



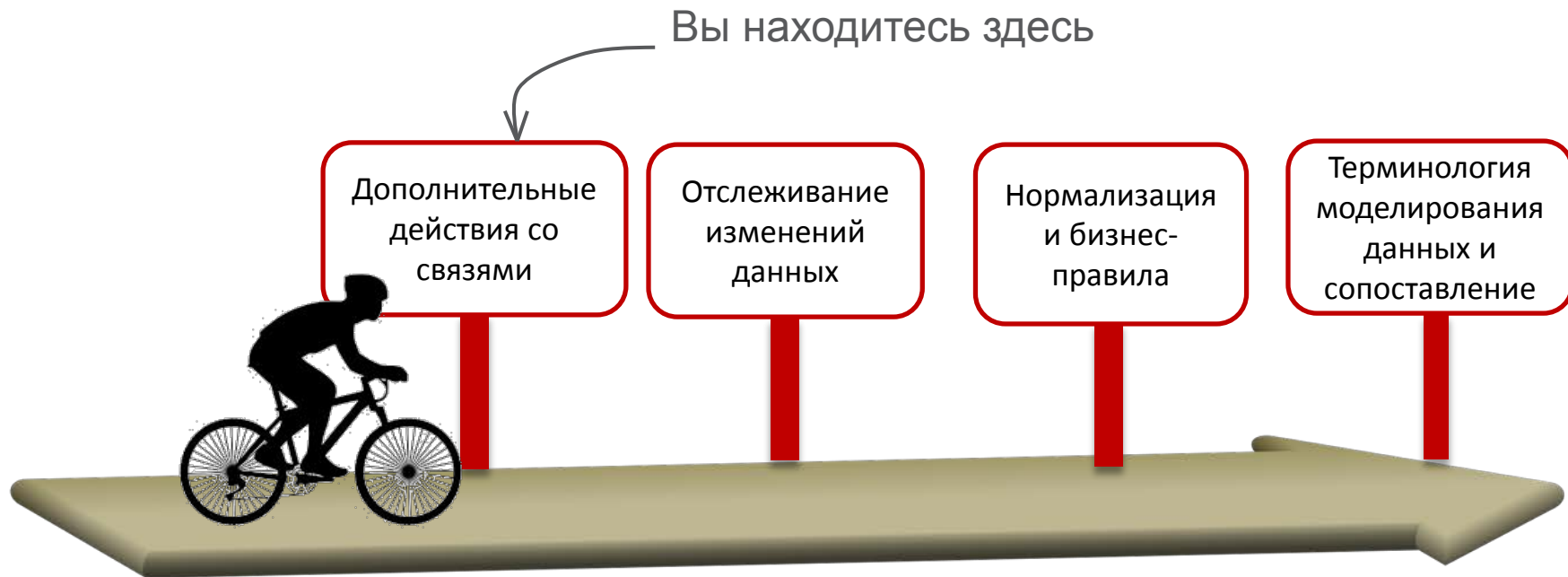
# Принципы баз данных

3-1

Дополнительные действия со связями



# Дорожная карта



# Цели

Цели этого урока:

- Определение ассоциативных связей
- Решение связей типа M:M
- Определение и примеры непередаваемых связей
- Определение и построение объектов с супертипом и подтипом
- Определение иерархических, рекурсивных и дуговых связей

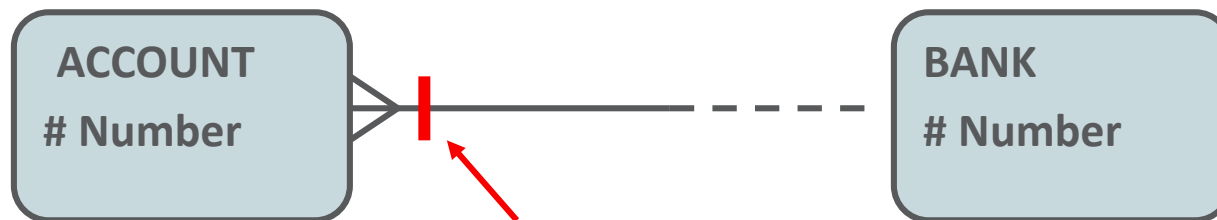


# Определяющие (ассоциативные) связи

- Какой UID у объекта ACCOUNT?



