

КАРАГАНДИНСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ АБЫЛКАСА САГИНОВА

Кафедра ИВС

Функциональные зависимости

Определение, вывод и замыкание множеств

Дисциплина: Проектирование и разработка приложений БД
ОП: «Информационные системы», «IT-медицина»

Авторы: ст.преп. Саданова Б.М., Олейникова А.В.

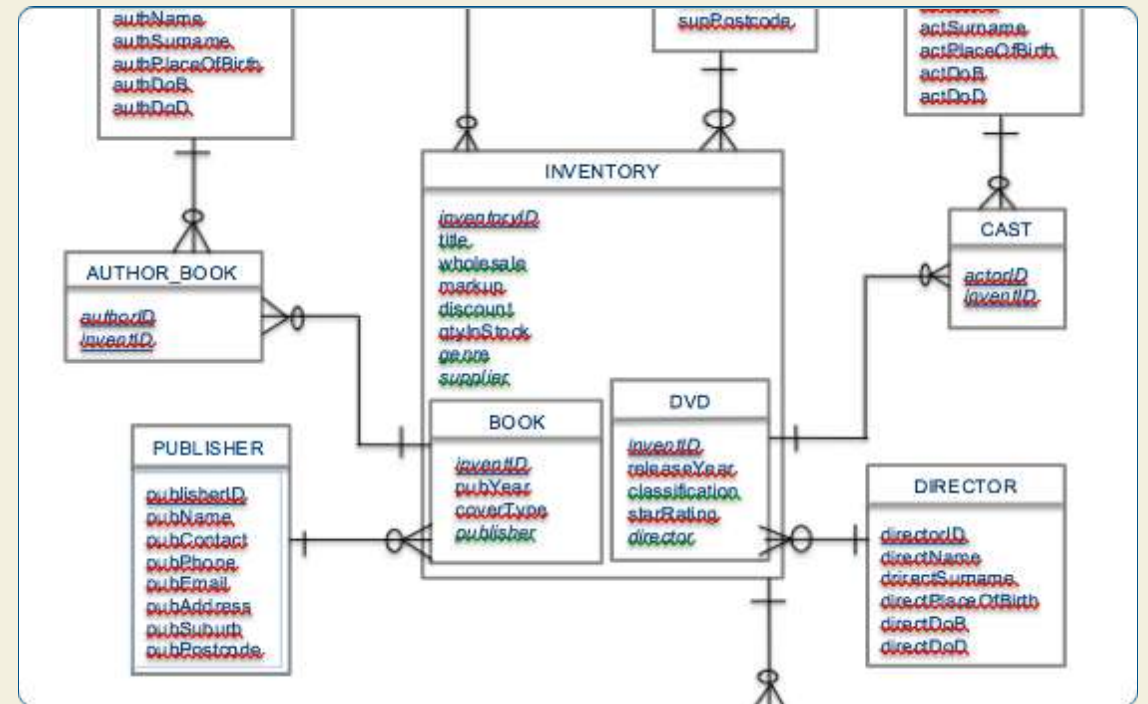


Функциональная зависимость: Концепция

Функциональная зависимость (ФЗ) — это утверждение о связи между атрибутами отношения. Она выражает бизнес-правило предметной области.

Если атрибут B функционально зависит от A , то для любого значения A в базе данных существует ровно одно соответствующее значение B в текущий момент времени.

Интуиция: ФЗ подобна математической функции $y = f(x)$, где аргумент x однозначно определяет результат y .



Математическое Определение ФЗ

Пусть дана схема отношения R и подмножества атрибутов $X, Y \subseteq R$

Говорят, что X функционально определяет Y ($X \rightarrow Y$), если для любого отношения $r(R)$ и любых кортежей $t_1, t_2 \in r$ выполняется условие:

$$t_1[X] = t_2[X] \Downarrow t_1[Y] = t_2[Y]$$

Важное примечание: ФЗ должна соблюдаться для *всех* возможных состояний отношения (инстансов), разрешенных правилами бизнеса, а не только для текущего набора данных.

