

Инфологическое моделирование. ER-диаграммы

Проект базы данных надо начинать с анализа предметной области и выявления требований к ней отдельных пользователей (сотрудников организации, для которых создается база данных). Проектирование обычно поручается человеку (группе лиц) – администратору базы данных (АБД). Им может быть как специально выделенный сотрудник организации, так и будущий пользователь базы данных, достаточно хорошо знакомый с машинной обработкой данных.

Объединяя частные представления о содержимом базы данных, полученные в результате опроса пользователей, и свои представления о данных, которые могут потребоваться в будущих приложениях, АБД сначала создает обобщенное неформальное описание создаваемой базы данных. Это описание, выполненное с использованием естественного языка, математических формул, таблиц, графиков и других средств, понятных всем людям, работающим над проектированием базы данных, называют **инфологической моделью данных**

Инфологическое моделирование. ER-диаграммы

Такая модель полностью независима от физических параметров среды хранения данных. Поэтому инфологическая модель не должна изменяться до тех пор, пока какие-то изменения в реальном мире не потребуют изменения в ней некоторого определения, чтобы эта модель продолжала отражать предметную область.

□ Модель «сущность-связь» (entity-relationship model) предложена американским исследователем в области баз данных Питером Ченом в 1976 году. С тех пор она расширилась и модифицировалась как самим Ченом, так и многими другими исследователями. В различных вариантах она вошла в состав многих автоматизированных средств поддержки проектирования информационных систем.

□ Существует много различных систем построения моделей ER.

□ В настоящее время нет единого стандарта этой модели, но
есть набор общих конструкций, лежащих в основе

Инфологическое моделирование. ER-диаграммы

Базовыми элементами инфологических моделей являются сущности, связи между ними и их свойства (атрибуты).

□ **Сущность** – любой различимый объект (объект, который мы можем отличить от другого), информацию о котором необходимо хранить в базе данных.

Сущностями могут быть люди, места, самолеты, рейсы, цвет и т.д. Необходимо различать такие понятия, как тип сущности и экземпляр сущности.

Понятие **тип сущности** относится к набору однородных личностей, предметов, событий или идей, выступающих как целое.

Экземпляр сущности относится к конкретной вещи в наборе. Например, типом сущности может быть ГОРОД, а экземпляром – Москва, Киев и т.д.

В реляционных БД сущности реализуются с помощью таблиц.

Инфологическое моделирование. ER-диаграммы

Атрибут – поименованная характеристика сущности. Его наименование должно быть уникальным для конкретного типа сущности, но может быть одинаковым для различного типа сущностей (например, ЦВЕТ может быть определен для многих сущностей: СОБАКА, АВТОМОБИЛЬ, ДЫМ и т.д.).

Атрибуты используются для определения того, какая информация должна быть собрана о сущности. Примерами атрибутов для сущности АВТОМОБИЛЬ являются ТИП, МАРКА, НОМЕРНОЙ ЗНАК, ЦВЕТ и т.д.

Здесь также существует различие между типом и экземпляром. Тип атрибута ЦВЕТ имеет много экземпляров или значений: Красный, Синий, Банановый, Белая ночь и т.д., однако каждому экземпляру сущности присваивается только одно значение атрибута.

В реляционных базах данных атрибуты реализуются с помощью столбцов.

Инфологическое моделирование. ER-диаграммы

Абсолютное различие между типами сущностей и атрибутами отсутствует. Атрибут является таковым только в связи с типом сущности. В другом контексте атрибут может выступать как самостоятельная сущность. Например, для автомобильного завода цвет – это только атрибут продукта производства, а для лакокрасочной фабрики цвет – тип сущности.

□ **Ключ – минимальный набор атрибутов, по значениям которых можно однозначно найти требуемый экземпляр сущности.**

Минимальность означает, что исключение из набора любого атрибута не позволяет идентифицировать сущность по оставшимся.

Для сущности Расписание ключом является атрибут Номер_рейса или набор: Пункт_отправления, Время_вылета и Пункт_назначения (при условии, что из пункта в пункт вылетает в каждый момент времени один самолет).

Инфологическое моделирование. ER-диаграммы

Выделяют **уникальные** ключи (потенциальные ключи) и **неуникальные**. Значение уникального ключа не может встретиться у двух экземпляров сущности. Оно указывает на один и только один экземпляр (**НомерСтудбилета**, **НомерАудитории**). Значение неуникального ключа указывает на множество экземпляров (**ФамилияПреподавателя** = Иванов указывает на всех Ивановых, преподающих в ВУЗе).

Ключом может быть не любой атрибут сущности. Например, **ДатаНайма** или **Должность** преподавателя вряд ли могут использоваться для идентификации преподавателей.

Сущность может иметь несколько уникальных и неуникальных ключей. Атрибут нельзя назначить уникальным ключом сущности. Он либо является таковым, либо не является.

В реляционных базах данных ключи реализуются с

помощью первичных ключей

Инфологическое моделирование. ER-диаграммы

Связь – ассоциирование двух или более сущностей. Если бы назначением базы данных было только хранение отдельных, не связанных между собой данных, то ее структура могла бы быть очень простой. Однако одно из основных требований к организации базы данных – это обеспечение возможности отыскания одних сущностей по значениям других, для чего необходимо установить между ними определенные связи. А так как в реальных базах данных нередко содержатся сотни или даже тысячи сущностей, то теоретически между ними может быть установлено более миллиона связей. Наличие такого множества связей и определяет сложность инфологических моделей.

Описание сущностей и их связей – это и есть (с точки зрения проектировщика БД) основная часть инфологической модели.

Идея Чена, благодаря которой его имя стало широко известным в кругах проектировщиков баз данных, состоит в том, что **сущности и связи следует представлять графически**. Тогда модель требований пользователя будет компактной и наглядной.

Классификация сущностей и связей.

