

# Раздел 3. Проектирование баз данных

## Тема 3.2.

### Модель «сущность – связь».

#### Вопросы лекции:

1. **Основные понятия метода.**
2. Виды нотаций ER-диаграмм.
3. Нотация IDEF1X.

Конкретные явления реального мира, представляющие **интерес для проводимого исследования**, называют **предметной областью**.

**Адекватность** предметной области выражается в том, что база данных должна позволять **выполнять все необходимые операции, которые объективно нужны в реальной жизни** в контексте той работы, для которой предназначена база данных.

Как и было сказано в определении **инфологического уровня моделирования**, **основной его целью** является создание **модели, отражающей сущности** предметной области, **их атрибуты и связи** (возможно, пока не все) между сущностями.

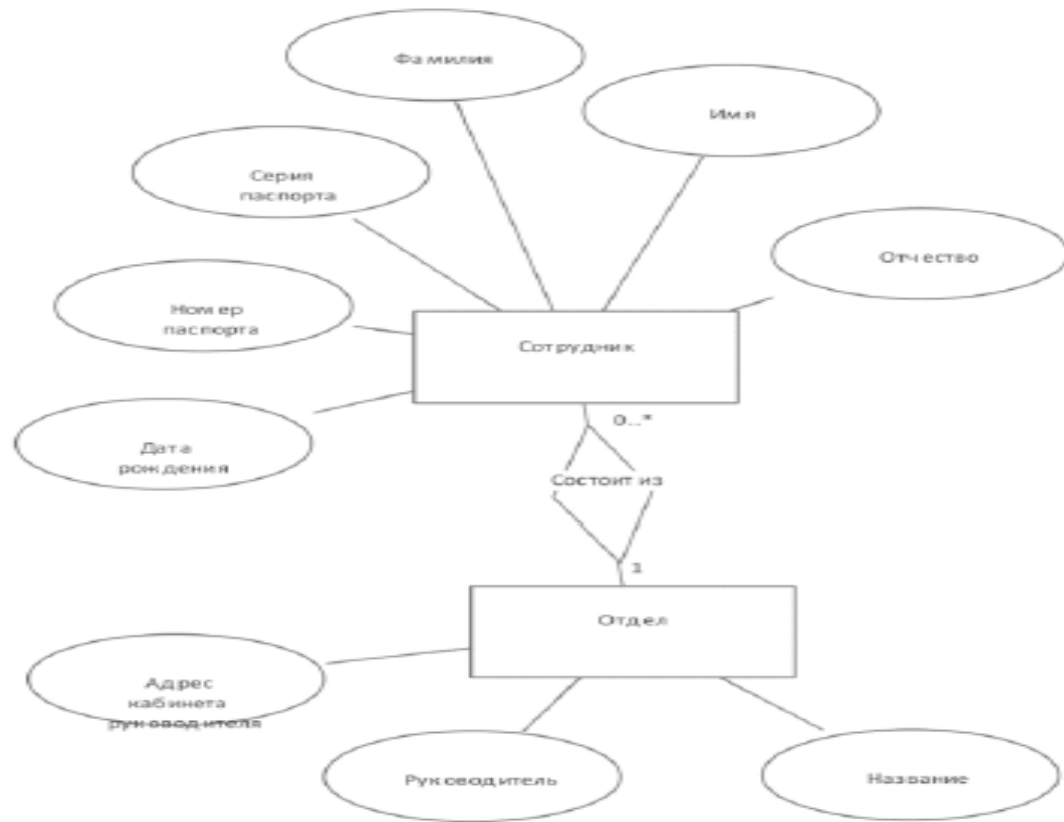
**Текстовое** представление модели **предметной области** удобно не только своей привычностью, но и своей практичностью.

Однако, у текстового представления есть **ряд серьёзных недостатков**:

- оно **неудобно для представления взаимосвязи** сущностей (фактически, это можно делать только комментариями);
- оно **не компактно** (может занимать несколько десятков страниц там, где другие формы представления заняли бы пару экранов);
- оно **допускает разночтение** технических аспектов реализации базы данных (а чаще всего эти аспекты и вовсе оказываются упущены);
- попытки устранить обозначенные выше недостатки делают **текстовое представление перегруженным информацией** и постепенно сводят на нет его преимущества.

Альтернативами текстовому представлению являются **графические формы** — в виде семантических моделей, графовых моделей и UML-диаграмм.

Статус официального стандарта получили нотации **IDEF** и язык **UML**



ER-диаграмма



UML-диаграмма

# Раздел 3. Проектирование баз данных

## Тема 3.2.

### Модель «сущность – связь».

#### Вопросы лекции:

1. Основные понятия метода.
2. **Виды нотаций ER-диаграмм.**
3. Нотация IDEF1X.

Метод использования **графической модели** сущность-связь называют также **методом «ER-диаграмм» (ERD)**:

- во-первых, **ER** - аббревиатура от слов **Entity** (сущность) и **Relationship** (связь),
- во-вторых, **метод основан на использовании диаграмм**, называемых соответственно **диаграммами ER-экземпляров** и **диаграммами ER-типа**.

Существуют различные **варианты ER-диаграмм**, отличающиеся способами графического представления сущностей и связей:

- нотация **Питера Чена** (предложена первая в 1976 году);
- нотация **Мартина**;
- нотация **IDEF1X** ;
- нотация **Баркера** и др.

