

# Лекция 13

## Семантическое моделирование

МГТУ им. Н.Э.Баумана. ФФ12

## Уникальные идентификаторы типов сущности

При определении типа сущности необходимо гарантировать, что каждый экземпляр сущности является отличимым от любого другого экземпляра той же сущности.

БД для поддержки работы книжного склада.

На складе могут храниться произвольные части тиража любого издания любой книги.

Прообраз типа сущности — набор одноименных книг одного автора, вышедших в одном издании.

<b>КНИГА</b>
автор
название
номер издания
издательство
год издания
<b>ISBN</b>
количество книг на складе

Рис. 10.1.

При проектировании БД для поддержки работы библиотеки прообраз типа сущности — книга, т.к. ей присваивается уникальный библиотечный номер.

В ER-модели у экземпляра типа сущности не может быть назначаемого пользователем имени или назначаемого системой внешнего уникального идентификатора.

Экземпляр типа сущности идентифицируется только своими индивидуальными характеристиками: значениями свойств и экземплярами типов связи.

Уникальным идентификатором сущности может быть свойство, комбинация свойств, связь, комбинация связей или комбинация связей и свойств, уникально отличающая любой экземпляр сущности от других экземпляров сущности того же типа.

# Нормальные формы ER-диаграмм (1-3)

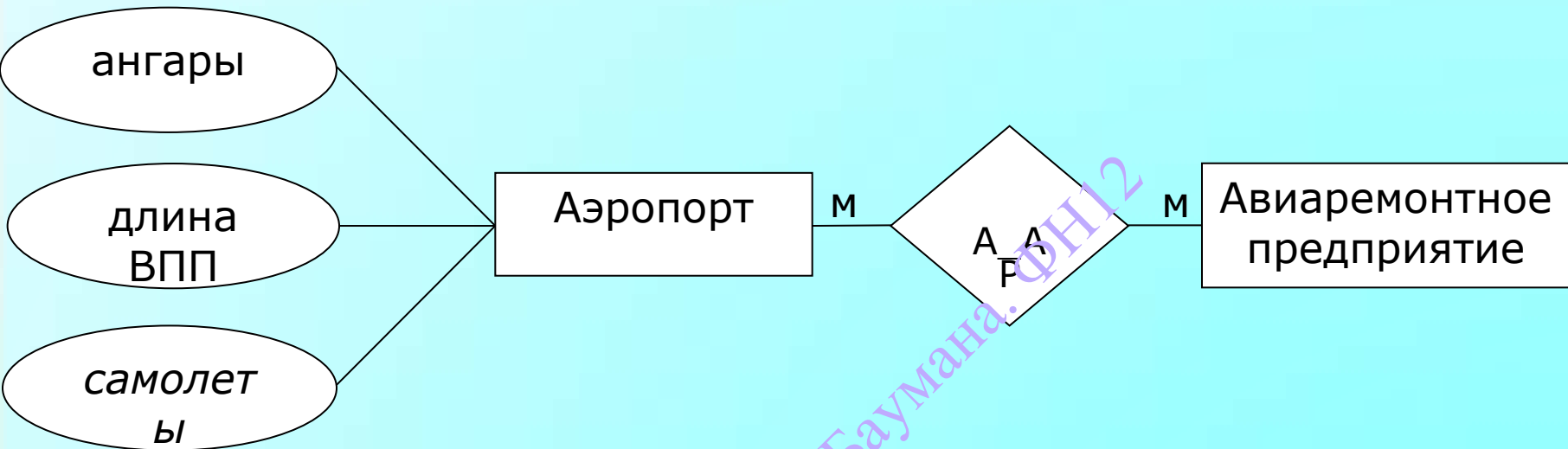


Рис. 10.2 ER-диаграмма.

Тип сущности АЭРОПОРТ не удовлетворяет требованию *1NF*

Свойство самолеты — **многозначное** свойство характеризует самолеты.

На ER-диаграмме не отражен тот факт, что авиаремонтное предприятие ремонтирует самолеты, а не аэропорты.

## Первая нормальная форма ER-диаграммы

В **первой нормальной форме** ER-диаграммы устраняются свойства, содержащие множественные значения.

Выделим тип сущности САМОЛЕТ.

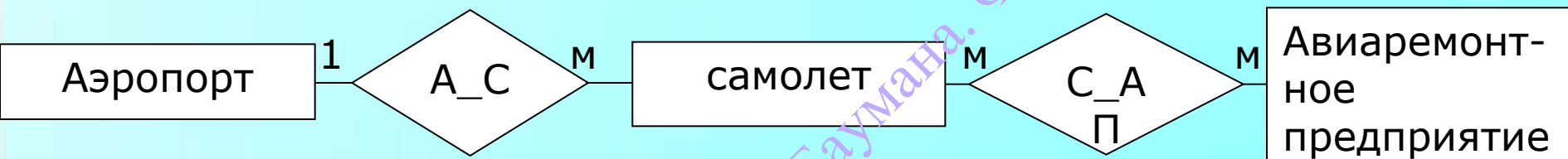


Рис. 10.3 ER-диаграмма в 1НФ

Связь А\_С показывает, что к одному аэропорту приписывается несколько самолетов.

Связь С\_А П показывает, что каждый самолет из группы самолетов обслуживается любым транспортным предприятием из группы предприятий.

## Вторая нормальная форма ER-диаграммы

Во **второй нормальной форме** устраняются свойства, зависящие только от части уникального идентификатора. Эта часть уникального идентификатора определяет отдельную сущность.

БД СЛУЖЕБНОЕ РАСПИСАНИЕ предназначена для хранения данных о рейсах самолетов.

Требования к БД:

- 1) у каждого рейса имеется уникальный заранее приписанный номер;
- 2) характеристикой рейса является дата и время его совершения,
- 3) бортовой номер самолета определяется парой (номер рейса, дата-время вылета).

Тип сущности ЭЛЕМЕНТ РАСПИСАНИЯ

Уникальный идентификатор типа сущности ЭЛЕМЕНТ РАСПИСАНИЯ

—пара свойств *<номер рейса, дата-время вылета>*.

