

Реляционная модель данных

Технологии баз данных. Лекция 4

Семантическое моделирование

- **Семантическое моделирование** – разработка модели предметной области, представляющей смысл данных этой предметной области.

Понятия и описания

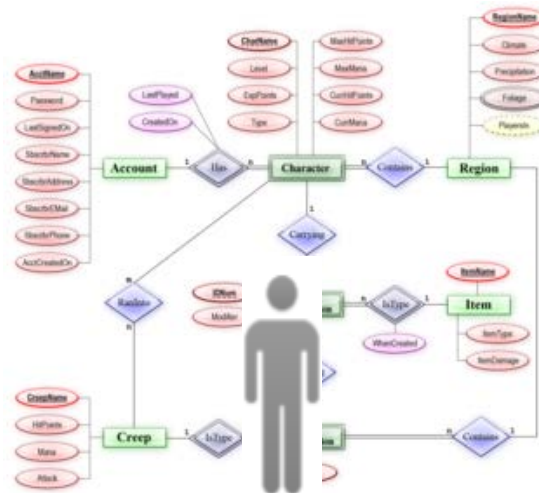
Схема базы данных

время

Модель предметной области



Пользователь

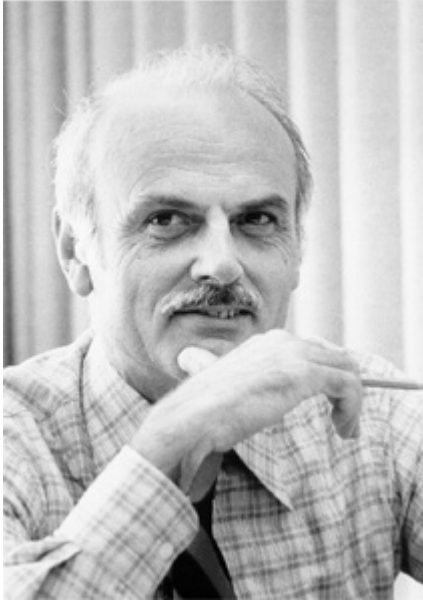


Аналитик

CUSTOMER	LINEORDER	PART
CUSTKEY	ORDERKEY	PARTKEY
NAME	LINENUMBER	NAME
ADDRESS	CUSTKEY	MFGR
CITY	PARTKEY	CATEGOTY
NATION	SUPPKEY	BRAND1
REGION	ORDERDATE	COLOR
PHONE	ORDPRIORITY	TYPE
MKTSEGMENT	SHPPRIORITY	SIZE
<i>Size=scalefactor x 30,000</i>	QUANTITY	CONTAINER
	EXTENDEDPRICE	<i>Size=200,000 x (1 + log₂ scalefactor)</i>
	ORDTOTALPRICE	DATEKEY
	DISCOUNT	DATE
	QFVENUE	DAYOFWEEK
	PLYCOST	MONTH
	MITDATE	YEAR
	MODE	YEARMONTHNUM
	<i>=scalefactor x 2,000</i>	YEARMONTH
		DAYNUMWEEK
		<i>... (9 add'l attributes)</i>
		<i>Size= 365 x 7</i>

АБД

Историческая справка



Эдгар Франк Кодд
1923-2003

- **Реляционная модель данных (РМД)** изобретена Т. Коддом в 1970 г.
 - Codd E.F. The Relation Model for Large Shared Data Banks // Communications of the ACM. 1970. Vol. 13, No. 6. P. 377-387.
- За разработку реляционной модели данных Кодд был удостоен звания "IBM Fellow" (1976) и премии Тьюринга (1981).

