

ПРОЕКТИРОВАНИЕ РЕЛЯЦИОННОЙ БАЗЫ ДААННЫХ

3

Инфологическое проектирование

- 1.1. Анализ предметной области (на примере Проектной организации)
- БД создаётся для информационного обслуживания руководства организации, руководителей проектов и участников проектов. БД должна содержать данные об отделах организации, сотрудниках и проектах.

Инфологическое проектирование

В соответствии с представлением о предметной области, РБД строится с учётом следующих особенностей:

- Каждый штатный сотрудник работает в определённом отделе, в каждом отделе могут работать несколько сотрудников.
- Каждый проект относится к определённому отделу, каждый отдел может отвечать за выполнение нескольких проектов.
- Каждый сотрудник может принимать участие в выполнении нескольких проектов, над каждым проектом может трудиться несколько сотрудников.

Инфологическое проектирование

- Для каждого проекта назначается руководитель из числа сотрудников того отдела, к которому относится проект.
- Каждый проект должен быть выполнен в заданные сроки, каждый проект может состоять из нескольких этапов. Если проект состоит из одного этапа, то сроки выполнения этапа должны совпадать со сроками выполнения проекта в целом.
- Оклад сотрудника зависит от занимаемой должности, за участие в проектах сотрудник получает дополнительное вознаграждение.

Инфологическое проектирование

- Виды участия сотрудников в проектах: руководитель, консультант, исполнитель.
- Каждый отдел занимает одно или несколько помещений (комнат), в каждом помещении может быть один или несколько стационарных телефонов.

Примечание. Описание особенностей ПрО должно быть достаточным для создания ER–диаграмм.

Инфологическое проектирование

Для создания ER-модели необходимо выделить сущности предметной области:

1. **Отделы.** Атрибуты: название, аббревиатура, комнаты, телефоны.
2. **Сотрудники.** Атрибуты: ФИО, паспортные данные, дата рождения, пол, ИНН (индивидуальный номер налогоплательщика), номер пенсионного страхового свидетельства, адреса, телефоны (рабочий, домашний, мобильный), данные об образовании (вид образования (высшее, среднее специальное и т.д.), специальность, номер диплома, дата окончания учебного заведения), должность, оклад, логин (имя пользователя).

Примечания: 1. Логин потребуется для назначения дифференцированных прав доступа.

- 2. Не предусмотрена полная информационная поддержка сотрудников отдела кадров, поэтому не будем отражать в БД такие сведения как дату поступления сотрудника на работу, его переводы с одной должности на другую, уход в отпуск и т.п.

Инфологическое проектирование

- 3. Проекты.** Атрибуты: номер договора; полное название проекта; сокращённое название проекта; дата подписания договора; заказчик; контактные данные заказчика; дата начала проекта; дата завершения проекта; сумма по проекту; дата реальной сдачи проекта; сумма, полученная по проекту на текущую дату.
- 4. Этапы проекта.** Атрибуты: номер по порядку, название, дата начала этапа, дата завершения этапа, форма отчетности, сумма по этапу, дата реальной сдачи этапа; сумма, полученная по этапу на текущую дату.

Исходя из выявленных сущностей, построим ER–диаграмму (рис. 2). Напомним, что пометки у линий означают степень связи: 1:1, 1:N и N:M.

Инфологическое проектирование

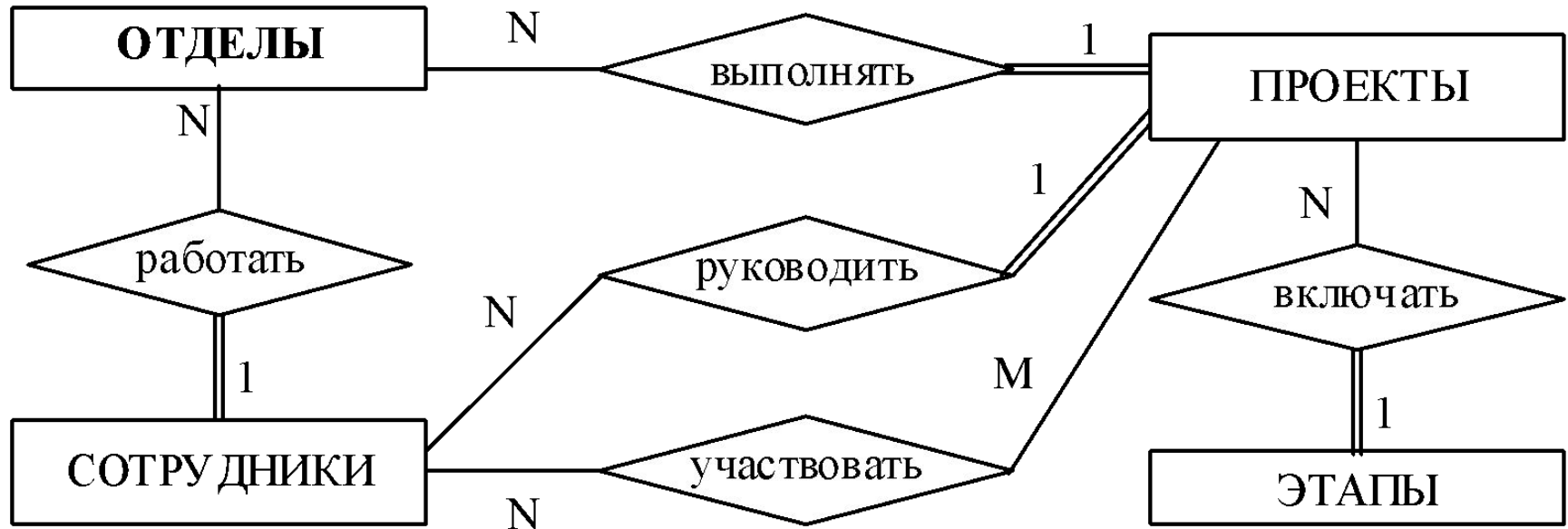


Рис. 2. ER–диаграмма ПрО «Проектная организация»

Инфологическое проектирование

1.2. Анализ информационных задач и круга пользователей системы

Определим группы пользователей, их основные задачи и запросы к БД:

1. Руководители организации:

- заключение новых договоров;
- назначение руководителей проектов;
- получение списка всех участников проектов;
- изменение должностных окладов и штатного расписания;
- получение полной информации о проектах;
- внесение изменений в данные о проектах;
- архивирование данных по завершённым проектам.

