

§8. ПРОЕКТИРОВАНИЕ БД

1. Принципы концептуального проектирования **БД**
2. Метод нормальных форм - классический метод проектирования реляционных **БД**

ВВЕДЕНИЕ

Проектирование **БД** – одна из наиболее ответственных и трудных задач, связанных с созданием информационной системы. В результате ее решения должны быть определены и содержание **БД**, и эффективный с точки зрения всего сообщества будущих пользователей способ ее организации, и инструментальные средства управления данными.

1. Принципы концептуального проектирования БД

Процесс проектирования включает в себя следующие этапы:

- инфологическое проектирование;
- определение требований к операционной обстановке;
- выбор СУБД;
- логическое проектирование;
- физическое проектирование.

1.1. Этап инфологического проектирования

Первой задачей инфологического проектирования является определение *предметной области* системы, позволяющее изучить информационные потребности будущих пользователей.

Формирование взгляда на предметную область с позиций уже сформировавшегося или потенциального сообщества будущих пользователей **БД** является **второй задачей** стадии инфологического проектирования.

Такое представление предметной области – *ее инфологическая модель* – выражается в терминах не отдельных объектов предметной области и связей между ними, а их типов, связанных с ними ограничений целостности, а также тех процессов, которые приводят к переходу из одного состояния системы в другое. В простейших случаях проектировщик **БД** ограничивается содержательным описанием модели предметной области на ***естественном языке***.

Наряду с использованием естественного языка проектировщик может также применять разнообразные выразительные средства для изображения структуры предметной области, например, **графы**, вершинам которых соответствуют типы объектов, а ребрам – типы связей между ними (диаграммы Бахмана, ER-диаграммы сущностей-связей и др.). Можно также использовать средства автоматизации проектирования, например, **CASE-средства** (интерфейс **Erwin**).

1.2. Этап определения требований к операционной обстановке

Этот этап предполагает оценку требований к вычислительным ресурсам, необходимым для функционирования проектируемой информационной системы, определение типа и конфигурации ЭВМ, на которой она будет функционировать, выбор типа и версии операционной системы.

1.3. Выбор СУБД

Проектировщики БД при выборе инструментальных средств (СУБД) руководствуются чаще всего собственными интуитивными экспертными оценками требований к выбираемой СУБД по нескольким важным количественным и качественным характеристикам. К числу таких характеристик относятся:

- 1) тип модели данных, которую поддерживает выбранная **СУБД**, ее адекватность потребностям моделирования рассматриваемой предметной области;
- 2) характеристики производительности системы;
- 3) запас функциональных возможностей выбранной **СУБД** для дальнейшего развития информационной системы;
- 4) степень оснащенности **СУБД** инструментарием для персонала администрирования данными;
- 5) удобство и надежность **СУБД** в эксплуатации.

1.4. Этап логического проектирования

Этот этап проектирования заключается в создании схемы БД, в разработке ее логической структуры в соответствии с инфологической моделью, в определении числа и структуры отношений, формировании запросов к БД, определении типов отчетных документов, разработке алгоритмов обработки информации, создании форм для ввода и редактирования данных в БД.

1.5. Этап физического проектирования

Проектирование на физическом уровне во многом зависит от используемой СУБД, зачастую автоматизировано и скрыто от пользователя. В ряде случаев пользователю предоставляется возможность настройки отдельных параметров системы, которая обычно не составляет большой проблемы.

2. Метод нормальных форм – классический метод проектирования реляционных БД

Проектирование **БД** является одним из этапов **жизненного цикла** информационной системы. Основной задачей в процессе проектирования **БД** является задача **нормализации** ее отношений.

Метод нормальных форм основан на фундаментальном в теории реляционных **БД** понятии зависимости между атрибутами отношений.

2.1. Функциональная зависимость между атрибутами

Определение 1. Атрибут **B** функционально зависит от атрибута **A**, если каждому значению атрибута **A** соответствует в точности одно значение атрибута **B**.

Обозначение: $A \rightarrow B$.

Атрибуты **A** и **B** могут быть составными, то есть состоять из двух и более атрибутов.

