Основы

Стержневая сущность – сущность, не зависящая от других сущностей. Внутри себя содержит необходимое кол-во атрибутов, но может также подключать к себе характеристическую сущность и ассоциативную сущность.

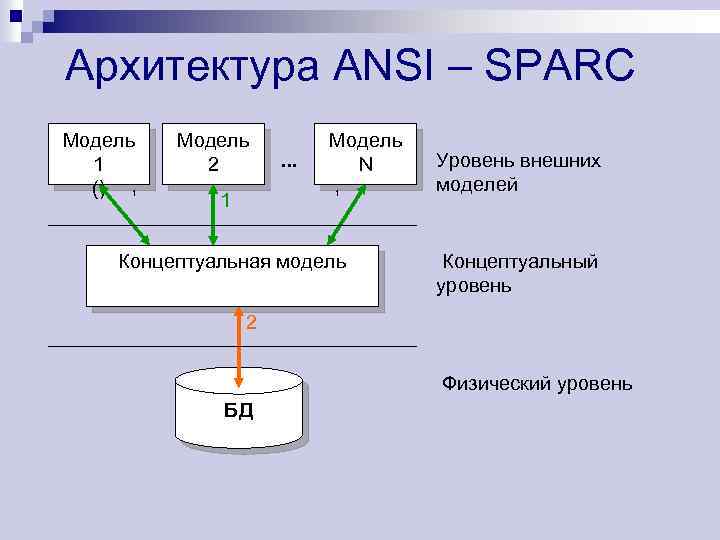
Характеристическая сущность – связь вида «многие-к-одной» или «одна-к-одной». Вы должны использовать характеристические сущности там, где экземпляров сущности имеет смысл хранить различные наборы атрибутов.

Ассоциативная сущность – содержат первичные ключи 2 или более других сущностей. Ассоциативные сущности всегда зависимы. Они используются для разрешения отношений «многие-ко-многим».

Инфологическая модель – ориентированная на человека и не зависимая от типа СУБД модель предметной области, определяющая совокупности информационных объектов

Даталогическая модель – модель, ориентированная на реализацию БД в конкретной СУБД, т. е. инфологическая модель, трансформированная с учетом требований и ограничений конкретной СУБД.

# 1.Архитектура ANSI-SPARC



Архитектура ANSI-SPARC – принципы, согласно которым рекомендуется строить системы управления базами данных (СУБД).

Согласно принципам, выделяется 3 уровня системы:

1. внешний (пользовательский)

2. промежуточный (концептуальный)

3. внутренний (физический)

• Определяет принцип организации СУБД (необязательно РМД)

• Цель: введение уровней абстракции для пользователей и архитекторов БД

# 2. Модель «Сущность-Связь». Классификация сущностей

Один из вариантов для построения инфологической модели: **ER-диаграммы**

(Entity-Relationship – **модель сущность-связь**).

Данная модель состоит из:

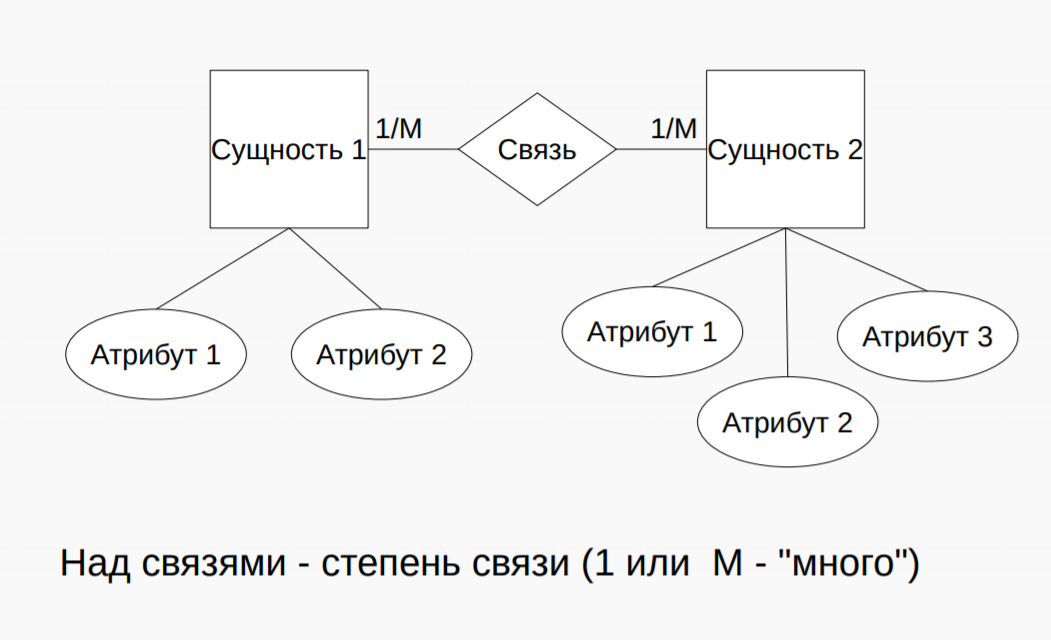
1. **Сущность** – класс объектов, фактов, явлений, предметов, элементы которых будут храниться в базе данных.

2. **Экземпляр сущности** относится к конкретной вещи в наборе. Например, типом сущности (сущностью) может быть СТУДЕНТ , а экземпляром — Иван Иванов и т. д.

3. **Атрибут** – важная характеристика (свойство) сущности, которой присваивается имя.

4. **Связь** – ассоциирование двух или более сущностей, выражающая форму взаимодействия между ними.

Сущность изображается прямоугольником, атрибут – овалом, связь – ромбом.



Помимо визуализации Чена существует еще и другие способы визуализации, например, перевернутая лапка или обратная стрелка. Ну и вы сами можете придумать свой способ визуализации.

Главное преимущество этой модели в том, что она не привязана ни к конкретной СУБД, ни даже к базам данных, она может применяться и в других областях, но просто наиболее частое применение у нее именно в БД.

Для всех дальнейших примеров будем рассматривать MySQL – реляционная БД, визуализация – перевернутая лапка.

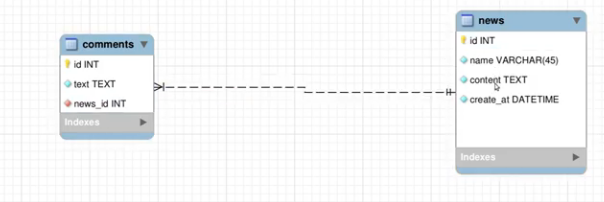
В модели сущность-связь очень важно, чтобы имена сущностей пересекались с именами, которые использует бизнес в своей терминологии. Это называется высокая связность. Сущностей может быть очень много, но связей между сущностями, условно существует всего 3.

Типы связей:

(!!! «1» здесь – это 1 или 0 !!!)

**1. 1 ко многим**

Одному экземпляру первой сущности соответствуют 0, 1 или несколько экземпляров второй сущности; Одному экземпляру второй сущности соответствует 0 или 1 экземпляров первой сущности;

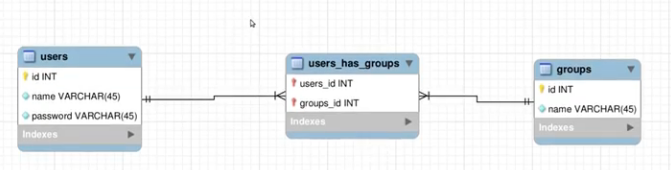


Допустим, есть новость, у которой есть название, какой-то текстовый контент и дата создания.

Есть комментарий с id, текстом комментария и внешним ключом на id новости.

С левой стороны указано много стрелочек. Это означает связь «М» – много. С правой стороны перечеркнуто 2-мя перпендикулярными чертами – это значит «один». Получается у *1 новости может быть много комментариев*.