Инфологическое проектирование

2. Руководитель проекта:

- назначение участников проекта;
- получение списка сотрудников, работающих над конкретным проектом;
- получение полной информации о проекте, руководителем которого он является;
- получение сведений о сотрудниках, которые могут стать участниками проекта;
- определение размера дополнительного вознаграждения сотрудников по конкретному проекту;
- внесение изменений в данные об этапах проекта.

Инфологическое проектирование

3. Сотрудники отдела кадров:

- приём/увольнение сотрудников;
- внесение изменений в данные о сотрудниках.

4. Бухгалтеры:

- получение ведомости на выплату зарплаты.

5. Сотрудники – участники проектов:

- просмотр данных о других участниках проекта;
- просмотр данных о сроках сдачи проекта и форме отчётности.

Определение требований к операционной обстановке

Для выполнения этого этапа необходимо знать (ориентировочно) объём работы организации (количество проектов и сотрудников), иметь представление о характере и интенсивности запросов.

Объём внешней памяти, необходимый для функционирования системы, складывается из двух составляющих: память, занимаемая модулями СУБД (ядро, утилиты, вспомогательные программы), и память, отводимая под данные (МД). Для реальных баз данных обычно наиболее существенным является МД.

Определение требований к операционной обстановке

На основе результатов анализа ПрО можно приблизительно оценить объём памяти, требуемой для хранения данных. Примем ориентировочно, что: одновременно осуществляется порядка десяти проектов, работа над проектом продолжается в среднем год (по 1К на каждый проект); каждый проект состоит в среднем из четырёх этапов (по 0,5К на этап); в компании работают 100 сотрудников (по 0,5К на каждого сотрудника); в выполнении каждого проекта в среднем участвуют 10 сотрудников (по 0,2К); устаревшие данные переводятся в архив (накапливаются в архиве БД).

Определение требований к операционной обстановке

Объём памяти для хранения данных за первый год составит:

$$M_d = 2(10 \cdot 1 + 10 \cdot 4 \cdot 0,5 + 100 \cdot 0,5 + (10 \cdot 10 \cdot 0,2)) = 200 \text{ К,}$$

Коэффициент 2 необходим для того, чтобы учесть необходимость выделения памяти под дополнительные структуры (например, индексы). Объём памяти будет увеличиваться ежегодно на столько же при сохранении объёма работы.

Требуемый объём оперативной памяти определяется на основании анализа интенсивности запросов и объёма результирующих данных. Для проектируемой БД требуется относительно небольшой объём ресурсов запоминающих устройств, поэтому никаких специальных требований к объёму внешней и оперативной памяти компьютера не предъявляется.

Выбор СУБД и других программных средств

Анализ задач показывает, что для реализации требуемых функций подходят различные СУБД (MS Access, Firebird, MySQL и др.). Все они поддерживают реляционную модель данных и предоставляют разнообразные возможности для работы с данными.

Объём внешней и оперативной памяти, требующийся для функционирования СУБД, обычно указывается в сопроводительной документации.

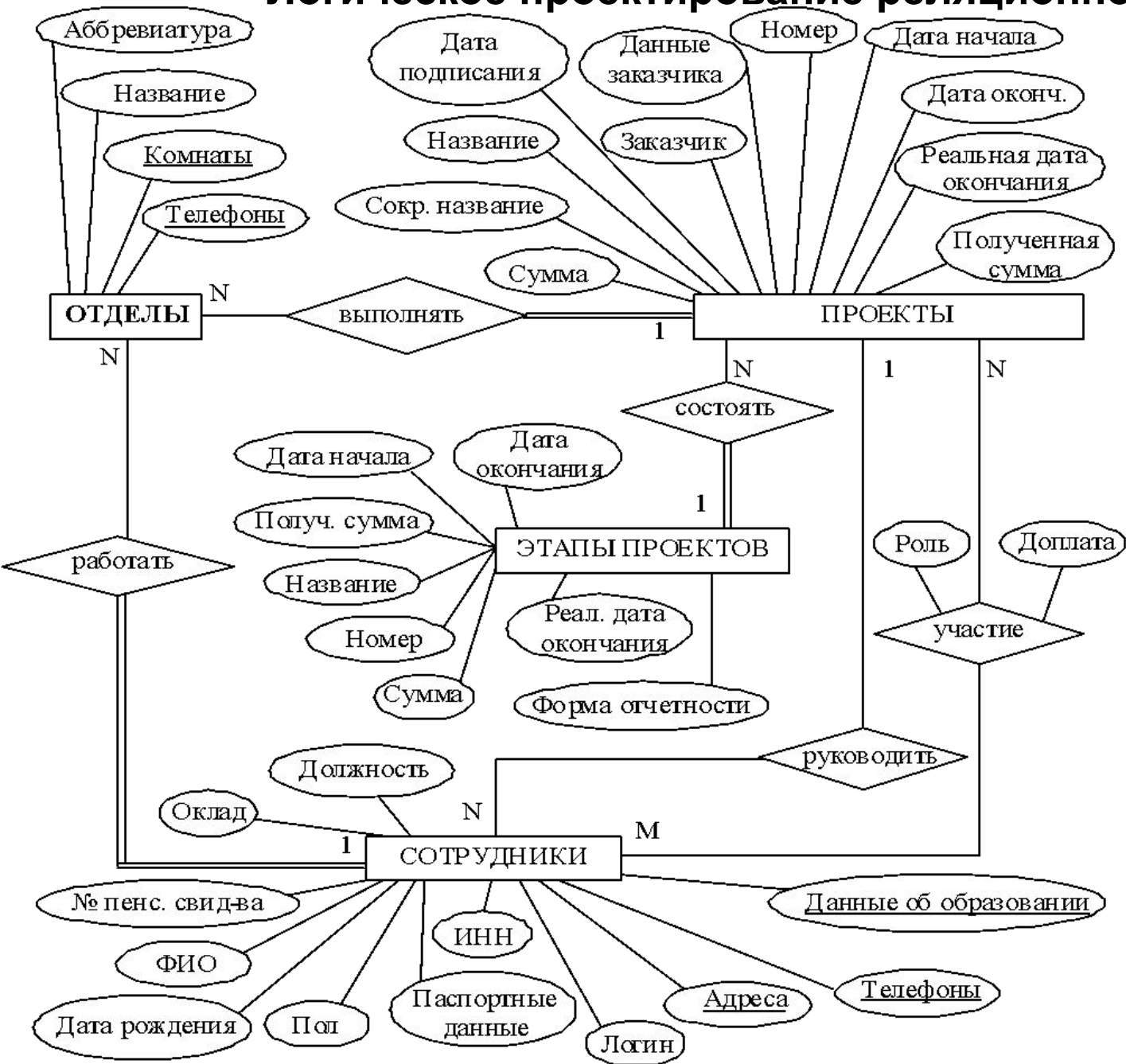
Логическое проектирование реляционной БД