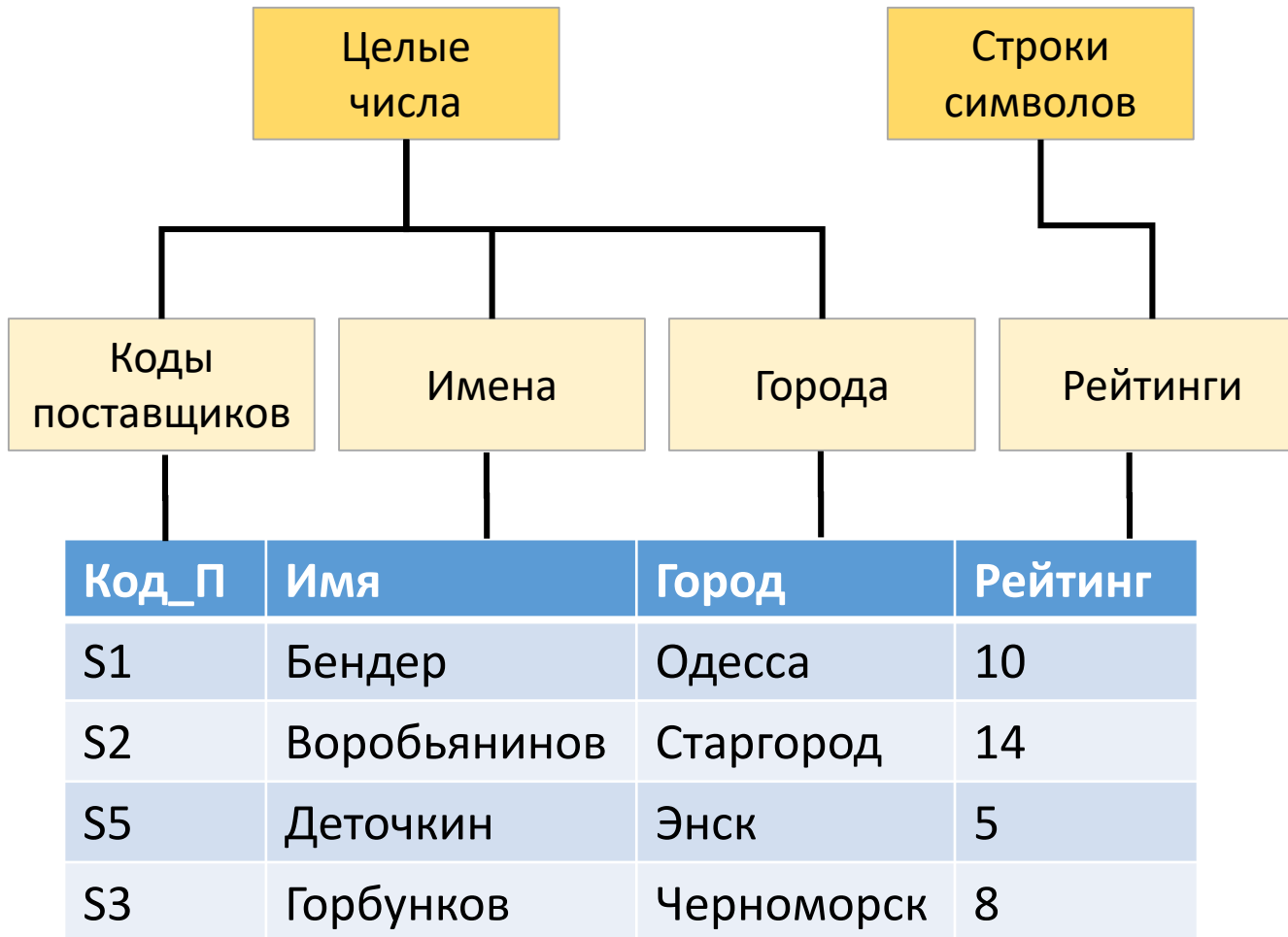
Состав РМД

- Структурный аспект
 - Данные в базе данных представляют собой набор отношений.
- Аспект целостности
 - Отношения отвечают определенным условиям целостности.
 - РМД поддерживает декларативные ограничения целостности уровня домена (типа данных), уровня отношения и уровня базы данных.
- Аспект манипулирования
 - РМД поддерживает операторы манипулирования отношениями (реляционная алгебра, реляционное исчисление).
- Аспект нормализации
 - Ограничения на структуру отношений базы данных, улучшающие эффективность работы с базой данных.

Структура реляционных данных

Термин РМД	Англ. термин	Неформальный термин
Отношение	Relation	Таблица
Кортеж	Tuple	Запись таблицы
Атрибут	Attribute	Столбец таблицы
Домен	Domain	Тип данных у значений в столбце таблицы
Первичный ключ	Primary key	Поле - уникальный идентификатор записи

