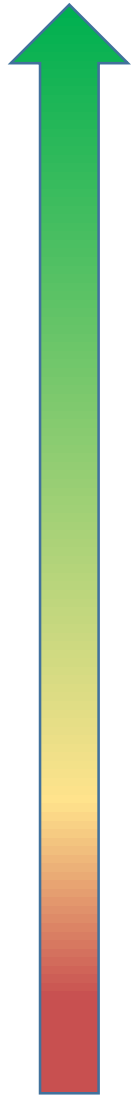


Проектирование информационных систем

Евгений Александрович Мирошниченко
mir@tpu.ru

Уровни абстракции

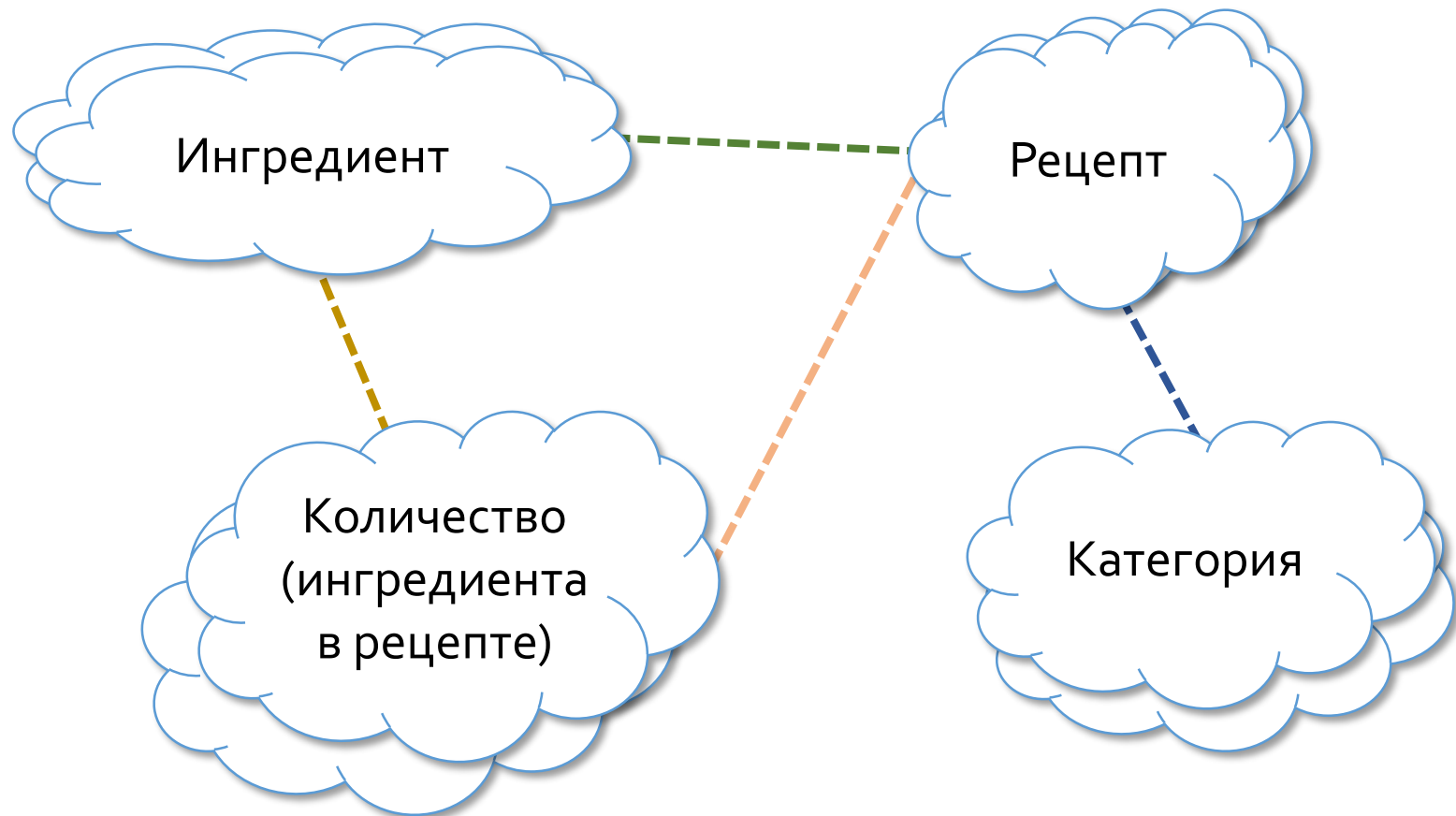


- Описание предметной области (текст, спецификации, «на пальцах»)
- Формальная модель предметной области (семантическая/ концептуальная схема/модель)
- Конкретная схема в рамках выбранной модели данных (логическая схема/модель)
- Конкретная схема в рамках выбранной СУБД (физическая схема/модель)
- БД как то, к чему адресуются запросы (БД на уровне представления, на логическом уровне)
- БД как файлы, байты (БД на уровне реализации, на физическом уровне)
- БД как материальный объект (уровень носителя: жёсткий диск, флэш-карта)