1.4. Преобразование ER–диаграммы в схему базы данных

База данных создаётся на основании схемы БД. Для преобразования ER–диаграммы в схему БД приведём уточнённую ER–диаграмму, содержащую атрибуты сущностей (рис. 3).

Примечание. Многозначные атрибуты на рисунке выделены подчеркиванием.

Логическое проектирование реляционной БД



Логическое проектирование реляционной БД

Преобразование ER–диаграммы в схему БД выполняется путем сопоставления каждой сущности и каждой связи, имеющей атрибуты, отношения (таблицы) БД. Связь типа 1:n (один-ко-многим) между отношениями реализуется с использованием внешнего ключа. Ключ вводится для того отношения, к которому осуществляется множественная связь. Внешнему ключу должен соответствовать первичный или уникальный ключ основного (родительского) отношения.

Для схемы БД будем использовать обозначения, представленные на рис. 4.

Логическое проектирование реляционной БД

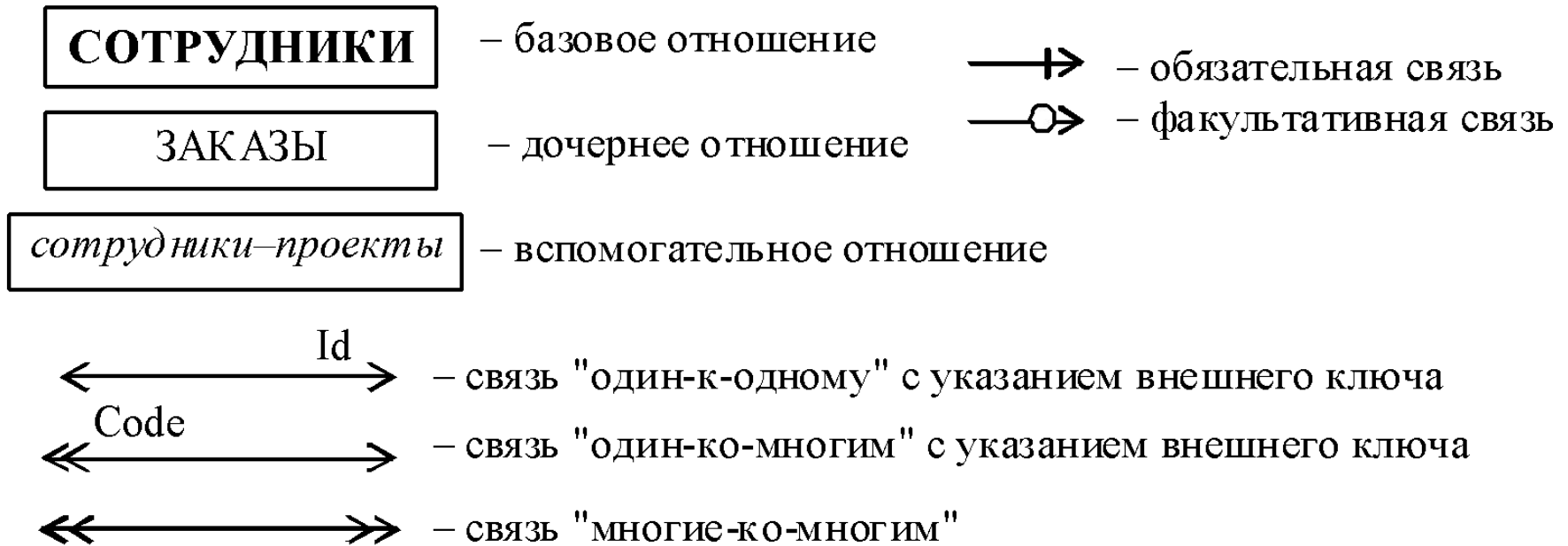


Рис. 4. Обозначения, используемые на схеме базы данных

Полученная схема реляционной базы данных (РБД) приведена на рис. 5.