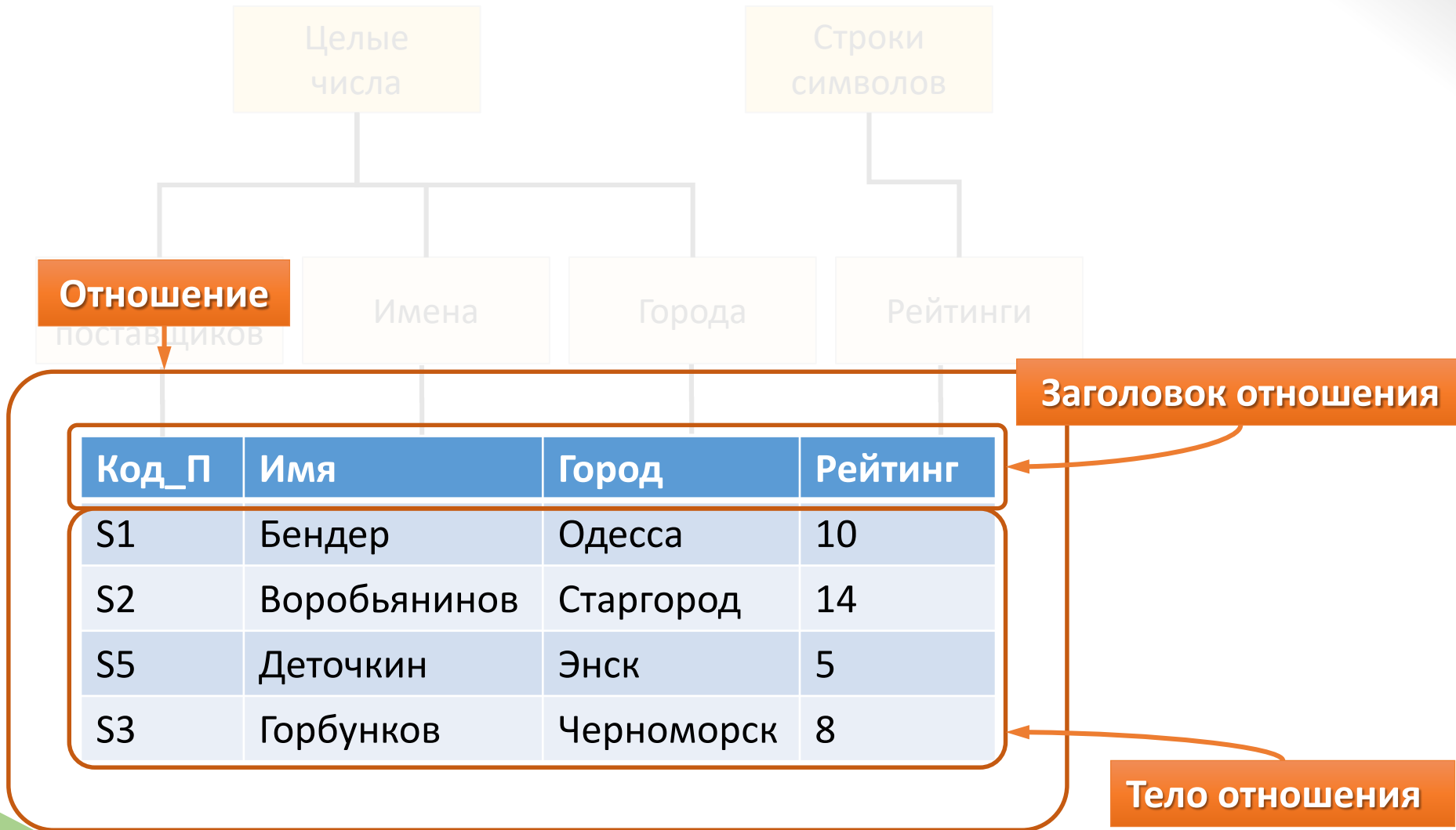
Структура реляционных данных



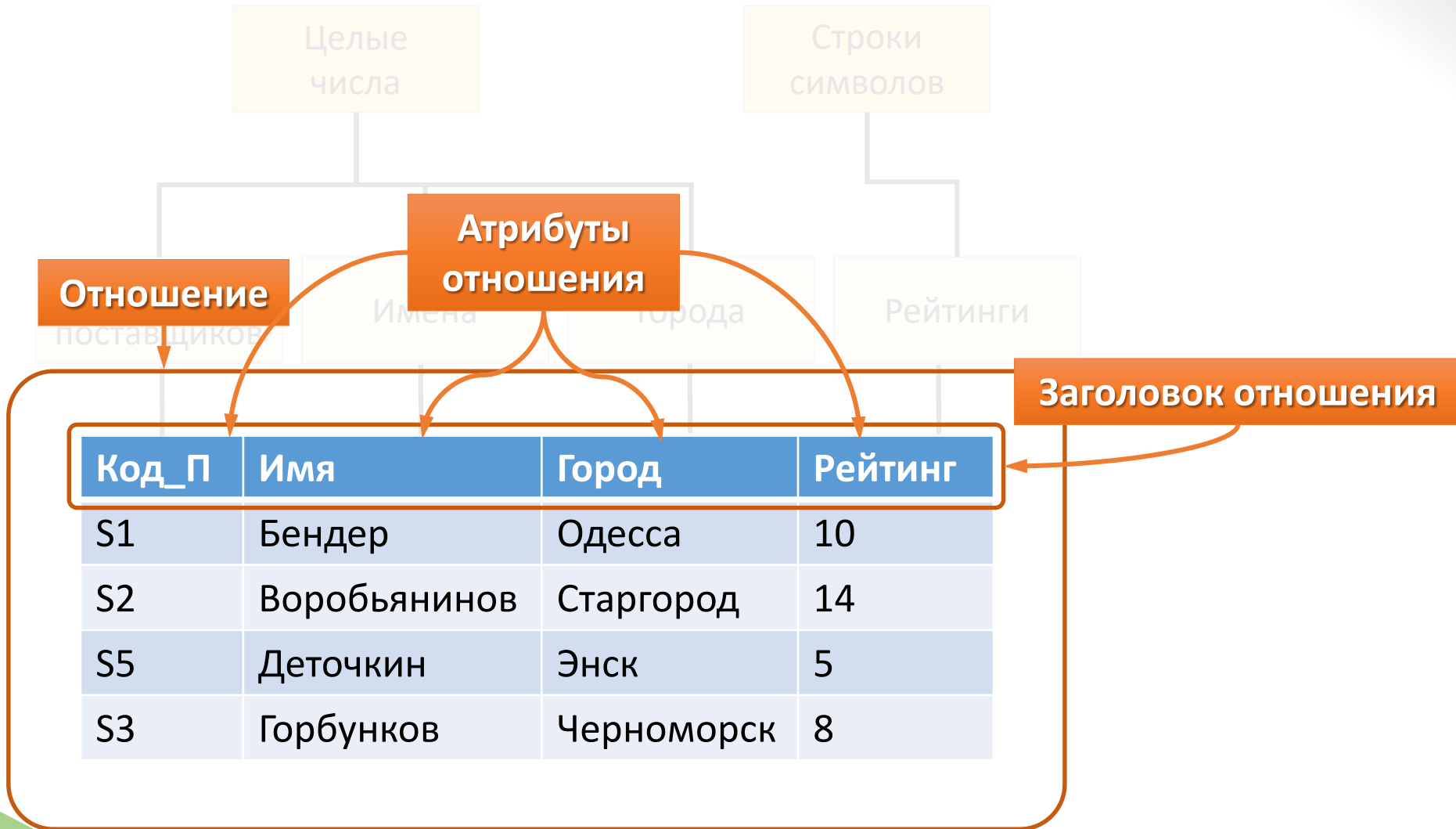
Структура реляционных данных



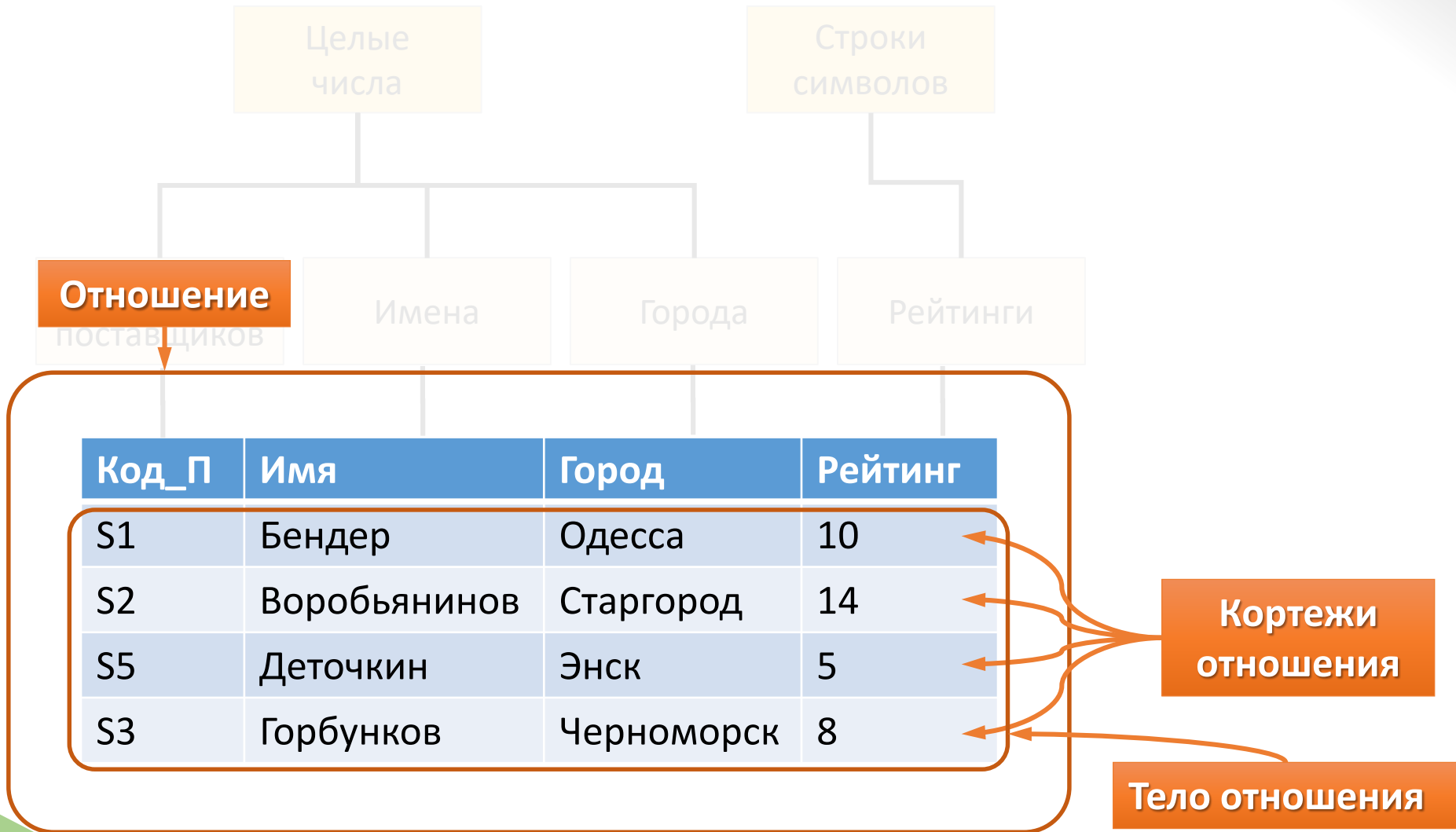
Структура реляционных данных



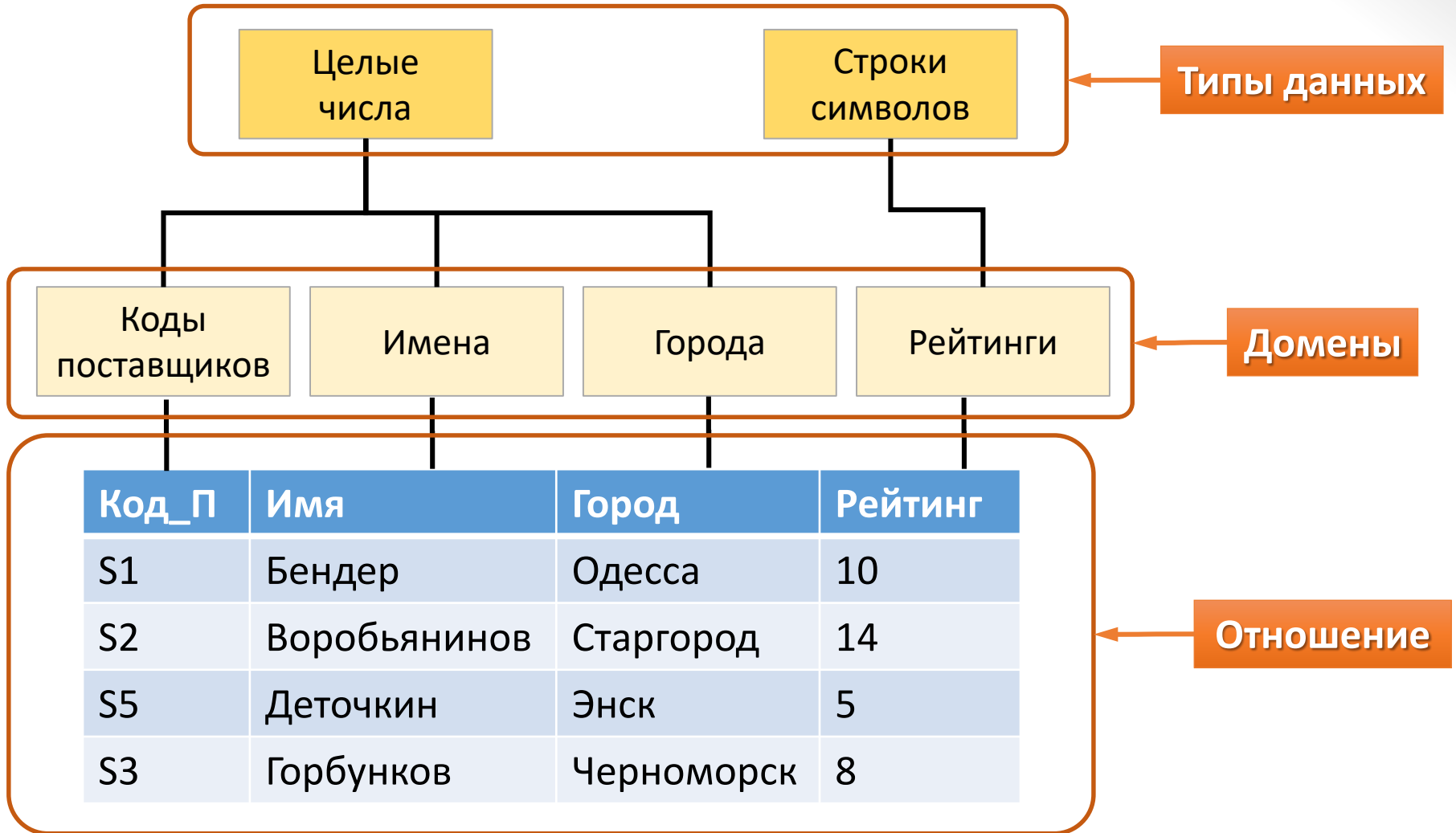
Структура реляционных данных



Структура реляционных данных



Структура реляционных данных



Домен

- *Домен* – именованное множество скалярных значений одного типа.
 - Скалярное (атомарное) значение не имеет внутренней структуры.
 - ФИО: строка[30] – скаляр
 - ФИО: { строка[10], строка[10], строка[10] } – не скаляр
 - Создание /удаление домена
 - create domain Код_Д char(5)
 - drop domain Код_Д

Для чего нужны домены?

Домены ограничивают сравнения:

- `select Имя_Д
from P, SP
where P.Код_Д=SP.Код_Д`
- `select Имя_Д
from P, SP
where P.Вес=SP.Количество`

Ошибка! Разные домены!

Отношение

- *Отношение R* , определенное на множестве доменов $D=\{D_1, \dots, D_k\}$ состоит из двух частей: заголовков и тело.