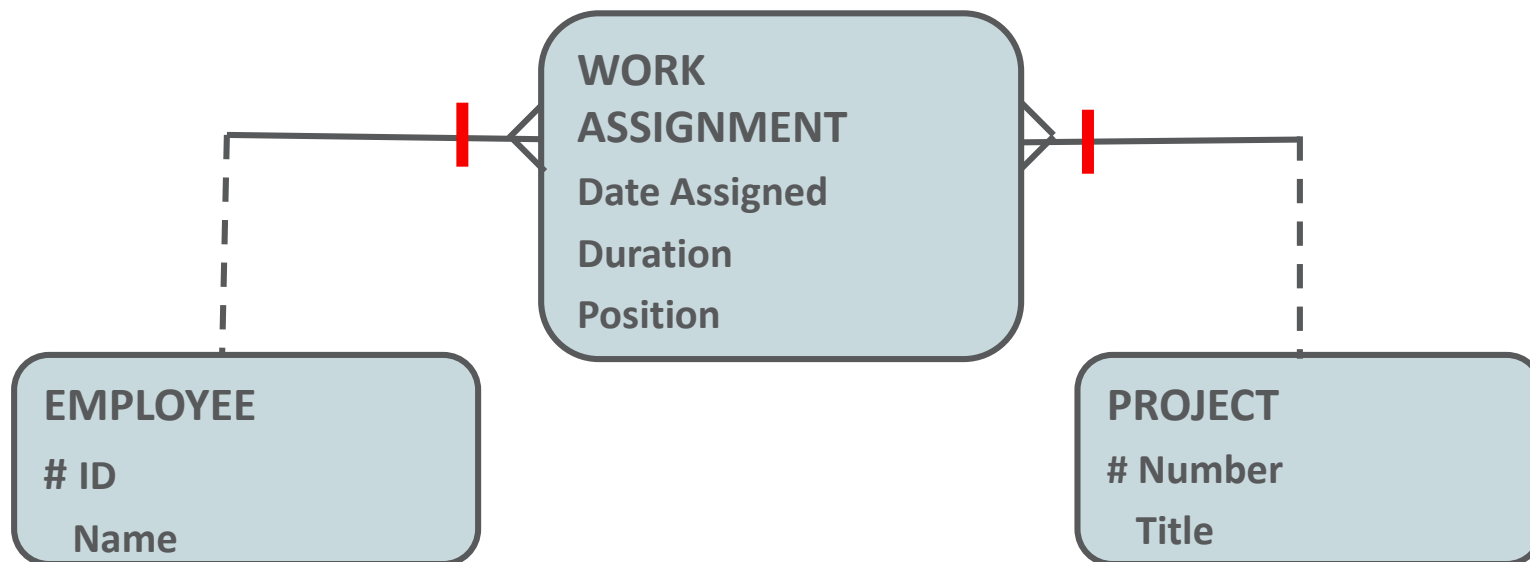
- Для UID требуются атрибут ACCOUNT Number и связь между объектами BANK и ACCOUNT.



- UID для объекта ACCOUNT состоит из атрибута ACCOUNT Number, а также атрибута BANK Number, который представлен полосой на линии связи.