Тип сущности ГОРОД. Связь ЭЛЕМЕНТ РАСПИСАНИЯ — ГОРОД

Каждый экземпляр связи определяется значением свойства *Рейс#*.

Имеются ФЗ:

1. {рейс#, дата-время вылета} → Борт#;
2. Рейс# → аэропорт вылета;
3. Рейс# → аэропорт назначения;
4. Борт# → тип самолета;
5. Рейс# определяет экземпляр связи Р\_Г;

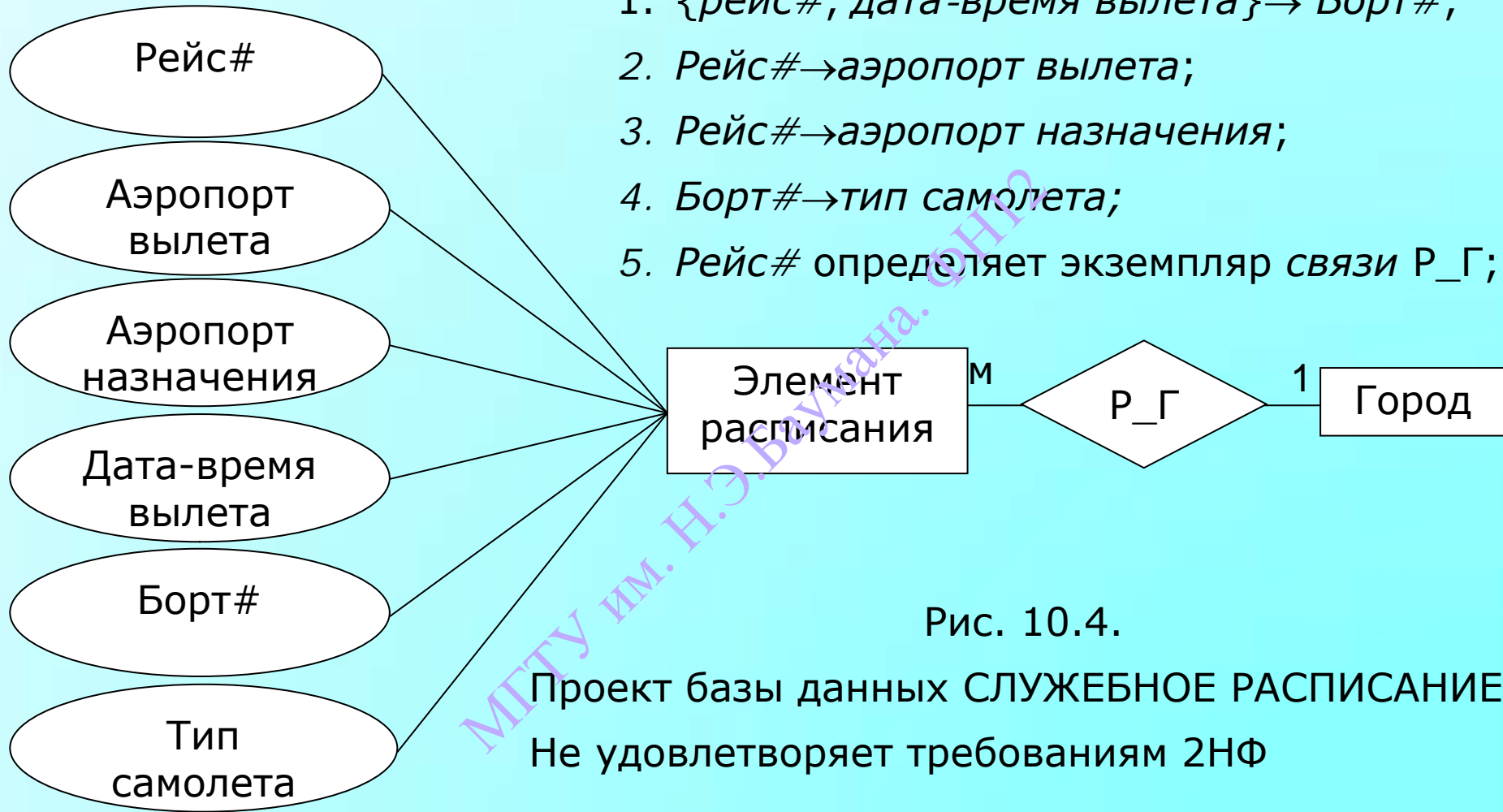


Рис. 10.4.

Проект базы данных СЛУЖЕБНОЕ РАСПИСАНИЕ.  
Не удовлетворяет требованиям 2НФ

Избыточность: значения свойств *аэропорт вылета* и *аэропорт назначения* повторяются при каждом повторении номера рейса.

Сущность ЭЛЕМЕНТ РАСПИСАНИЯ включает в себя тип сущности РЕЙС.

- РЕЙС (рейс#, аэропорт вылета, аэропорт назначения);
- ЭЛЕМЕНТ РАСПИСАНИЯ (дата-время вылета, борт#, тип самолета);
- ГОРОД.