- *Заголовок отношения* – множество пар **<имя-атрибута : имя-домена>**, т.е. множество $\{<A_1:D_1>, \dots, <A_n:D_n>\}$, где имена атрибутов A_j различны, домены атрибутов $D_j \in D$ не обязательно различны.
- *Тело отношения* – множество кортежей.
- *Кортеж* – множество пар **<имя-атрибута : значение-атрибута>**, т.е. множество $\{<A_1:v_{i1}>, \dots, <A_n:v_{in}>\}$, где $i \in \{1, \dots, m\}$.
- Значение m – *кардинальное число отношения*.
- Значение n – *степень (арность) отношения*.

Отношение

Код_П	Имя	Город	Рейтинг
S1	Бендер	Одесса	10
S2	Воробьянинов	Старгород	14
S5	Деточкин	Энск	5
S3	Горбунков	Черноморск	8

- { Код_П : Код_П, Имя : Имя, Город : Город, Рейтинг : Рейтинг }
- {
 - {<Код_П : S1>, <Имя : Бендер>, <Город : Одесса>, <Рейтинг : 10>},
 - {<Код_П : S2>, <Имя : Воробьянинов>, <Город : Старгород>, <Рейтинг : 14>}
 - {<Код_П : S5>, <Имя : Деточкин>, <Город : Энск>, <Рейтинг : 5>}
 - {<Код_П : S3>, <Имя : Горбунков>, <Город : Черноморск>, <Рейтинг : 8>}
- }

Свойства отношений

- Не может содержать одинаковых кортежей (тело отношения – множество)
- Кортежи не упорядочены (тело отношения – множество)
- Атрибуты не упорядочены (заголовок отношения – множество)
- Значения атрибутов кортежей атомарные (или отношение *нормализовано*)

Виды отношений

- *Базовое отношение* – именованное отношение (переменная отношения), которое постоянно хранится в базе данных.
- *Производное отношение* – отношение, определенное через базовые посредством реляционного выражения.
- *Представление* – виртуальное (не хранящееся постоянно в базе данных) именованное производное отношение. Применяется для реализации внешнего уровня ANSI/SPARC архитектуры систем баз данных.
- *Снимок* – именованное производное отношение, которое постоянно хранится в базе данных. Применяется для реконструкции базы данных после сбоев.
- *Результат запроса* – неименованное производное отношение.