К.Дейт [3] определяет три основные класса сущностей: **стержневые, ассоциативные и характеристические**, а также подкласс ассоциативных сущностей – **обозначения**.

Стержневая сущность (стержень) – независимая сущность.

Например – это "Студент", "Группа", "Мужчины", "Врач"

На ЯИМ стержневые сущности представляются предложениями вида:

СУЩНОСТЬ (атрибут_1, атрибут_2 , ..., атрибут_n)

где атрибуты, входящие в ключ, должны быть выделены с помощью подчеркивания или жирным шрифтом.

Сущности ВРАЧ и ПАЦИЕНТ можно описать на ЯИМ следующим образом:

Врач (**Номер_врача**, Фамилия, Имя, Отчество, Должность, ...)

Пациент (**Регистрационный_номер**, Фамилия,

Имя, Отчество, Адрес, Дата рождения, Пол)

Стержень

Классификация сущностей и связей.

Ассоциативная сущность (ассоциация) – это связь вида "многие-ко-многим" ("ко-многим" и т.д.) между двумя или более сущностями или экземплярами сущности.

Ассоциации рассматриваются как полноправные сущности: они могут участвовать в других ассоциациях и обозначениях точно так же, как стержневые сущности; могут обладать свойствами, т.е. иметь не только набор ключевых атрибутов, необходимых для указания связей, но и любое число других атрибутов, характеризующих связь.

Ассоциация

Ассоциация

АТРИБУТ

ПЕРВИЧНЫЙ КЛЮЧ

Классификация сущностей и связей.

Ассоциации на **ЯИМ** представляются предложениями вида:

СУЩНОСТЬ (атрибут 1, атрибут 2, ..., атрибут n) **АССОЦИАЦИЯ**

[СУЩНОСТЬ S1, СУЩНОСТЬ S2, ...] (атрибут 1, атрибут 2, ..., атрибут n)

где S – степень связи, а **атрибуты, входящие в ключ, должны**

быть выделены с помощью подчеркивания или жирным шрифтом.

Ассоциации рассматриваются как полноправные сущности:

1) они могут участвовать в других ассоциациях и обозначениях точно так же, как стержневые сущности;

2) могут обладать свойствами, т.е. иметь не только набор ключевых атрибутов, необходимых для указания связей, но и любое число других атрибутов, характеризующих связь. Например, ассоциации "Брак" содержат ключевые атрибуты "Табельный номер мужа", "Табельный номер жены", а также уточняющие атрибуты "Номер свидетельства", "Дата регистрации", "Место_регистрации", "Номер записи в книгу ЗАГС"

Классификация сущностей и связей.

Пример связей между сущностями может быть описан на ЯИМ следующим образом:

Лечащий_врач [Врач 1, Пациент M] (Номер_врача, Рег_номер)

Консультант [Врач M, Пациент N] (Номер_врача, Рег_номер).



Соответствующая схема "Таблица - Связь"

Классификация сущностей и связей.

Характеристическая сущность (характеристика) – это связь вида "многие-к-одной" или "одна-к-одной" между двумя сущностями (частный случай ассоциации).

Единственная цель **характеристики** в рамках рассматриваемой предметной области состоит в описании или уточнении некоторой другой сущности. Необходимость в них возникает в связи с тем, что сущности реального мира имеют иногда многозначные свойства. Книга может иметь несколько характеристик переиздания (исправленное, дополненное, переработанное, ...) и т.д.

Для описания характеристики используется новое предложение ЯИМ, имеющее в общем случае вид:

ХАРАКТЕРИСТИКА (атрибут 1, атрибут 2, ...
{СПИСОК ХАРАКТЕРИЗУЕМЫХ СУЩНОСТЕЙ}

ХАРАКТЕРИСТИКА

Классификация сущностей и связей.

Расширим также язык ER-диаграмм, введя для изображения характеристики трапецию.

Обозначающая сущность или **обозначение** – это связь вида "многие-к-одной" или "одна-к-одной" между двумя сущностями и отличается от характеристики тем, что не зависит от обозначаемой сущности.

Отделы (Номер отдела, Название отдела, ...) Служащие (Табельный номер, Фамилия, ... , Номер отдела, Дата зачисления) [Отделы]

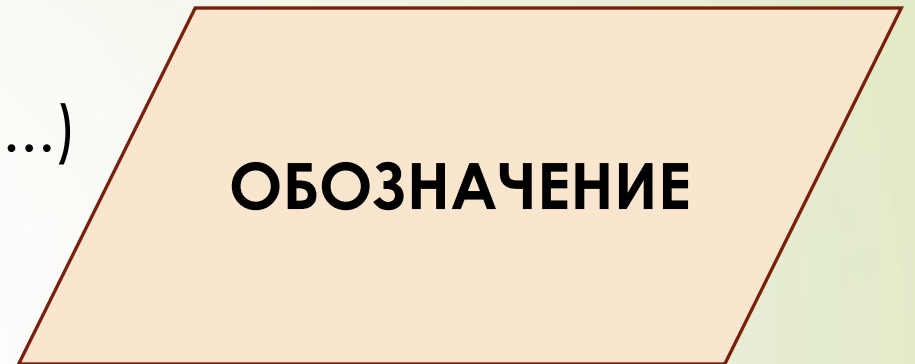
В данном примере служащие имеют независимое существование (если удаляется отдел, то из этого не следует, что также должны быть удалены служащие такого отдела). Поэтому они не могут быть характеристиками отделов и названы обозначениями.

Обозначения используют для хранения повторяющихся значений больших текстовых атрибутов: "кодификаторы" изучаемых студентами

Классификация сущностей и связей.

Описание обозначения внешне отличается от описания характеристики только тем, что обозначаемые сущности заключаются не в фигурные скобки, а в квадратные:

ОБОЗНАЧЕНИЕ (атрибут 1, атрибут 2, ...)
[СПИСОК ОБОЗНАЧАЕМЫХ СУЩНОСТЕЙ].