# Определение связей с несколькими объектами

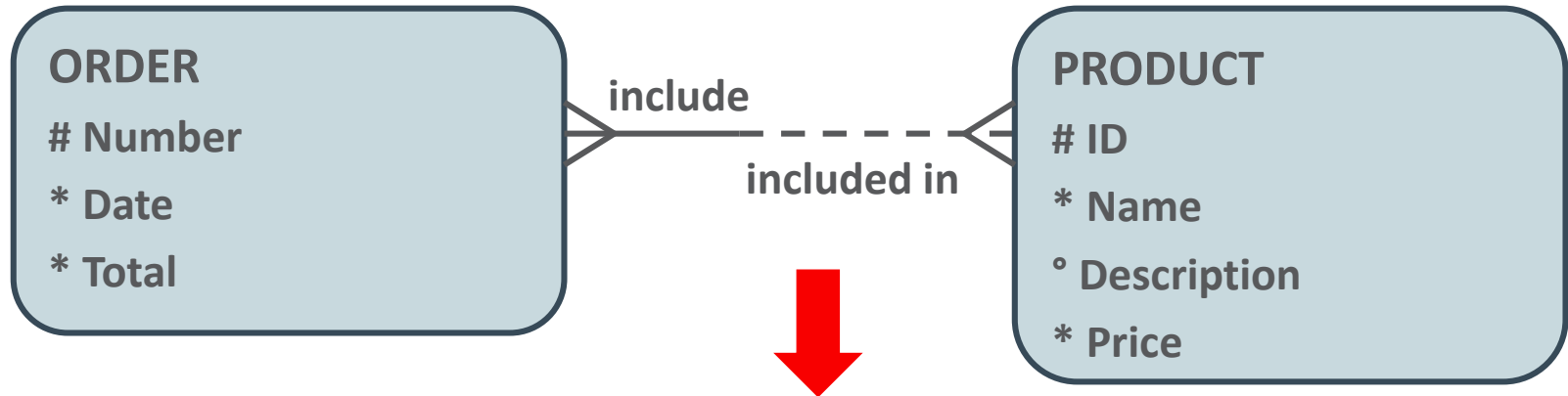
- Объект можно определить уникальным образом через несколько связей.



- UID для объекта WORK ASSIGNMENT — EMPLOYEE ID и PROJECT Number (представлены полосами ассоциаций).

# Связи типа M:M

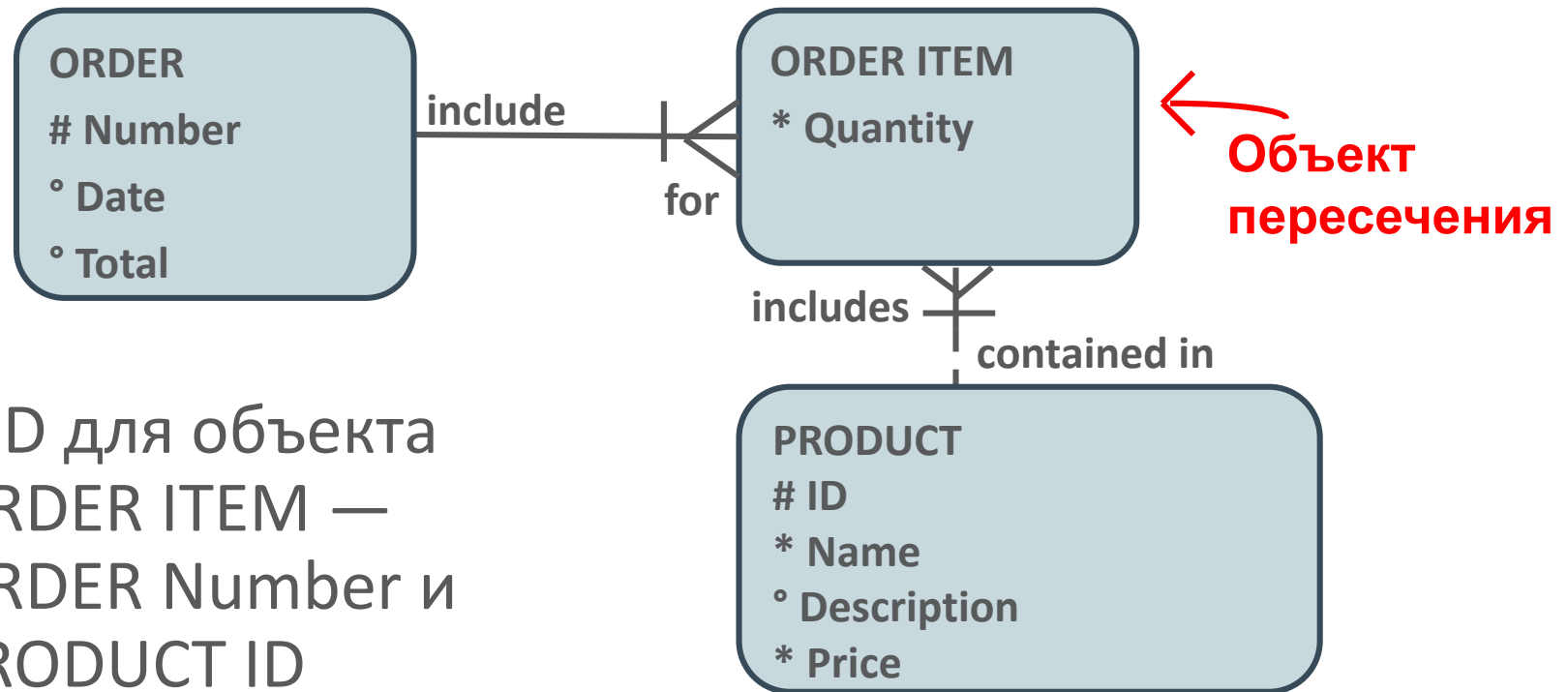
- Атрибуты описывают только объекты.
- Если атрибуты описывают связь, значит, эту связь необходимо решить.