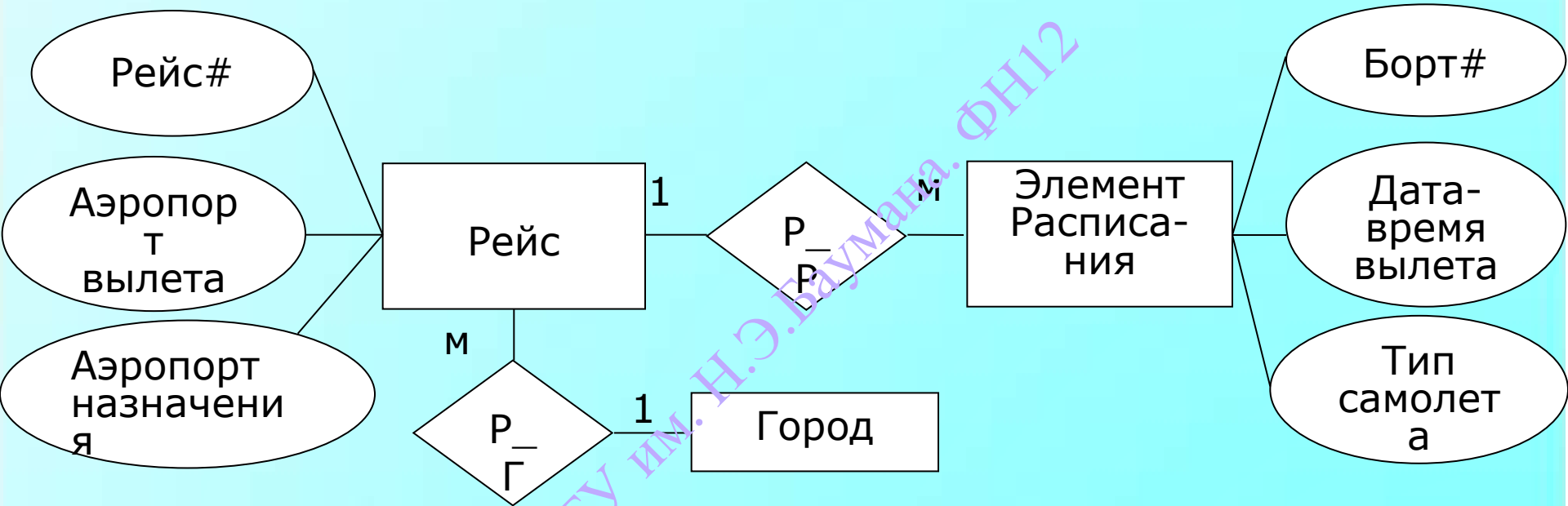


Рис. 10.5(а) Проект базы данных СЛУЖЕБНОЕ РАСПИСАНИЕ II.

Уникальный идентификатор сущности РЕЙС — Рейс#,  
уникальный идентификатор ЭЛЕМЕНТ РАСПИСАНИЯ — (дата-время вылета, Борт #, Тип самолета, связь P\_P (указывающая рейс)).

Все сущности находятся во второй нормальной форме.

## Третья нормальная форма ER-диаграммы

В **третьей нормальной форме** устраняются свойства, зависящие от свойств, не входящих в уникальный идентификатор.

Эти свойства являются основой отдельной сущности.

Тип сущности ЭЛЕМЕНТ РАСПИСАНИЯ.

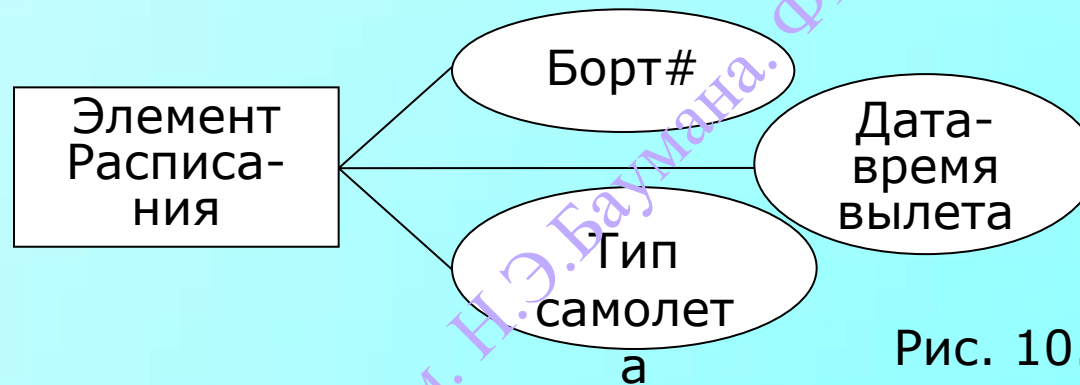


Рис. 10.6

Имеются ФЗ:

- 1) {P\_R, *дата-время вылета*} → *тип самолета*;
- 2) {P\_R, *дата-время вылета*} → Борт#;
- 3) Борт# → *тип самолета*;
- 4) {P\_R, *дата вылета*} → *тип самолета*; Транз.ФЗ

Требования третьей нормальной формы нарушены.

Тип сущности ЭЛЕМЕНТ РАСПИСАНИЯ включает в себя тип сущности САМОЛЕТ. Это вызывает избыточность хранения.