Классификация зависимостей



Тривиальные ФЗ

Зависимость $X \rightarrow Y$ тривиальна, если $Y \subseteq X$

Они всегда истинны и не несут полезной информации для проектирования (например, $\{ \text{ID}, \text{Name} \} \rightarrow \text{Name}$).



Нетривиальные ФЗ

Зависимость $X \rightarrow Y$ нетривиальна, если Y содержит хотя бы один атрибут, не входящий в X

Полностью нетривиальна, если $X \cap Y = \emptyset$.

Полная и Частичная зависимость

Эти понятия критически важны для понимания 2-й нормальной формы (2NF).

Полная ФЗ: Атрибут зависит от всего составного ключа целиком и не зависит ни от какой его части.

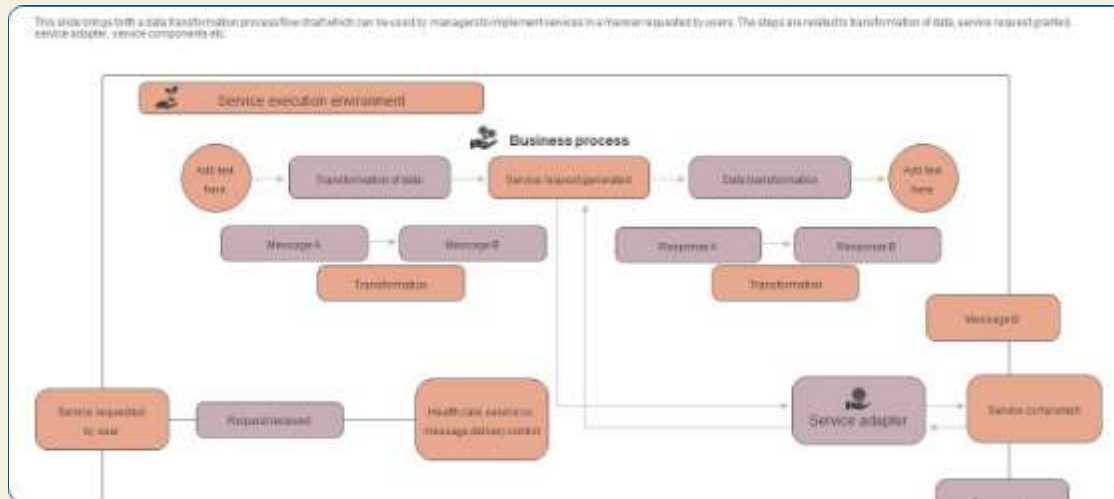
Частичная ФЗ: Атрибут зависит только от части составного ключа.

Пример: Студент-Курс

Схема: {Студент_ID, Курс_ID, Оценка, Кафедра_Курса}

- {Студент_ID, Курс_ID} → Оценка (Полная ФЗ)
- Курс_ID → Кафедра_Курса (Частичная ФЗ от ключа)

Частичные зависимости приводят к аномалиям избыточности и должны устраняться.



Аксиомы Армстронга (Armstrong's Axioms)

Система правил вывода, предложенная Уильямом Армстронгом в 1974 году. Позволяет логически выводить новые зависимости из исходного набора.

Правило	Название	Описание
Рефлексивность	Reflexivity	Если $Y \subseteq X$ то $X \rightarrow Y$ (Основа тривиальных зависимостей)
Пополнение	Augmentation	Если $X \rightarrow Y$ то для любого Z выполняется $XZ \rightarrow YZ$
Транзитивность	Transitivity	Если $X \rightarrow Y$ и $Y \rightarrow Z$ то $X \rightarrow Z$
		.

Свойства: Эти правила являются *полными* (можно вывести все следствия) и *корректными* (выведенные зависимости будут истинны).

Производные правила вывода

Объединение (Union)

Если $X \rightarrow Y$ и $X \rightarrow Z$, то $X \rightarrow YZ$

Позволяет компактно записывать зависимости от одного и того же детерминанта.

Декомпозиция (Decomposition)

Если $X \rightarrow YZ$, то $X \rightarrow Y$ и $X \rightarrow Z$

Обратное правило объединению: мы можем рассматривать зависимость каждого атрибута по отдельности.