В частности, программное **CASE-средство** для автоматизации проектирования БД на основе ER-диаграмм **AllFusion ERwin Data Modeler** использует **IDEF1X**-нотацию, CASE-средства других разработчиком могут также использовать и другие варианты ER-диаграммы.

# Нотация ER-диаграмм Питера Чена

Основными понятиями метода **сущность-связь** являются следующие:

- сущность, • атрибут сущности, • ключ сущности,
- связь между сущностями, • степень связи,
- класс принадлежности экземпляров сущности,
- диаграммы **ER-экземпляров**, • диаграммы **ER-типа**.

Различаются **сильные** и **слабые** типы сущностей.

**Слабый** тип сущности определяется как тип, существование которого **зависит от какого-то другого типа сущности**. Слабые сущности могут быть дочерними, подчиненными и зависимыми.

**Сильный** тип сущности – как тип, существование которого **не зависит от всех других типов сущностей**. Сильные сущности могут быть родительскими, владельцами или доминантными.

# Нотация ER-диаграмм Питера Чена

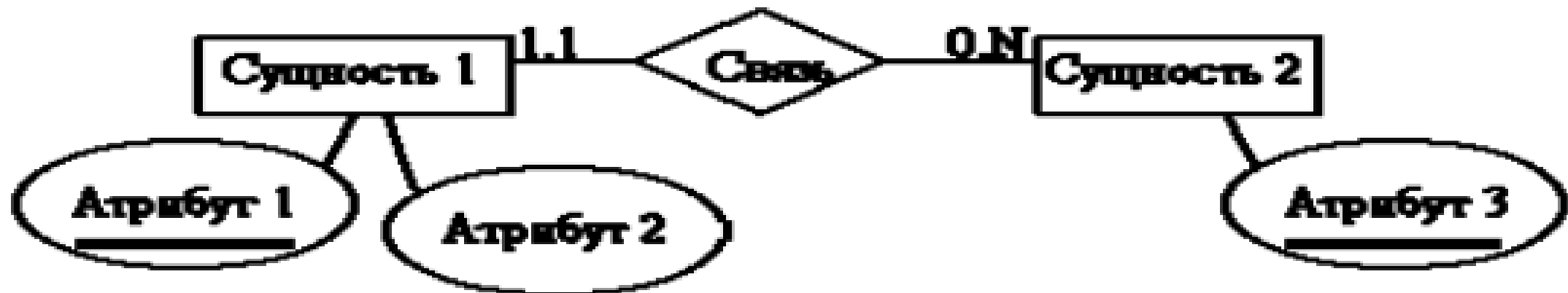
**Сущность** представляет собой объект, информация о котором хранится в БД. Экземпляры сущности отличаются друг от друга и **однозначно идентифицируются**.

Названиями сущностей **являются, как правило, существительные**, например: ПРЕПОДАВАТЕЛЬ, ДИСЦИПЛИНА, КАФЕДРА, ГРУППА.

**Атрибут** представляет собой **свойство** сущности. Это понятие **аналогично понятию атрибута в отношении**.

Так, атрибутами сущности ПРЕПОДАВАТЕЛЬ может быть его Фамилия, Должность, Стаж (преподавательский) и т. д.

## ПРИМЕР:





## Нотация ER-диаграмм Питера Чена

*Ключ сущности* - атрибут или набор атрибутов, используемый для идентификации экземпляра сущности.

Как видно из определения, понятие ключа сущности аналогично понятию ключа отношения.

*Связь двух или более сущностей* - предполагает зависимость между атрибутами этих сущностей. Название связи обычно представляется глаголом.

Примерами связей между сущностями являются следующие:

ПРЕПОДАВАТЕЛЬ *ВЕДЕТ* ДИСЦИПЛИНУ - (Иванов *ВЕДЕТ* «Базы данных»),

ПРЕПОДАВАТЕЛЬ *ПРЕПОДАЕТ*-В ГРУППЕ -(Иванов *ПРЕПОДАЕТ*-В 256 группе).

ПРЕПОДАВАТЕЛЬ *РАБОТАЕТ*-НА КАФЕДРЕ - (Иванов *РАБОТАЕТ*-НА кафедре ИСУ).