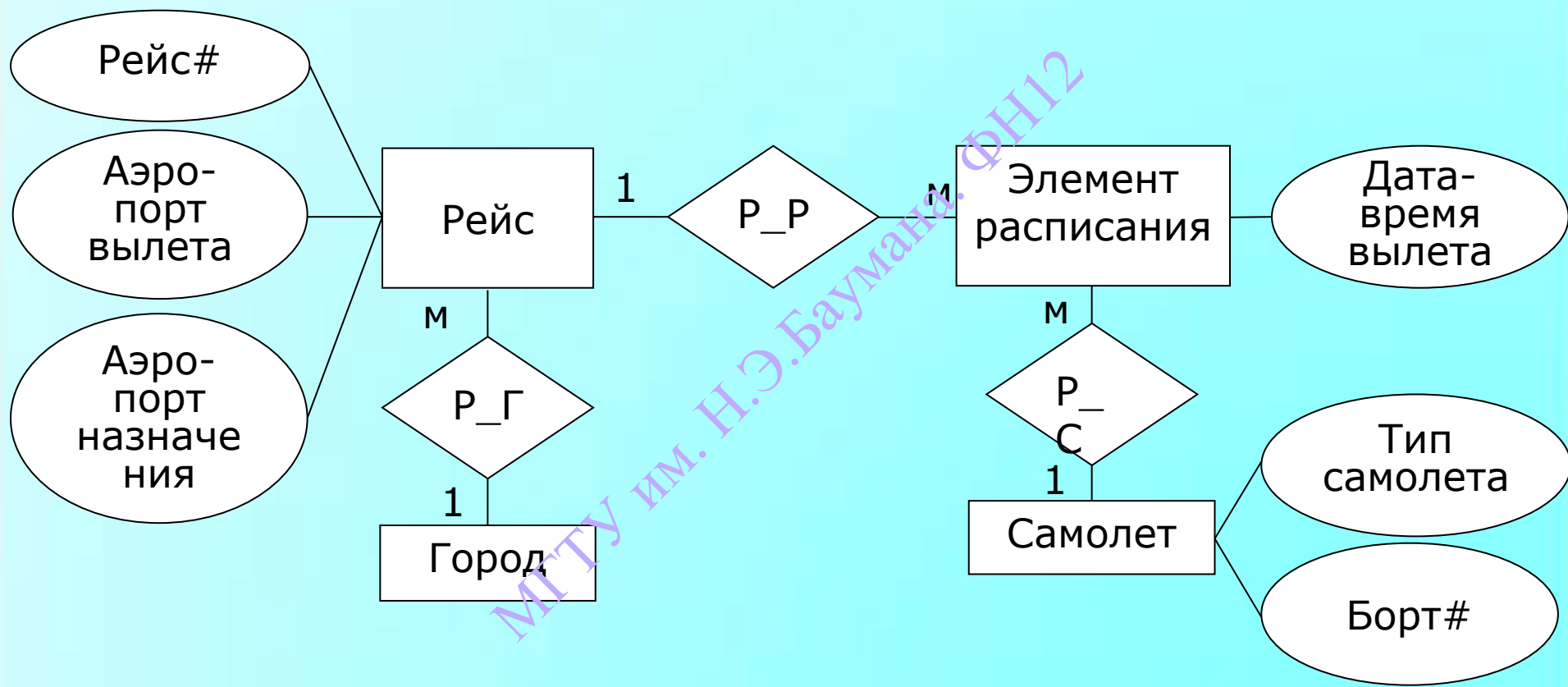


Рис. 10.7. Пример приведения ER-диаграммы к 3НФ

## Более сложные элементы ER-модели

- **Подтипы и супертипы сущностей.** В *ER-модели* поддерживается возможность наследования *типа сущности*, исходя из одного или нескольких **супертипов**.
- **Домены.** В *ER-модели* разрешается определять потенциально допустимое множество значений свойств сущности (домена).
- **Уточняемые степени связи.** В *ER-модели* разрешается указывать на *конце связи* ее максимальную или обязательную степень.
- **Взаимно исключающие связи.** Для заданного типа сущности можно определить такой набор типов связи с другими *типами сущности*, что для каждого *экземпляра* заданного типа сущности может или должен существовать экземпляр только одной *связи* из этого набора.  
Если набор связей является необязательным, то экземпляр только одной связи может существовать, если набор связей обязателен, то экземпляр только одной связи должен существовать.

• **Каскадные удаления экземпляров сущностей.** Некоторые *связи* бывают настолько сильными (в случае связи 1:M), что при удалении опорного экземпляра сущности (соответствующего концу связи «один») нужно удалить и все экземпляры сущности, соответствующие концу связи «многие». Соответствующее требование каскадного удаления специфицируют при определении связи.

## Подтипы и супертипы сущностей

Тип сущности, на основе которого определяются подтипы, называется **супертипом**.

Подтипы должны образовывать полное множество, т. е. любой экземпляр супертипа должен относиться к некоторому подтипу.

Если у типа сущности  $A$  имеются подтипы  $B_1, B_2, \dots, B_n$ , то:

- любой экземпляр типа сущности  $B_1, B_2, \dots, B_n$  является экземпляром типа сущности  $A$  (включение);
- если  $a$  является экземпляром типа сущности  $A$ , то  $a$  является экземпляром некоторого подтипа сущности  $B_i$  ( $i = 1, 2, \dots, n$ ) (отсутствие собственных экземпляров у супертипа сущности);
- ни для каких подтипов  $B_i$  и  $B_j$  ( $i, j = 1, 2, \dots, n$ ) не существует экземпляра, типом которого одновременно являются типы сущности  $B_i$  и  $B_j$  (разъединенность подтипов).

## Иерархия типов сущности

Сущность-подтип является типом сущности, может иметь собственные подтипы. Некоторый тип сущности, его непосредственные подтипы, подтипы этих подтипов и т.д. все вместе образуют **иерархию типов сущности**.

