

# Нормализация баз данных

**Первая нормальная форма (1NF)** — базовая нормальная форма отношения в реляционной модели данных.

Переменная отношения находится в первой нормальной форме тогда и только тогда, когда в любом допустимом значении отношения каждый его кортеж содержит только одно значение для каждого из атрибутов.

В реляционной модели отношение всегда находится в первой нормальной форме по определению понятия *отношение*.

Что же касается различных *таблиц*, то они могут не быть правильными представлениями отношений и, соответственно, могут не находиться в 1NF. В соответствии с определением К. Дж. Дейта для такого случая, таблица нормализована (эквивалентно — находится в первой нормальной форме) тогда и только тогда, когда она является прямым и верным представлением некоторого отношения. Конкретнее, рассматриваемая таблица должна удовлетворять следующим пяти условиям:

1. Нет упорядочивания строк сверху-вниз (другими словами, порядок строк не несет в себе никакой информации).
2. Нет упорядочивания столбцов слева-направо (другими словами, порядок столбцов не несет в себе никакой информации).
3. Нет повторяющихся строк.
4. Каждое пересечение строки и столбца содержит ровно одно значение из соответствующего домена (и больше ничего).
5. Все столбцы являются обычными

«Обычность» всех столбцов таблицы означает, что в таблице нет «скрытых» компонентов, которые могут быть доступны только в вызове некоторого специального оператора взамен ссылок на имена регулярных столбцов, или которые приводят к побочным эффектам для строк или таблиц при вызове стандартных операторов. Таким образом, например, строки не имеют идентификаторов кроме обычных значений потенциальных ключей (без скрытых «идентификаторов строк» или «идентификаторов объектов»). Они также не имеют скрытых временных меток

# Нормализация баз данных

## Пример приведения к 1NF

Исходная ненормализованная (то есть не являющаяся правильным представлением некоторого отношения) таблица:

Сотрудник	Номер телефона
Иванов И.И.	283-56-82 390-57-34
Петров П.П.	708-62-34

Таблица, приведённая к 1NF (являющаяся правильным представлением некоторого отношения):

Сотрудник Фамилия	Сотрудник Имя	Сотрудник Отчество	Номер телефона 1	Номер телефона 2
Иванов	Иван	Иванович	283-56-82	390-57-34
Петров	Петр	Петрович	708-62-34	

Таблица, находящаяся в 1 НФ обладает следующими недостатками:

- добавление новых данных требует ввода для всех столбцов, даже если в таблице уже существуют необходимые данные, то есть присутствует избыточность данных, а это всегда приводит в ошибкам ввода (появлению 2 разных значений одного поля);
- если какие либо данные изменяются, то необходимо просматривать всю таблицу и вносить эти изменения во всех строках, где она встречается;
- при удалении одной строки в БД может быть потеряна важная для всей обработки информация.

## **Вторая нормальная форма (2NF)**

Переменная отношения находится во второй нормальной форме тогда и только тогда, когда она находится в первой нормальной форме и каждый неключевой атрибут неприводимо (функционально полно) зависит от ее потенциального (первичного) ключа.

Таблица, находящаяся во 2НФ, должна удовлетворять следующим правилам:

- должна содержать данные об одном типе объектов;
- каждая таблица должна содержать одно или несколько полей, образующих уникальный идентификатор (или первичный ключ) для каждой строки;
- все поля, не имеющие ключа должны определяться полным уникальным идентификатором данной таблицы.

Если таблица имеет простой первичный ключ, состоящий только из одного столбца, то она автоматически находится во второй нормальной форме. Если же первичный ключ составной, то таблица необязательно находится во второй нормальной форме и тогда ее необходимо разделить на 2 или более таблиц таким образом, чтобы первичный ключ однозначно идентифицировал значение в любом столбце.