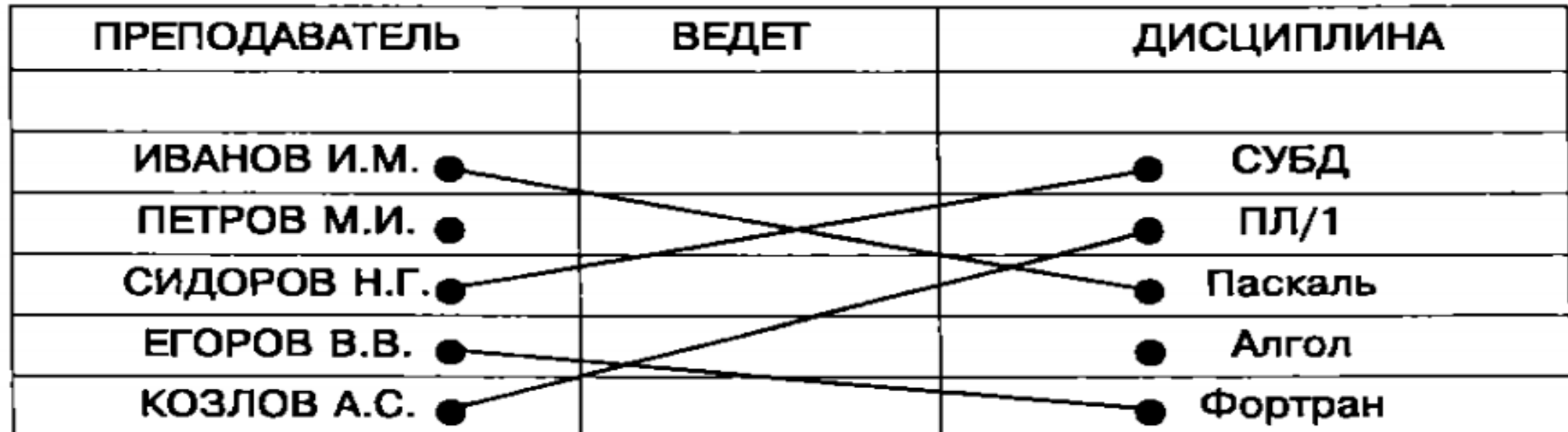
# Нотация ER-диаграмм Питера Чена

С целью повышения наглядности и удобства проектирования для представления сущностей, экземпляров сущностей и связей между ними используются следующие **графические средства**:

- диаграммы *ER-экземпляров*,
- диаграммы *ER-типа*, или *ER-диаграммы*.

Например **диаграмма ER-экземпляров** для сущностей ПРЕПОДАВАТЕЛЬ и ДИСЦИПЛИНА со связью *ВЕДЕТ* будет выглядеть так:

ПРЕПОДАВАТЕЛЬ	ВЕДЕТ	ДИСЦИПЛИНА
ИВАНОВ И.М. ●		● СУБД
ПЕТРОВ М.И. ●		● ПЛ/1
СИДОРОВ Н.Г. ●		● Паскаль
ЕГОРОВ В.В. ●		● Алгол
КОЗЛОВ А.С. ●		● Фортран



## Нотация ER-диаграмм Питера Чена

**Диаграмма ER-типа** в обобщенном виде показывает, какие связи существуют между сущностями того или иного типа .

Связи отличаются по степени (**кардинальности**) и классу принадлежности (**степени участия**) – будут рассмотрены далее.

Пример диаграммы **ER-типа**, соответствующей рассмотренной диаграмме ER-экземпляров.



## Нотация ER-диаграмм Питера Чена

На начальном этапе проектирования БД обязательно **выделяются атрибуты**, составляющие **ключи сущностей**.

На основе анализа диаграмм ER-типа формируются отношения проектируемой БД - связи.

Связям могут также присваиваться **ролевые имена** для однозначного определения назначения каждой связи.

При этом учитывается *степень связи сущностей* и *класс их принадлежности*, которые, в свою очередь, определяются на **основе анализа диаграмм ER-экземпляров** соответствующих сущностей.

**Степень связи (кардинальность)** является характеристикой связи между сущностями, которая может быть типа: **1:1**, **1:M**, **M:1**, **M:M**

# Нотация ER-диаграмм Питера Чена

Класс принадлежности или **степень участия** сущности может быть:

- **Обязательным** (полная степень участия)
- **Необязательным** (частичная степень участия).

Класс принадлежности сущности является **обязательным**, если все экземпляры этой сущности обязательно участвуют в рассматриваемой связи,

в противном случае класс принадлежности сущности является **необязательным**.

Варьируя классом принадлежности сущностей для каждого из названных типов связи, можно получить несколько вариантов диаграмм ER-экземпляров.

## Пример 1. Нотация ER-диаграмм Питера Чена

Степень связи между сущностями **1:1**, а класс принадлежности обеих сущностей **необязательный**.

Действительно:

- каждый преподаватель ведет не более одной дисциплины, а каждая дисциплина ведется не более чем одним преподавателем (степень связи **1:1**);
- некоторые преподаватели не ведут ни одной дисциплины и имеются дисциплины, которые не ведет ни один из преподавателей (класс принадлежности обеих сущностей **необязательный**).

ПРЕПОДАВАТЕЛЬ	ВЕДЕТ	ДИСЦИПЛИНА
ИВАНОВ И.М. ●		● СУБД
ПЕТРОВ М.И. ●		● ПЛ/1
СИДОРОВ Н.Г. ●		● Паскаль
ЕГОРОВ В.В. ●		● Алгол
КОЗЛОВ А.С. ●		● Фортран

## Пример 2. Нотация ER-диаграмм Питера Чена

Связи типа **1:1** и класс принадлежности **обязательный**.

В этом случае:

- каждый преподаватель **ведет одну** дисциплину и
- каждая дисциплина **ведется одним** преподавателем.

а) ER-экземпляров

ПРЕПОДАВАТЕЛЬ	ВЕДЕТ	ДИСЦИПЛИНА
ИВАНОВ И.М.		СУБД
ПЕТРОВ М.И.		ПЛ/1
СИДОРОВ Н.Г.		Паскаль
ЕГОРОВ В.В.		Алгол
КОЗЛОВ А.С.		Фортран

б) ER-типов

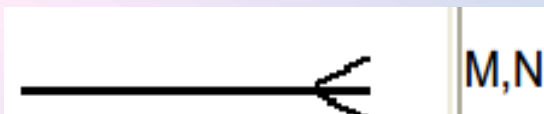
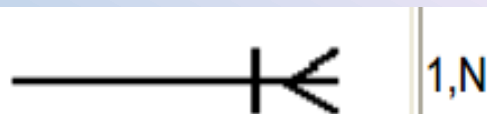
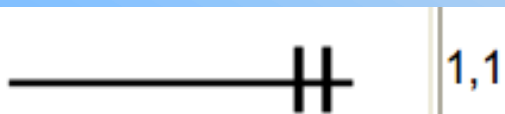


