



Нормализация ОТНОШЕНИЙ

*"Сложная система, спроектированная наспех,
никогда не работает, и исправить её,
чтобы заставить работать,
невозможно".*

Законы Мерфи. 16-й закон системантики

Аномалии модификации данных

При неправильно спроектированной схеме реляционной БД могут возникнуть аномалии выполнения операций модификации данных.

Рассмотрим эти аномалии на примере следующего отношения:

ПОСТАВКИ (*Номер поставки, Название товара, Цена товара, Количество, Дата поставки, Название поставщика, Адрес поставщика*)

Различают аномалии обновления, удаления и добавления.

Аномалия обновления может возникнуть в том случае, когда информация дублируется. Другие аномалии возникают тогда, когда две и более сущности объединены в одно отношение.

Например:

- **Аномалия обновления:** изменился адрес поставщика. Если от него было несколько поставок, то придется менять несколько записей.
- **Аномалия удаления:** при удалении в архив записей обо всех поставках определённого поставщика все данные об этом поставщике (название, адрес) будут утеряны.
- **Аномалия добавления:** нельзя добавить сведения о поставщике, пока от него нет ни одной поставки.

Для решения проблемы аномалии модификации данных при проектировании РБД проводится нормализация отношений.

Нормализация схемы отношения

Нормализация схемы отношения выполняется путём декомпозиции схемы. **Декомпозицией** схемы отношения R называется замена её совокупностью схем отношений A_i таких, что

$$R = \boxtimes_i A_i$$

и не требуется, чтобы отношения A_i были непересекающимися.

Декомпозиция отношения не должна приводить к потере зависимостей между атрибутами сущностей. Для декомпозиции должна существовать операция реляционной алгебры, применение которой позволит восстановить исходное отношение.

Покажем нормализацию на примере отношения КНИГИ :

Id – идентификатор (первичный ключ),

Code – шифр рубрики (по ББК – библиотечно-библиографической классификации),

Theme – название рубрики (по ББК),

Title – название книги,

Author – автор(ы),

Editor – редактор(ы),

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

Year – год издания,

Pg – количество страниц.

Пример для демонстрации нормализации

В таблице приведен пример содержимого исходного отношения КНИГИ:

Таблица 8.1. Исходное отношение КНИГИ

<u>ID</u>	<u>Code</u>	<u>Theme</u>	<u>Author</u>	<u>Title</u>	<u>Editor</u>	<u>Type</u>	<u>Year</u>	<u>Pg</u>
20	22.18	МК	Бочков С.	Язык программирования СИ	Садчиков П.	учебник	1990	384
			Субботин Д.		Седов П.			
10	22.18	МК	<u>Джехани Н.</u>	Язык АДА	Красилов А.	учебник	1988	552
					<u>Перминов О.</u>			
35	32.97	ВТ	Соловьев Г.	Операционные системы ЭВМ		учебное пособие	1992	208
			Никитин В.					
11	32.81	<u>Кибернетика</u>	Попов Э.В.	Общение с ЭВМ на естественном языке	Некрасов А.	учебник	1982	360
44	32.97	ВТ		ПУ для ПЭВМ	<u>Витенберг Э.</u>	<u>справочник</u>	1992	208
89	32.973	ЭВМ	<u>Коутс Р.</u>	Интерфейс «человек-компьютер»	Шаньгин В.	учебник	1990	501
			<u>Влейминк И.</u>					

Первая нормальная форма

Введём понятие простого и сложного атрибута:

Простой атрибут – это атрибут, значения которого атомарны (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 (1НФ).

Отношение приведено к 1НФ, если все его атрибуты простые.

Приведение к 1НФ

Таблица 8.2. Отношение КНИГИ, приведённое к 1НФ

<u>ID</u>	<u>Code</u>	<u>Theme</u>	<u>Author</u>	<u>Title</u>	<u>Editor</u>	<u>Type</u>	<u>Year</u>	<u>Pg</u>
20	22.18	МК	Бочков С.	Язык программирования СИ	Садчиков П.	учебник	1990	384
20	22.18	МК	Субботин Д.	Язык программирования СИ	Седов П.	учебник	1990	384
10	22.18	МК	<u>Джехани Н.</u>	Язык АДА	Красилов А.	учебник	1988	552
10	22.18	МК	<u>Джехани Н.</u>	Язык АДА	<u>Перминов О.</u>	учебник	1988	552
35	32.97	ВТ	Соловьев Г.	Операционные системы ЭВМ		учебное пособие	1992	208
35	32.97	ВТ	Никитин В.	Операционные системы ЭВМ		учебное пособие	1992	208
11	32.81	<u>Кибер-нетика</u>	Попов Э.В.	Общение с ЭВМ на естественном языке	Некрасов А.	учебник	1982	360
44	32.97	ВТ		ПУ для ПЭВМ	<u>Витенберг Э.</u>	<u>справочник</u>	1992	208
89	32.973	ЭВМ	<u>Коутс Р.</u>	Интерфейс «человек-компьютер»	Шаньгин В.	учебник	1990	501
89	32.973	ЭВМ	<u>Влейминк И.</u>	Интерфейс «человек-компьютер»	Шаньгин В.	учебник	1990	501