Логическое проектирование реляционной БД

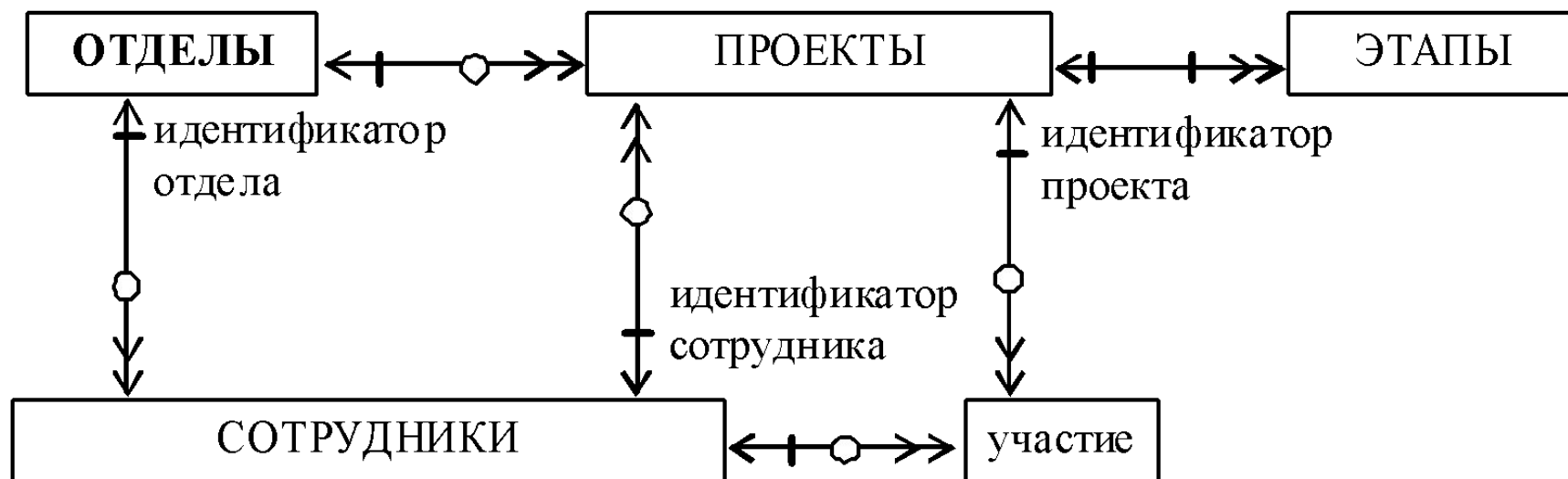
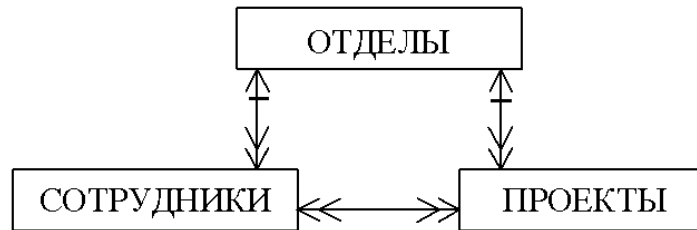


Рис. 5. Схема РБД, полученная из ER-диаграммы проектной организации

Логическое проектирование реляционной БД

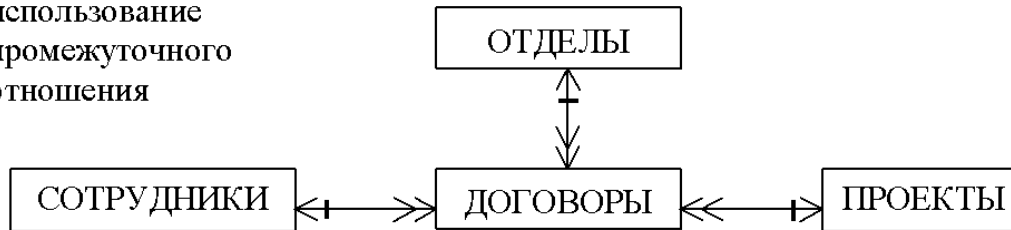
а) пример цикла



б) разрыв связи



в) использование промежуточного отношения



г) миграция внешних ключей

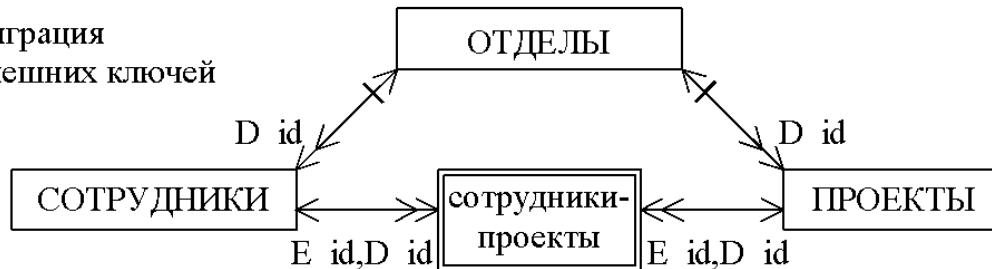


Рис.6. Некоторые способы разрешения циклов в схеме базы данных

Логическое проектирование реляционной БД

1.5. Составление реляционных отношений

Каждое реляционное отношение соответствует одной сущности (объекту ПрО) и в него вносятся все атрибуты этой сущности. Для каждого отношения определяются первичный ключ и внешние ключи (в соответствии со схемой БД). В том случае, если базовое отношение не имеет потенциальных ключей, вводится *суррогатный первичный ключ*, который не несёт смысловой нагрузки и служит только для идентификации записей.

Логическое проектирование реляционной БД

Отношения приведены в табл. 1..5. Для каждого отношения указаны атрибуты с их внутренним названием, типом и длиной. Типы данных обозначаются так: N – числовой, C – символьный тип фиксированной длины, V – символьный тип переменной длины, D – дата (этот тип имеет стандартную длину, зависящую от СУБД, поэтому она не указывается).

Потенциальными ключами отношения ОТДЕЛЫ являются атрибуты Аббревиатура и Название отдела. Первый занимает меньше места, поэтому выбираем его в качестве первичного ключа.