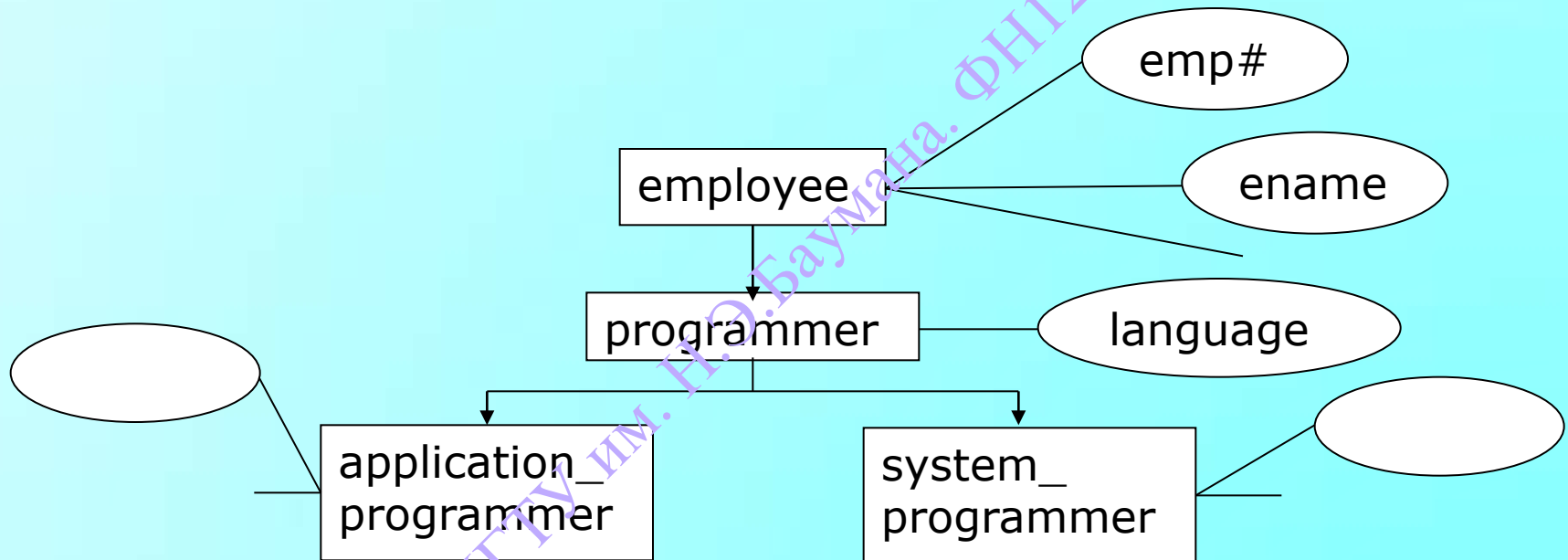


Рис. 10.8 Иерархия типов сущности EMPLOYEE

## Взаимно исключающие связи

Самолет может находиться в рабочем состоянии, и тогда у него имеется один и только один пилот.

Самолет может находиться на ремонте на одном из нескольких возможных авиаремонтных предприятий (каждое предприятие может производить ремонт нескольких самолетов).

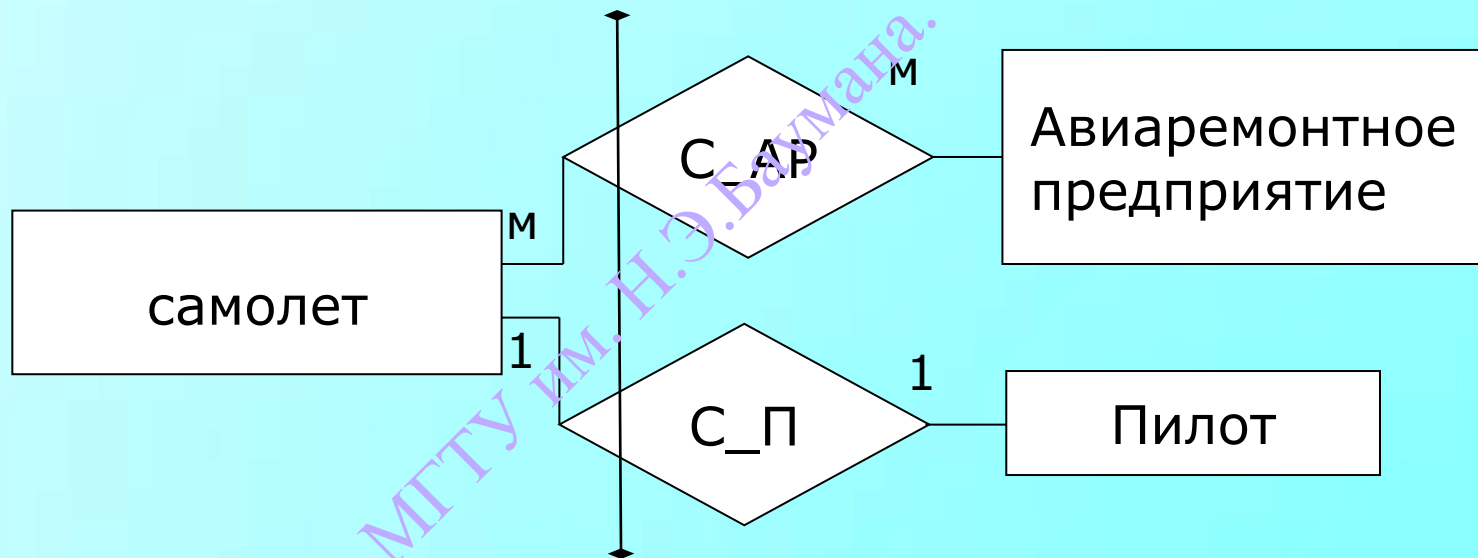


Рис. 10.9. Пример ER-диаграммы со взаимно исключающими связями

Для каждого экземпляра типа сущности САМОЛЕТ должен существовать экземпляр одной из указанных связей.

Диаграмма со взаимно исключающими связями может быть преобразована к диаграмме без взаимно исключающих связей путем введения подтипов.