Виды отношений

- Базовое отношение

- create **base relation** S (
Код_П domain (Код_П),
Имя domain (Имя),
Город domain (Город),
primary key (Код_П))

- Представление

- create **view** Надежные_П as
select Код_П, Имя
from S
where Рейтинг >= 15

- СНИМОК

- create **snapshot** Поставляемые_Детали as
select Код_Д, Имя
from P, SP
where P.Код_Д=SP.Код_Д
refresh every day

- Промежуточный результат (выражение реляционной алгебры)

- ((S JOIN SP) WHERE Код_Д='P1')[Код_П, Имя_П]

Целостность реляционных данных

- Ключи (целостность сущностей)
 - Потенциальные, первичные и альтернативные
 - Простые и составные
 - Внешние
- Ссылочная целостность
- NULL-значения

Потенциальный ключ

- Подмножество K атрибутов отношения R является *потенциальным ключом*, если K обладает следующими свойствами:
 1. Уникальность
 - Никакие два кортежа в R не могут иметь одинаковое значение в K .
 2. Неизбыточность
 - Никакое подмножество K не обладает свойством уникальности.
- Потенциальных ключей может быть несколько.

Выбор потенциальных ключей

- На основе определения и семантики предметной области.
- Пример:

выбрать потенциальные ключи отношения Персона

- Фамилия
- Имя
- Отчество
- Пол
- Дата рождения
- ИНН
- СНИЛС
- Серия паспорта
- Номер паспорта
- Место и дата выдачи паспорта
- Индекс
- Город
- Адрес

Назначение потенциальных ключей

- Потенциальный ключ позволяет адресовать кортежи отношения.

- `select Имя
from S
where Код_П='S1'`

- Результат:

Имя
Бендер

Потенциальный
ключ



Код_П*	Имя	Город	Рейтинг
S1	Бендер	Одесса	15
S2	Воробьянинов	Старгород	14
S5	Деточкин	Энск	5
S3	Горбунков	Черноморск	8
S10	Бендер	Москва	20

Виды потенциальных ключей

- Если потенциальный ключ состоит из более чем одного атрибута, он называется *составным*, иначе – *простым*.
- Один из потенциальных ключей должен быть выбран в качестве *первичного ключа*. Остальные потенциальные ключи называются *альтернативными ключами*.

Внешний ключ

- *Внешний ключ FK* в отношении $R2$ – это подмножество атрибутов $R2$ такое, что
 - существует отношение $R1$ (не обязательно отличное от $R2$) с потенциальным ключом $СК$
 - для каждого значения FK существует кортеж в $R1$ с совпадающим значением $СК$.

Внешние ключи

Поставщики

Код_П*	Имя_П
S1	Бендер
S2	Воробьянинов
S5	Деточкин
S3	Горбунков
S10	Бендер

Внешний ключ

Внешний ключ

Поставки

Код_П	Код_Д	К-во
S1	P1	300
S1	P5	500
S3	P2	900
S3	P1	100
S10	P2	400

Детали

Код_Д*	Имя_Д
P1	Гайка
P2	Болт
P5	Шуруп
P7	Дюбель

Внешние ключи

- Внешний ключ должен определяться на тех же доменах, что и соответствующий потенциальный ключ.
- Внешний ключ не обязательно должен входить в первичный (потенциальный) ключ, но это желательно.

Внешние ключи

- Потенциальный ключ может содержать значение, не являющееся значением внешнего ключа.

Поставщики

Код_П*	Имя_П
S1	Бендер
S2	Воробьянинов
S5	Деточкин
S3	Горбунков
S10	Бендер

Поставки

Код_П*	Код_Д*	К-во
S1	P1	300
S1	P5	500
S3	P2	900
S3	P1	100
S10	P2	400

Детали

Код_Д*	Имя_Д
P1	Гайка
P2	Болт
P5	Шуруп
P7	Дюбель

Внешние ключи

- ***Внешний ключ составной (простой)***, если соответствующий потенциальный ключ составной (простой).

Внешние ключи

- Внешний ключ и соответствующий потенциальный ключ могут принадлежать одному и тому же отношению (ссылающееся и целевое отношение совпадают).

Код_Сотр*	ФИО	Начальник
001	Бендер О.И.	NULL
002	Воробьянинов И.М.	001
005	Деточкин Ю.И.	002
007	Бонд Д.	001

Внешние ключи

- create base relation SP (
Код_П domain (Код_П),
Код_Д domain (Код_Д),
Количество domain (Количество),
primary key (Код_П, Код_Д),
foreign key (Код_П) references S (Код_П),
foreign key (Код_Д) references P (Код_Д)
)

Правила внешних ключей

- Что делать в случае попытки удалить (обновить) потенциальный ключ, на который ссылается внешний ключ?
 - Удаление поставщика, имеющего хотя бы одну поставку.
 - Изменение кода детали, входящей хотя бы в одну поставку.