*Хочу создать базу данных «кулинарная книга».
В неё буду вносить точные рецепты (чего и
сколько), и чтобы их можно было распределять
по категориям (салаты, напитки и т.п.)
Да, и чтобы я могла сделать что-то вроде:
«Выдай-ка мне все рецепты, в которых
используется сельдерей».*

Концептуальная схема/модель (неформальная)

Связь Рецепт—Ингредиент лишняя! Ведь ингредиент всегда можно связать с рецептами через количество, у количества есть связи и с рецептом, и с ингредиентом.



Концептуальная схема/модель БД: как уточнить?

- Нужно формализовать «язык» моделирования (средство), которым описывать модель БД (результат)
- Нужна графическая нотация, чтобы рисовать.

Примеры семантических моделей (средств моделирования):

- Модель «сущность — связь», Entity-Relationship Model, ER-Model, ER-модель
- Модель UML (Unified Modeling Language).

Автор — Петер Пин-Шен Чен (Peter Pin-Shan Chen) — предложил ER-модель в 1976 г. и уточнял много раз.

ER-модель стала наиболее популярным средством формального описания концептуальных схем БД.

Основные элементы:

- сущность (entity)
- атрибут, свойство (property)
- связь (relationship).



Сущность (entity)

- Некоторый различимый *тип объекта*:
рецепт, категория, ингредиент, количество...
- Имеет *свойства*:
название блюда, значение, единица измерения...
- Потенциально состоит из множества *экземпляров*:
рецепт салата «Оливье», рецепт торта «Наполеон»...

Атрибут, свойство (property)

- Имеет название
- Имеет тип (data type, domain)
- У экземпляра сущности имеет значение (value)
- Имеет признак обязательности значения (mandatory)
- Может иметь правила (ограничения на значения, ограничения целостности)
- Может иметь значение по умолчанию (default value)
- Может быть идентификатором или входить в составной идентификатор
 - основной (первичный) идентификатор (primary identifier)
 - неосновной (альтернативный) идентификатор (alternative identifier).

Связь (relationship)

- Нечто, что связывает две или более сущностей
поставщик—товар, работник—отдел, супруг—супруга, студент—предмет—семестр...
- В отличие от сущности не имеет собственных свойств
- Тип (мощность) связи характеризует, сколько может быть экземпляров сущностей — участников связи:
 - один к одному (1:1)
 - один ко многим (1: N) или многие к одному (N :1)
 - многие ко многим (N : M)
- Обязательность связи данной сущности характеризует, должен ли в неё обязательно входить хотя бы один экземпляр этой сущности.

- **Сущность: Рецепт**
 - Свойство: Название
 - Свойство: Описание
 - Свойство: Фотография
- **Сущность: Категория**
 - Свойство: Название
 - Свойство: Описание
- **Сущность: Ингредиент**
 - Свойство: Название
 - Свойство: Описание
- **Сущность: Количество (ингредиента в рецепте)**
 - Свойство: Значение
 - Свойство: Единица измерения

- **Связь: Категория—Рецепт**

- Тип связи: один ко многим
- Обязательность рецепта для категории: нет
- Обязательность категории для рецепта: да

- **Связь: Ингредиент—Количество**

- Тип связи: один ко многим
- Обязательность количества для ингредиента: нет
- Обязательность ингредиента для количества: да

- **Связь: Рецепт—Количество**

- Тип связи: один ко многим
- Обязательность рецепта для количества: да
- Обязательность количества для рецепта: нет

Пример: уточнение свойств сущности «Рецепт»

- **Свойство: Название**

- Тип данных: строка(200)
- Обязательность?: да
- Первичный идентификатор?: да
- Ограничения: нет

- **Свойство: Описание**

- Тип данных: CLOB
- Обязательность?: да
- Первичный идентификатор?: нет
- Ограничения: нет

- **Свойство: Фотография**

- Тип данных: BLOB
- Обязательность?: нет
- Первичный идентификатор?: нет
- Ограничения: нет

- **Свойство: Значение**

- Тип данных: действительное число
- Обязательность?: да
- Первичный идентификатор?: нет
- Ограничения: больше нуля

- **Свойство: Единица измерения**

- Тип данных: ?
- Обязательность?: да
- Первичный идентификатор?: нет
- Ограничения: ?