Логическое проектирование реляционной БД

Таблица 1. Схема отношения ОТДЕЛЫ (Departs)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Аббревиатура отдела	D_ID	C(10)	первичный ключ
Название отдела	D_NAME	V(100)	обязательное поле
Комнаты	D_ROOMS	V(20)	обязательное многозначное поле
Телефоны	D_PHONE	V(40)	обязательное многозначное поле

Логическое проектирование реляционной БД

Таблица 2. Схема отношения СОТРУДНИКИ (Employees)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Номер	E_ID	N(4)	суррогатный первичный ключ
Фамилия, имя, отчество	E_NAME	V(50)	обязательное поле
Дата рождения	E_BORN	D	обязательное поле
Пол	E_SEX	C(1)	обязательное поле, 'м' или 'ж'
Паспортные данные	E_PASP	V(50)	обязательное поле
ИНН	E_INN	C(12)	обязательное уникальное поле
Номер пенсионного страхового свидетельства	E_PENS	C(14)	обязательное уникальное поле
Отдел	E_DEPART T	C(10)	внешний ключ (к Departs)
Должность	E_POST	V(30)	обязательное поле
Оклад	E_SAL	N(8,2)	обязательное поле, > 4500 руб.
Данные об образовании	E_EDU	V(200)	обязательное многозначное поле
Адреса	E_ADDR	V(100)	многозначное поле
Телефоны	E_PHONE	V(30)	многозначное поле
Логин	E_LOGIN	V(30)	

Логическое проектирование реляционной БД

Таблица 3. Схема отношения ПРОЕКТЫ (Projects)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Номер проекта	P_ID	N(6)	обязательное уникальное поле
Название проекта	P_TITLE	V(100)	обязательное поле
Сокращённое название	P_ABBR	C(10)	первичный ключ
Отдел	P_DEPART	C(10)	внешний ключ (к Departs)
Заказчик	P_COMPANY	V(40)	обязательное поле
Данные заказчика	P_LINKS	V(200)	обязательное поле
Руководитель	P_CHIEF	N(4)	внешний ключ (к Employees)
Дата начала проекта	P_BEGIN	D	обязательное поле
Дата окончания проекта	P_END	D	обязательное поле, больше даты начала проекта
Реальная дата окончания	P_FINISH	D	
Стоимость проекта	P_COST	N(10)	обязательное поле
Полученная сумма	P_SUM	N(10)	обязательное поле, значение по умолчанию – 0

Логическое проектирование реляционной БД

Таблица 4. Схема отношения ЭТАПЫ ПРОЕКТА (Stages)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>	
Проект	S_PRO	C(10)	внешний ключ (к Projects)	составной первичный ключ
Номер этапа	S_NUM	N(2)		
Название этапа	S_TITLE	V(200)	обязательное поле	
Дата начала этапа	S_BEGIN	D	обязательное поле	
Дата окончания этапа	S_END	D	обязательное поле, > даты начала	
Реальная дата окончания	S_FINISH	D	больше даты начала этапа	
Стоимость этапа	S_COST	N(10)	обязательное поле	
Полученная сумма по этапу	S_SUM	N(10)	обязательное поле, значение по умолчанию – 0	
Форма отчётности	S_FORM	V(100)	обязательное поле	

Логическое проектирование реляционной БД

Таблица 5. Схема отношения УЧАСТИЕ (Job)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания *</i>
Проект	J_PRO	C(10)	внешний ключ (к Projects)
Сотрудник	J_EMP	N(4)	внешний ключ (к Employees)
Роль	J_ROLE	V(20)	обязательное поле
Доплата	J_BONUS	N(2)	

* – в отношении УЧАСТИЕ первичный ключ состоит из первых 3-х полей этого отношения.

Логическое проектирование реляционной БД

1.6. Нормализация полученных отношений (до 3НФ)

Механизм нормализации подразумевает определённую последовательность преобразования отношений к третьей нормальной форме.

1НФ. Для приведения таблиц к 1НФ требуется составить прямоугольные таблицы (одно значение атрибута – одна ячейка таблицы) и разбить сложные атрибуты на простые.

Логическое проектирование реляционной БД

Разделим атрибут Фамилия, имя, отчество на два атрибута Фамилия и Имя, отчество, Паспортные данные на Номер паспорта (уникальный), Дата выдачи и Кем выдан, а Данные об образовании – на Вид образования, Специальность, Номер диплома и Год окончания учебного заведения.

Многозначные атрибуты Комнаты и Телефоны из отношения ОТДЕЛЫ вынесем в отдельное отношение КОМНАТЫ, а домашние и мобильные телефоны и адреса сотрудников – в отношение АДРЕСА-ТЕЛЕФОНЫ. Так как в комнате может не быть телефона, первичный ключ отношения КОМНАТЫ не определен (ПК не может содержать null-значения), но на этих атрибутах можно определить составной уникальный ключ. В отношении АДРЕСА-ТЕЛЕФОНЫ также нет потенциальных ключей: оставим это отношение без первичного ключа, т.к. на это отношение никто не ссылается. Данные об образовании сотрудников также вынесем в отдельное отношение.

Логическое проектирование реляционной БД

Что касается рабочих телефонов сотрудников, то один из этих номеров – основной – определяется рабочим местом сотрудника (рассматриваются только стационарные телефоны). Будем хранить этот номер в атрибуте Рабочий телефон. Наличие других номеров зависит от того, есть ли в том же помещении (комнате) другие сотрудники, имеющие стационарные телефоны. Добавим в отношение СОТРУДНИКИ атрибут Номер комнаты, чтобы дополнительные номера телефонов сотрудника можно было вычислить из других кортежей с таким же номером комнаты.

Связь между отношениями СОТРУДНИКИ и КОМНАТЫ реализуем через составной внешний ключ (Номер комнаты, Рабочий телефон). Мы также удалим вычисляемый атрибут Полученная сумма из отношения ПРОЕКТЫ, т.к. он является суммой значений аналогичного атрибута из отношения ЭТАПЫ ПРОЕКТОВ. Но атрибут Стоимость проекта оставим, т.к. она фигурирует в документации по проекту. А для обеспечения логической целостности данных предусмотрим в приложении проверку того, что сумма по всем этапам совпадает со стоимостью проекта.