**2. Многие ко многим**



1-я сущность – users. У каждого пользователя есть атрибуты в виде имени и пароля.

2-я сущность – groups – группы пользователей: администраторы, модераторы и т. д.

1 пользователь может одновременно находиться в нескольких группах.

Для связи «многие ко многим» используется вспомогательная таблица (вспомогательная сущность) – users\_has\_groups – она говорит в каких группах состоит пользователь или какие пользователи состоят в группе. Здесь есть 2 колонки – указатели на ключи пользователя и группы.

Здесь видно по лапкам, что у 1 пользователя может быть много групп и в 1 группе может быть много пользователей.

**3. 1 к 1**



Он встречается довольно редко.

Его применяют чаще всего для оптимизации, когда 1 огромная таблица и ее нужно разделить на несколько логических элементов.

Рассмотрим пример.

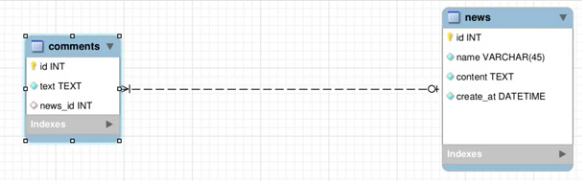
Сущность people – у каждого человека есть имя и номер паспорта.

Сущность passport.

*У 1 человека может быть только 1 паспорт, а у 1-го паспорта только 1 человек.*

На этом в теории модель сущность-связь заканчивается, но на самом деле при работе с реляционными БД существуют еще и другие разновидности связи. Вот эти 3 – это основные. Но каждая из них еще подразделяется на несколько.

Например, если мы посмотрим на первую связь «1 ко многим» между комментарием и новостями, если мы там в таблице comments поле news\_id уберем галочку NOT NULL, то само поле будет визуализировано по-другому и у связи появится пустой кругляшек (у news). Это значит, что значение news\_id может ссылаться на пустое значение (на пустой кругляшек).



Т. е.

1) любая связь, о которой говорили выше, может быть преобразована к тому, что появится пустой кругляшек.

2) Вы заметили, что какие-то связи нарисованы прямой линией, а какие-то пунктиром. Это тоже имеет значение.

Пунктирная линия рисуется в том случае, если, к примеру, колонка news\_id не является primary key. Если поставить галочку в news\_id на primary key, то на картинке выше линия станет сплошной. Это означает, что колонка news\_id является внешним ключом и в то же время она является primary key для данной таблицы. Связка id комментария и id новости должны быть уникальны. Здесь это смысла не имеет, но в других случаях может понадобиться.

Сущность — класс объектов, фактов, явлений, предметов, элементы которых будут храниться в базе данных.

Экземпляр сущности относится к конкретной вещи в наборе. Например, типом сущности (сущностью) может быть СТУДЕНТ , а экземпляром — Иван Иванов и т. д.

Атрибут — важная характеристика (свойство) сущности, которой присваивается имя.

Связь — ассоциирование двух или более сущностей, выражающая форму взаимодействия между ними.

Классификация сущностей

Стержневые

Ассоциативные

Характеристические

Сильные

Слабые

Типы

Подтипы

# 3. DDL

Data Definition Language (язык описания данных) – это семейство компьютерных языков, используемых в компьютерных языков, используемых в компьютерных программах для описания структуры баз данных. Наиболее популярным языком DDL является **SQL**, используемый для получения и манипулирования данными РСУБД, и сочетающий в себе элементы DDL, DML и DCL.

Функции языков DDL определяются первым словом в предложении (запрос), которое почти всегда является глаголом.

• CREATE создает объект БД

• ALTER изменяет объект

• DROP удаляет объект

• TRUNCATE

• COMMENT

• RENAME

# 4. DML

Data Manipulation Language – занимается манипулированием данными и включает в себя наиболее распространенные операторы SQL. Он используется для хранения, изменения, извлечения, удаления и обновления данных в БД.

• SELECT

• INSERT

• UPDATE

• DELETE

• MERGE