

**Проблема выбора
рациональных схем
отношений**

Проблема выбора рациональных схем отношений

Выбор схем отношений, представляющих концептуальную схему, в значительной степени определяет эффективность БД.

Рассмотрим схему отношений и проанализируем её недостатки. Предположим, что данные о студентах, факультетах, специальностях, включены в таблицу со следующей структурой :
СТУДЕНТ (Код студента, Фамилия, Название факультета, Название специальности).

Эта схема отношений определяет следующие недостатки БД:

Проблема выбора рациональных схем отношений

- Дублирование информации (избыточность). У студентов, обучающихся на одном факультете, возможны повторы названий факультетов, специальностей.
- Потенциальная противоречивость данных (аномалии обновления). Если изменится название специальности, то необходимо вносить изменения во всех аналогичных кортежах (для всех студентов, обучающихся по данной специальности).
- Возможность потери информации (аномалии удаления). При удалении информации о всех студентах, поступающих на определенную специальность, можем потерять сведения об этой специальности и возможность зачисления на указанную специальность в будущем.

В теории реляционных баз данных существуют формальные методы построения реляционной модели базы данных, в которой отсутствует избыточность и аномалии обновления, удаления и включения.

Нормализация. Первая нормальная форма

Построение схем отношений, обладающих оптимальными свойствами при операциях включения, модификации и удаления данных, осуществляется нормализацией схем отношений. Нормализация производится в несколько этапов. На начальном этапе схема отношений должна находиться в первой нормальной форме (1НФ). Отношение находится в первой нормальной форме, если все атрибуты отношения принимают простые значения (атомарные или неделимые).

Нормализация. Первая нормальная форма

Таблица представляет сущность ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ

| Код студента | Фамилия | Код экзамена | Предмет и дата | Оценка |
|--------------|---------|--------------|--------------------|--------|
| 1 | Сергеев | 1 | Математика 5.06.08 | 4 |
| 2 | Иванов | 1 | | 5 |
| 1 | Сергеев | 2 | Физика 9.06.08 | 5 |
| 2 | Иванов | 2 | | 5 |

Данная таблица не находится в первой нормальной форме: на пересечении кортежей и четвертого атрибута располагается более одного значения.

Для перехода к первой нормальной форме перенесем значения предмета и даты в соответствующие столбцы

Нормализация. Первая нормальная форма

| Код студента | Фамилия | Код экзамена | Предмет | Дата | Оценка |
|--------------|---------|--------------|------------|---------|--------|
| 1 | Сергеев | 1 | Математика | 5.08.03 | 4 |
| 2 | Иванов | 1 | Математика | 5.08.03 | 5 |
| 1 | Сергеев | 2 | Физика | 9.08.03 | 5 |
| 2 | Иванов | 2 | Физика | 9.08.03 | 5 |

На пересечении любого кортежа и любого поля находится одно значение и, следовательно, данная таблица находится в первой нормальной форме.

Нормализация таблиц БД

Для улучшения структуры реляционной БД (устранения возможных аномалий), необходимо привести все таблицы БД к третьей нормальной форме или к более высокой нормализации: к НФ «Бойса-Кодда», к четвертой и пятой нормальным формам (если это возможно).

Нормализация таблиц БД

Отношение, представленное в первой нормальной форме, преобразуется во вторую нормальную форму (2НФ).

Таблица БД, не являющаяся таблицей в третьей нормальной форме, преобразуется в совокупность таблиц, соответствующих третьей нормальной форме (декомпозиция).

Нормализация таблиц БД

| Код студента | Фамилия | Код экзамена | Предмет | Дата | Оценка |
|--------------|---------|--------------|------------|---------|--------|
| 1 | Сергеев | 1 | Математика | 5.08.03 | 4 |
| 2 | Иванов | 1 | Математика | 5.08.03 | 5 |
| 1 | Сергеев | 2 | Физика | 9.08.03 | 5 |
| 2 | Иванов | 2 | Физика | 9.08.03 | 5 |

Первичным ключом данного отношения будет совокупность атрибутов – Код студента и Код экзамена.