Вторая нормальная форма

Введём понятие **функциональной зависимости**. Пусть X и Y – атрибуты (группы атрибутов) некоторого отношения. Говорят, что Y функционально зависит от X , если в любой момент времени каждому значению $X=x$ соответствует единственное значение $Y=y$ ($X \rightarrow Y$). (При этом любому значению $Y=y$ может соответствовать несколько значений $X=(x_1, x_2, \dots)$).

Атрибут X в функциональной зависимости $X \rightarrow Y$ называется **детерминантом** отношения.

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.

Вторая нормальная форма (2НФ).

Отношение находится во 2НФ, если оно приведено к 1НФ и каждый неключевой атрибут функционально полно зависит от составного ключа.

Для того чтобы привести отношение ко 2НФ, нужно:

1. построить его проекцию, исключив атрибуты, которые не находятся в функционально полной зависимости от составного ключа;
2. построить дополнительные проекции на часть составного ключа и атрибуты, функционально зависящие от этой части ключа.

Приведение к 2НФ

Таблица 8.3. Отношение *КНИГИ*, приведённое к 2НФ

<u>ID</u>	<u>Code</u>	<u>Theme</u>	<u>Title</u>	<u>Type</u>	<u>Year</u>	<u>Pg</u>
20	22.18	МК	Язык программирования СИ	учебник	1990	384
10	22.18	МК	Язык АДА	учебник	1988	552
35	32.97	ВТ	Операционные системы ЭВМ	учебное пособие	1992	208
11	32.81	Кибернетика	Общение с ЭВМ на естественном языке	учебник	1982	360
44	32.97	ВТ	ПУ для ПЭВМ	справочник	1992	208
89	32.973	ЭВМ	Интерфейс «человек-компьютер»	учебник	1990	501

Таблица 8.4. Отношение *КНИГИ–АВТОРЫ–РЕДАКТОРЫ* (2НФ)

<u>ID</u>	<u>Author</u>	<u>Editor</u>
20	Бочков С.	Садчиков П.
20	Субботин Д.	Седов П.
10	<u>Джихани Н.</u>	Красилов А.
10		<u>Перминов О.</u>
35	Соловьев Г.	
35	Никитин В.	
11	Попов Э.В.	Некрасов А.
44		<u>Витенберг Э.</u>
89	<u>Коутс Р.</u>	Шаньгин В.
89	<u>Влейминк И.</u>	

Третья нормальная форма

Рассмотрим понятие **транзитивной зависимости**.

Пусть X, Y, Z – атрибуты некоторого отношения. При этом $X \rightarrow Y$ и $Y \rightarrow Z$, 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 ($X \twoheadrightarrow Z$).

Третья нормальная форма (3НФ).

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

Исключение:

если для атрибутов X, Y, Z есть транзитивная зависимость $X \twoheadrightarrow Z$, и при этом $Y \rightarrow X$ или $Z \rightarrow Y$, то такая зависимость не требует декомпозиции отношения.

Например, для отношения АВТОМОБИЛИ с первичным ключом Государственный номерной знак и полями № кузова и № двигателя очевидно, что номера кузова и двигателя зависят как друг от друга, так и от первичного ключа. Но эта зависимость взаимно однозначная, поэтому декомпозиция отношения не нужна.

Приведение к 3НФ

Таблица 8.5. Отношения КНИГИ, приведённые к 3НФ

<u>ID</u>	<u>Code</u>	<u>Title</u>	<u>Type</u>	<u>Year</u>	<u>Pg</u>
20	22.18	Язык программирования СИ	учебник	1990	384
10	22.18	Язык АДА	учебник	1988	552
35	32.97	Операционные системы ЭВМ	учебное пособие	1992	208
11	32.81	Общение с ЭВМ на естественном языке	учебник	1982	360
44	32.97	ПУ для ПЭВМ	справочник	1992	208
89	32.973	Интерфейс «человек-компьютер»	учебник	1990	501

Табл. 8.6. Отношение РУБРИКАТОР

<u>Code</u>	<u>Theme</u>
22.18	МК
32.97	ВТ
32.973	ЭВМ
32.81	Кибернетика

Отношения, находящиеся в 3НФ, свободны от аномалий модификации.