# Нотация ER-диаграмм Мартина

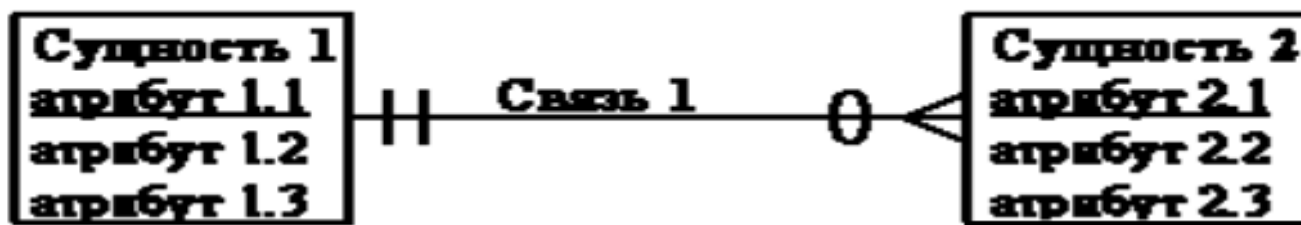
- Список атрибутов приводится внутри **прямоугольника**, обозначающего сущность.
- **Ключевые атрибуты подчеркиваются**.
- Связи обозначаются линиями, соединяющими сущности, вид линии в месте соединения с сущностью обозначает кардинальность связи.
- **Имя связи** указывается на линии, ее обозначающей.

**В отличие от диаграмм Чена** - линии связи не заканчиваются точкой, а указывают ее степень (**кардинальность**) символом «гусиная лапка».

**Обязательность связи** - указывается поперечными линиями, необязательность – окружностью вокруг линии связи.



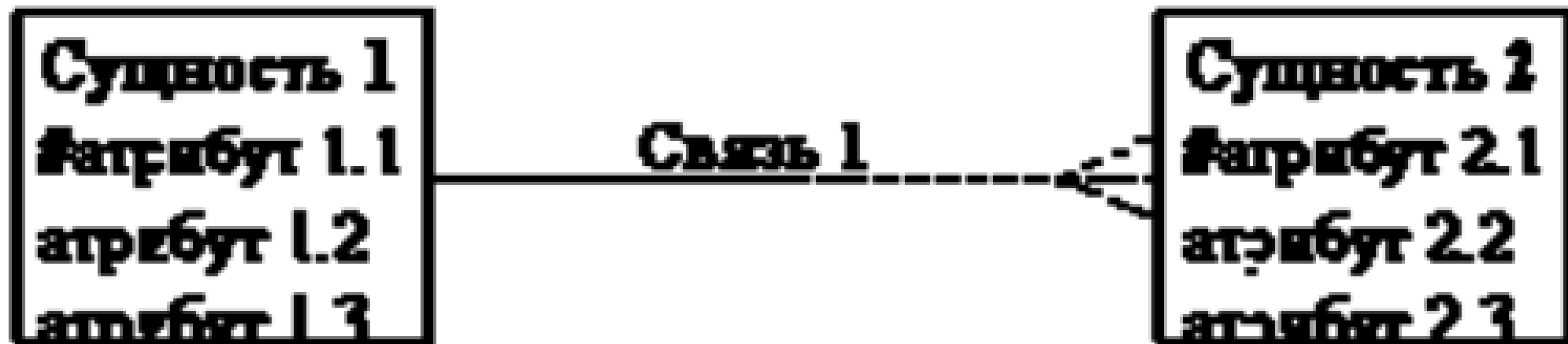
Пример:





# Нотация ER-диаграмм Баркера

- Сущности обозначаются **прямоугольником**, внутри которых приводится список атрибутов.
- **Ключевые** атрибуты обозначаются **символом #** (решетка).
- Связи обозначаются линиями с именами.
- **Также как в диаграммах Мартина** линии связи указывают ее степень символом «гусиная лапка».
- Связь, допускающая **отсутствие значения атрибута** (необязательный класс принадлежности) для элемента данных **обозначается пунктиром**.



# Раздел 3. Проектирование баз данных

## Тема 3.2.

### Модель «сущность – связь».

#### Вопросы лекции:

1. Основные понятия метода.
2. Виды нотаций ER-диаграмм.
3. **Нотация IDEF1X.**

## Нотация **IDEF1X**

**IDEF** – методология семейства **ICAM** (Integrated Computer-Aided Manufacturing) для решения задач моделирования сложных систем, позволяет отображать и анализировать модели деятельности широкого спектра систем в различных разрезах.

При этом широта и глубина обследования процессов в системе определяется самим разработчиком, что позволяет не перегружать создаваемую модель излишними данными.

Принципиальным требованием при разработке данного семейства методологий была возможность эффективного обмена информацией между всеми специалистами программы ICAM

(поэтому название: Integrated **DEF**inition).

# Нотация IDEF1X

Семейство стандартов IDEF достаточно большое:

Наибольшее распространение получили следующие нотации структурного моделирования:

- **IDEF0** — функциональные модели, основанные на методе структурного анализа и проектирования SADT (Structured Analysis and Design Technique) Дугласа Росса;
- **IDEF1** — методология моделирования информационных потоков внутри системы, позволяющая отображать и анализировать их структуру и взаимосвязи через **DFD (Data Flow Diagrams)** — диаграммы потоков данных;
- **IDEF1X** — модели данных, основанные на диаграммах «сущность-связь» (**ERD, Entity-Relationship Diagrams**);
- **IDEF2** — методология динамического моделирования развития систем;
- **IDEF3** — диаграммы потоков работ (Work Flow Diagrams);
- **IDEF4** — методология построения объектно-ориентированных систем.