Виды функциональных зависимостей

- **Определение 2.** Атрибут **C** зависит от атрибута **A** **транзитивно**, если существует атрибут **B** такой, что если $A \rightarrow B$ и $B \rightarrow C$, то $A \rightarrow C$.

Обозначают: $A \rightarrow B \rightarrow C$.

Например, оклад преподавателя транзитивно зависит от его должности (здесь **A** – фамилия, имя, отчество преподавателя, **B** – должность, **C** – оклад).

• **Определение 3.** В некотором отношении атрибут **B** **многозначно** зависит от атрибута **A**, если каждому значению атрибута **A** соответствует множество значений атрибута **B**, не связанных с другими атрибутами из данного отношения. Многозначные зависимости могут быть вида **1:M**, **M:1** и **M:M**. Обозначения соответственно:

$$A \Rightarrow B, A \Leftarrow B, A \Leftrightarrow B.$$

Например, пусть преподаватель ведет несколько дисциплин, и каждая дисциплина может вестись несколькими преподавателями. Тогда имеется зависимость **ФИО \leftrightarrow Дисциплина (M:M)**.

Функциональная зависимость может быть **частичной**, когда неключевой атрибут зависит от части составного ключа. Зависимость между атрибутами может быть **взаимно однозначной**, то есть вида **1:1 (A \leftrightarrow B)**. Выделяют также **взаимно независимые** атрибуты.

Основной способ определения функциональных зависимостей – внимательный анализ семантики атрибутов.

Предлагаем самостоятельно определить вид функциональной зависимости в отношении «Преподаватель»:

ФИО - Оклад; ФИО – Должность;
ФИО-Стаж; ФИО – Н_Должность; ФИО –
Кафедра; Должность – Н_Должность;
ФИО. Дисциплина. Группа – Вид_занятия,
где Н_Должность означает надбавку за
должность.

2.2. Общие сведения о нормальных формах

Иногда отношения содержат избыточное дублирование данных, которое является причиной *аномалий редактирования*. Различают *избыточность явную* и *неявную*.

Например, в отношении «Преподаватель» кортежи с данными о преподавателе, проводящему занятия в нескольких группах, могут повторяться несколько раз.

Поэтому, если у этого преподавателя изменится значение какого-то атрибута, то этот факт должен быть отражен во всех кортежах, где он задействован. В противном случае будет иметь место противоречие в данных, что является примером аномалии редактирования, обусловленной явной избыточностью данных.

Неявная избыточность в отношении «Преподаватель» может проявиться в одинаковых окладах у всех преподавателей и в одинаковых надбавках к окладу за одинаковую должность.

Действительно, если изменятся оклады за должность у всех преподавателей, кроме одного, то БД станет противоречивой.

Нормализация отношений и является как раз средством исключения избыточности.

Поскольку зависимость между атрибутами является причинной аномалией, стараются провести декомпозицию отношения с зависимостями атрибутов на несколько отношений. В результате образуется совокупность связанных отношений с различными видами связей. Таким образом, связи между таблицами отражают зависимости между атрибутами различных отношений.

После того, как выделены все функциональные зависимости, следует проверить их *согласованность* с данными исходного отношения.

Процесс проектирования БД с использованием метода нормальных форм является *итеративным* и заключается в последовательном переводе отношений из первой нормальной формы в нормальные формы более высокого порядка по определенным правилам.

Каждая следующая нормальная форма (НФ) ограничивает определенный тип функциональных зависимостей, устраняет соответствующие аномалии при выполнении операций над отношениями БД и сохраняет свойства предшествующих нормальных форм. Выделяют следующую последовательность нормальных форм: **1НФ**, **2НФ**, **3НФ**, **усиленная 3НФ** или нормальная форма **Бойса-Кодда (БКНФ)**, **4НФ** и **5НФ**.

2.3. Первая нормальная форма

Отношение R находится в 1НФ, если все его атрибуты являются простыми (имеют единственное значение).

Исходное отношение строится таким образом, чтобы оно было в 1НФ.