Четвертая нормальная форма

Введём понятие **многозначной зависимости**.

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

Различают тривиальные и нетривиальные многозначные зависимости. *Тривиальной* называется многозначная зависимость $X \twoheadrightarrow Y$, для которой $Y \subset X$ или $X \cup Y = R$, где R – рассматриваемое отношение.

Тривиальная многозначная зависимость не нарушает 4НФ. Если хотя бы одно из двух этих условий не выполняется (т.е. Y не является подмножеством X или $X \cup Y$ состоит не из всех атрибутов R), то такая многозначная зависимость называется *нетривиальной*.

Четвертая нормальная форма (4НФ).

Отношение находится в 4НФ, если оно находится в 3НФ и в нем отсутствуют нетривиальные многозначные зависимости.

Приведение к 4НФ

Таблица 8.7. Отношение КНИГИ-АВТОРЫ (4НФ)

<u>ID</u>	<u>Author</u>
200	Бочков С.
200	Субботин Д.
100	Джихани Н.
300	Крон Г.
876	Гик Е.Я.
385	Фролов Г.
385	Кузнецов Э.

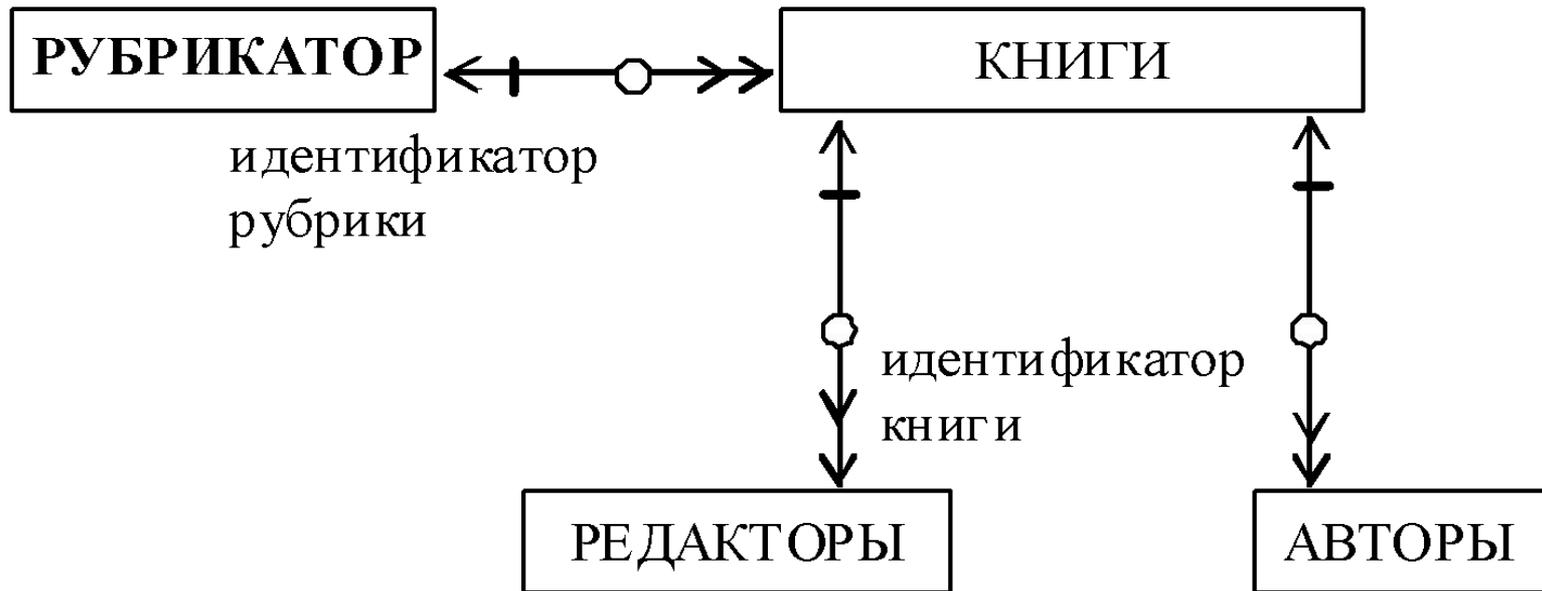
Таблица 8.8. Отношение КНИГИ-РЕДАКТОРЫ (4НФ)

<u>ID</u>	<u>Editor</u>
200	Садчиков П.
300	Баранов А.
876	Кикоин И.
876	Капица С.
440	Витенберг Э.
385	Храмов А.
385	Рожков П.