Куда добавить атрибут Quantity?

# Решение связей M:M: Пример 1

- Решение связи M:M с новым объектом пересечения и двумя ассоциативными связями 1:M.



- UID для объекта **ORDER ITEM** — **ORDER Number** и **PRODUCT ID**



# Характеристики объекта пересечения:

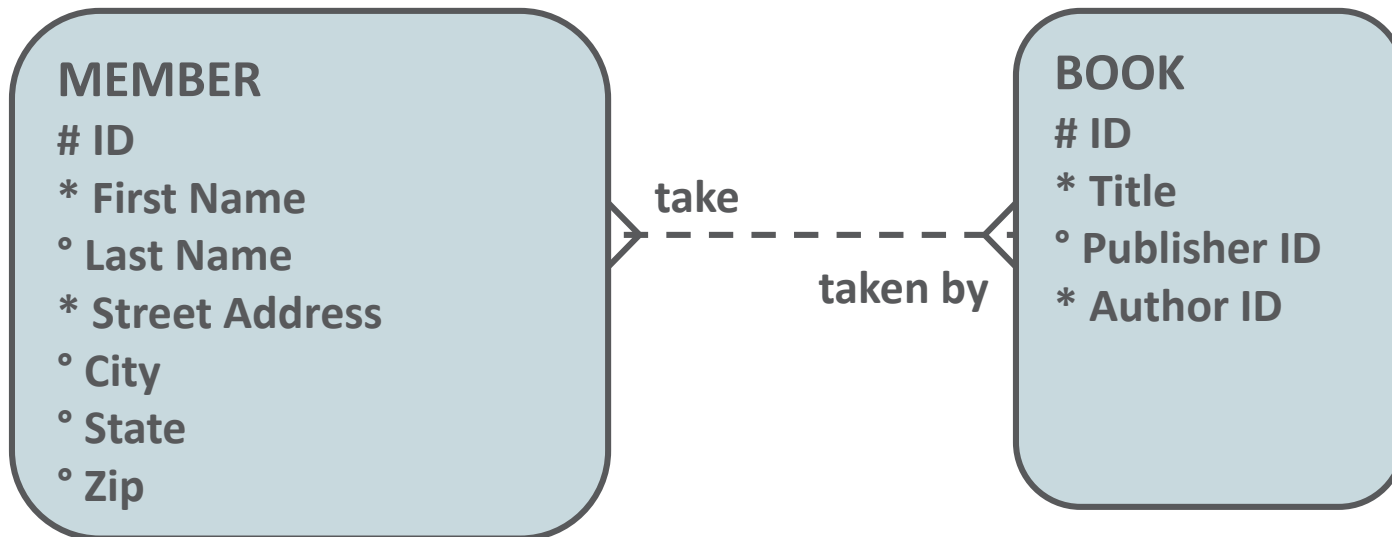
- Связи из объекта пересечения всегда обязательные.
- Объекты пересечения обычно содержат расходные материалы, например использованное количество и даты. Обычно это временные объекты большого объема.
- Объект пересечения определяется по своим двум исходным связям (определяющим связям).

# Практический сценарий: Решение связей М:М



Мэтт, как решить следующую связь между объектами MEMBER и BOOK?