Псевдотранзитивность (Pseudotransitivity)

Если $X \rightarrow Y$ и $WY \rightarrow Z$, то $WX \rightarrow Z$

Доказательство: из $X \rightarrow Y$ пополнением получаем $WX \rightarrow WY$, далее по транзитивности с $WY \rightarrow Z$ получаем $WX \rightarrow Z$.

Замыкание множества атрибутов X^+

Поиск всех зависимостей (замыкание множества ФЗ) F^+ — задача экспоненциальной сложности. Вместо этого мы используем замыкание атрибутов.

Замыкание множества атрибутов x относительно F (обозначается x^+) — это набор всех атрибутов A таких, что $x \rightarrow A$ логически выводимо из F .

Зачем это нужно?

- Проверка, является ли X ключом (если x^+ содержит все атрибуты схемы).
- Проверка, выводима ли конкретная ФЗ $x \rightarrow y$ (просто проверьте, входит ли y в x^+).
- Минимизация наборов атрибутов и проектирование ключей.

Концепция нормализации и её задачи

Нормализация — это пошаговый процесс организации данных в базе данных, который включает создание таблиц и установление связей между ними по определенным правилам. Эти правила предназначены для защиты данных и для обеспечения гибкости базы данных за счет исключения избыточности и несогласованных зависимостей.



Минимизация избыточности

Устранение дублирования фактов в разных местах системы. Экономия памяти и упрощение обслуживания.



Целостность данных

Формализация правил, защищающих от противоречий. Модель сама диктует логику хранения.



Гибкость структуры

Нормализованная база легче масштабируется и изменяется под новые требования бизнеса.

Почему это важно: Аномалии данных



Неправильно спроектированная база данных страдает от **аномалий** — ситуаций, когда операции над данными приводят к некорректным результатам:

- **Аномалия вставки:** невозможно добавить данные об объекте, пока нет данных о другом связанном объекте (например, нельзя добавить курс без студентов).
- **Аномалия обновления:** изменение одного факта требует изменения множества строк (дублирование), риск рассинхронизации.
- **Аномалия удаления:** удаление одной записи ведет к случайному уничтожению важной информации о другом объекте (удаляя студента, удаляем данные о группе).

Первая нормальная форма (1НФ)

Критерии 1НФ

Отношение находится в первой нормальной форме тогда и только тогда, когда каждое поле в каждой строке является **атомарным** (неделимым).

Основные требования:

- Никаких списков, массивов или повторяющихся групп в ячейках.
- В каждой таблице должен быть определен первичный ключ.
- Все строки должны иметь одинаковое количество столбцов.
- Тип данных в каждом столбце должен быть неизменным для всей таблицы.



Декомпозиция до 1НФ: Пример

Нарушение (Неатомарность):

Студент	Дисциплины
Иванов	БД, Сети, ОС
Петров	БД, ОС

Решение в 1НФ (Атомарность):