Обозначения и характеристики не являются полностью независимыми сущностями, поскольку они предполагают наличие некоторой другой сущности, которая будет "обозначаться" или "характеризоваться".

Однако они все же представляют собой частные случаи сущности и могут, конечно, иметь свойства, могут участвовать в ассоциациях, обозначениях и иметь свои собственные (более низкого уровня) характеристики.

Классификация сущностей и связей.

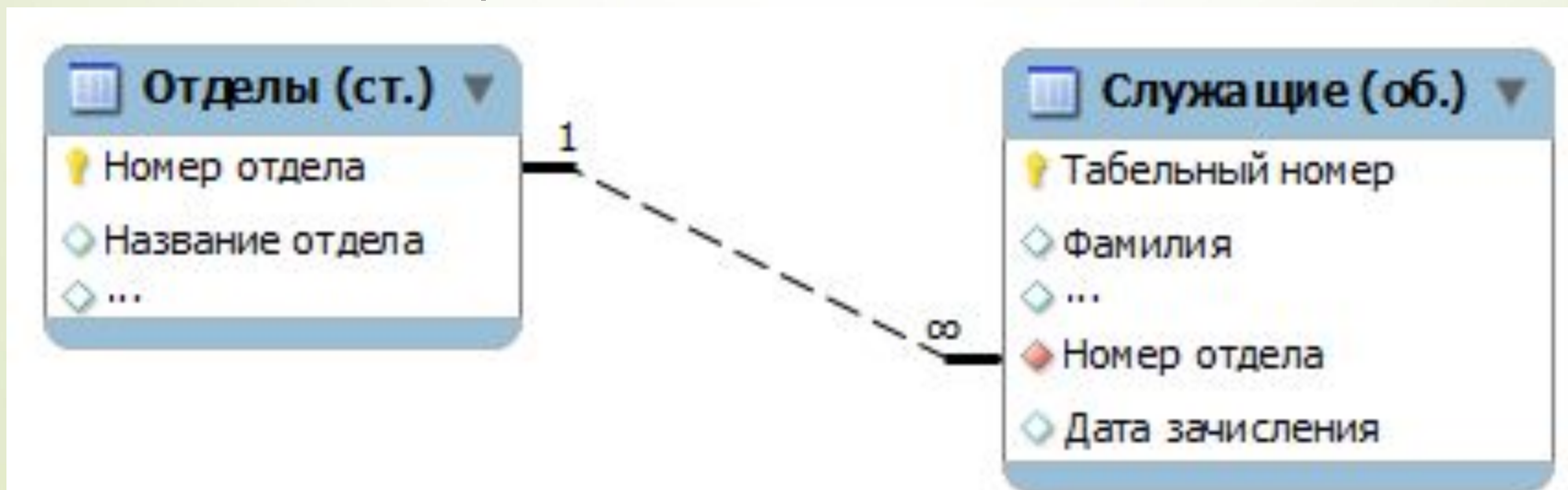
Рассмотрим пример, связанный с зачислением сотрудников в различные отделы организации. Здесь возможны 2 варианта:

1. Если сотрудник может одновременно зачисляться в несколько отделов или не зачисляться ни в один отдел, можно создать описание на ЯИМ с ассоциацией Зачисление:

Отделы (**Номер отдела**, Название отдела, ...)

Служащие (**Табельный номер**, Фамилия, ...)

Зачисление [Отделы M, Служащие N] **Номер отдела, Табельный номер**, Дата зачисления).



Классификация сущностей и связей.

2. Если каждый из сотрудников должен быть обязательно зачислен в один из отделов, можно создать описание с обозначением Служащие:

Отделы (**Номер отдела**, Название отдела, ...)

Служащие (**Табельный номер**, Фамилия, ... , Номер отдела, Дата зачисления) [Отделы]

В данном примере служащие имеют независимое существование (если удаляется отдел, то из этого не следует, что также должны быть удалены служащие такого отдела). Поэтому они не могут быть характеристиками отделов и названы обозначениями.

Обозначения используют для хранения повторяющихся значений больших текстовых атрибутов: "кодификаторы" изучаемых студентами дисциплин, наименований организаций и их отделов, перечней товаров и т.п.

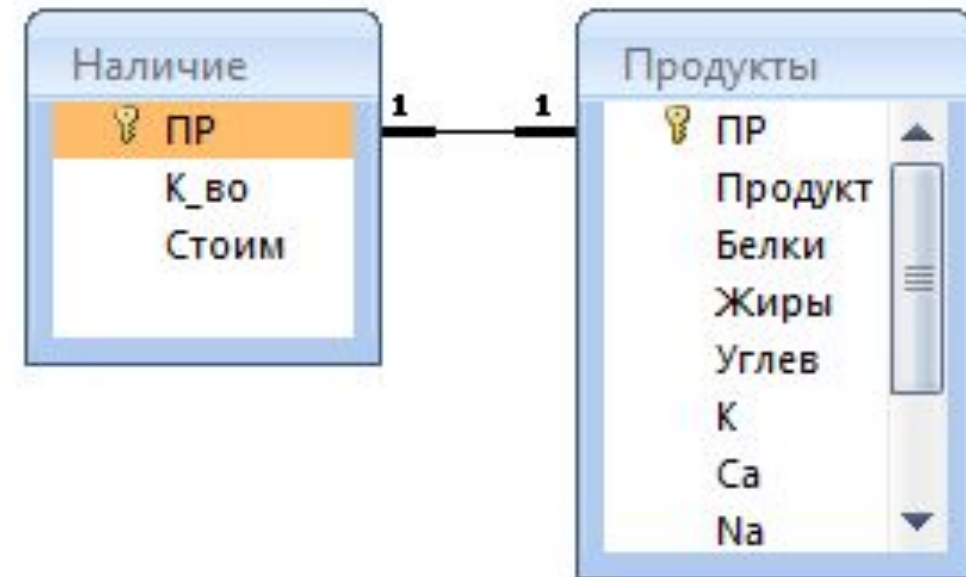
Классификация сущностей и связей.