Преподаватель

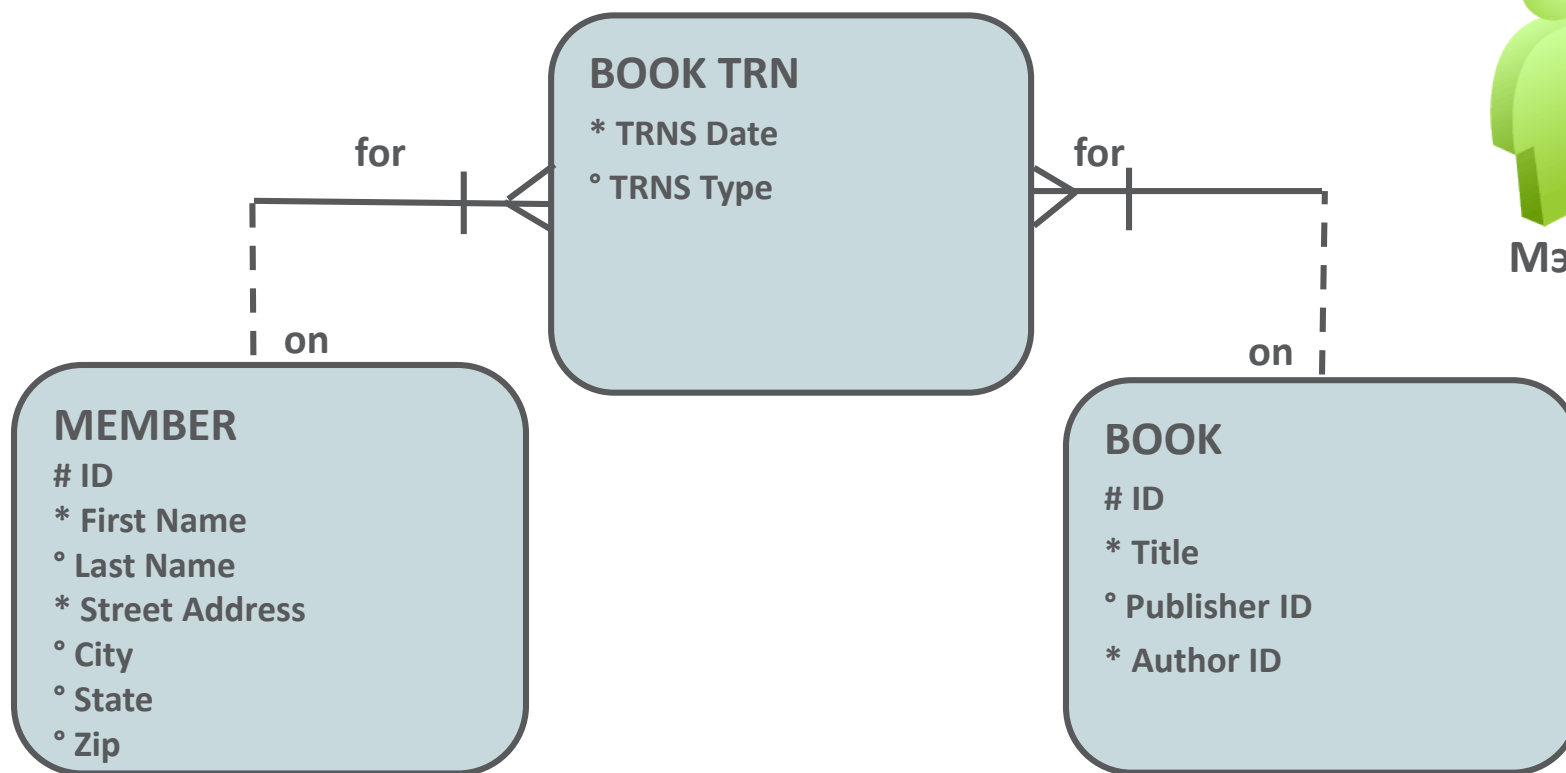


# Практический сценарий: Создание объекта пересечения

Я создам объект пересечения с определяющими связями с исходными объектами.