Свойство «Единица измерения», вариант 1

- Тип данных: строка (20)
- Ограничения:
 - значение = 'штука' или
 - значение = 'грамм' или
 - значение = 'миллилитр' или
 - значение = 'пучок' или
 - значение = 'чайная ложка' или
 - значение = 'столовая ложка' или
 - ...

Свойство «Единица измерения», вариант 2

- Убираем его как свойство из сущности «Количество»
- Добавляем сущность «Единица измерения» со свойствами
 - Название
 - Тип данных: строка (20)
 - Обязательность?: да
 - Первичный идентификатор?: да
 - Сокращение
 - Тип данных: строка (10)
 - Обязательность?: да
 - Альтернативный идентификатор?: да
- Добавляем связь Количество—Единица измерения
 - Тип связи: многие ко одному
 - Обязательность количества для ЕИ: нет
 - Обязательность ЕИ для количества: да

Сильные (обычные) и слабые сущности

- **Сильная (обычная) сущность** имеет естественный идентификатор, который однозначно отличает её экземпляры
- **Слабая (зависимая) сущность** не имеет такого естественного идентификатора, и её экземпляры уникальны только вместе со связью с экземпляром сильной сущности (или нескольких) и не могут существовать без связи с экземпляром сильной сущности (или нескольких).

Пример:

- Сущность **Дом** (Город, Улица, Номер, Материал, Количество_этажей)
- Сущность **Квартира** (Номер, Этаж, Количество_комнат, Общая площадь)
- Связь **Дом—Квартира**

Здесь Дом — сильная сущность, Квартира — слабая сущность.

Идентифицирующая связь

- **Неидентифицирующая (обычная) связь** связывает независимые (обычные) сущности
- **Идентифицирующая связь** связывает слабую сущность с нужной её сильной.

Связь Рецепт—Категория — обычная, так как это обычные сущности.

Связи Рецепт—Количество и Ингредиент—Количество — идентифицирующие, так как без них экземпляры Количества не идентифицировать!

По составу атрибутов

- простой (Name)
- составной (First Name, Last Name).

По «осмысленности» значений

- естественный, натуральный, интеллектуальный (Name)
- искусственный, суррогатный (ID).

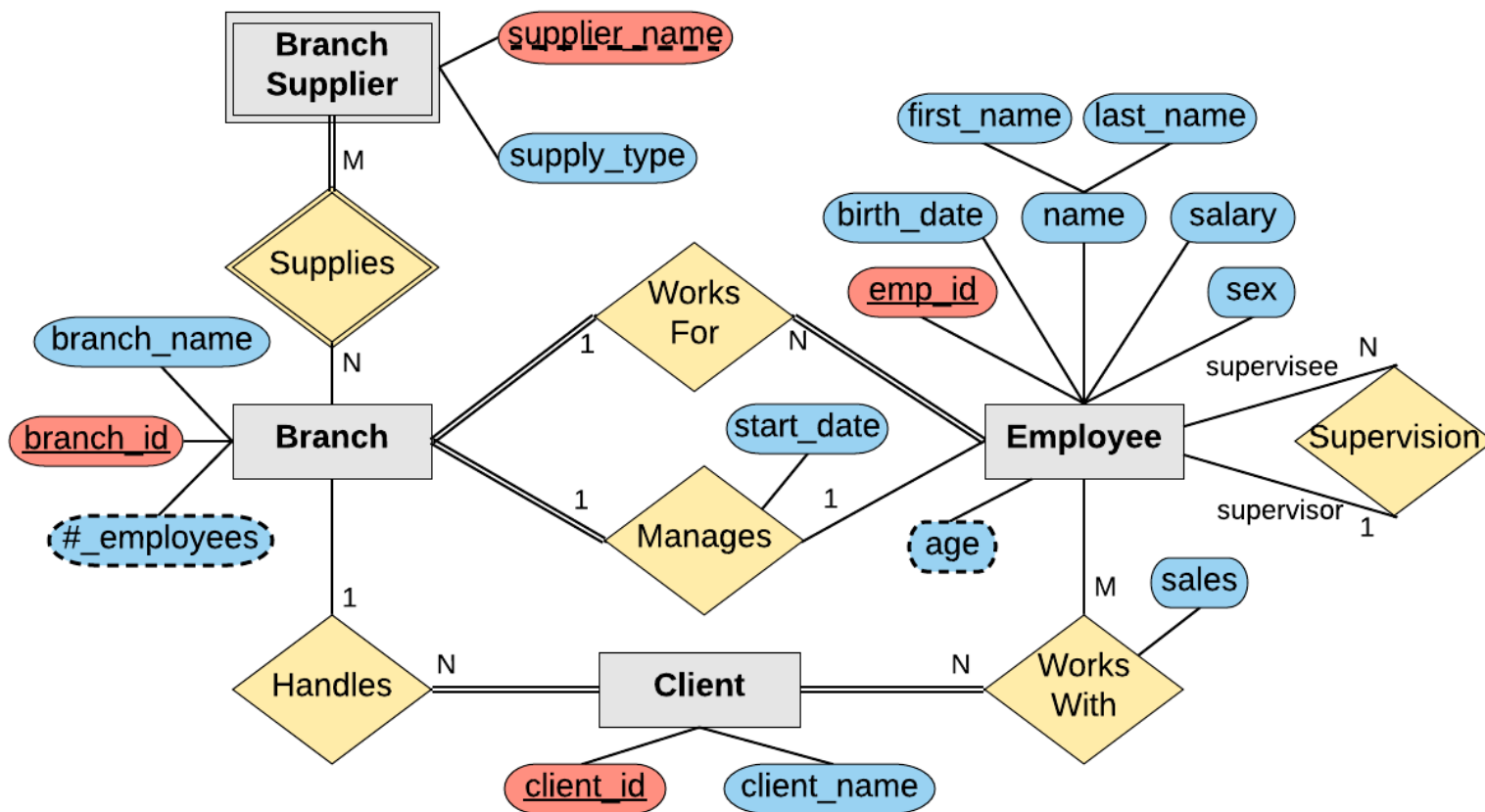
Важно: даже если вы добавите искусственный идентификатор, это не отменяет необходимости описать все остальные (альтернативные).