Между двумя сущностям, например, А и В возможны четыре вида связей.

Первый тип – связь **ОДИН-К-ОДНОМУ (1:1)**: в каждый момент времени каждому представителю (экземпляру) сущности А соответствует 1 или 0 представителей сущности В:

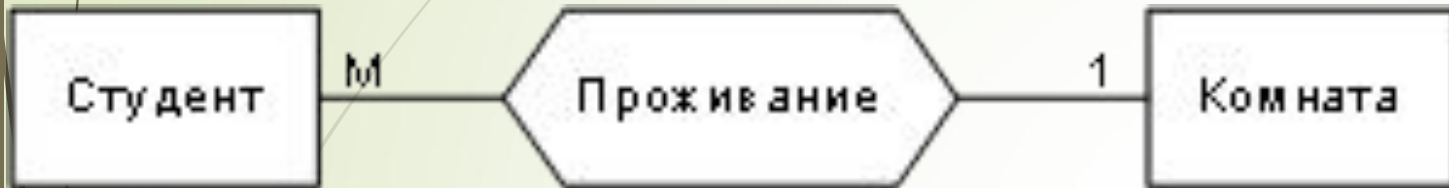


Примечание. В реляционных базах данных связь ОДИН-К-ОДНОМУ, как правило, реализуется с помощью двух таблиц.

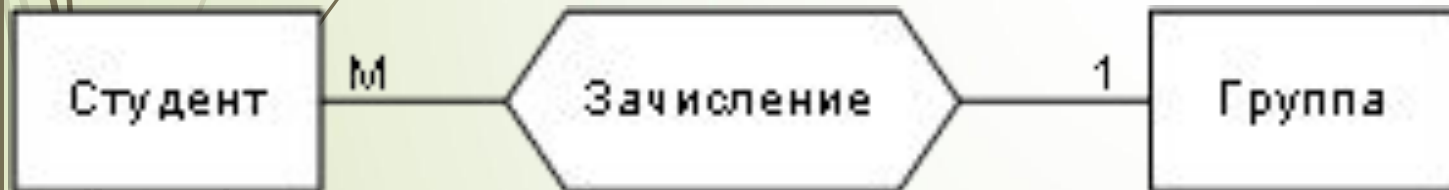


Классификация сущностей и связей.

Второй тип – СВЯЗЬ **ОДИН-КО-МНОГИМ (1:M)**: одному представителю сущности А соответствуют 0, 1 или несколько представителей сущности В.

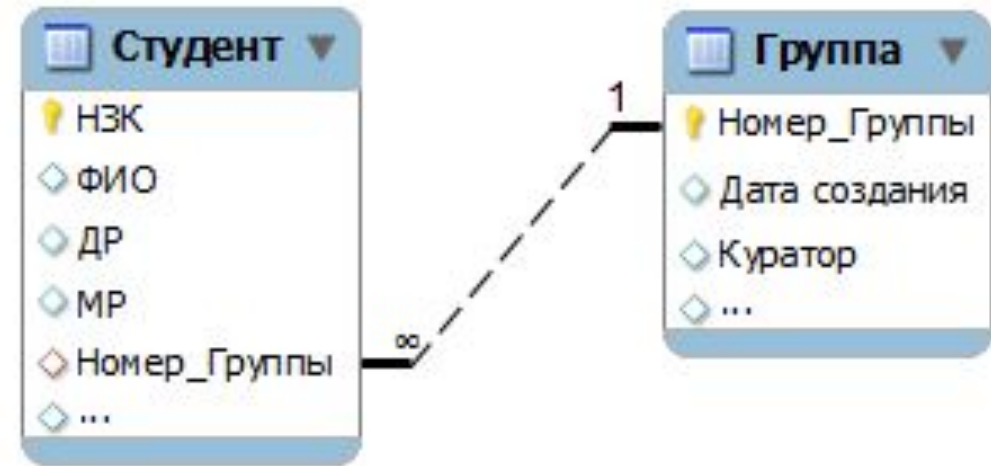


Квартира может пустовать, в ней может жить один или несколько жильцов.