В отношениях, полученных после приведения к 4НФ, первичный ключ (ПК) состоит из всех атрибутов отношения.

Нормализация сокращает дублирование данных, но появление новых отношений усложняет схему базы данных.

Схема после нормализации (до 4НФ)



Усиленная ЗНФ

Нормальная форма Бойса-Кодда (НФБК):

Все детерминанты отношения должны быть его потенциальными ключами.

Покажем нарушение НФБК на примере отношения "Переговоры" (Interview):

IdClient – номер клиента,

IdStaff – номер сотрудника,

Room – номер комнаты для переговоров,

Vdate – дата встречи,

Vtime – время встречи.



Анализ на соответствие НФБК

Данные в этой таблице должны удовлетворять следующим правилам:

1. С каждым клиентом в день происходит не более одной встречи.
2. В каждый момент времени в одной комнате происходит только одна встреча.
3. Каждый сотрудник встречается с клиентом один на один.
4. На один день сотруднику выделяется одна комната для переговоров.

В соответствие с правилами в отношении "Работа с клиентами" (Interview) есть 4 детерминанта:

fd1: IdClient, Vdate → IdStaff, Room, Vtime.

fd2: Room, Vdate, Vtime → IdStaff, IdClient.

fd3: IdStaff, Vdate, Vtime → IdClient, Room.

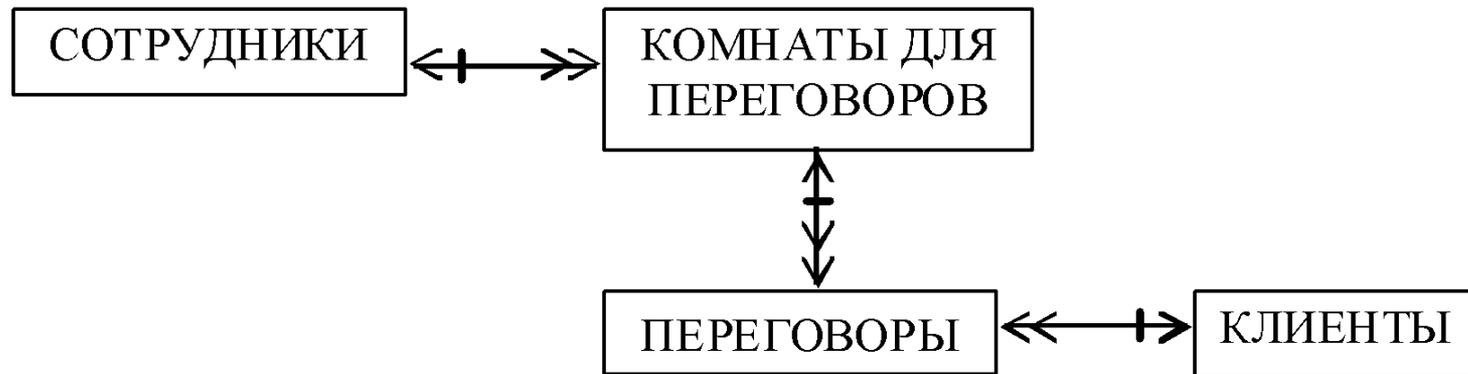
fd4: IdStaff, Vdate → Room.

Вопрос: все ли детерминанты являются потенциальными ключами?

Первые три являются потенциальными ключами, а fd4 – нет. Это может привести к аномалии обновления данных при смене номера комнаты.

Приведение к НФБК

Приведение к НФБК в данном случае выполняется путём декомпозиции на 2 отношения: "Комнаты для переговоров" (Id, IdStaff, Vdate, Room), где Id – суррогатный ПК, и "Переговоры" (Id, IdClient, Vtime), где Id – внешний ключ.



Но такая декомпозиция обычно приводит к потере функциональных зависимостей (в нашем случае – fd1, fd2, fd3), поэтому в реальных системах этой нормальной формой пользуются редко.

Проективно-соединительная нормальная форма (5НФ)

5НФ относится к ситуации нескольких отношений, связанных друг с другом связью 1:n или n:m.

Например:

Белов ведет:
в А-11 – ДМ,
в А-12 – МА;

Осипов ведет:
в А-12 – ДМ,
в А-11 – МА.