- ER-диаграмма
- IDEF₁X
- Crow's Foot
- Information Engineering
- OMT
- Oracle Designer
- Bachman's
- UML
- ...

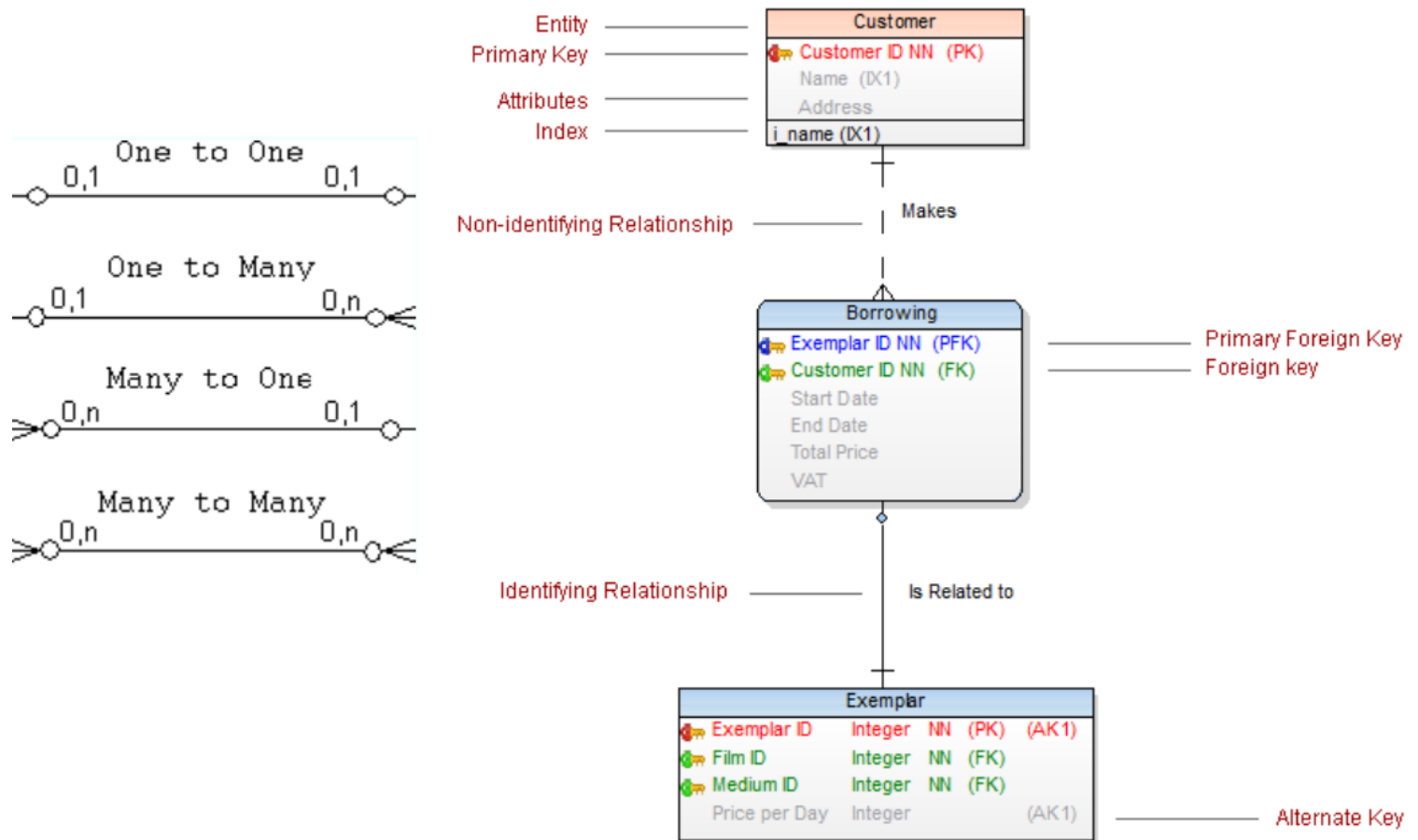
Инструменты проектирования БД

- PowerDesigner (SAP)
- Toad Data Modeler (Quest Software)
 - <https://www.toadworld.com/download/toad-data-modeler/freeware>
- ERwin Data Modeler (erwin, Inc.)
- и т.д., см., например,
https://en.wikipedia.org/wiki/Comparison_of_data_modeling_tools