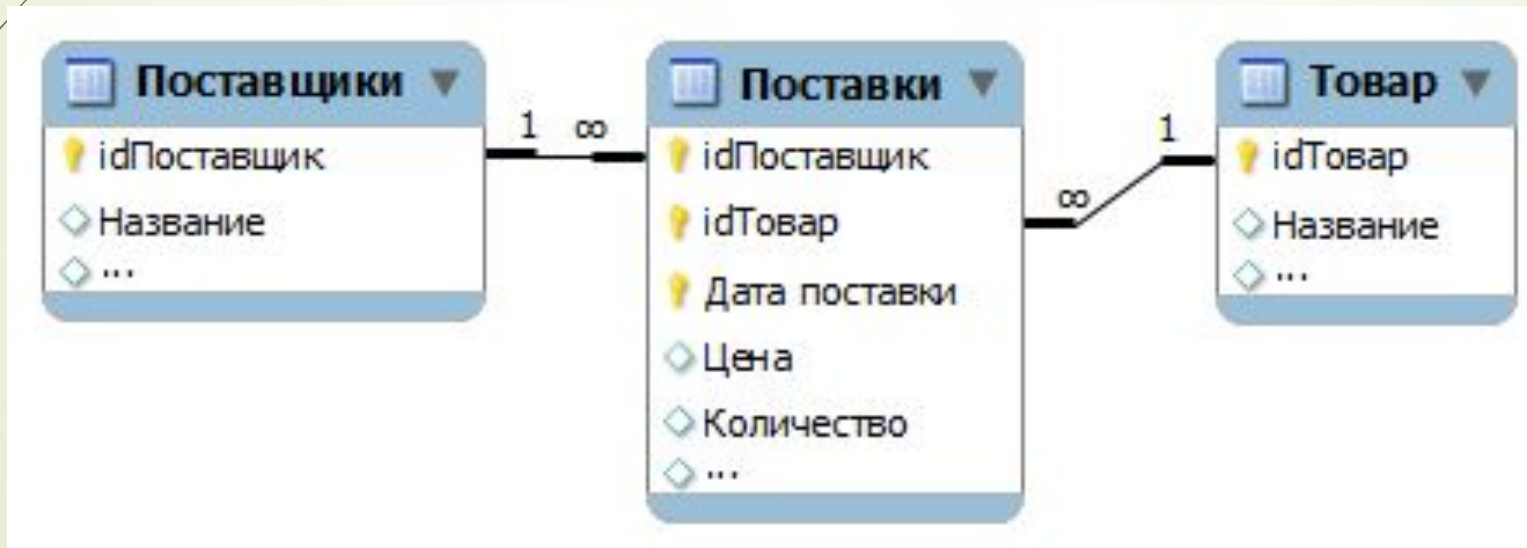
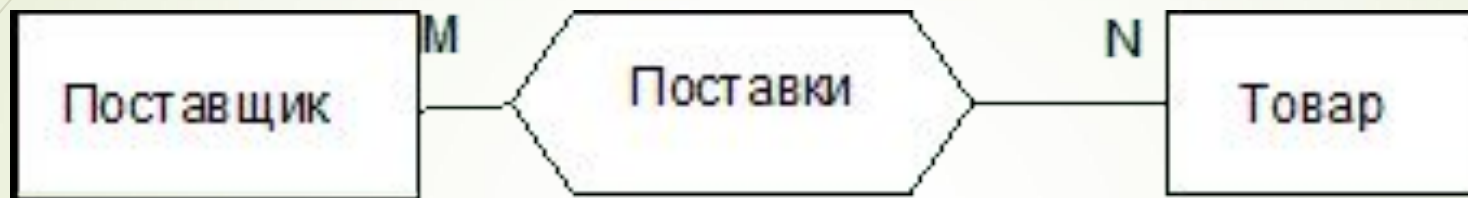
По правилам Минвуза *каждый* студент может быть зачислен *ровно* в одну учебную группу. Поэтому кардинальность этой связи со стороны сущности СТУДЕНТ равна единице.

В реляционных базах данных связь **ОДИН-КО-МНОГИМ**, как правило, реализуется с помощью двух таблиц.



Классификация сущностей и связей.

Так как между двумя сущностями возможны связи в обоих направлениях, то существует еще два типа связи: Связь МНОГИЕ-К-ОДНОМУ (M:1) и Связь МНОГИЕ-КО-МНОГИМ (M:N).

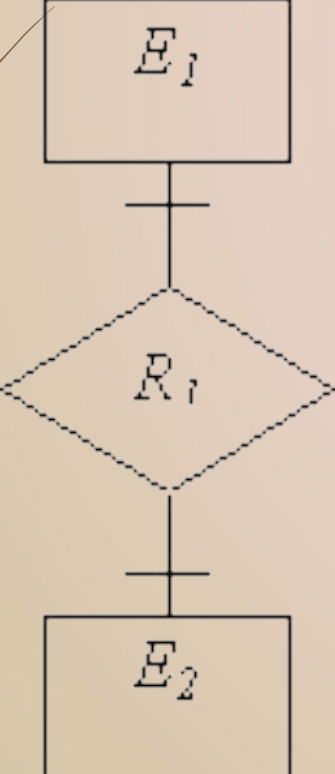
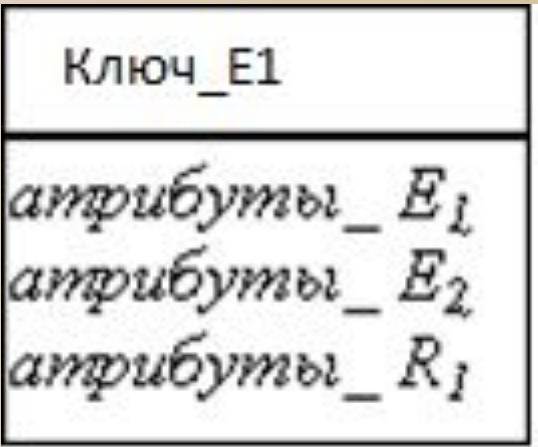


В реляционных базах данных связь МНОГИЕ-КО-МНОГИМ, как правило, реализуется с помощью трех таблиц.

Классификация сущностей и связей.