Нормализация таблиц БД

Для записи процесса нормализации введем следующие обозначения:

КС – код студента, КЭ – код экзамена, Ф – фамилия, П – предмет, Д – дата, О - оценка.

Выпишем функциональные зависимости:

КС, КЭ \square Ф, П, Д, О;

КС, КЭ \square Ф;

КС, КЭ \square П;

КС, КЭ \square Д;

КС, КЭ \square О;

КЭ \square П;

КЭ \square Д;

КС \square Ф.

(Транзитивность: Если $X \square Y$ и $Y \square Z$, то $X \square Z$)

Нормализация таблиц БД

По определению, отношение находится во второй нормальной форме, если оно находится в 1НФ и каждый неключевой атрибут зависит от первичного ключа и не зависит от части ключа. Атрибуты П, Д, Ф зависят от части ключа. Чтобы избавиться от этих зависимостей, необходимо произвести декомпозицию отношения. Отношение находится в 3НФ, если оно находится во 2НФ и дополнительно:

- каждый ключевой атрибут нетранзитивно зависит от первичного ключа;
- все неключевые атрибуты отношения взаимно независимы и полностью зависят от первичного ключа.

3НФ исключает избыточность и аномалии включения и удаления (не предотвращает всех возможных аномалий модификации).

Нормализация таблиц БД

Отношение **ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ** может быть определено следующим образом:

$R(КС, Ф, КЭ, П, Д, О)$.

На основе зависимости $КС \square Ф$ может быть построено отношение $R1(КС, Ф)$.

Первичным ключом отношения $R1(КС, Ф)$ будет ключ $КС$, не являющийся составным.

- Отношение $R1(КС, Ф)$ находится в первой нормальной форме, т. к. на пересечении любого кортежа и любого поля находится одно значение.
- Отношение $R1(КС, Ф)$ находится во 2НФ, т.к. в этом отношении:
 - неключевой атрибут $Ф$ зависит от первичного ключа и не зависит от части ключа.
- Отношение $R1(КС, Ф)$ находится в 3НФ, т.к. в этом отношении:
 - нет транзитивных зависимостей;
 - все неключевые атрибуты взаимно независимы и полностью зависят от первичного ключа

Нормализация таблиц БД

Первичным ключом отношения R_2 (КС, КЭ, П, Д, О) будет ключ КС, КЭ являющийся составным.

В отношении R_2 определена зависимость $КЭ \twoheadrightarrow П$, $КЭ \twoheadrightarrow Д$, $КЭ \twoheadrightarrow П, Д$. Атрибуты П, Д зависят от части ключа, следовательно отношение не находится во 2НФ.

На основе функциональной зависимости $КЭ \twoheadrightarrow П, Д$ может быть построено отношение $R_3(КЭ, П, Д)$.

Первичным ключом отношения $R_3(КЭ, П, Д)$ будет ключ КЭ, не являющийся составным.

- Отношение $R_3(КЭ, П, Д)$ находится в первой нормальной форме, т.к. на пересечении любого кортежа и любого поля находится одно значение.
- Отношение $R_3(КЭ, П, Д)$ находится во 2НФ, т.к. в этом отношении:
 - неключевой атрибут Ф зависит от первичного ключа и не зависит от части ключа.
- Отношение $R_3(КЭ, П, Д)$ находится в 3НФ, т.к. в этом отношении:
 - нет транзитивных зависимостей;
 - все неключевые атрибуты взаимно независимы и полностью зависят от первичного ключа

Нормализация таблиц БД

На основе зависимости $КС, КЭ \square O$ может быть построено отношение $R4(КС, КЭ, O)$.

Первичным ключом отношения $R4(КС, КЭ, O)$ будет ключ $(КС, КЭ)$, являющийся составным.

Отношение $R4(КС, КЭ, O)$ находится в первой нормальной форме, т.к. на пересечении любого кортежа и любого поля находится одно значение.