# Вторая нормальная форма: пример



## **Третья нормальная форма (3NF)**

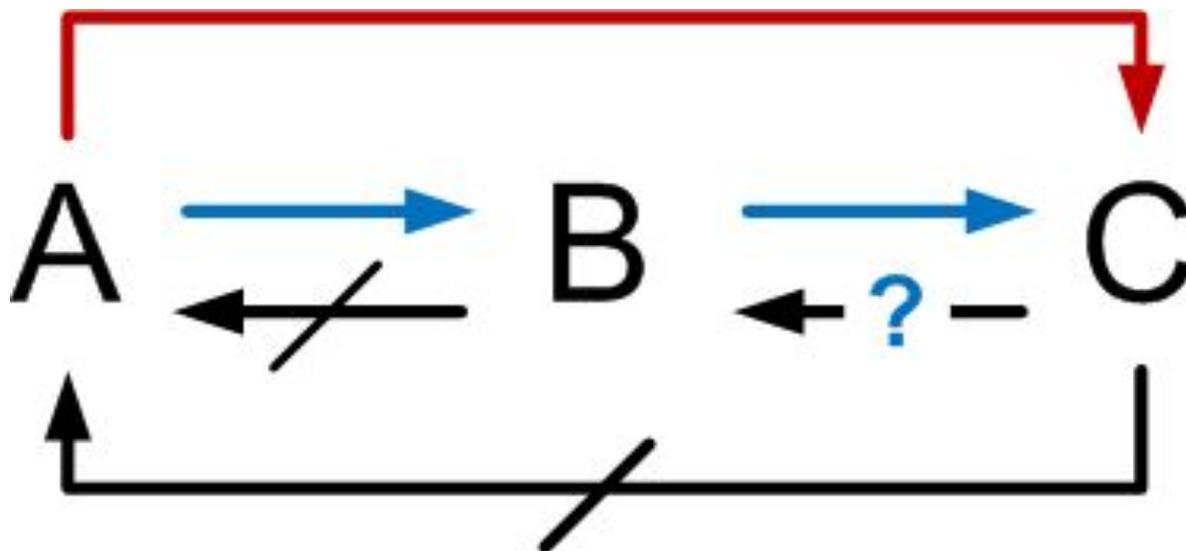
Переменная отношения находится в третьей нормальной форме тогда и только тогда, когда она находится во второй нормальной форме и отсутствуют транзитивные функциональные зависимости неключевых атрибутов от ключевых.

# Транзитивная зависимость

## Транзитивная функциональная зависимость:

Пусть A, B, C – три атрибута некоторого отношения R.

Схема транзитивной зависимости:



# Третья нормальная форма:

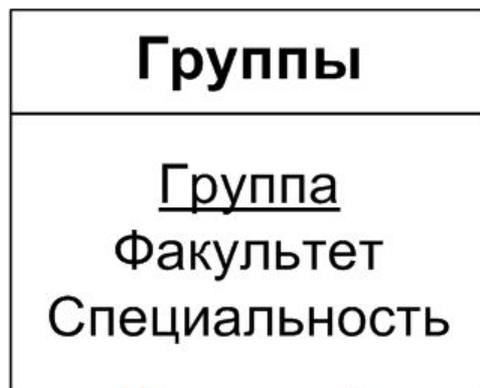
**Транзитивная**

**зависимость:**

Группа  $\rightarrow$  Специальность

Специальность  $\rightarrow$  Факультет

2 НФ



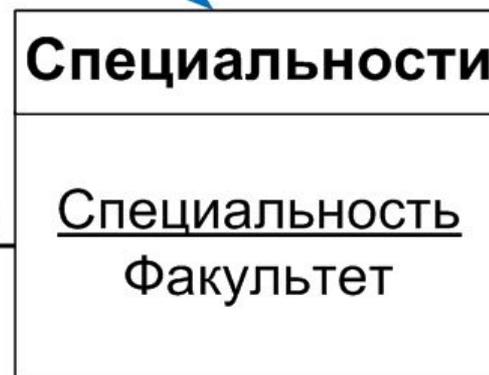
Проекция

3 НФ



$\infty$

1



Т.о. Таблица находится в 3НФ, если она удовлетворяет определению 2НФ и ни одно из ее не ключевых полей функционально не зависит от любого другого не ключевого поля. Можно сказать, что таблица находится в 3НФ, если она находится во 2НФ и каждое не ключевое поле не транзитивно зависит от первичного ключа.

Требование 3НФ сводится к тому, чтобы все не ключевые поля зависели только от первичного ключа и не зависели друг от друга. Должна существовать возможность изменять значение любого не ключевого поля, не изменяя значения любого другого поля БД. Это требование исключает любое поле, значения в котором получаются как результат вычислений, использующий значения других полей.