Бинарные связи

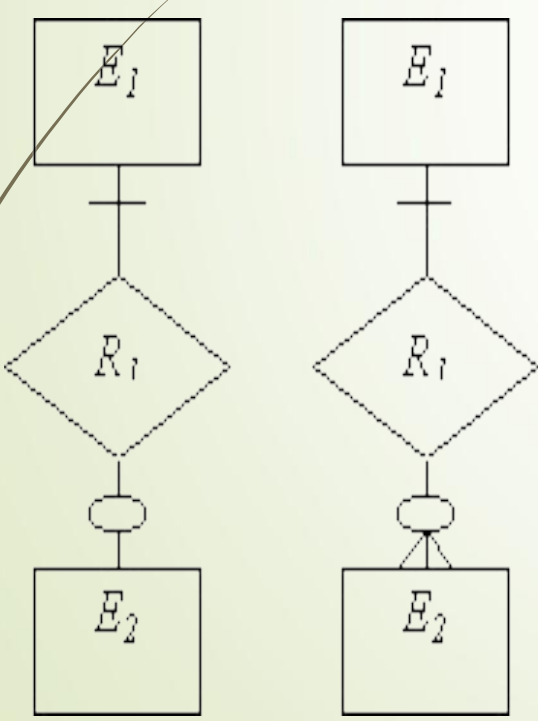
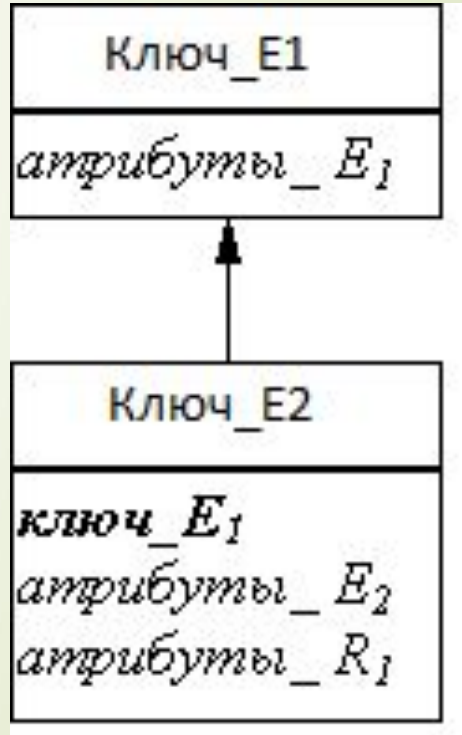
Правила порождения реляционных отношений из модели «сущность-связь.»

Тип связи	Пример связи	Правило построения отношений	Рел тношения
(1,1): (1,1)		Требуется только одно отношение. Первичным ключом данного отношения может быть ключ любой из сущностей.	

Классификация сущностей и связей.

Правила порождения реляционных отношений из модели «сущность-связь.»

Бинарные связи

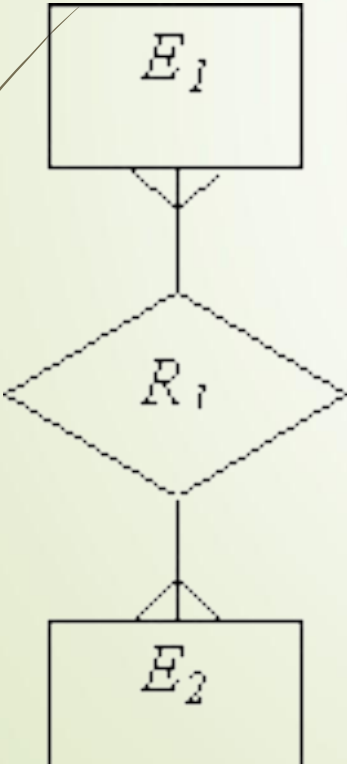
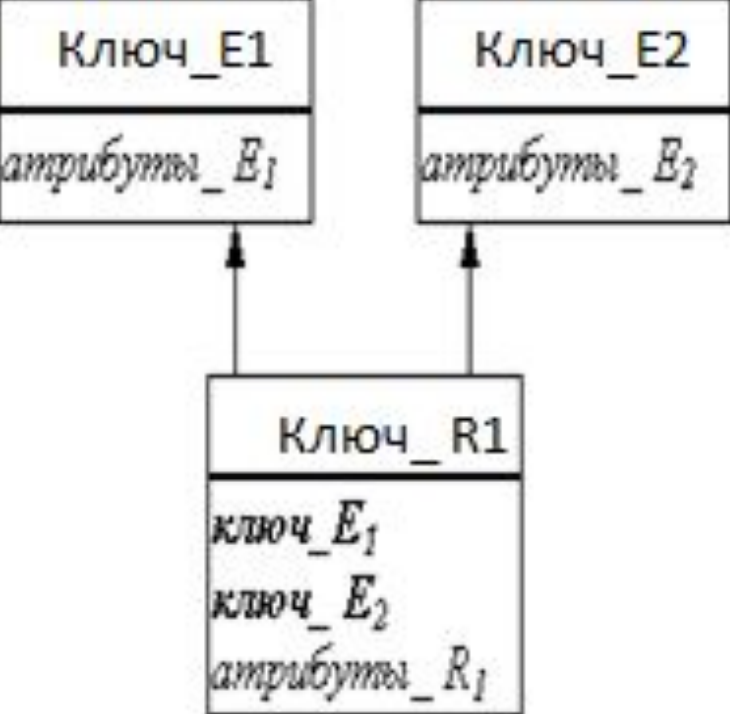
Тип связи	Пример связи	Правило построения отношений	Рел тношения
(1,1): (0,1) (1,1): (0,n)		Для каждой сущности создается свое отношение, при этом ключи сущностей служат ключами соответствующих отношений. Кроме того, ключ сущности с обязательным классом принадлежности добавляется в качестве внешнего ключа в отношение, созданное для сущности с необязательным классом принадлежности.	

Классификация сущностей и связей.

Правила порождения реляционных отношений из модели

«СУЩНОСТЬ-СВЯЗЬ.»

Бинарные связи

Тип связи	Пример связи	Правило построения отношений	Рел тношения
$n : m$	 <p>The diagram shows two rectangular boxes representing entities, labeled E_1 and E_2. A diamond-shaped box representing a relationship, labeled R_1, is positioned between them. Lines connect each entity to the relationship, indicating a binary association.</p>	<p>В этом случае всегда используются три отношения: по одному для каждой сущности, причем ключ каждой сущности служит первичным ключом соответствующего отношения, и одно отношение для связи. Последнее отношение должно иметь среди своих атрибутов внешние ключи, по одному от каждой сущности.</p>	 <p>The diagram shows three tables. The top two tables are for entities: 'Ключ_E1' (primary key) and 'атрибуты_E1', and 'Ключ_E2' (primary key) and 'атрибуты_E2'. The bottom table is for the relationship: 'Ключ_R1' (primary key), 'ключ_E1' (foreign key), 'ключ_E2' (foreign key), and 'атрибуты_R1'. Arrows point from the relationship table to the entity tables, indicating foreign key relationships.</p>

Классификация сущностей и связей.