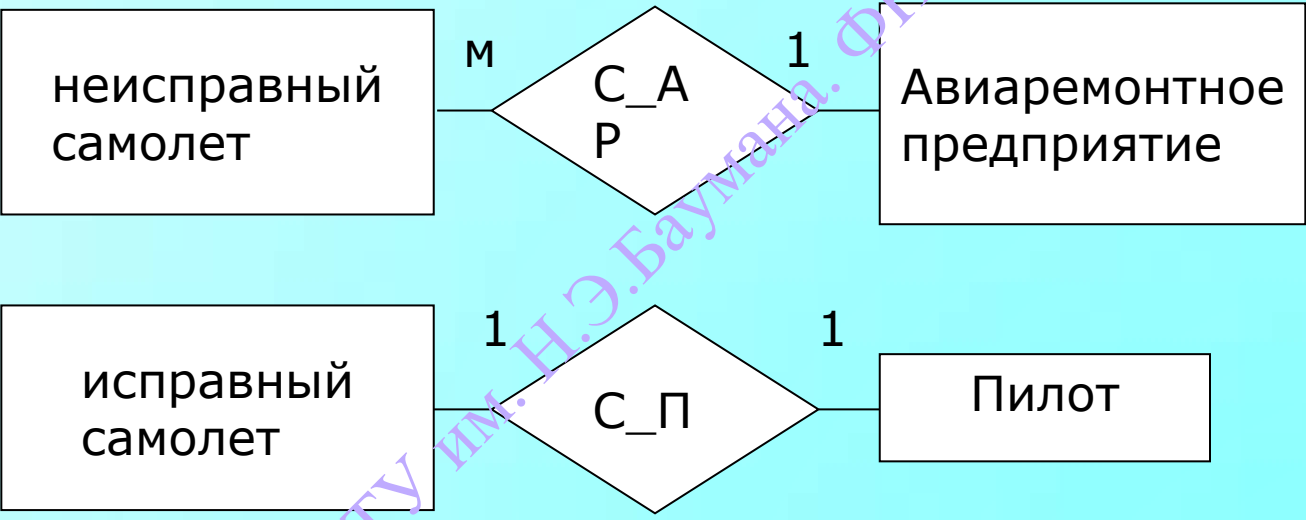


Рис. 10.10 ER-диаграмма без взаимно исключающих связей

# Проектирование базы данных с помощью метода ER-моделирования

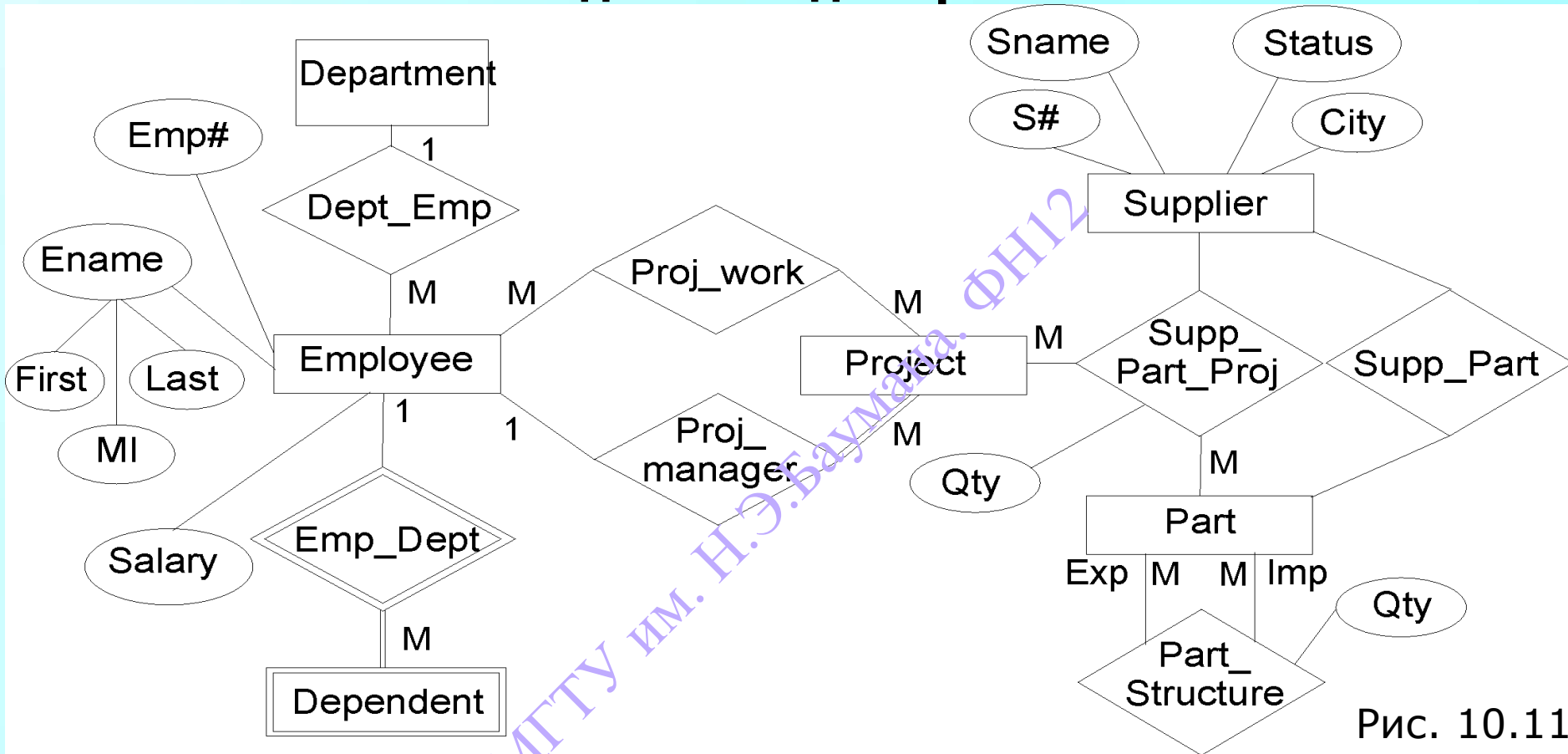


Рис. 10.11

## Сильные типы сущностей:

- DEPARTMENT
- EMPLOYEE
- SUPPLIER
- PART
- PROJECT

## Базовые переменные-отношения:

- DEPT(DEPT# PRIMARY KEY ..., ... );
- EMP (EMP#, ..., ... );
- P(P#, ..., ... );
- S(S#, ..., ... );
- J(J#, ..., ... );

## Связи типа "многие ко многим (М:М) "

- PROJ\_WORK (между работниками и проектами)
- SUPP\_PART (между поставщиками и деталями)
- SUPP\_PART\_PROJ (между поставщиками, деталями и проектами)
- PART STRUCTURE (между деталями-узлами и деталями, входящими в их состав)

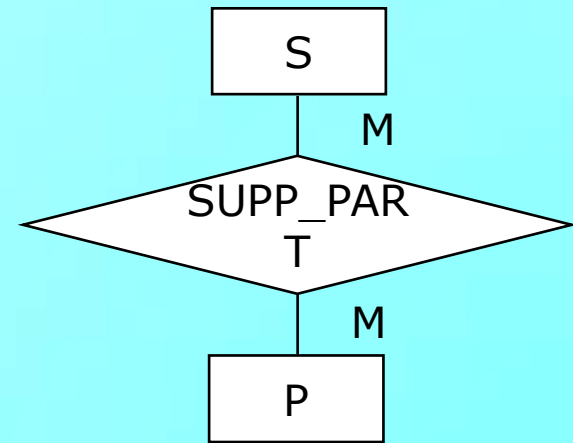
Каждая такая связь отображается и в базовую переменную-отношение.