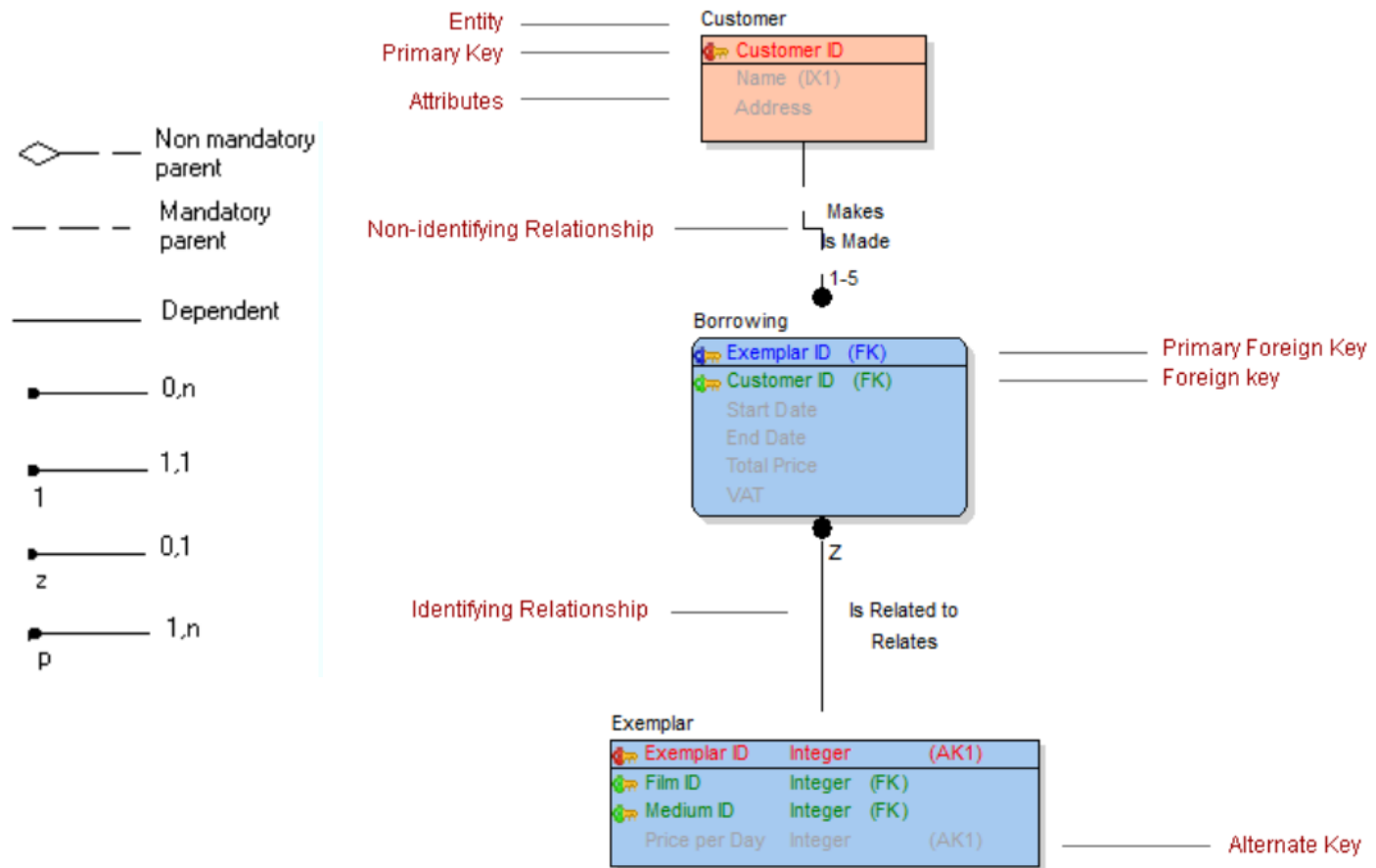
ER-диаграмма



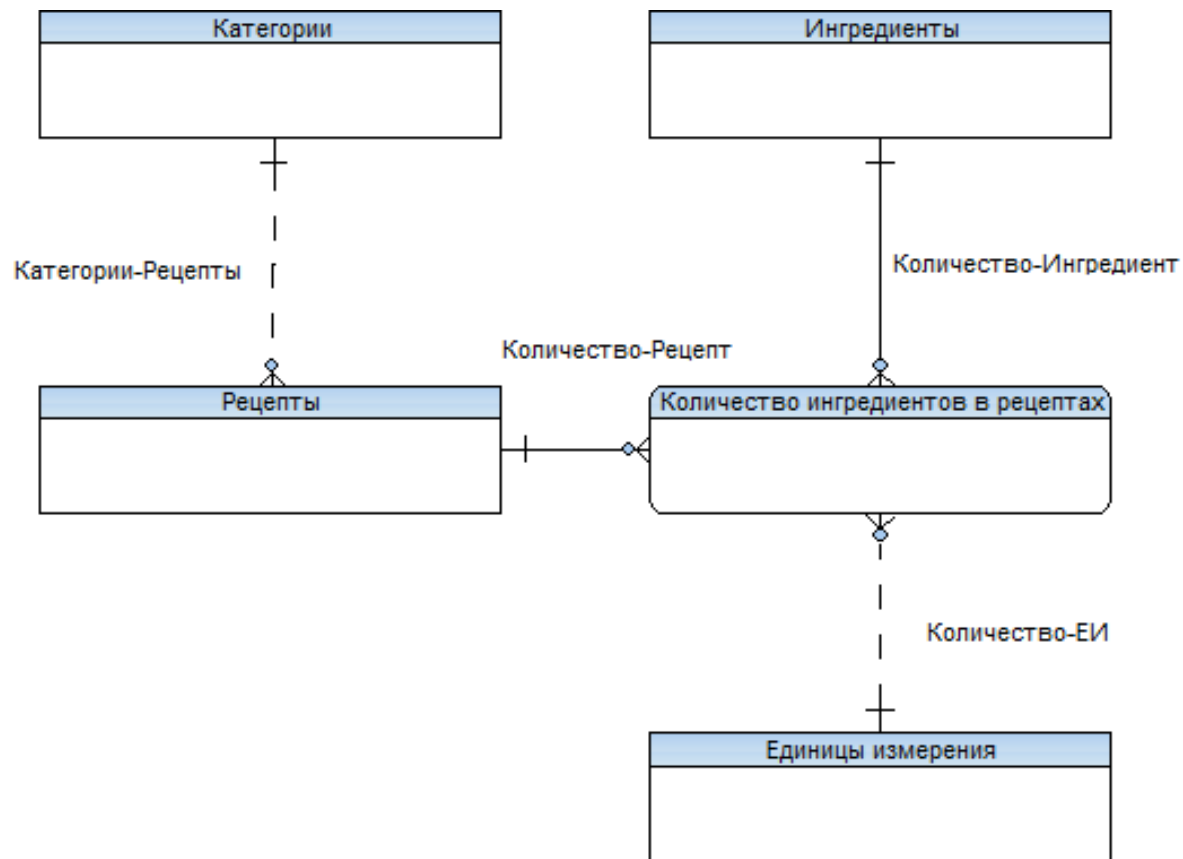
Information Engineering, Crow's Foot



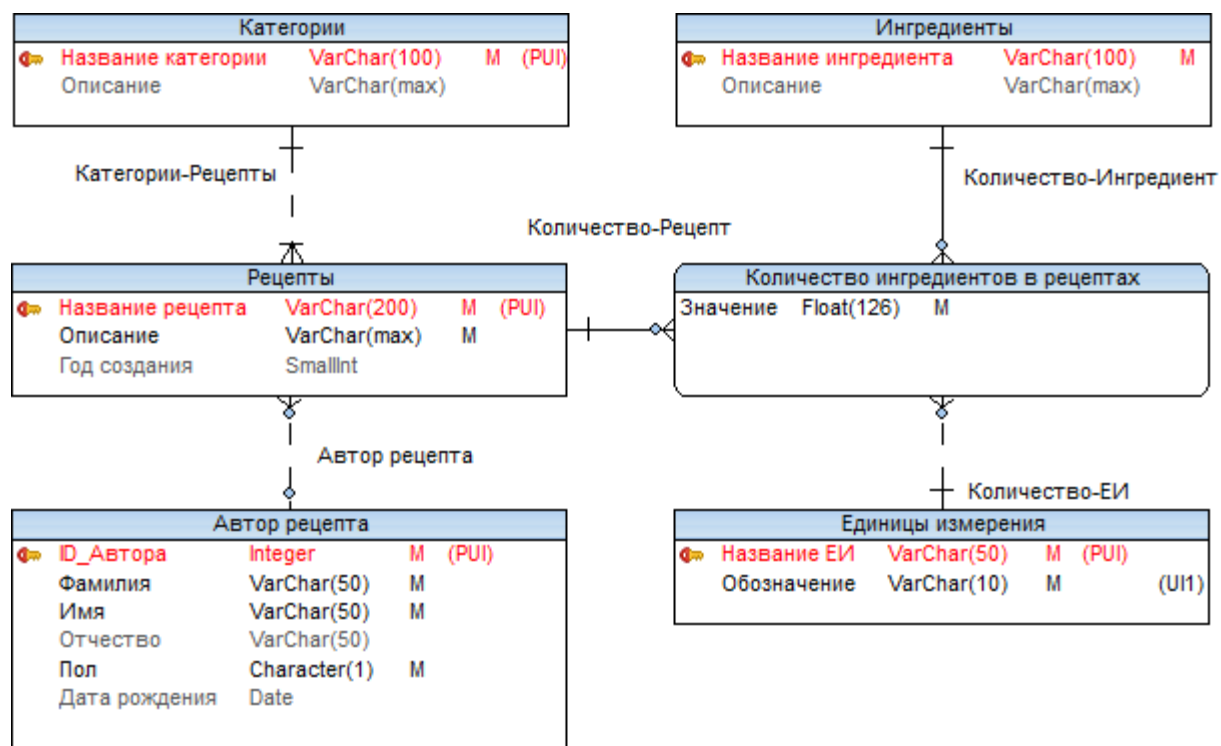
IDEF1X



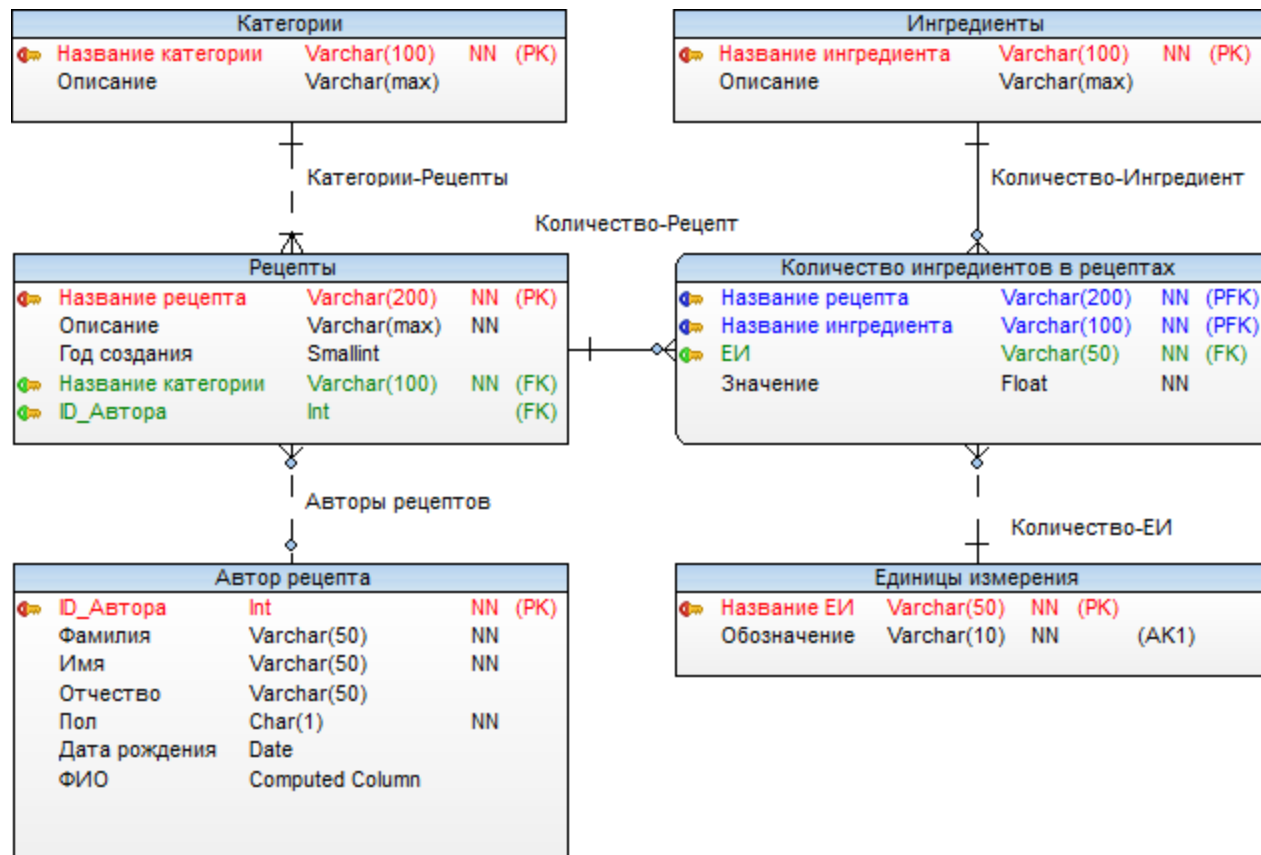
Пример без атрибутов



Пример с атрибутами



Физическая модель



Типы данных (примерно)

- Логический
- Текстовые
 - строка фиксированной длины
 - строка максимальной длины
 - строка неограниченной длины (character large object, CLOB)
- Численные
 - Целое число
 - Действительное число
 - без ограничения количества знаков до и после запятой