## Нормальная форма Бойса — Кодда

**Нормальная форма Бойса-Кодда** (англ. *Boyce-Codd normal form*; сокращённо **BCNF**) — одна из возможных нормальных форм отношения в реляционной модели данных.

Иногда нормальную форму Бойса-Кодда называют *усиленной третьей нормальной формой*, поскольку она во всех отношениях сильнее (строже) по сравнению с ранее определённой ЗНФ<sup>[1]</sup>.

Названа в честь Рэя Бойса и Эдгара Кодда, хотя Кристофер Дейт указывает, что на самом деле строгое определение «третьей» нормальной формы, эквивалентное определению нормальной формы Бойса-Кодда, впервые было дано Иэном Хитом (англ. *Ian Heath*) в 1971 году, поэтому данную форму следовало бы называть «нормальной формой Хита»<sup>[</sup>

## **Четвертая нормальная форма**

При использовании 1НФ запрещается таблицам иметь неатомарные, или многозначные, атрибуты. Однако на практике существует множество ситуаций моделирования, требующих многозначных атрибутов. Например, в университете один преподаватель проводит нескольких дисциплин. Таблица имеет 4НФ, если она имеет 3НФ и не содержит многозначных зависимостей. Так как проблема многозначных зависимостей возникает в связи с многозначными атрибутами, то можно решить ее, поместив каждый многозначный атрибут в свою собственную таблицу вместе с ключом, от которого атрибут зависит.

## **Пятая нормальная форма. Нормальная форма область/ключ.**

5НФ была предложена для исключения аномалии, связанные с особым типом ограничительных условий называемых совместными зависимостями. Эта форма не имеет практического применения.

НФОК была предложено в 1981 г., основана на определениях ключей и областей атрибутов. Таблица имеет НФОК если любое ограничительное условие является следствием определений областей и ключей. Однако, общий метод приведения таблицы к НФОК не разработан.

## **Процессы, протекающие на протяжении жизненного цикла информационной системы**

Жизненный цикл информационно системы представляет собой непрерывный процесс, начинающийся с момента принятия решения о создании информационной системы и заканчивается в момент полного изъятия ее из эксплуатации.

Существует международный стандарт, регламентирующий жизненный цикл информационных систем — ISO/IEC 12207.

ISO — International Organization of Standardization (международная организация по стандартизации). IEC — International Electrotechnical Commission (международная комиссия по электротехнике).

Стандарт ISO/IEC 12207 определяет структуру жизненного цикла, содержащую процессы, действия и задачи, которые должны быть выполнены во время создания информационной системы. Согласно данному стандарту структура жизненную цикла основывается на трех группах процессов:

- основные процессы жизненного цикла (приобретение, поставка, разработки, эксплуатация, сопровождение);
- вспомогательные процессы, обеспечивающие выполнение основных процессов (документирование, управление конфигурацией, обеспечение качества, аттестация, оценка, аудит, разрешение проблем);
- организационные процессы (управление проектами, создание инфраструктуры проекта, определение, оценка и улучшение самого жизненного цикла, обучение).

# Основные процессы жизненного цикла

## Разработка

Разработка информационной системы включает в себя все работы по созданию информационного программного обеспечения и его компонентов в соответствии с заданными требованиями. Разработка информационного программного обеспечения также включает:

- оформление проектной и эксплуатационной документации;
- подготовку материалов, необходимых для проведения тестирования разработанных программных продуктов;
- разработку материалов, необходимых для организации обучения персонала.

Разработка является одним из важнейших процессов жизненного цикла информационной системы и, как правило, включает в себя стратегическое планирование, анализ, проектирование и реализацию (программирование).

## Эксплуатация

Эксплуатационные работы можно подразделить на подготовительные и основные. К подготовительным относятся:

- конфигурирование базы данных и рабочих мест пользователей;
- обеспечение пользователей эксплуатационной документацией;
- обучение персонала.

Основные эксплуатационные работы включают:

- непосредственно эксплуатацию;
- локализацию проблем и устранение причин их возникновения;
- модификацию программного обеспечения;
- подготовку предложений по совершенствованию системы;
- развитие и модернизацию системы.