Отношение $R4(КС, КЭ, O)$ находится во 2НФ, т.к. в этом отношении:

- неключевой атрибут O зависит от первичного ключа и не зависит от части ключа.

Отношение $R4(КС, КЭ, O)$ находится в 3НФ, т.к. в этом отношении:

- нет транзитивных зависимостей;
- все неключевые атрибуты взаимно независимы и полностью зависят от первичного ключа

Нормализация таблиц БД

Таким образом, исходное отношение R приведено в к трем отношениям, каждое из которых находится в третьей нормальной форме $R1(КС, Ф)$, $R3(КЭ, П, Д)$, $R4(КС, КЭ, О)$. В отношении $R4$ атрибуты $КС$, $КЭ$ являются внешними ключами, используемыми для установления связей с другими отношениями.

Представление модели в виде диаграммы объектов-связей

Представим полученную модель в виде диаграммы объектов (сущностей)-связей (ER-диаграммы).

Для наглядности и возможности последующего программирования перейдем к английским названиям объектов (отношений) и атрибутов.

Отношение $R1$ представляет объект `student` с атрибутами `id_st` (первичный ключ), `surname`.

Отношение $R3$ представляет объект `exam_st` с атрибутами `id_ex` (первичный ключ), `subject`, `date`.

Отношение $R4$ представляет объект `mark_st` с атрибутами `id_st` (внешний ключ), `id_ex` (внешний ключ), `mark`.

Первичный ключ: (`id_st`, `id_ex`).