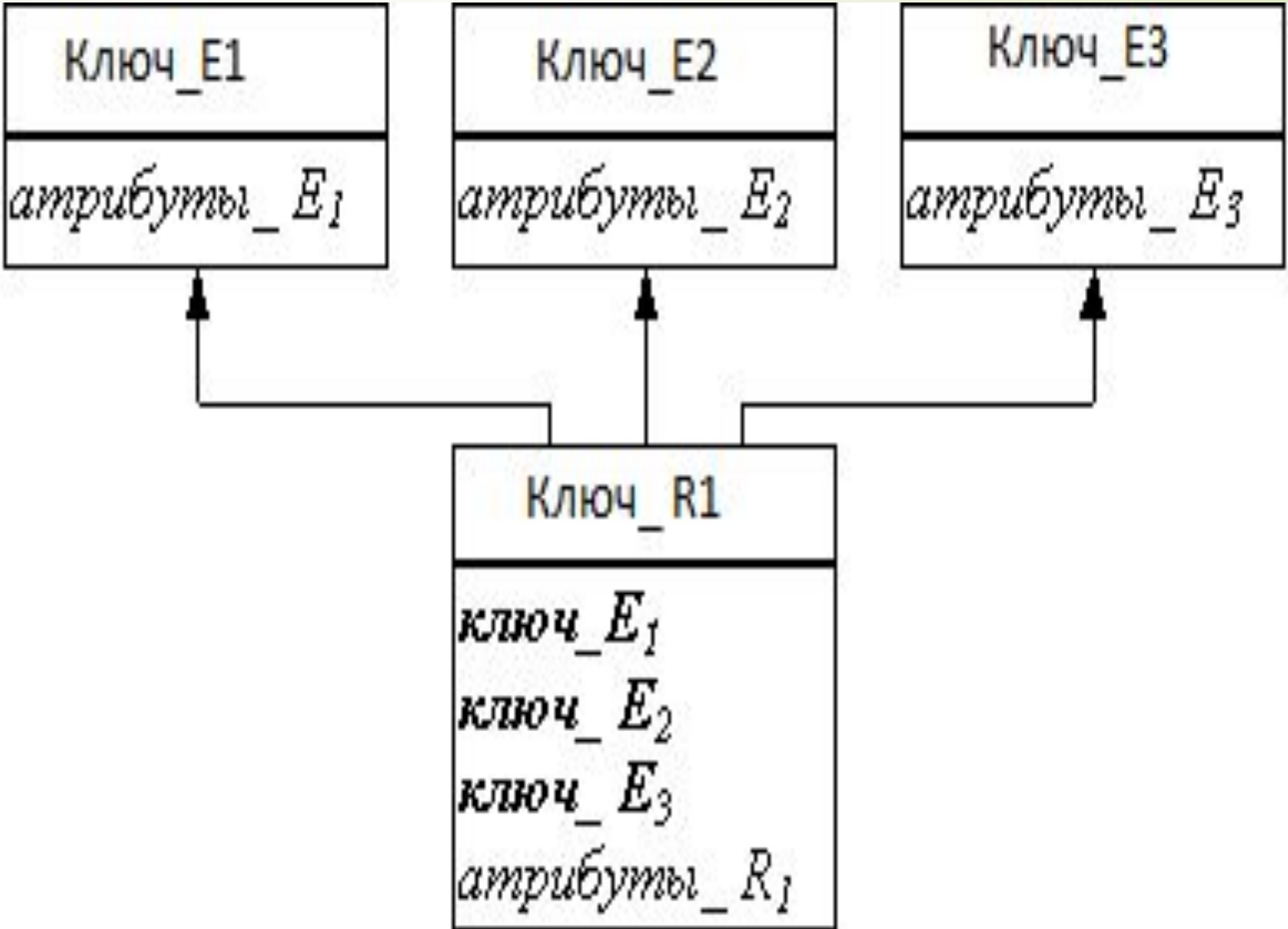
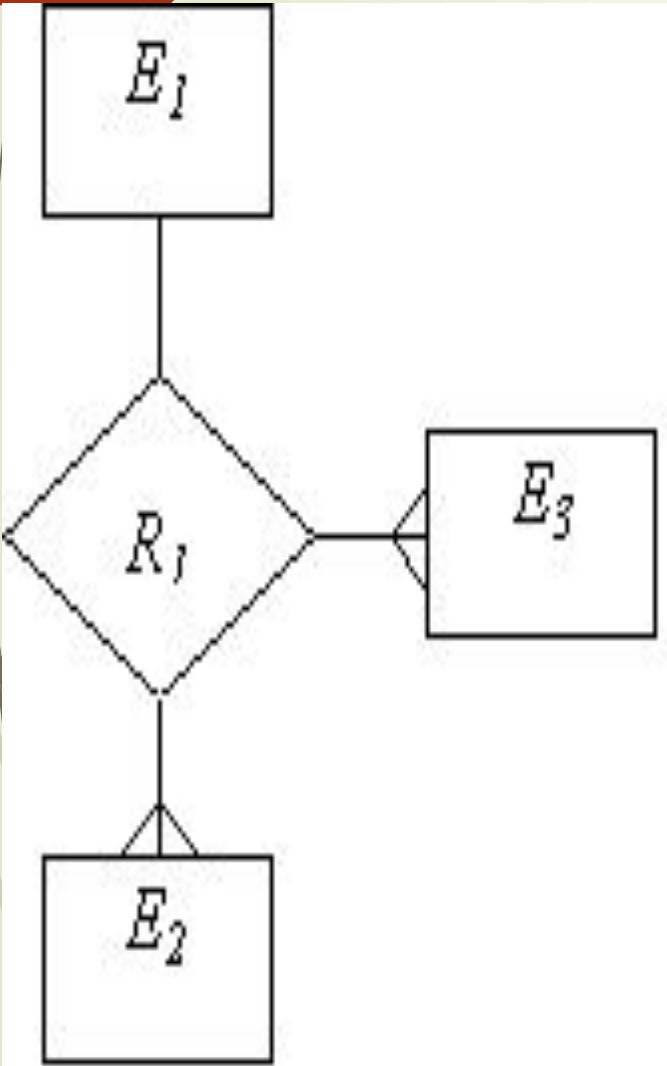
Правила порождения реляционных отношений из модели «сущность-связь.»