Поставщики

Код_П*	Имя_П
S1	Бендер
S2	Воробьянинов
S5	Деточкин
S3	Горбунков
S10	Бендер

Поставки

Код_П*	Код_Д*	К-во
S1	P1	300
S1	P5	500
S3	P2	900
S3	P1	100
S10	P2	400

Детали

Код_Д*	Имя_Д
P1	Гайка
P2	Болт
P5	Шуруп
P7	Дюбель

Правила внешних ключей

- Ограничить
- Каскадировать
- Установить в NULL
- Установить по умолчанию

Правила внешних ключей: ограничить

- **Ограничить** – запретить удаление (обновление) кортежей ссылочного отношения до момента, когда в ссылающемся отношении не будут отсутствовать кортежи с соответствующим значением внешнего ключа.

Правила внешних ключей: ограничить

НЕ могут быть обновлены (удалены)

Поставщики

Код_П*	Имя_П
S1	Бендер
S2	Воробьянинов
S5	Деточкин
S3	Горбунков
S10	Бендер

Поставки

Код_П*	Код_Д*	К-во
S1	P1	300
S1	P5	500
S3	P2	900
S3	P1	100
S10	P2	400

Детали

Код_Д*	Имя_Д
P1	Гайка
P2	Болт
P5	Шуруп
P7	Дюбель

Могут быть обновлены (удалены)

Правила внешних ключей: каскадировать

- ***Каскадировать*** – удалить (обновить) кортежи ссылочного отношения с соответствующим значением внешнего ключа.

Правила внешних ключей: каскадировать

Поставщики

Код_П*	Имя_П
007	Бендер
S2	Воробьянинов
S5	Деточкин
S3	Горбунков
S10	Бендер

Поставки

Код_П*	Код_Д*	К-во
007	P1	300
007	P5	500
УДАЛЕН		
S3	P1	100
УДАЛЕН		

Детали

Код_Д*	Имя_Д
P1	Гайка
УДАЛЕН	
P5	Шуруп
P7	Дюбель



Ограничение и каскад

- create base relation SP (
Код_П domain (Код_П),
Код_Д domain (Код_Д),
Количество domain (Количество),
primary key (Код_П, Код_Д),
foreign key (Код_П) references S (Код_П)
on delete restrict
on update cascade,
foreign key (Код_Д) references P (Код_Д)
on delete restrict
on update cascade)

Правила внешних ключей

- **Установить в NULL** – удалить (обновить) кортежи ссылочного отношения и в ссылающемся отношении установить у соответствующих кортежей неопределенное значение внешнего ключа (NULL).
- **Установить по умолчанию** – удалить (обновить) кортежи ссылочного отношения и в ссылающемся отношении установить у соответствующих кортежей значение внешнего ключа по умолчанию (в этой базе данных).

Установка в умолчание и NULL

- create base relation EMP (
Код_Сотр domain (Код_Сотр),
ФИО domain (ФИО),
Начальник domain (Код_Сотр),
primary key (Код_Сотр),
foreign key (Начальник) references EMP (Код_Сотр)
on delete set NULL
on update set default '????')

NULL-значения

- *NULL-значение* – специальное значение, показывающее отсутствие информации.
 - $NULL \neq \{ "", ' ', ' ', 0 \}$
 - $NULL \approx \emptyset$
 - create base relation R (
Attr domain (Domain) nulls [not] allowed)

NULL-значения и потенциальные ключи

- Ни один элемент первичного ключа базового отношения не может принимать NULL-значение.
 - Записываемые в отношение кортежи должны быть идентифицируемы!
- Результирующее отношение может иметь NULL-значение в первичном ключе.
- Для альтернативных ключей NULL значения могут быть разрешены или запрещены.
 - Если альтернативный ключ может принимать NULL-значения, то он не может быть выбран в качестве первичного.

NULL-значения и внешние ключи

- ***FK* может принимать неопределенное значение **NULL**, а для каждого отличного от **NULL** значения *FK* в *R1* существует кортеж с совпадающим значением *СК*.**

Правила внешних ключей: установить в NULL

Код_П*	Имя_П
007	Бендер
S2	Воробьянинов
S5	Деточкин
S3	Горбунков
S10	Бендер

Код_ПД*	Код_П	Код_Д	К-во
SP1	NULL	P1	300
SP2	NULL	P5	500
SP5	S3	NULL	900
SP9	S3	P1	100
SP4	S10	NULL	400
SP5	NULL	NULL	700