Имеются также стандарты IDEF 5 – IDEF 14, но они имеют частное применение и еще дорабатываются.

# Нотация IDEF1X

## Основные элементы ERD :

- *Сущность (таблица, в РБД – отношение)* – реальный либо воображаемый объект, имеющий существенное значение для рассматриваемой ПрО, информация о котором подлежит хранению. Если выразаться точнее, то это не объект, а набор объектов (класс) с одинаковыми свойствами.
- *Экземпляр сущности (запись, строка, в РБД – кортеж)* – уникально идентифицируемый объект.
- *Связь* – некоторая ассоциация между двумя сущностями, значимая для рассматриваемой предметной области..
- *Атрибут (столбец, поле)* – свойство сущности или связи.

**Сущность** называется **зависимой** (**слабой, дочерней, подчиненной**), если ее существование зависит от других сущностей.

**Терминология** «родительская – дочерняя» или «владелец – подчиненный» также **может использоваться в отношении двух зависимых сущностей**, если экземпляры одной из них (дочерней, подчиненной) могут быть однозначно определены с использованием экземпляров другой (родительской, владельца), несмотря на то, что вторая сущность в свою очередь может зависеть от третьей сущности и т.д.

# Нотация IDEF1X

В графической нотации IDEF1X для обозначения сущности используются такие же обозначения, как и у других нотаций, но дальше **имеются отличия**.



– независимая сущность



– зависимая сущность

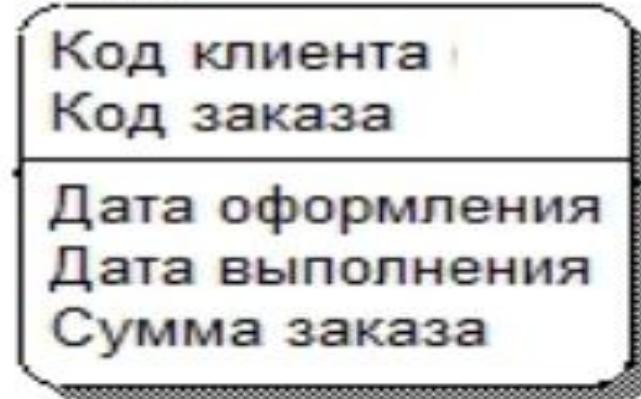
В нотации IDEF1X **атрибуты сущности** изображаются в виде списка их имен внутри блока сущности.

Атрибуты, **определяющие первичный ключ**, размещаются наверху списка и **отделяются от других атрибутов горизонтальной чертой**. (иногда еще и выделяются **жирным шрифтом**)

Клиент



Заказ



Выявленные атрибуты могут быть следующих видов:

- **простой** (атомарный, неделимый) – состоит из одного компонента с независимым существованием (например, «должность работника», «зарплата» и т. д.);
- **составной** (псевдоатомарный) – состоит из нескольких компонентов (например, «ФИО», «адрес», и т. д.)
- **однозначный** – содержит только одно значение для одного экземпляра сущности (например, пол, должность, дата рождения и т. д.);
- **многозначный** – содержит несколько значений (например, у одного отделения компании может быть несколько контактных телефонов);
- **производный** (вычисляемый) – значение атрибута может быть определено по значениям других атрибутов (например, «возраст» может быть определен по «дате рождения» и текущей дате, установленной на компьютере);
- **ключевой** – служит для уникальной идентификации экземпляра сущности (входит в состав первичного ключа);
- **неключевой** (описательный) – не входит в первичный ключ;
- **обязательный** – при вводе нового экземпляра в сущность или редактировании обязательно указывается допустимое значение атрибута, т.е. оно после редактирования не может быть неопределенным (NOT NULL).

После определения атрибутов **задаются их домены** (область допустимых значений атрибута), а также тип, размер и формат атрибутов, например:

- наименование должности – набор из букв русского алфавита длиной не более 60 символов;
- пол – допустимые значения «М» и «Ж»;
- возраст – положительное число не более 2 цифр.



# Нотация IDEF1X

На основании выделенного множества атрибутов для сущности **определяется набор ключей**.

**Ключ** – один или несколько атрибутов сущности, служащих для **однозначной идентификации** ее экземпляров или для их **быстрого поиска**.

Выделяют следующие **типы ключей**:

- **суперключ** (*superkey*) – атрибут или множество атрибутов, которое единственным образом идентифицирует экземпляр сущности.
- **потенциальный ключ** (*potential key*) – суперключ, который не содержит подмножества, также являющегося суперключом данной сущности. Если ключ состоит из нескольких атрибутов, то он называется составным ключом;
- **первичный ключ** (*primary key*) – потенциальный ключ, который выбран для уникальной идентификации экземпляров внутри сущности;
- **альтернативные ключи** (*alternative key*) – потенциальные ключи, которые не выбраны в качестве первичного ключа.
- **суррогатный ключ** (*искусственный ключ, surrogate key*) – если в сущности нет ни одной комбинации атрибутов, подходящей на роль потенциального ключа, то в сущность добавляют отдельный атрибут.
- **внешний ключ** (*foreign key*) – атрибут в дочерней сущности для поддержки связи (соответствует первичному ключу родительской сущности).