Логическое проектирование реляционной БД

2НФ. В нашем случае составные первичные ключи имеют отношения ЭТАПЫ ПРОЕКТА и УЧАСТИЕ. Неключевые атрибуты этих отношений функционально полно зависят от составных первичных ключей.

Логическое проектирование реляционной БД

ЗНФ. В отношении ПРОЕКТЫ атрибут Данные заказчика зависит от атрибута Заказчик, а не от первичного ключа, поэтому его следует вынести в отдельное отношение ЗАКАЗЧИКИ. Но при этом первичным ключом нового отношения станет атрибут Заказчик, т.е. длинная символьная строка. Целесообразнее перенести в новое отношение атрибуты Заказчик и Данные заказчика и ввести для него суррогатный ПК. Так как с каждым заказчиком может быть связано несколько проектов, связь между отношениями ЗАКАЗЧИКИ и ПРОЕКТЫ будет 1:n и суррогатный ПК станет внешним ключом для отношения ПРОЕКТЫ.

В отношении СОТРУДНИКИ атрибут Оклад зависит от атрибута Должность. Поступим с этой транзитивной зависимостью так же, как в предыдущем случае: создадим отношение ДОЛЖНОСТИ, перенесём в него атрибуты Должность и Оклад, а первичным ключом сделаем название должности.

В отношениях СОТРУДНИКИ и ОБРАЗОВАНИЕ атрибуты (Дата выдачи и Кем выдан) и (Номер диплома и Год окончания учебного заведения) зависят не от первичного ключа, а от атрибутов соответственно Номер паспорта и Специальность. Но если мы выделим их в отдельное отношение, то получим связи типа 1:1. Следовательно, здесь декомпозиция нецелесообразна.

Логическое проектирование реляционной БД

4НФ. Отношение АДРЕСА-ТЕЛЕФОНЫ нарушают 4НФ, т.к. не всякий телефон привязан к конкретному адресу (т.е. мы имеем две многозначных зависимости в одном отношении). Но выделять Телефоны в отдельное отношение не стоит, т.к. эти сведения носят справочный характер и не требуется их автоматическая обработка. Отношения, полученные после нормализации, приведены в табл. 6-15.

Логическое проектирование реляционной БД

Таблица 6. Схема отношения ОТДЕЛЫ (Departs)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Аббревиатура отдела	D_ID	V(12)	первичный ключ
Название отдела	D_NAME	V(100)	обязательное поле

Таблица 7. Схема отношения КОМНАТЫ (Rooms)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Отдел	R_DEPART	V(12)	внешний ключ (к Departs)
Номер комнаты	R_ROOM	N(4)	составной уникальный ключ
Телефон	R_PHONE	V(20)	

Таблица 8. Схема отношения ДОЛЖНОСТИ (Posts)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Название должности	P_POST	V(30)	первичный ключ
Оклад	P_SAL	N(8,2)	обязательное поле, > 4500 руб.

Логическое проектирование реляционной БД

Таблица 9. Схема отношения СОТРУДНИКИ (Employees)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Идентификатор сотрудника	E_ID	N(4)	суррогатный первичный ключ
Фамилия	E_FNAME	V(25)	обязательное поле
Имя, отчество	E_LNAME	V(30)	обязательное поле
Дата рождения	E_BORN	D	обязательное поле
Пол	E_SEX	C(1)	обязательное поле
Серия и номер паспорта	E_PASP	C(10)	обязательное уникальное поле
Когда выдан паспорт	E_DATE	D	обязательное поле
Кем выдан паспорт	E_GIVEN	V(50)	обязательное поле
ИНН	E_INN	C(12)	обязательное уникальное поле
Номер пенсионного страхового свидетельства	E_PENS	C(14)	обязательное уникальное поле
Отдел	E_DEPART	V(12)	внешний ключ (к Departs)
Должность	E_POST	V(30)	внешний ключ (к Posts)
Номер комнаты	E_ROOM	N(4)	составной внешний ключ (к Rooms)
Рабочий телефон	E_PHONE	V(20)	
Логин	E_LOGIN	V(30)	

Логическое проектирование реляционной БД

Таблица 10. Схема отношения ОБРАЗОВАНИЕ (Edu)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Идентификатор сотрудника	U_ID	N(4)	внешний ключ (к Employees)
Вид образования	U_TYPE	V(20)	обязательное поле
Специальность	U_SPEC	V(40)	
Номер диплома	U_DIPLOM	V(15)	
Год окончания учебного заведения	U_YEAR	N(4)	обязательное поле

Таблица 11. Схема отношения АДРЕСА-ТЕЛЕФОНЫ (AdrTel)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Идентификатор сотрудника	A_ID	N(4)	внешний ключ (к Employees)
Адрес	A_ADDR	V(50)	
Телефон	A_PHONE	V(30)	

Логическое проектирование реляционной БД

Таблица 12. Схема отношения ЗАКАЗЧИКИ (Clients)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Номер заказчика	C_ID	N(4)	суррогатный первичный ключ
Заказчик	C_COMPANY	V(40)	обязательное поле
Адрес заказчика	C_ADR	V(50)	обязательное поле
Контактное лицо	C_PERSON	V(50)	обязательное поле
Телефон	C_PHONE	V(30)	

таблица 13. Схема отношения ПРОЕКТЫ (Projects)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Номер проекта	P_ID	N(6)	обязательное уникальное поле
Название проекта	P_TITLE	V(100)	обязательное поле
Сокращённое название	P_ABBR	C(10)	первичный ключ
Отдел	P_DEPART	V(12)	внешний ключ (к Departs)
Заказчик	P_COMPANY	N(4)	внешний ключ (к Clients)
Руководитель	P_CHIEF	N(4)	внешний ключ (к Employees)
Дата начала проекта	P_BEGIN	D	обязательное поле
Дата окончания проекта	P_END	D	обязательное поле, больше даты начала проекта
Реальная дата окончания	P_FINISH	D	больше даты начала проекта
Стоимость проекта	P_COST	N(10)	обязательное поле, > 0