2.4. Декомпозиция без потерь

Под *декомпозицией без потерь* или *проецированием без потерь* данного отношения понимается такой способ декомпозиции этого отношения, при котором оно полностью и без избыточности восстанавливается путем естественного соединения полученных отношений.

Такая декомпозиция должна обеспечивать одинаковый результат запроса к исходному отношению и к отношениям, получаемым в результате декомпозиции.

2.5. Вторая нормальная форма

Отношение R находится во 2НФ, если каждый его неключевой атрибут функционально полно зависит от составного первичного ключа.

Для устранения частичной избыточности и перевода отношения R во 2НФ необходимо, используя операцию проекции, разложить его (то есть произвести операцию декомпозиции без потерь) на несколько отношений следующим образом:

- построить проекцию отношения **R** без атрибутов, находящихся в частичной функциональной зависимости от первичного ключа;
- построить проекции на части составного первичного ключа и атрибуты, зависящие от этих частей.

2.6. Третья нормальная форма

Существует два равносильных определения 3НФ:

- **Отношение R находится в 3НФ, если каждый его неключевой атрибут не транзитивно зависит от первичного ключа.**
- **Отношение R находится в 3НФ в том и только в том случае, если все его неключевые атрибуты взаимно независимы и полностью зависят от первичного ключа.**

2.7. Третья усиленная нормальная форма

Отношение R находится в НФ Бойса-Кодда (БКНФ), если в нем отсутствуют зависимости ключей (атрибутов составного ключа) от неключевых атрибутов.

При этом, как и в предыдущих определениях нормальных форм, должны сохраняться все свойства предыдущих нормальных форм.

2.8. Теорема Фейджина

В произвольном отношении $R(A, B, C)$ может одновременно существовать многозначная зависимость вида $A \Rightarrow B$ и $A \Rightarrow C$ (обозначение: $A \Rightarrow B|C$).

Переход к 4НФ основывается на теореме Фейджина: **отношение $R(A, B, C)$ можно спроецировать без потерь на отношения $R_1(A, B)$ и $R_2(A, C)$ в том и только в том случае, когда существует зависимость $A \Rightarrow B|C$.**

2.9. Четвертая нормальная форма

Отношение R находится в 4НФ в том и только в том случае, когда существует многозначная зависимость вида $A \Rightarrow B$, а все остальные атрибуты отношения R функционально зависят от атрибута A .

Заметим, что в общем случае не всякое отношение, находящееся в 4НФ, можно восстановить к исходному отношению.

Результатом нормализации всех предыдущих схем является два новых отношения. Иногда это сделать не удастся, либо получаемые отношения заведомо имеют нежелательные свойства. В этом случае выполняют декомпозицию исходного отношения на отношения, количество которых больше двух.

Введем определение **зависимости соединения**. Говорят, что **отношение** $R(X, Y, \dots, Z)$ **удовлетворяет зависимости соединения** в том и только в том случае, если R **восстанавливается без потерь** путем соединения своих проекций на атрибуты X, Y, \dots, Z .

Зависимость соединения является обобщением функциональной и многозначной зависимостей.

2.10. Пятая нормальная форма

Отношение R находится в 5НФ (или нормальной форме проекции-соединения – PJ/NF) в том и только в том случае, когда любая зависимость соединения в отношении R следует из существования некоторого возможного ключа в R .

Условия получения пятой нормальной формы весьма нетривиальны и поэтому на практике она используется редко. Обычно ограничиваются структурой **БД, соответствующей **ЗНФ** или **БКНФ**. Поэтому процесс нормализации отношений методом нормальных форм предполагает последовательное удаление из исходного отношения следующих межатрибутных зависимостей:**

- 1) частичных зависимостей неключевых атрибутов от ключевых (удовлетворение требований **2НФ**);
- 2) нетранзитивных зависимостей неключевых атрибутов от ключа (удовлетворение требований **3НФ**);
- 3) зависимости ключей (атрибутов составных ключей) от неключевых атрибутов (удовлетворение требований **БКНФ**).

Заметим, что метод нормальных форм используется для проектирования небольших БД. Для больших БД чаще используется метод **«сущность-связь»**.