# Нотация IDEF1X

Если потенциальных ключей несколько, то для выбора первичного ключа рекомендуется придерживаться следующих правил:

- количество атрибутов, входящих в ключ, должно быть минимальным (желательно, чтобы ключ был атомарным, т. е. состоял из одного атрибута);
- размер ключа в байтах должен быть как можно короче;
- тип домена ключа – числовой;
- вероятность изменения значений ключа была наименьшей (например, «Номер пенсионного страхового свидетельства» более постоянный параметр, чем «ИНН» или «Номер паспорта»).

Пусть имеется таблица, содержащая сведения о студенте, со следующими столбцами:

- фамилия;
- имя;
- отчество;
- дата рождения;
- место рождения;
- номер группы;
- ИНН;
- номер пенсионного страхового свидетельства (НПСС);
- номер паспорта;
- дата выдачи паспорта;
- организация, выдавшая паспорт.

Для каждого экземпляра (записи) в качестве суперключа может быть выбран весь набор атрибутов. Потенциальными ключами (уникальными идентификаторами) могут быть:

- ИНН;
- номер пенсионного страхового свидетельства;
- номер паспорта.

В качестве уникального идентификатора можно было бы выбрать совокупность атрибутов «Фамилия»+«Имя»+«Отчество», если вероятность учебы в вузе двух полных тезок была бы равна нулю.

## Определение связей

Наиболее **характерными типами связей** между сущностями являются:

- *связи типа «часть–целое»*, определяемые обычно глаголами «состоит из», «включает» и т.п.;
- *классифицирующие связи* (например, «тип – подтип», «множество – элемент», «общее – частное» и т. п.);
- *производственные связи* (например, «начальник–подчиненный»);
- *функциональные связи*, определяемые обычно глаголами «производит», «влияет», «зависит от», «вычисляется по» и т. п.

# Нотация IDEF1X

## Определение связей

Связь характеризуется следующим набором параметров:

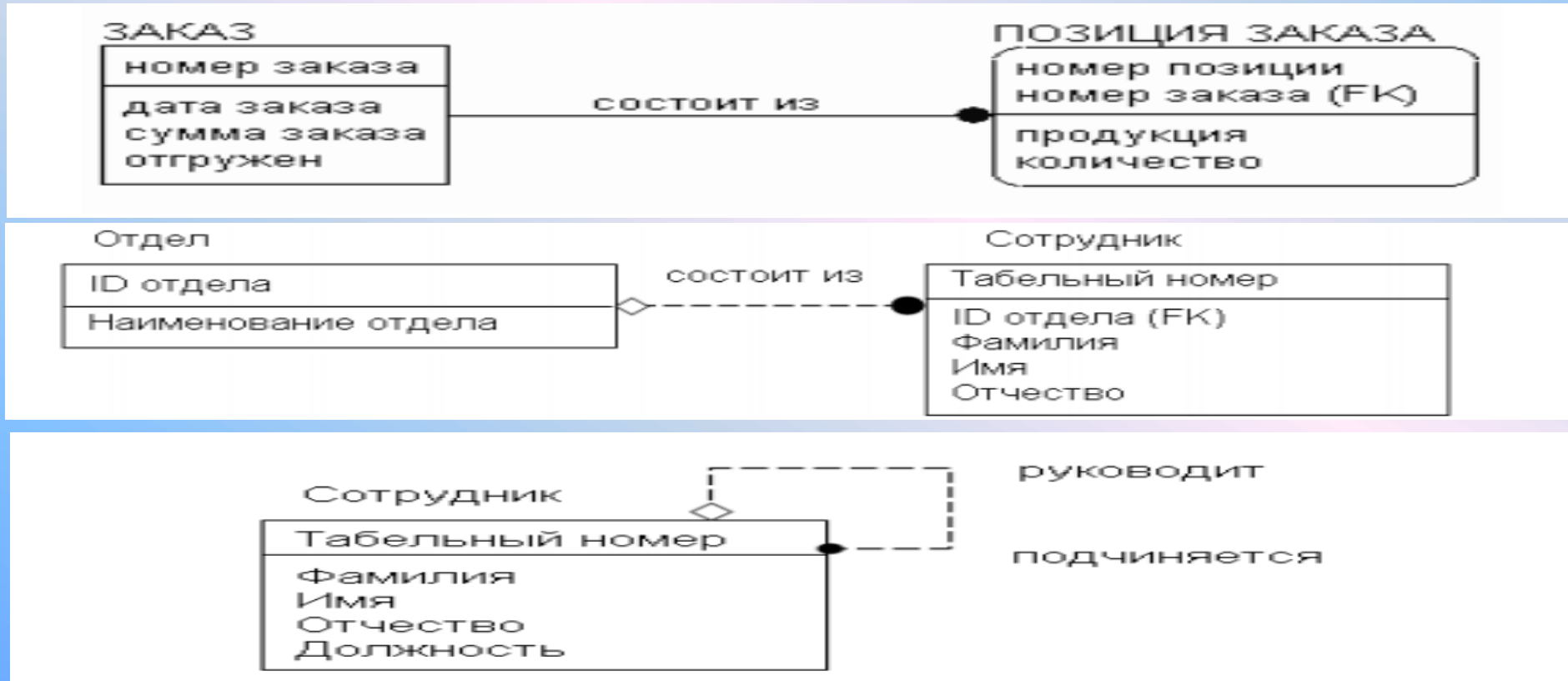
- *именем* – указывается в виде глагола и определяет семантику (смысловую подоплеку) связи;
- *кратностью* (кардинальность, мощность):
  - один-к-одному (1:1),
  - один-ко-многим (1:N),
  - многие-ко-многим (N:M,  $N = M$  или  $N \neq M$ ).
- *типом*:
  - идентифицирующая (атрибуты одной сущности, называемые внешним ключом, входят в состав дочерней и служат для идентификации ее экземпляров, т.е. входят в ее первичный ключ),
  - неидентифицирующая (внешний ключ имеется в дочерней сущности, но не входит в состав первичного ключа);
- *обязательностью*:
  - обязательная (при вводе нового экземпляра в дочернюю сущность заполнение атрибутов внешнего ключа обязательно, для введенных значений должен существовать экземпляр в родительской сущности),
  - необязательная (заполнение атрибутов внешнего ключа в экземпляре дочерней сущности необязательно или введенным значениям не соответствует экземпляр в родительской сущности);
- *степенью участия* – количеством сущностей, участвующих в связи.
  - унарная (рекурсивная) – сущность может быть связана сама с собой. Например, в таблице «Работники» могут быть записи и по подчиненным, и по их начальникам. Тогда возможна связь «начальник» – «подчиненный», определенная на одной таблице;
  - тернарная – связывает три сущности. Например, «Студент» на «Сессии» получил «Оценку по дисциплине»;
  - квaternарная и т.д.