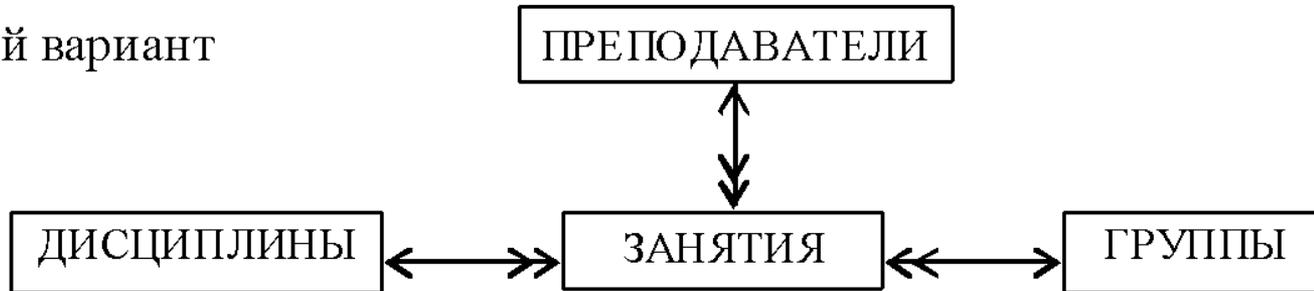
Таблицы не содержат неверных данных, но при соединении таблиц могут выдаваться "лишние" строки.



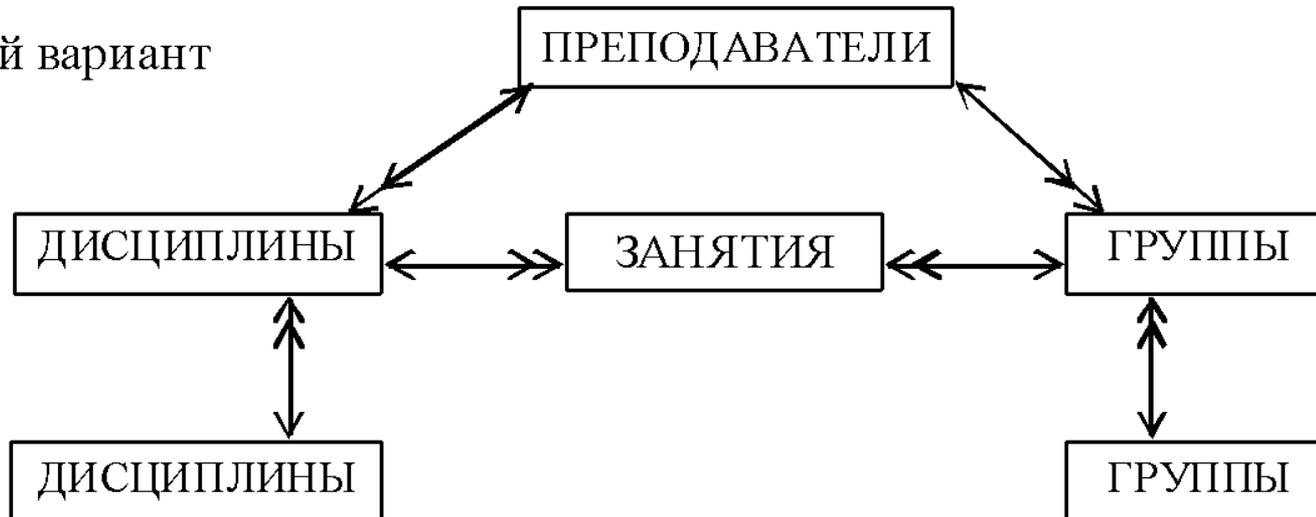
Приведение к 5НФ

Преобразование схемы БД:

1-й вариант



2-й вариант



Денормализация отношений

Обоснованием денормализации может служить требование обеспечения определённой производительности для критических запросов.

Денормализация бывает нескольких видов:

1. Восходящая.

Подразумевает перенос некоторой информации из подчинённого отношения в родительское.

2. Нисходящая.

В этом случае информация переносится из родительского отношения в подчинённое.

3. Разбиение одного отношения на два.

Обычно применяется в случае, когда запись имеет большую длину за счёт наличия атрибутов большого объёма (графические данные, текстовые описания и проч.). Если данные этих атрибутов редко используются, то можно выделить в отдельное отношение атрибуты большого объёма.

Для связи с исходным отношением вводится уникальный внешний ключ. А для получения исходного отношения создаётся представление (view), которое является соединением двух полученных отношений.