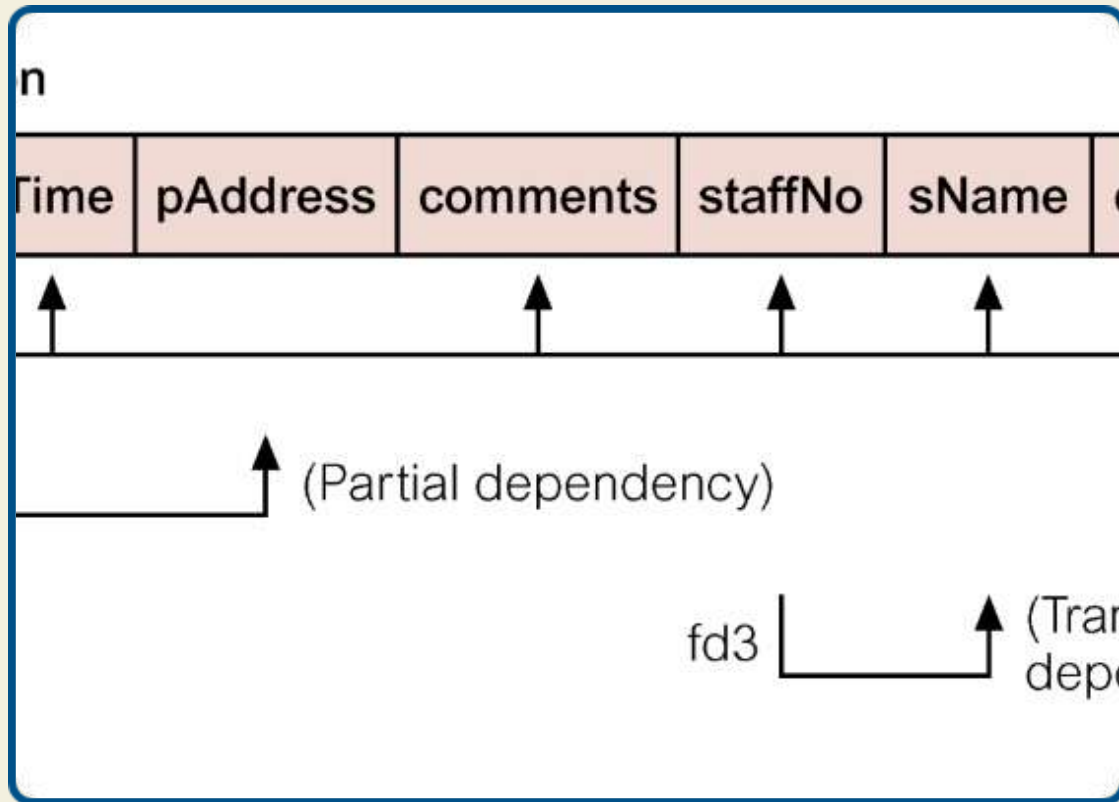
Студент	Дисциплина
Иванов	БД
Иванов	Сети
Иванов	ОС
Петров	БД

В 1НФ мы "выпрямляем" данные. Если у студента несколько предметов, мы создаем отдельную строку для каждой пары (Студент, Предмет).

Результат: Теперь по полю "Дисциплина" можно легко искать и фильтровать записи.



Вторая нормальная форма (2НФ)



Критерии 2НФ

Отношение находится во второй нормальной форме, если оно:

1. Уже находится в 1НФ.
2. Каждый неключевой атрибут **полностью** функционально зависит от первичного ключа.

Проблема: Частичные зависимости. Если ключ состоит из двух полей (A, B), а атрибут C зависит только от A — это нарушение 2НФ. Такие зависимости порождают избыточность, так как данные C будут повторяться для каждого B.

Процесс декомпозиции до 2НФ

Декомпозиция — это процесс разделения одного "плохого" отношения на два или более "хороших" без потери информации.

Алгоритм декомпозиции 2НФ:

1. Выявить все атрибуты, которые зависят только от **части** составного ключа
2. Вынести эти атрибуты в новое отношение вместе с той частью ключа, от которой они зависят.
3. В исходном отношении оставить только те атрибуты, которые зависят от всего ключа целиком.

Результат: Мы устраняем повторение описательных характеристик объектов, связанных через много-ко-многим или составные сущности.

Практический пример 2НФ

Дано: Заказы(ID_Заказа, ID_Товара, Кол-во, Цена_Товара, Название_Товара).

Здесь {ID_Заказа, ID_Товара} — первичный ключ.

Но **Название_Товара** зависит только от **ID_Товара** (частичная зависимость).

Таблица 1: Заказанные_Позиции

ID_Заказа, ID_Товара, Кол-во

Остались только атрибуты полной зависимости.

Таблица 2: Справочник_Товаров

ID_Товара, Название, Цена

Вынесена частичная зависимость в отдельный словарь.