В методологии IDEF1X степень участия может быть только **унарной или бинарной**. Связи большей степени приводятся к бинарному виду.

# Нотация IDEF1X

## Определение связей

1. Идентифицирующая связь отображается сплошной линией.
2. Необязательность обозначается ромбиком, обязательность - кружком.
3. Рекурсивная (иерархическая связь – указывает на связь сущности саму на себя.

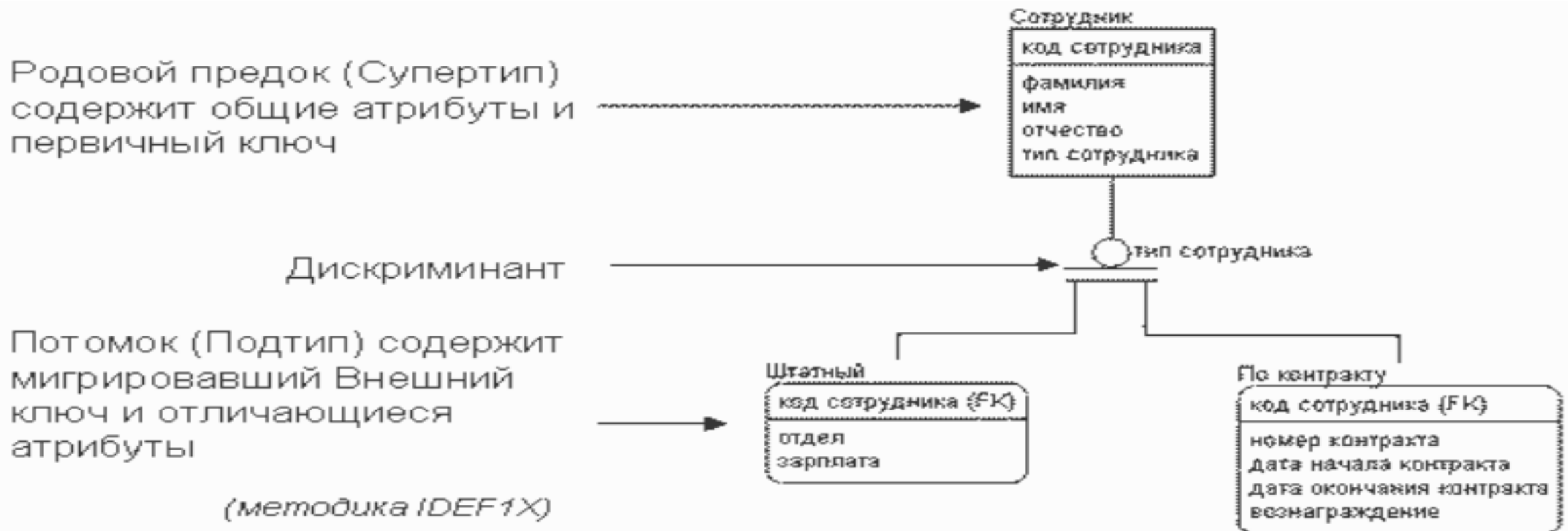


# Нотация IDEF1X

Определение **суперклассов** (супертипов) и **подклассов** (подтипов).

В тех случаях, когда две и более сущностей по набору атрибутов незначительно отличаются друг от друга, можно применять в модели **конструкцию – иерархию наследования (категорий)**, включающую в себя суперклассы и подклассы.

**Суперкласс** – сущность, включающая в себя **подклассы**.



# Нотация IDEF1X

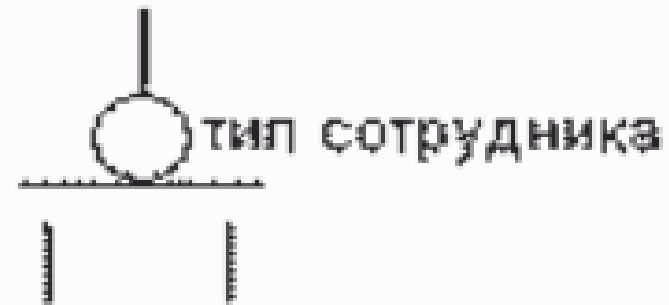
Определение **суперклассов** (супертипов) и **подклассов** (подтипов).