Связь SUPP\_PART — базовая переменная-отношение SP.

SP (S# ... , P#..., ... )

FOREIGN KEY (S#) REFERENCES S

FOREIGN KEY (P#) REFERENCES P



Для каждого из ВК задается набор правил внешних ключей (правило обновления UPDATE и правило удаления DELETE). Эти правила нельзя вывести или установить с только помощью ER-диаграммы.

SP (S#... , P# ... , ... ) FOREIGN KEY (S#) REFERENCES S  
ON DELETE RESTRICT  
ON UPDATE CASCADE  
FOREIGN KEY (P#) REFERENCES P  
ON DELETE RESTRICT  
ON UPDATE CASCADE  
PRIMARY KEY (S#, P#);

Способы задания первичного ключа

1. Комбинация внешних ключей, идентифицирующих участников связи

Это возможно в случае, если:

- a) данная комбинация имеет уникальное значение для каждого экземпляра данной переменной-отношения;
- b) разработчик базы данных не возражает против использования составных первичных ключей;

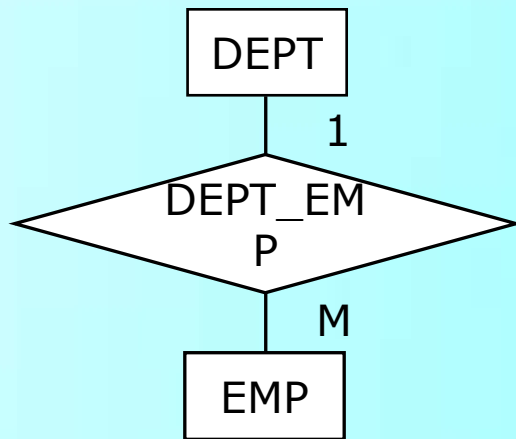
2. Новое несоставное суррогатное свойство "номер поставки".

## Связи типа "многие к одному (М:1)"

- PROJ\_MANAGER (между проектами и их менеджерами (сильные типы сущностей))
- DEPT\_EMP (между работниками и отделами (сильные типы сущностей))
- EMP\_DEP (руководящими работниками и между подчиненными (слабый тип сущности DEPENDENT))

Пример. Связь DEPT\_EMP.

В переменную-отношение EMP (сторона "многие"), введем внешний ключ, ссылающийся на переменную-отношение DEPT(сторона "один").



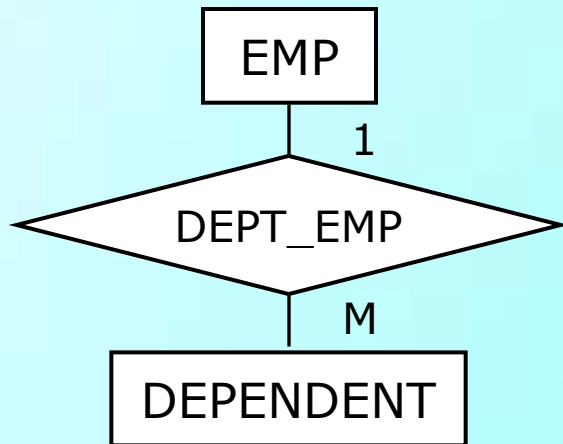
EMP (EMP#, DEPT#... , ...) PRIMARY KEY(EMP#)

FOREIGN KEY (DEPT#) REFERENCES DEPT

ON DELETE ...

ON UPDATE ...;

## Слабые сущности



DEPENDENT (EMP#, ...)  
PRIMARY KEY (EMP#, DEP\_NAME)  
FOREIGN KEY ( EMP# ) REFERENCES EMP  
ON DELETE CASCADE  
ON UPDATE CASCADE ;

Есть два варианта выбора первичного ключа.

1. комбинация внешнего ключа и ключевого свойства слабой сущности, представленного на ER-диаграмме.
2. введение нового несоставного суррогатного атрибута.

## Свойства

Каждое показанное на ER-диаграмме свойство отображается в отдельный атрибут в соответствующей переменной-отношении.

Исключение — многозначное свойство. Для многозначного свойства требуется создать новую переменную-отношение, это следует из принципов нормализации.

Атрибуты, соответствующие необязательным свойствам, могут содержать неопределенные значения;  
атрибуты, соответствующие обязательным свойствам, – нет.

# Алгоритм получения реляционной схемы из ER-диаграммы

1. Каждый простой тип сущности отображается в таблицу. Имя сущности становится именем таблицы. Экземплярам типа сущности соответствуют строки соответствующей таблицы.
2. Каждое свойство отображается столбцом таблицы с тем же именем.
3. Компоненты уникального идентификатора сущности отображаются в первичный ключ таблицы.

Если имеется несколько возможных уникальных идентификаторов, для первичного ключа выбирается наиболее характерный.

Если в состав уникального идентификатора входят связи, к числу столбцов первичного ключа добавляется копия уникального идентификатора сущности, находящейся на дальнем конце связи.

Для именованния этих столбцов используются имена концов связей и/или имена парных типов сущностей.

4. Связи «М:1» становятся внешними ключами. Столбцы, соответствующие уникальному идентификатору сущности на конце связи «один» составляют внешний ключ таблицы, соответствующей типу сущности на конце связи «многие».

Необязательные связи соответствуют столбцам внешнего ключа, допускающим наличие неопределенных значений;  
обязательные связи – столбцам, не допускающим неопределенных значений.

5. Если между двумя типами сущности А и В имеется связь «1:1», то соответствующий внешний ключ может быть объявлен как в таблице А, так и в таблице В.

Ограничение «степень конца связи должна равняться единице» — соответствующий столбец должен быть дополнительно специфицирован как возможный ключ таблицы (спецификация UNIQUE).