## Сопровождение

Службы технической поддержки играют весьма заметную роль в жизни любой корпоративной информационной системы. Наличие квалифицированного технического обслуживания на этапе эксплуатации информационной системы является необходимым условием для решения поставленных перед ней задач, причем ошибки обслуживающего персонала могут приводить к явным или скрытым финансовым потерям, сопоставимым со стоимостью самой информационной системы.

Основными предварительными действиями при подготовке к организации технического обслуживания информационного системы являются следующие:

- выделение наиболее ответственных узлов системы. Это позволит выделить наиболее критичные составляющие информационной системы и оптимизировать распределение ресурсов для технического обслуживания;
- определение задач технического обслуживания и их разделение на внутренние (решаемые силами обслуживающего подразделения) и внешние (решаемые специализированными сервисными организациями). Таким образом, производится четкое определение круга исполняемых функций и разделение ответственности;
- проведение анализа имеющихся внутренних и внешних ресурсов, необходимых для организации технического обслуживания в рамках описанных задач и разделения компетенции. Основные критерии для анализа: наличие гарантии на оборудование, состояние ремонтного фонда, квалификация персонала;
- подготовка плана организации технического обслуживания, в котором необходимо определить этапы исполняемых действий, сроки их исполнения, затраты на этапах, ответственность исполнителей.

Обеспечение качественного технического обслуживания информационной системы требует привлечения специалистов высокой квалификации, которые в состоянии решать не только каждодневные задачи администрирования, но и быстро восстанавливать работоспособность системы при сбоях.

## **Вспомогательные процессы**

Среди вспомогательных процессов одно из главных мест занимает управление конфигурацией. Это один из вспомогательных процессов, поддерживающих основные процессы жизненного цикла информационной системы, прежде всего процессы разработки и сопровождения. При разработке проектов сложных информационных систем, состоящих из многих компонентов, каждый из которых может разрабатываться независимо и, следовательно, иметь несколько вариантов реализации и/или несколько версий одной реализации, возникает проблема учета их связей и функций, создания единой структуры и обеспечения развития всей системы. Управление конфигурацией позволяет организовывать, систематически учитывать и контролировать внесение изменений в различные компоненты информационной системы на всех стадиях ее жизненного цикла.

## Организационные процессы

Управление проектом связано с вопросами планирования и организации работ, создания коллективов разработчиков и контроля за сроками и качеством выполняемых работ. Техническое и организационное обеспечение проекта включает:

- выбор методов и инструментальных средств для реализации проекта;
- определение методов описания промежуточных состояний разработки;
- разработку методов и средств испытаний созданного программного обеспечения;
- обучение персонала.

Обеспечение качества проекта связано с проблемами верификации, проверки и тестирования компонентов информационной системы. **Верификация** — это процесс определения соответствия текущего состояния разработки, достигнутого на данном этапе, требованиям этого этапа. **Проверка** — это процесс определения соответствия параметров разработки исходным требованиям. Проверка отчасти совпадает с тестированием, которое проводится для определения различий между действительными и ожидавшимися результатами и оценки соответствия характеристик информационной системы исходным требованиям.

## *Модели жизненного цикла информационной системы*

Моделью жизненного цикла информационной системы будем называть некоторую структуру, определяющую последовательность осуществления процессов, действий и задач, выполняемых на протяжении жизненного цикла информационной системы, а также взаимосвязи между этими процессами, действиями и задачами.

В стандарте ISO/IEC 12207 не конкретизируются в деталях методы реализации и выполнения действий и задач, входящих в процессы жизненного цикла информационной системы, а лишь описываются структуры этих процессов. Это вполне понятно, так как регламенты стандарта являются общими для любых моделей жизненного цикла, методологий и технологий разработки. Модель же жизненного цикла зависит от специфики информационной системы и условий, в которых она создается и функционирует. Поэтому не имеет смысла предлагать какие-либо конкретные модели жизненного цикла и методы разработки информационных систем для общего случая, без привязки к определенной предметной области.

К настоящему времени наибольшее распространение получили следующие две основные модели жизненного цикла:

- каскадная модель, иногда также называемая моделью «водопад» (waterfall);
- спиральная модель.



Водопадная модель процесса разработки