Для каждой категории требуется указать *дискриминатор* – атрибут родового предка, который показывает, как отличить одну сущность от другой. В приведенном примере дискриминатор – атрибут «Тип сотрудника».

Виды **иерархии категорий** в методике IDEF1X:



Полная  
иерархия  
категорий



Неполная  
иерархия  
категорий

# Нотация IDEF1X

Определение **суперклассов** (супертипов) и **подклассов** (подтипов).

Виды **иерархии категорий** в методике IDEF1X:

Иерархия наследования (неполная категория)



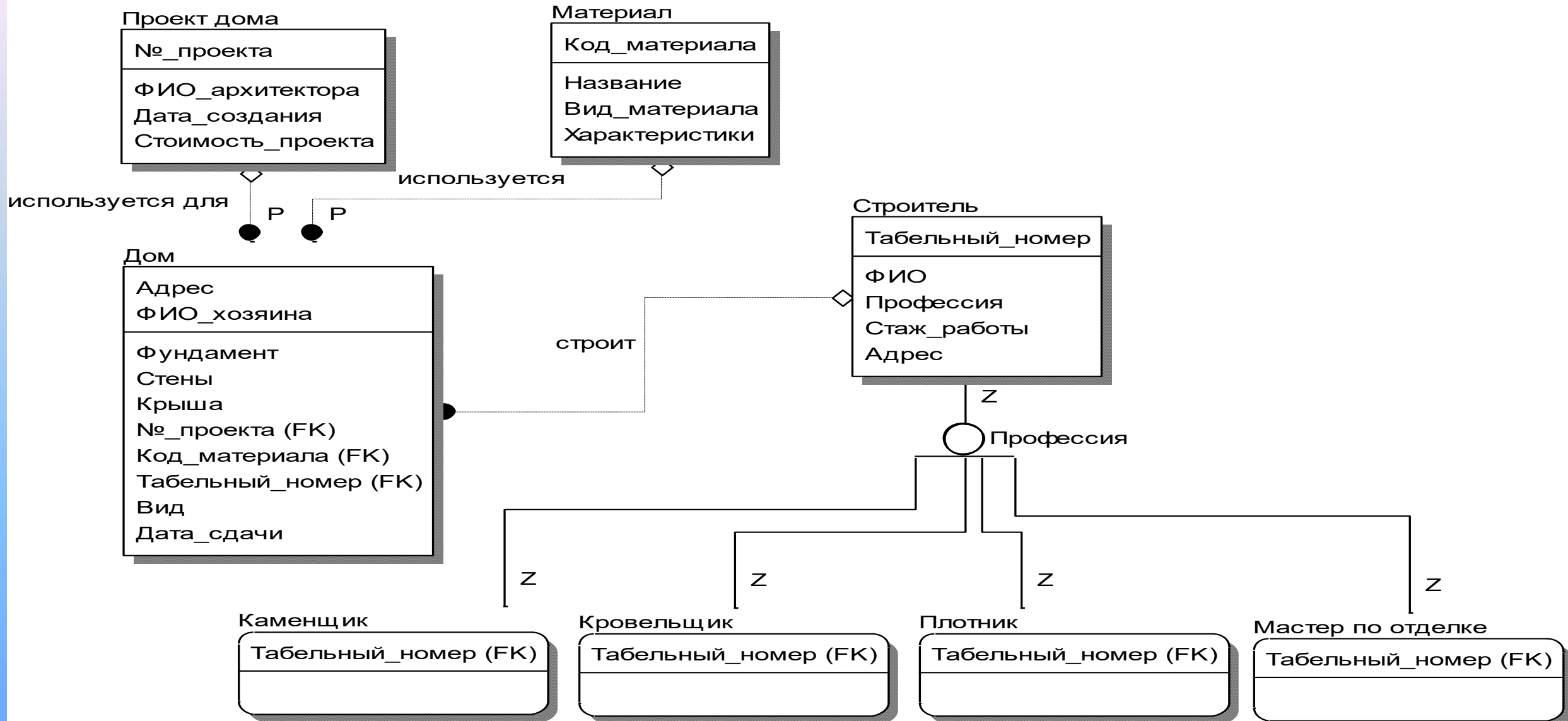
Если категория еще не выстроена полностью и в родовом предке могут существовать экземпляры, которые не имеют соответствующих экземпляров в потомках, то такая категория будет **неполной**.



В **полной категории** одному экземпляру родового предка обязательно соответствует экземпляр в каком-либо потомке, т. е. в примере сотрудник обязательно является либо совместителем, либо консультантом, либо постоянным сотрудником.



# Нотация IDEF1X



Пример ER-диаграммы в нотации IDEF1X



## CASE-инструменты для создания моделей

- Computer Associates AllFusion (**ERWin**, BPWin, Paradigm Plus) - **IDEF, ER, DFD**
- Rational Rose - **UML**
- ARIS Toolset - **DFD, UML, eEPC, Industrial and Office process, Value-added chain diagram (VAD)**
- Microsoft Visio - **IDEF, ER, DFD, WorkFlow, UML, Basic Flowchart, Cross- Functional Flowchart (SwimLine)**
- Design/IDEF - **IDEF, DFD**
- Case – аналитик - **IDEF, DFD**

**Спасибо за внимание !**