Логическое проектирование реляционной БД

Таблица 14. Схема отношения ЭТАПЫ ПРОЕКТА (Stages)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>	
Проект	S_PRO	C(10)	внешний ключ (к Projects)	составной первичный ключ
Номер этапа	S_NUM	N(2)		
Название этапа	S_TITLE	V(200)	обязательное поле	
Дата начала этапа	S_BEGIN	D	обязательное поле	
Дата окончания этапа	S_END	D	обязательное поле, больше даты начала этапа	
Реальная дата окончания	S_FINISH	D	больше даты начала этапа	
Стоимость этапа	S_COST	N(10)	обязательное поле	
Полученная сумма по этапу	S_SUM	N(10)	обязательное поле, значение по умолчанию – 0	
Форма отчётности	S_FORM	V(100)	обязательное поле	

Таблица 15. Схема отношения УЧАСТИЕ (Job)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>	
Проект	J_PRO	C(10)	внешний ключ (к Projects)	составной ПК
Сотрудник	J_EMP	N(4)	внешний ключ (к Employees)	
Роль	J_ROLE	V(20)	обязательное поле	
Доплата	J_BONUS	N(2)		

Логическое проектирование реляционной БД

Таблица 14. Схема отношения ЭТАПЫ ПРОЕКТА (Stages)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>	
Проект	S_PRO	C(10)	внешний ключ (к Projects)	составной первичный ключ
Номер этапа	S_NUM	N(2)		
Название этапа	S_TITLE	V(200)	обязательное поле	
Дата начала этапа	S_BEGIN	D	обязательное поле	
Дата окончания этапа	S_END	D	обязательное поле, больше даты начала этапа	
Реальная дата окончания	S_FINISH	D	больше даты начала этапа	
Стоимость этапа	S_COST	N(10)	обязательное поле	
Полученная сумма по этапу	S_SUM	N(10)	обязательное поле, значение по умолчанию – 0	
Форма отчётности	S_FORM	V(100)	обязательное поле	

Таблица 15. Схема отношения УЧАСТИЕ (Job)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>	
Проект	J_PRO	C(10)	внешний ключ (к Projects)	составной ПК
Сотрудник	J_EMP	N(4)	внешний ключ (к Employees)	
Роль	J_ROLE	V(20)	обязательное поле	
Доплата	J_BONUS	N(2)		

Логическое проектирование реляционной БД

Схема базы данных после нормализации приведена на рис. 7.

1.7. Определение дополнительных ограничений целостности, которые не указаны в табл. 6–15.

1. Вид образования может принимать одно из следующих значений: 'начальное', 'среднее', 'среднее - специальное', 'высшее'.
2. Атрибут Роль может принимать одно из двух значений: 'исполнитель' или 'консультант'.
3. В поле Доплата хранится величина доплаты сотруднику за участие в проекте (в процентах к его окладу). Значение поля больше либо равно 0.
4. Нумерация в поле Номер этапа начинается с 1 и является непрерывной для каждого проекта.

