

Нормализация баз данных

4 модуль

Нормализация

Создание набора отношений на основе некоторых требований

Основная задача: Найти наилучшую группировку атрибутов по отношениям в данной схеме

Процесс нормализации

Идентификация отношений по:

- Первичному (или потенциальному) ключу
- Функциональной зависимости

Проверка отдельных отношений на соответствие правилам нужной нормальной формы.

Нормальные формы

- ННФ - ненормальная форма - без требований
- 1НФ - первая нормальная форма - обязательное требование реляционных БД
- 2НФ, 3НФ, НФБК - устранение часто возникающих аномалий
- 4НФ - решение проблем с многозначной зависимостью
- 5НФ - решение проблем с зависимостями соединения

Функциональная зависимость

Описывает связь между атрибутами отношения

Если в отношении один атрибут функционально зависит от другого, то каждое значение первого атрибута связано только с одним значением второго.

Детерминант

Атрибут, от которого полностью функционально зависит некоторый другой атрибут.

Первая нормальная форма

Ненормализованная форма - отношение содержит повторяющиеся группы данных в одной ячейке.

Первая нормальная форма (1НФ) - отношение, в котором на пересечении строки и столбца содержится только одно значение

Нормализация

Цель нормализации: проверка корректности модели для конкретной нормальной формы.

Конкретная нормальная форма дает защиту от дублирования данных и возникновения аномалий.

Функциональная зависимость

Отношение содержит атрибуты A и B :

Каждое значение A связано только с одним значением атрибута B :

B функционально зависит от A ($A \rightarrow B$)

Атрибуты A и B могут быть составными.

Детерминант

Атрибут или группа атрибутов в функциональной зависимости, которые являются независимыми в паре.

Процесс нормализации

1. Проверка каждого отношения на требования нормальной формы.
2. Выполнение действий нормализации, если необходимо.

Каждая последующая нормальная форма накладывает еще большие ограничения.

Первая нормальная форма

Ненормализованная форма - отношение содержит повторяющиеся группы данных в одной ячейке.

Первая нормальная форма (1НФ) - отношение, в котором на пересечении строки и столбца содержится только одно значение

Первая нормальная форма (пример)

Любые списки вместо одиночных значений

Задачи

Задача	Исполнитель
Нормализовать БД	Иванов, Петров
Составить ER-модель	Иванов

Первая нормальная форма (решения)

1. Декомпозировать записи продублировав остальные данные:

Задача	Исполнитель
Нормализовать БД	Иванов
Нормализовать БД	Петров
Составить ER-модель	Иванов

Первая нормальная форма (решения)

Выделить списки в отдельную сущность:

Задача

ID	Название
1	Нормализовать БД
2	Составить ER-модель

Исполнитель

ID	Имя
1	Иванов
2	Петров

ЗадачиИсполнители

ид_задачи	ид_исполнителя
1	1
1	2
2	1

Полная функциональная зависимость

В отношении атрибут B полностью функционально зависит от атрибута A , если:

- B функционально зависит от полного значения A (составные атрибуты)
- B не зависит от подмножества полного значения A

Вторая нормальная форма

1. Отношение находится в 1НФ.
2. Каждый атрибут, не входящий в первичный ключ, полностью функционально зависит от первичного ключа.

Вторая нормальная форма

Если у вас первичный ключ только один - *id*, то отношение уже в 2НФ.

Составной первичный ключ возникает при декомпозиции в 1НФ связей многие-ко-многим.

Система управления проектами

Отношение, получившееся после декомпозиции в 1НФ, чуть дополненное.
Отношение, которое связывает Задачи и Исполнителей:

ЗадачиИсполнители

- ид
- ид_задачи
- ид_исполнителя
- название_роли
- может_ли_делегировать
- дата_назначения_задачи
- дата_создания_задачи

Ключи

Ключевые атрибуты:

- ид_задачи и ид_исполнителя
- ид

Функциональные зависимости

Зависимости:

- **название_роли** от **ид_исполнителя** и **ид_задачи** (считаем, что исполнитель может иметь разные роли в разных задачах)
- **дата_назначения_задачи** от **ид_задачи** и **ид_исполнителя**
- **дата_создания_задачи** от **ид_задачи**
- **дата_назначения_задачи** также зависит и от другого ключа **ид**.
- **название_роли** тоже зависит и от обычного **ид**.

