные зависимости:

#### Отношение Выдача книг

Шифр книги

Авторы

Название

Год издания

Номер читательского билета

ФИО

Шифр группы

Специальность

Дата выдачи

Дата возврата

Ключом данного отношения являются поля: Шифр книги, Номер читательского билета.

Полная функциональная зависимость в этом отношении:

Шифр книги, Номер читательского билета → Дата выдачи, Дата возврата.

Частичные функциональные зависимости:

Шифр книги → Авторы, Название, Год издания.

Номер читательского билета → ФИО., Шифр группы, Специальность.

Частичные функциональные зависимости приводят к аномалиям:

- избыточно дублируются сведения о книгах (авторы, название, год издания), поскольку одинаковую книгу могут взять несколько читателей;
- избыточно дублируются сведения о читателях (ФИО, шифр группы, специальность), поскольку читатель может взять несколько книг;
- существует проблема контроля правильности данных: измененные сведения о книге или читателе приходится вносить более чем в один кортеж;

- существует проблема добавления записей: включить в таблицу кортеж о новой книге можно только, если эта книга взята; включить в таблицу кортеж о новом читателе можно только, если он взял книгу;

- существует проблема удаления записей: удаление кортежа может привести к потере сведений о книге или читателе.

Перевод отношения в следующую нормальную форму осуществляется методом «декомпозиции без потерь». Такая декомпозиция должна обеспечить то, что запросы (выборка данных по условию) к исходному отношению и к отношениям, получаемым в результате декомпозиции, дадут одинаковый результат. Основной операцией метода является операция проекции.

Чтобы устранить частичную зависимость и привести отношение во 2НФ, необходимо разложить отношение на новые отношения следующим образом:

- построить проекцию без атрибутов, которые находятся в частичной зависимости от составного ключа;

- построить проекцию на часть составного ключа и атрибуты, зависящие от этой части.

При разложении отношения Выдача книг получают три новых отношения:

1) Отношение Выдача книг

Шифр книги

Номер читательского билета

Дата выдачи

Дата возврата;

2) Отношение Книги

Авторы

Название

Год издания

3) Отношение Читатели

Номер читательского билета

ФИО

Шифр группы

Специальность

*В первом отношении ключом являются атрибуты – Шифр книги, Номер читательского билета, во втором отношении – шифр книги, в третьем отношении – номер читательского билета*

Третья нормальная форма. Отношение находится в третьей нормальной форме (3НФ), если оно находится во 2НФ и в нем отсутствуют транзитивные зависимости от не ключевых атрибутов от ключа.

*Если в отношении для атрибутов  $X, Y, Z$  имеют место функциональные зависимости  $X \rightarrow Y, Y \rightarrow Z$ , но обратная зависимость отсутствует, то атрибут  $Z$  транзитивно зависит от атрибута  $X$ .*

Рассмотрим отношение, в котором присутствуют транзитивные зависимости:

Отношение Сотрудники

Табельный номер

ФИО

Кафедра

Номер корпуса

Номер аудитории

Телефон кафедры

Должность

Категория

Оклад

Надбавка за должность.

Ключом данного отношения является Табельный номер.

В отношении Сотрудники имеют место следующие функциональные зависимости:

1) Табельный номер → ФИО, Кафедра, Номер корпуса, Номер аудитории, Телефон кафедры, Должность, Категория, Оклад, Надбавка за должность;

2) Кафедра → Номер корпуса, Номер аудитории, Телефон кафедры;

3) Должности → Надбавка за должности;

4) Категория → Оклад.

2-4 зависимости являются транзитивными.

Наличие транзитивных зависимостей порождает аномалии следующего характера:

- избыточно дублируется информация о кафедрах, должностях, категориях;
- сложное редактирование данных: изменение, например, оклада требует поиска и замены его во всех кортежах для сотрудников, имеющих одинаковую категорию;
- существует проблема удаления записей: удаление кортежа может привести к потере сведений о кафедре, должности, категории;
- существует проблема добавления записей: невозможно, например, включить сведения о новой должности, если эту должность никто не занимает.

Чтобы устранить транзитивную зависимость и привести отношение в 3НФ, необходимо:

- построить проекцию без атрибутов, находящихся в транзитивной зависимости от ключа;
- построить проекцию на не ключевые атрибуты, между которыми имеется функциональная зависимость.

В нашем примере получим:

Отношение Сотрудники

Табельный номер

ФИО

Кафедра

Должность

Категория

Отношение Кафедры

Кафедра

Номер корпуса

Номер аудитории

Номер телефона

Отношение Должности

Должность

Надбавка за должность

Отношение Категории

Категория

Оклад

В первом отношении ключ – Табельный номер, во втором – Кафедра, в третьем – Должность, в четвертом – Категория.

На практике процесс проектирования схем отношений заканчивается, когда отношения приведены к 3НФ.