Логическое проектирование реляционной БД

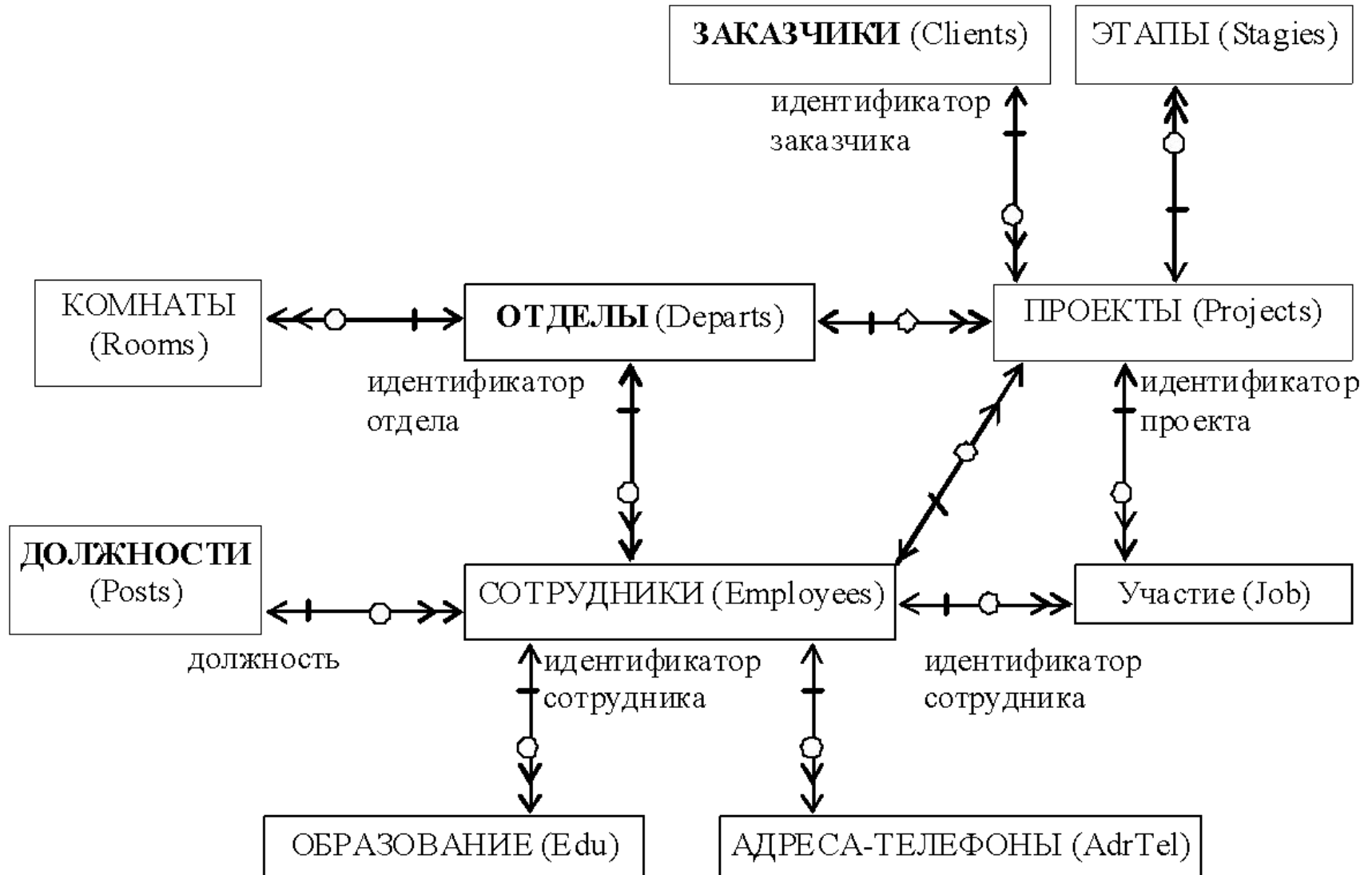


Рис. 7. Окончательная схема БД проектной организации

Логическое проектирование реляционной БД

5. Дата начала первого этапа проекта должна соответствовать началу проекта в целом, дата завершения последнего этапа должна соответствовать завершению проекта в целом. Этапы не должны пересекаться по времени и между ними не должно быть разрывов.
 6. Стоимость проекта должна быть равна сумме стоимостей всех этапов этого проекта.
- Ограничения 4-6 не реализованы в схеме отношений. В реальных БД подобные ограничения целостности реализуются вручную или программно (через внешнее приложение или специальную процедуру контроля данных – триггер).

Логическое проектирование реляционной БД

- 1.8. Описание групп пользователей и прав доступа

Опишем для каждой группы пользователей права доступа к каждой таблице. Права доступа должны быть распределены так, чтобы для каждого объекта БД был хотя бы один пользователь, который имеет право добавлять и удалять данные из объекта. Права приведены в табл. 16. Используются следующие сокращения:

- s – чтение данных (select);
- i – добавление данных (insert);
- u – модификация данных (update);
- d – удаление данных(delete).

Логическое проектирование реляционной БД

Таблица 16. Права доступа к таблицам для групп пользователей

Таблицы	Группы пользователей (роли)				
	Руководители организации	Сотрудники отд. кадров	Руководители проектов	Бухгалтеры	Участники проектов
Отделы	S	SIUD	S	S	
Комнаты	S	SUID	S	S	S
Должности	SIUD			S	
Сотрудники	S	SUID	S	S	
Адреса-телефоны	S	SUID	S	S	
Образование	S	SUID	S	S	
Заказчики	SIUD		S		
Проекты	SIUD		S		
Этапы проектов	SIUD		SUI		
Участие	S		S	S	

Логическое проектирование реляционной БД

- Права на изменение данных в таблице УЧАСТИЕ назначает руководитель проекта, администратор БД (или администратор безопасности).

Логическое проектирование реляционной БД

1.9. Реализация проекта базы данных

Создание таблиц

Отношение Departs (отделы):

- create table departs (
• d_id varchar(12) primary key,
• d_name varchar(100) not null);

Отношение Rooms (комнаты):

- create table rooms (
• d_dept varchar(12) references departs(d_id),
• r_room numeric(4) not null,
• r_phone varchar(20),
• unique(r_room, r_phone));

Отношение Posts (должности):

- create table posts (
• p_post varchar(30) primary key,
• p_salary numeric(8,2) not null check(p_salary >= 4500));

Логическое проектирование реляционной БД

Отношение Projects (проекты):

- create table projects (
 - p_id numeric(6) not null unique,
 - p_title varchar(100) not null,
 - p_abbr char(10) primary key,
 - p_depart varchar(12) references departs,
 - p_company numeric(4) references clients,
 - p_chief numeric(4) references employees,
 - p_begin date not null,
 - p_end date not null,
 - p_finish date,
 - p_cost numeric(10) not null check(p_cost>0),
 - check (p_end>p_begin),
 - check (p_finish is null or p_finish>p_begin));

Отношение Stages (этапы проектов):

- create table stages (
 - s_pro char(10) references projects,
 - s_num numeric(2) not null,
 - s_title varchar(200) not null,
 - s_begin date not null,
 - s_end date not null,
 - s_finish date,
 - s_cost numeric(10) not null,
 - s_sum numeric(10) not null,
 - s_form varchar(100) not null,
 - check (s_cost>0),
 - check (s_end>s_begin),
 - check (s_finish is null or s_finish>s_begin));

Логическое проектирование реляционной БД

1.9. Создание представлений (готовых запросов)

Приведём примеры нескольких готовых запросов (представлений):

- Список всех текущих проектов (`sysdate` – функция, возвращающая текущую дату, определена в СУБД Oracle; в других системах аналогичная функция может называться по-другому, например, `getdate()` в Transact-SQL, `now()` в MS Access, `currrdate()` в MySQL и т.д.):
 - create view **curr_projects** as
 - select *
 - from projects
 - where p_begin <= sysdate and sysdate <= p_end;
- Определение суммы по текущим проектам, полученной на текущую дату:
 - create or replace view **summ** (title, cost, total) as
 - select p_title, p_cost, sum(s_sum)
 - from curr_projects, stages
 - where p_abbrev=s_pro
 - group by p_title, p_cost;

Логическое проектирование реляционной БД

1.10. Создание индексов

Анализ готовых запросов показывает, что для повышения эффективности работы с данными необходимо создать индексы для всех внешних ключей. Приведём пример создания индексов:

- `create index p_chief on projects(p_chief);`