Функциональные зависимости

может_ли_делегировать полностью функционально зависит от *ключа*, но *транзитивно*:

полностью зависит от **название_роли**, которая в свою очередь зависит от полного ключа: **ид_задачи** и **ид_исполнителя**.

Не 2НФ

Атрибут **дата_создания_задачи** не полностью функционально зависит от ключевого атрибута, так как не зависит от подмножества состоящего из **ид_исполнителя**.

После преобразований

ЗадачиИсполнители:

- ид
- ид_задачи
- ид_исполнителя
- название_роли
- может_ли_делегировать
- дата_назначения_задачи

Задачи:

- ид
- ...
- дата_создания_задачи

Транзитивная зависимость

Если для атрибутов A , B и C существуют зависимости:

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

значит C транзитивно зависит от A через B .

* Если A функционально не зависит от B и C .

Третья нормальная форма

1. Находится во 2НФ.
2. Нет атрибутов, не входящих в первичный ключ, которые находятся в транзитивной зависимости от первичного ключа.

Все атрибуты зависят только от первичного ключа напрямую.

Третья нормальная форма

3НФ: Любой неключевой атрибут зависит непосредственно от ключа. + 2НФ (то есть зависит полностью).

Теперь и транзитивные зависимости нужно убрать.

может_ли_делегировать нужно вынести в сущность **Роль**, добавив туда первичный ключ.

После преобразований

ЗадачиИсполнители:

- ид
- ид_задачи
- ид_исполнителя
- ид_роли
- дата_назначения_задачи

Роль:

- ид_роли
- название_роли
- может_ли_делегировать

Нормальная форма Бойса-Кодда

1. Находится в 3НФ
2. Каждый детерминант является потенциальным ключом.

Найти все детерминанты и убедиться, что они являются потенциальными ключами, то есть однозначно определяют запись в таблице.

Если есть непотенциальные ключи, то вынести в отдельную сущность.

Нормальная форма Бойса-Кодда

НФБК: Убирать лишние ключи: в таблице должен остаться только один первичный ключ.

Убираем ид, так как все зависимости для обоих первичных ключей.

А **ид_задачи** и **ид_исполнителя** мы не можем выкинуть, потому что они также являются внешними ключами к другим сущностям.

После преобразований

ЗадачиИсполнители:

- ид_задачи
- ид_исполнителя
- ид_роли
- дата_назначения_задачи

Многозначная зависимость

Для каждого значения атрибута A

есть наборы значений атрибутов B и C

но входящие в эти наборы значения не зависят от B и C .

Тривиальная многозначная зависимость - атрибут B является подмножеством A или $A \cup B$ - все множество отношения.

Нетривиальная - не выполняются условия.

Четвертая нормальная форма

Отношение из базы магазина, который занимается реселлингом.

R1:

- поставщик
- заказчик
- магазин

Свойства отношения

- Атрибут **поставщик** не зависит от магазина, так как **поставщик** зависит от **заказчика**.
- **Заказчик** может быть привязан к любому **магазину** и любому **поставщику**.
- Первичный ключ - все три атрибута и больше атрибутов нет, поэтому отношение уже в НФБК.

Проблема обновления

Если нужно добавить к одному заказчику нового поставщика, то это нужно сделать для всех магазинов.

Если нужно удалить поставщика, то это тоже нужно сделать из всех магазинов, и это может привести к удалению заказчика (если он работал только с одним поставщиком).

Прийти к 4НФ

Не должно быть нетривиальных многозначных зависимостей.

или

Может быть многозначная зависимость, но она должна быть единственной (то есть быть функциональной) и отношение должно быть в НФБК (то есть должен быть один потенциальный ключ).

После декомпозиции

R1

- поставщик
- магазин

R2

- магазин
- заказчик