Третья нормальная форма (3НФ)

Критерии 3НФ

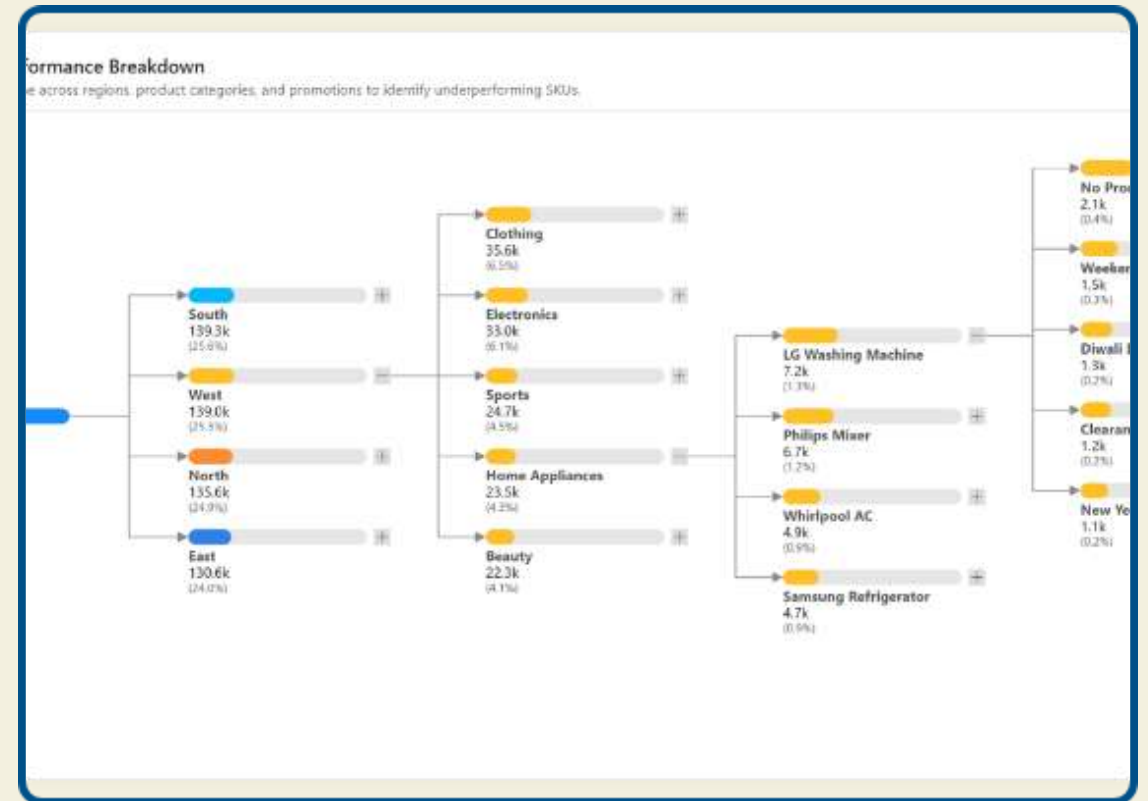
Отношение находится в третьей нормальной форме, если оно:

1. Находится во 2НФ.
2. Ни один неключевой атрибут не находится в транзитивной зависимости от первичного ключа.

Транзитивность: Если Ключ $\rightarrow$ Поле_А и Поле_А $\rightarrow$ Поле_В.

Следовательно, Поле_В зависит от Ключа через Поле_А. Это "косвенная" зависимость, которую нужно устранить.

Золотое правило: все атрибуты должны зависеть "от ключа, всего ключа и ничего кроме ключа".



Понятие детерминанта

Что такое детерминант?

Детерминант — это атрибут или набор атрибутов, от которого полностью функционально зависит какой-либо другой атрибут в отношении. В любой функциональной зависимости вида $X \rightarrow Y$, набор X является детерминантом.



Детерминанты-ключи: Когда детерминант является потенциальным ключом, зависимость считается «хорошей», так как она следует из природы идентификации строк.



Детерминанты не-ключи: Если детерминант определяет значение другого поля, но сам не может однозначно идентифицировать строку, это создает избыточность — один и тот же факт начинает храниться в нескольких местах.

Определение НФБК

Отношение находится в нормальной форме Бойса-Кодда тогда и только тогда, когда каждый его детерминант является потенциальным ключом.