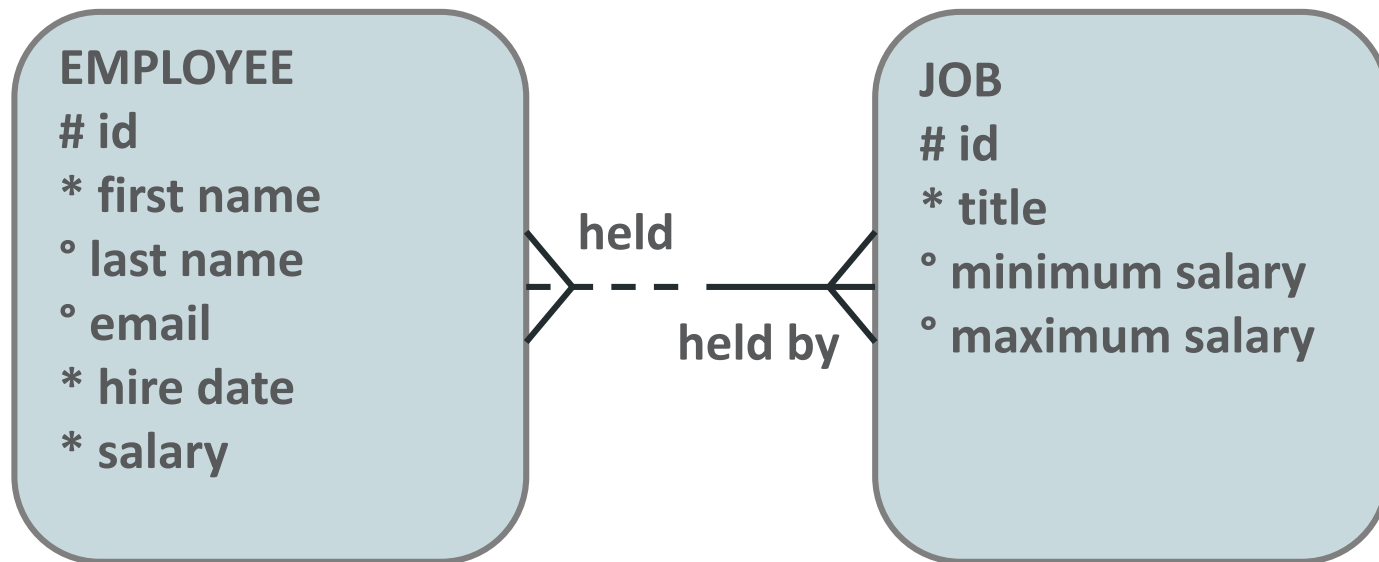


Мэтт

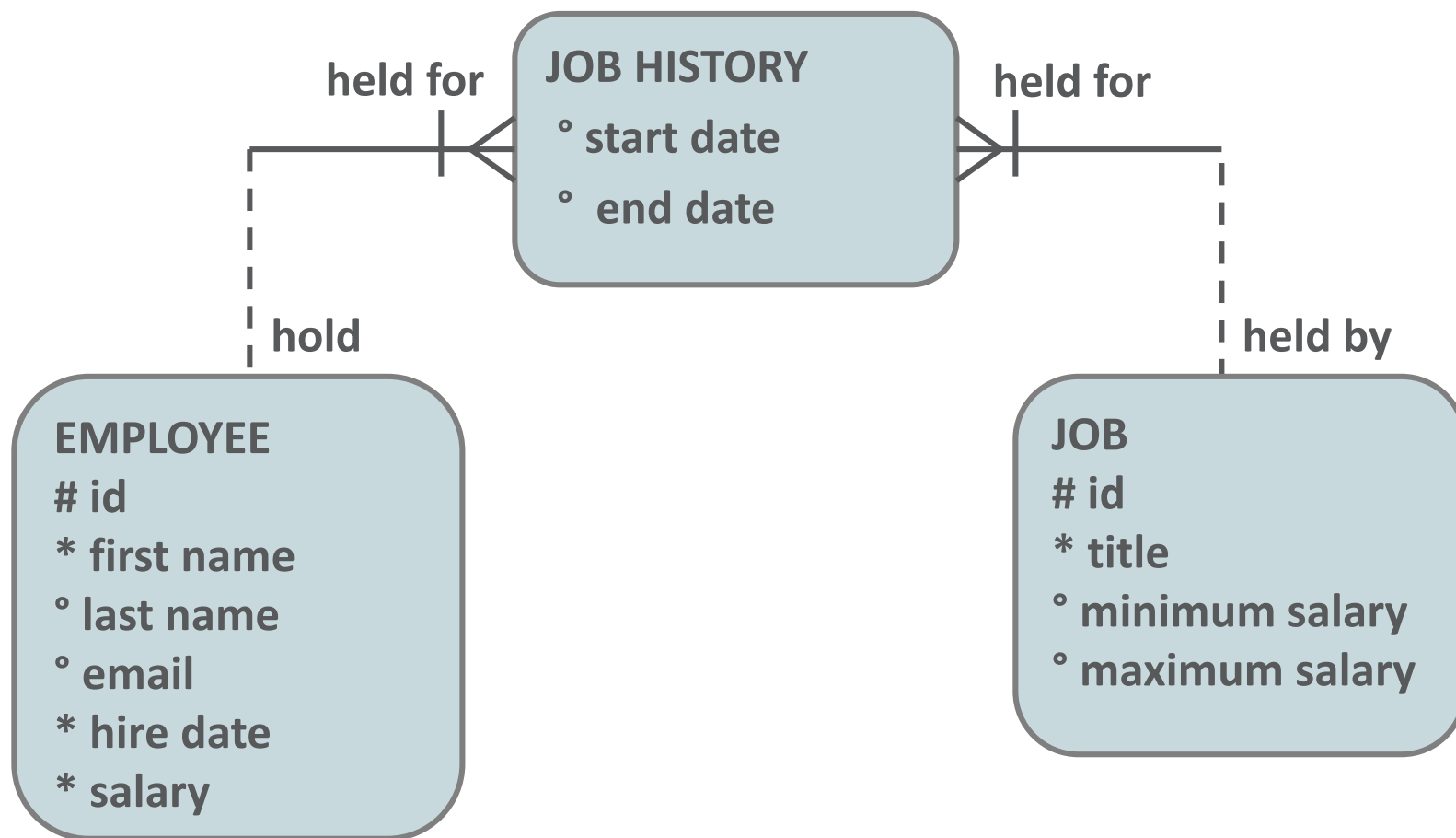


# Решение связей М:М: Пример 2

- Объекты EMPLOYEE и JOB не хранят историю работ сотрудника.