6. Для поддержки связи M:M между типами сущности A и B создается дополнительная таблица AB с двумя столбцами, один содержит уникальные идентификаторы экземпляров сущности A ( $UId(a_i)$ ), а другой – уникальные идентификаторы экземпляров сущности B ( $UId(b_j)$ ).

Если в экземпляре связи (M:M) участвуют экземпляры  $a_1, a_2, \dots, a_n$  типа сущности A и экземпляры  $b_1, b_2, \dots, b_m$  типа сущности B, то в таблице AB должны присутствовать все строки вида ( $UId(a_i), UId(b_j)$ ) ( $i=1, 2, \dots, n, j=1, 2, \dots, m$ ).

Индексы создаются для первичного ключа (уникальный индекс), внешних ключей и тех атрибутов, на которых предполагается в основном базировать запросы.

# Представление в реляционной схеме супертипов и подтипов сущности

Если в ER-диаграмме присутствуют подтипы, то возможны два способа их представления в реляционной схеме:

- a) собрать все подтипы в одной таблице;
- b) для каждого подтипа образовать отдельную таблицу.

МГТУ им. Н.Э.Баумана. ФН12

# Представление в реляционной схеме взаимно исключающих связей

Существуют два способа формирования схемы реляционной БД при наличии взаимно исключающих связей (связи «1:M», конец связи «многие» находится на стороне сущности, для которой связи являются взаимно исключающими):

1. общее хранение внешних ключей;
2. раздельное хранение внешних ключей.

## 1. Общее хранение внешних ключей

Внешние ключи всех связанных таблиц имеют общий формат. В таблице сущности, для которой связи являются взаимно исключающими, создаются столбец-идентификатор связи и столбец-идентификатор сущности. Столбец идентификатора связи используется для различения взаимно-исключающих связей. В столбце (столбцах) идентификатора сущности хранятся значения уникального идентификатора сущности на дальнем конце соответствующей связи.

## 2. Раздельное хранение внешних ключей

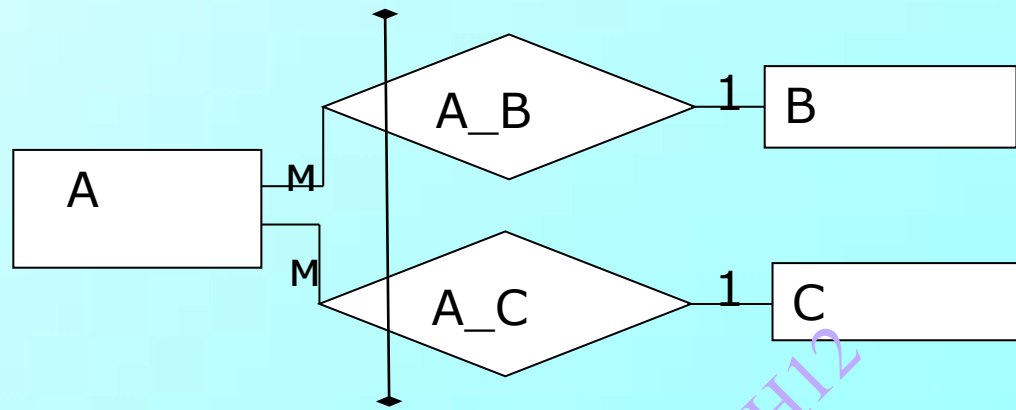
Результирующие внешние ключи не относятся к одному домену. Для каждой связи создаются явные столбцы внешних ключей; все эти столбцы могут содержать неопределенные значения.

**Преимущество подхода 1.** В таблице, соответствующей сущности, появляется всего два дополнительных столбца.

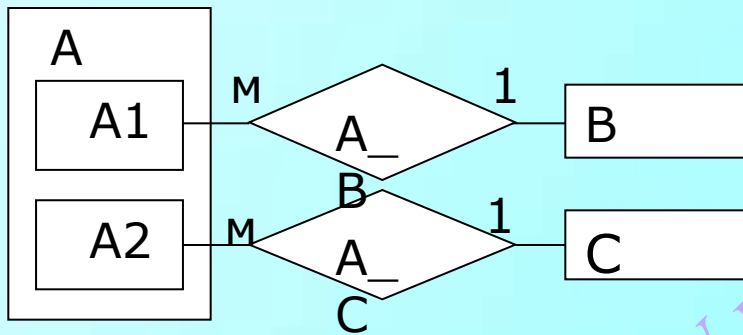
**Преимущество подхода 2.** Соединения являются естественными.

**Недостаток подхода 1.** Усложнение выполнения операции соединения: для соединения одной из альтернативных связей, нужно сначала произвести ограничение таблицы в соответствии с нужным значением столбца, содержащего идентификаторы связей.

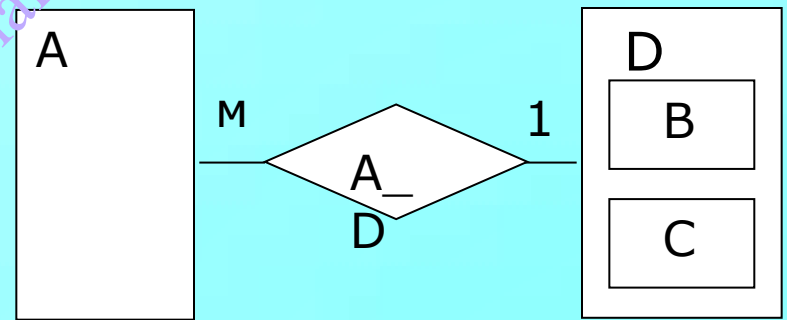
**Недостаток подхода 2.** Требуется иметь столько столбцов, сколько имеется альтернативных связей. В каждом столбце будет содержаться много неопределенных значений (расходы внешней памяти).



а) ER-диаграмма со взаимно-исключающими связями.



б) ER-диаграмма с подтипами сущности без альтернативных связей.



в) ER-диаграмма, в которой альтернативные связи заменены связью с супертипом сущности.

Рис. 10.12. Возможные модификации ER-диаграмм.