Представление модели в виде диаграммы объектов-связей

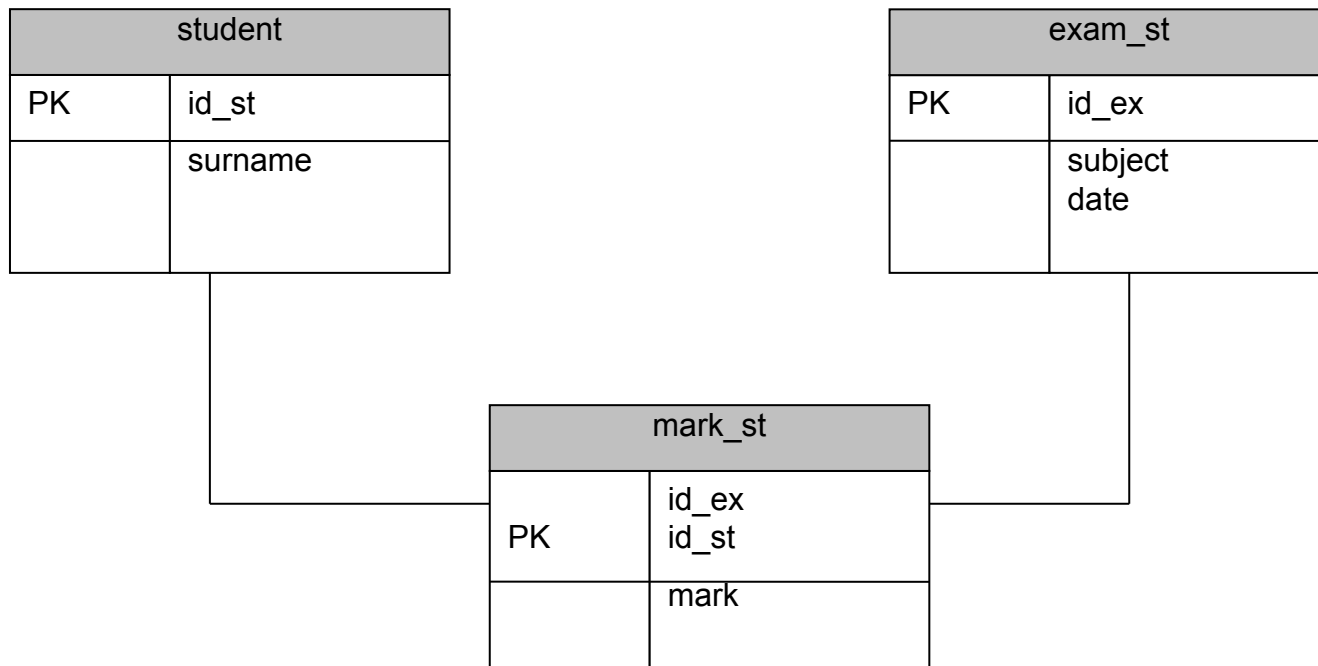


Рис. 8.1. ER-диаграмма, представляющая предметную область

Реализация условия целостности данных в современных СУБД

Под целостностью БД понимается то, что в ней содержится полная, непротиворечивая и адекватно отражающая предметную область информация. Поддержка целостности в реляционных БД основана на выполнении следующих требований.

1. Первое требование называется **требованием целостности сущностей**. Объекту или сущности реального мира в реляционных БД соответствуют кортежи отношений. Требование состоит в том, что любой кортеж любого отношения отличим от любого другого кортежа этого отношения, т.е., другими словами, любое отношение должно обладать определенным первичным ключом. Это требование автоматически удовлетворяется, если в системе не нарушаются базовые свойства отношений.

2. Второе требование называется **требованием целостности по ссылкам**. При соблюдении нормализованности отношений, сложные сущности реального мира представляются в реляционной БД в виде нескольких кортежей нескольких отношений. взаимосвязь между отношениями определяется схемой данных.

Реализация условия целостности данных в современных СУБД

Пример внешнего ключа.

СТУДЕНТ (Код студента, Фамилия) сдает ЭКЗАМЕН (Код студента, Предмет, Оценка).

Атрибут **Код студента** сущности **ЭКЗАМЕН** называется **внешним ключом**, поскольку его значения однозначно характеризуют сущности, представленные кортежами другого отношения (отношения Студент). Отношение, в котором определен внешний ключ, ссылается на соответствующее отношение, в котором такой же атрибут является первичным ключом.

Требование целостности по ссылкам или требование внешнего ключа состоит в том, что для каждого значения внешнего ключа в отношении, на которое ведет ссылка, должен найтись кортеж с таким же значением первичного ключа (или значение внешнего ключа должно быть неопределенным).

Реализация условия целостности данных в современных СУБД

Ограничения целостности сущности и по ссылкам должны поддерживаться СУБД. Для соблюдения целостности сущности достаточно гарантировать отсутствие в любом отношении кортежей с одним и тем же значением первичного ключа. (В Access для этого предназначена специальная реализация целочисленного поля – поле типа «Счетчик».) При обновлении ссылающегося отношения (при вставке новых кортежей или модификации значения внешнего ключа в существующих кортежах), необходимо контролировать корректность значений внешнего ключа.

Реализация условия целостности данных в современных СУБД

Существуют три подхода, поддерживающих целостность по ссылкам при удалении кортежа из отношения, на которое ведет ссылка:

1. Запрещается производить удаление кортежа, на который существуют ссылки (т.е. сначала нужно либо удалить ссылающиеся кортежи, либо соответствующим образом изменить значения их внешнего ключа);
2. При удалении кортежа, на который имеются ссылки, во всех ссылающихся кортежах значение внешнего ключа автоматически становится неопределенным;
3. Каскадное удаление состоит в том, что при удалении кортежа из отношения, на которое ведет ссылка, из ссылающегося отношения автоматически удаляются все ссылающиеся кортежи.

Проблема выбора рациональных схем отношений

Таблица представляет сущность ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ

| Код студента | Фамилия | Код экзамена | Предмет и дата | Оценка |
|--------------|---------|--------------|-----------------------|--------|
| 1 | Сергеев | 1 | Математика 5.06.08 | 4 |
| 2 | Иванов | 1 | | 5 |
| 1 | Сергеев | 2 | Физика 9.06.08 | 5 |
| 2 | Иванов | 2 | | 5 |

Данная таблица не находится в первой нормальной форме (не нормализована), так как на пересечении кортежей и четвертого столбца располагается более одного значения.

Для перехода к первой нормальной форме перенесем значения предмета и даты в соответствующие столбцы