# Решение связей М:М с помощью объекта пересечения с ассоциативными связями



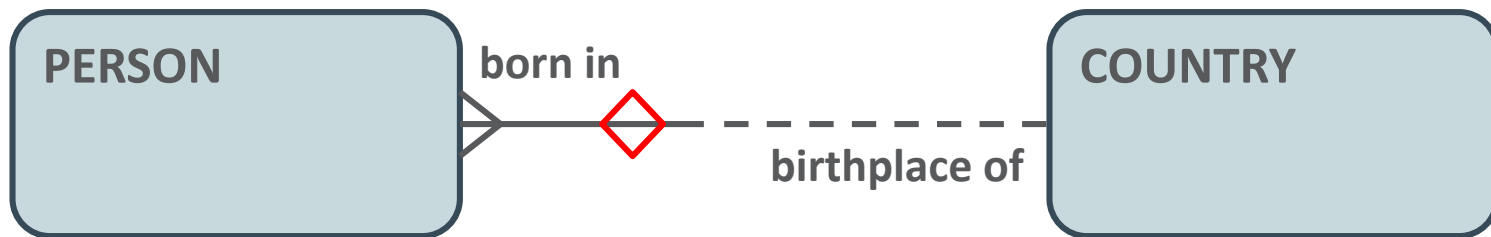
# Упражнение по проекту 1

## DFo\_3\_1\_1\_Project

- База данных магазина Oracle Baseball League решает связи «многие ко многим»