Характер связей между сущностями не ограничивается перечисленными. Существуют и более сложные связи: **N - арные связи**

Общее правило: для представления n-сторонней связи всегда требуется $n+1$ отношение.

Например, в случае трехсторонней связи необходимо использовать четыре отношения, по одному для каждой сущности (причем ключ сущности служит первичным ключом соответствующего отношения), и одно для связи. Отношение, порождаемой для связи, будет иметь среди своих атрибутов ключи от каждой сущности.

Классификация сущностей и связей.



Классификация сущностей и связей.

Иерархические связи

К сожалению, надо признать, что реляционная модель мало подходит для отображения отношений наследования между сущностями (иерархических связей). Напомним, что в таких связях дочерние сущности наследуют все атрибуты родительской, и каждая из них обладает своим уникальным набором дополнительных атрибутов.

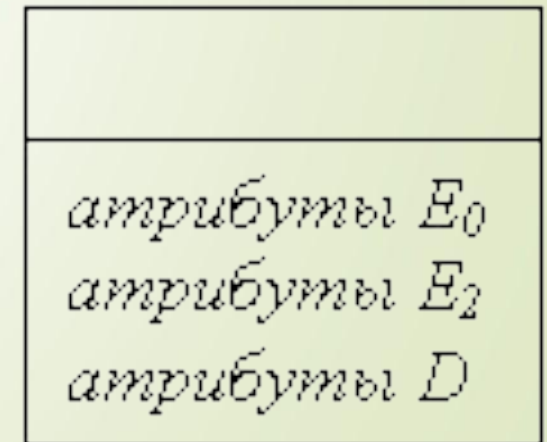
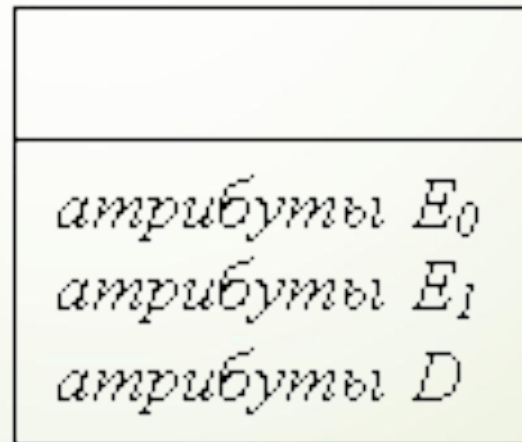
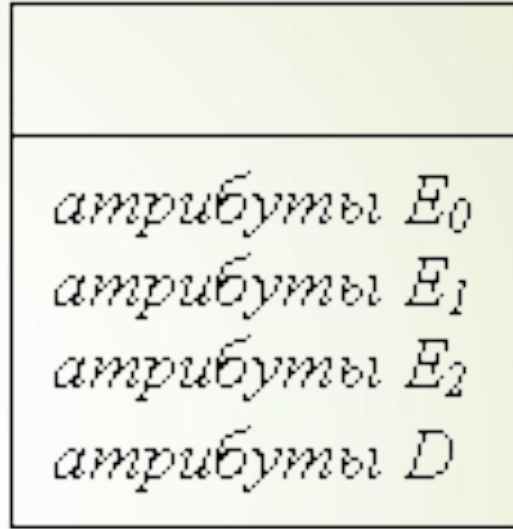
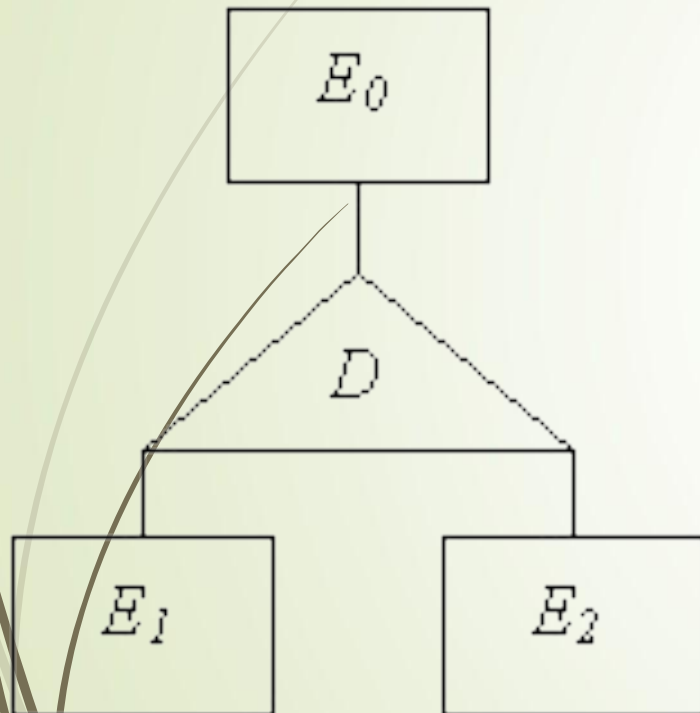
В этом случае возможны два варианта построения реляционных отношений.

Согласно первому для иерархической структуры создается одно отношение, которое содержит атрибуты связи и всех сущностей.

По второму способу генерируется по одному отношению для каждой дочерней сущности. Каждое из этих отношений включает атрибуты родительской сущности и связи кроме атрибутов - дискриминантов

Классификация сущностей и связей.

Иерархические связи



Классификация сущностей и связей.

Иерархические связи