Для любой нетривиальной функциональной зависимости $X \rightarrow Y$,
 X является надключом отношения.

НФБК = ЗНФ + отсутствие зависимостей ключевых атрибутов от неключевых детерминантов.

Это требование жестче, чем в ЗНФ. В ЗНФ разрешалась зависимость $X \rightarrow A$, если A — первичный атрибут (входит в состав какого-либо ключа). НФБК это запрещает, если X не является ключом сам по себе.

НФБК vs 3НФ: ключевое отличие

Критерий	Третья нормальная форма (3НФ)	Нормальная форма Бойса-Кодда (НФБК)
Базовое правило	Атрибуты зависят «от ключа и только от ключа».	Детерминанты — только ключи.
Допуск ($X \rightarrow A$)	Разрешено, если A — часть ключа (даже если X не ключ).	Запрещено, если X не является суперключом.
Область применения	Эффективна для большинства схем.	Необходима при перекрывающихся ключах.
Сложность	Проста в реализации.	Сложнее: может нарушать сохранение ФЗ.

Аномалии при нарушении НФБК



Вставка

Невозможно добавить информацию о зависимости детерминанта (например, какой преподаватель ведет какой курс), пока нет данных о главной сущности (студенте).



Обновление

Если преподаватель меняет курс, нам приходится обновлять данные во всех строках, где этот преподаватель связан со студентами. Риск рассинхронизации.



Удаление

При отчислении последнего студента с курса мы безвозвратно теряем информацию о том, какой преподаватель был назначен на данный предмет.

Четвертая нормальная форма (4НФ)

Отношение находится в 4НФ тогда и только тогда, когда оно находится в НФБК и все его нетривиальные многозначные зависимости являются функциональными зависимостями от потенциальных ключей.

Простыми словами: отношение в 4НФ не должно содержать более одной независимой многозначной зависимости. Каждая "многозначность" должна быть вынесена в отдельную таблицу.

Декомпозиция $R(X, Y, Z)$ в 4НФ приводит к созданию двух отношений: $R1(X, Y)$ и $R2(X, Z)$.



4НФ: Пример декомпозиции

Дано (Нарушение 4НФ): Курс(Название, Преподаватель, Учебник).
Преподаватели и учебники независимы, но оба относятся к курсу.

Курс	Преп.	Книга
БД	Иванов	Дейт
БД	Петров	Дейт
БД	Иванов	Коннолли
БД	Петров	Коннолли

× Избыточность: книги дублируются для каждого преподавателя.

Решение (Декомпозиция 4НФ):

R1: Курс_Преподаватели
{Название, Преподаватель}

R2: Курс_Учебники
{Название, Учебник}

✓ Число строк сократилось с 4 до 2+2. Аномалии устранены.

Зависимости соединения (ЗС)

Зависимость соединения — это более общее понятие, чем многозначная зависимость. ЗС говорит о том, что отношение **R** может быть восстановлено путем соединения его проекций **R1, R2, ..., Rn**.

Обозначение: $\bowtie (R_1, R_2, \dots, R_n)$

Многозначная зависимость — это частный случай ЗС для $n = 2$. Однако существуют ситуации, когда отношение нельзя разбить на 2 части без потерь, но можно разбить на 3 или более. Такие случаи крайне редки на практике, но важны для полной теоретической чистоты модели.



Пятая нормальная форма (5НФ/PJNF)

Определение 5НФ

Отношение находится в 5НФ (нормальная форма проекции-соединения), если оно находится в 4НФ и любая зависимость соединения в нем подразумевается потенциальными ключами этого отношения.

Суть 5НФ: Мы доводим декомпозицию до предела, когда дальнейшее разбиение либо невозможно, либо не имеет смысла, так как все зависимости уже определяются ключами. 5НФ гарантирует, что при соединении таблиц мы не получим "фантомных" данных, которые не существовали в реальности.

Практика: 5НФ часто называют "академическим пределом". В реальных проектах нормализацию обычно останавливают на 3НФ или НФБК, если нет явных проблем с многозначностью.