Код_Д*	Имя_Д
P1	Гайка
УДАЛЕН	
P5	Шуруп
P7	Дюбель

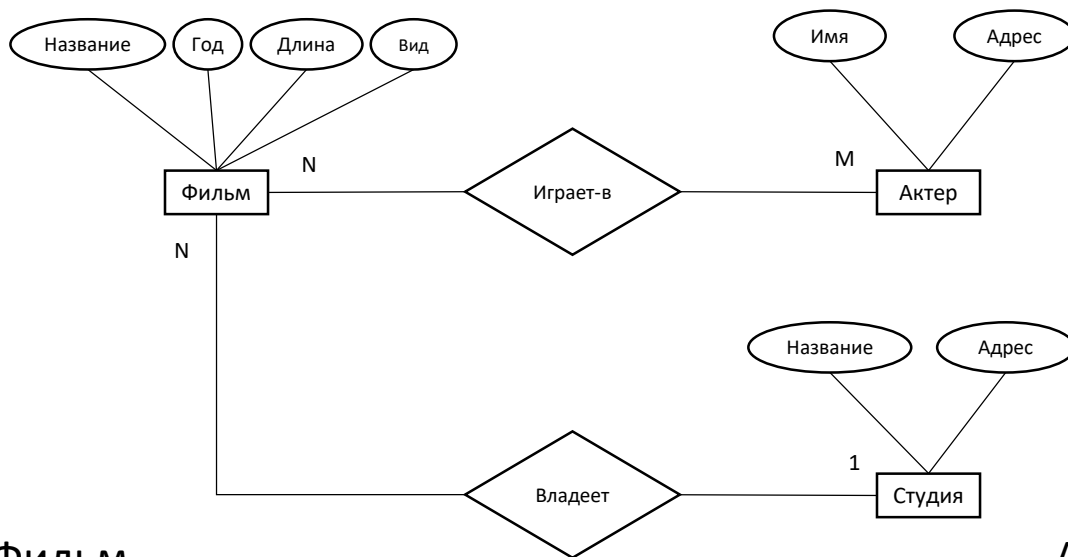
От ER-диаграмм к реляционным схемам

- 1) преобразовать каждую сущность в отношение с тем же набором атрибутов
- 2) заменить каждую связь отношением, атрибуты которого являются ключами сущностей, соединяемых этой связью

Особые ситуации:

- Иногда целесообразно объединить в составе одного отношения два других (например, сущность и связь «один-ко-многим»)

Преобразование сущностей



Фильм

Код_Ф*	Название	Год	Длина	Вид

Актер

Код_А*	Имя	Адрес

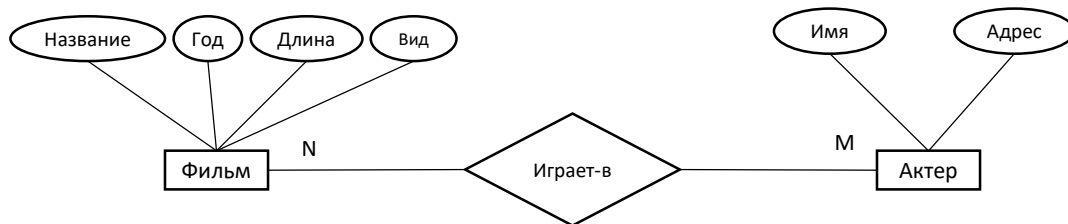
Студия

Код_С*	Название	Адрес

Преобразование связей

- В отношении для связи заносятся ключевые атрибуты каждого из множеств сущностей, соединяемых этой связью.
- Если связь обладает собственными атрибутами, они также вводятся в набор атрибутов отношения для этой связи.

Преобразование связей «многие-ко-многим»



Фильм

Код_Ф*	Название	Год	Длина	Вид

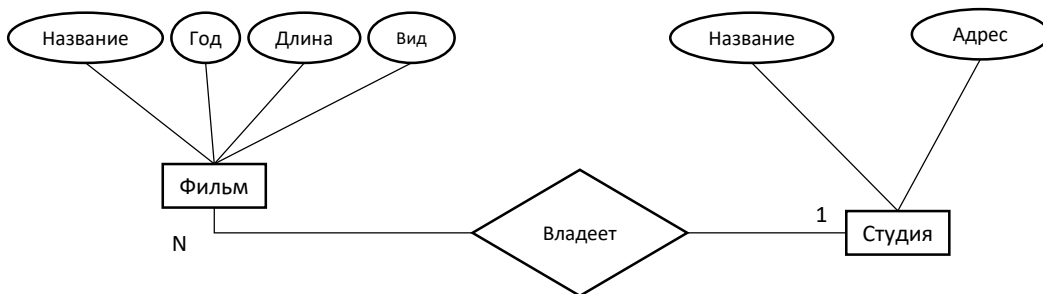
Актер

Код_А*	Имя	Адрес

Актер-Играет-в-Фильме

Код_Ф*	Код_А*

Преобразование связей «один-ко-многим»



Фильм

Код_Ф*	Название	Год	Длина	Вид

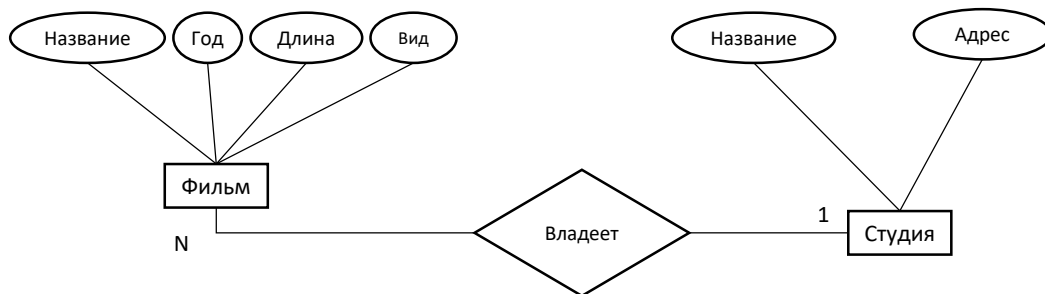
Студия

Код_С*	Название	Адрес

Студия-Владеет-Фильмом

Код_Ф*	Код_С*

Преобразование связей «один-ко-многим»



Фильм

Код_Ф*	Название	Год	Длина	Вид	Код_С

Студия

Код_С*	Название	Адрес

Код_Ф*	Код_С*