 - с ограничением количества знаков до и после запятой

Типы данных (примерно)

- Дата/время
 - дата с временем
 - дата без времени
 - время без даты
 - интервал
- Бинарные
 - набор байт фиксированной длины
 - набор байт максимальной длины
 - набор байт неограниченной длины (binary large object, BLOB)

- Специальные
 - GUID
 - XML
 - геометрический объект
 - географический объект
 - variant

Точные числа

Тип	Описание	Размер
bit	0, 1	1 байт
tinyint	От 0 до 255	1 байт
smallint	От -32 768 до 32 767	2 байта
int	От -2 147 483 648 до 2 147 483 647	4 байта
bigint	До 9 223 372 036 854 775 80	8 байт
decimal, numeric	decimal (точность, масштаб). Точность до 38	5–17 байт
smallmoney	От -214 748,3648 до 214 748,3647	4 байта
money	До 922 337 203 685 477,5807	8 байт

Приблизительные числа

Тип	Описание	Размер
real	=float(24). Точность — 7 цифр	4 байта
float [(N)]	N — количество битов, используемых для хранения мантиссы числа, от 1 до 53 Точность — 15 цифр	4–8 байт

Строковые типы:

- char(N)
- varchar(N)
- text = varchar(max)
- nchar(N)
- nvarchar(N)
- ntext = nvarchar(max)

Бинарные данные:

- `binary(N)`
- `varbinary(N)`
- `image = varbinary(max)`

Дата и время:

- `date`, от 1 января 1 года до 31 декабря 9999 года
- `datetime2` [(число знаков для точности секунд)], с 1 января 1 года нашей эры до 31 декабря 9999 года нашей эры
- `time`, от 00:00:00.0000000 до 23:59:59.999999
- `smalldatetime`, дата от 01.01.1900 до 06.06.2079, время без долей секунд
- `datetime`, дата с 1 января 1753 года до 31 декабря 9999 года, время до миллисекунд
- `datetimeoffset`, дата/время с учётом часового пояса в 24-часовом формате