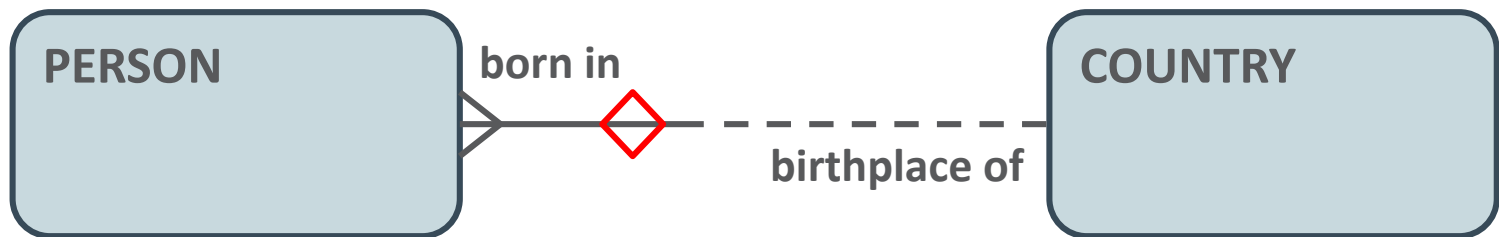
# Неперемещаемые связи

- Перемещаемость — это возможность связи между двумя экземплярами объекта меняться со временем.
- Неperемещаемую связь невозможно переместить между объектами, которые она соединяет.
- Неperемещаемая связь обозначается ромбом.



# Неперемещаемые связи

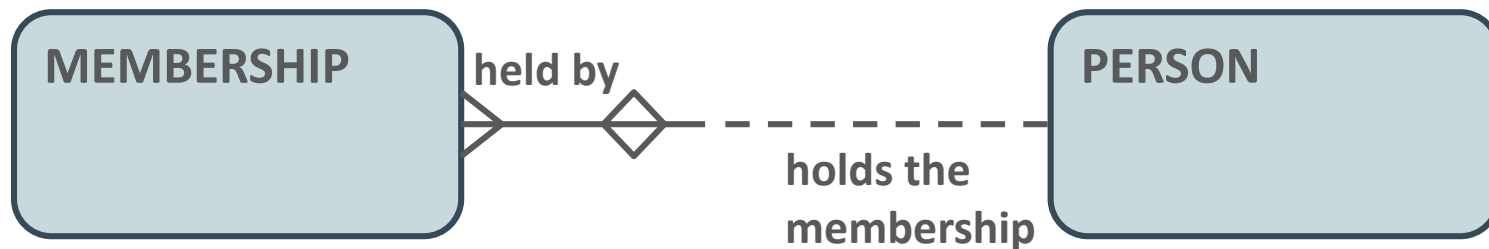
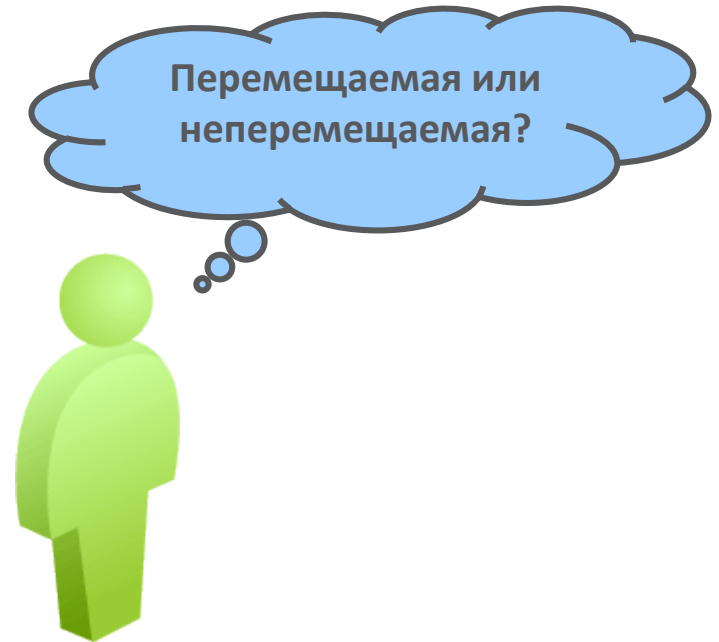
- Неperемещаемые связи могут быть только обязательными.
- Например, страна рождения для человека представляет собой неперемещаемую связь.





# Практический сценарий

- Связь membership должна принадлежать только одному объекту person.
- Связь membership нельзя переместить для другого объекта person.



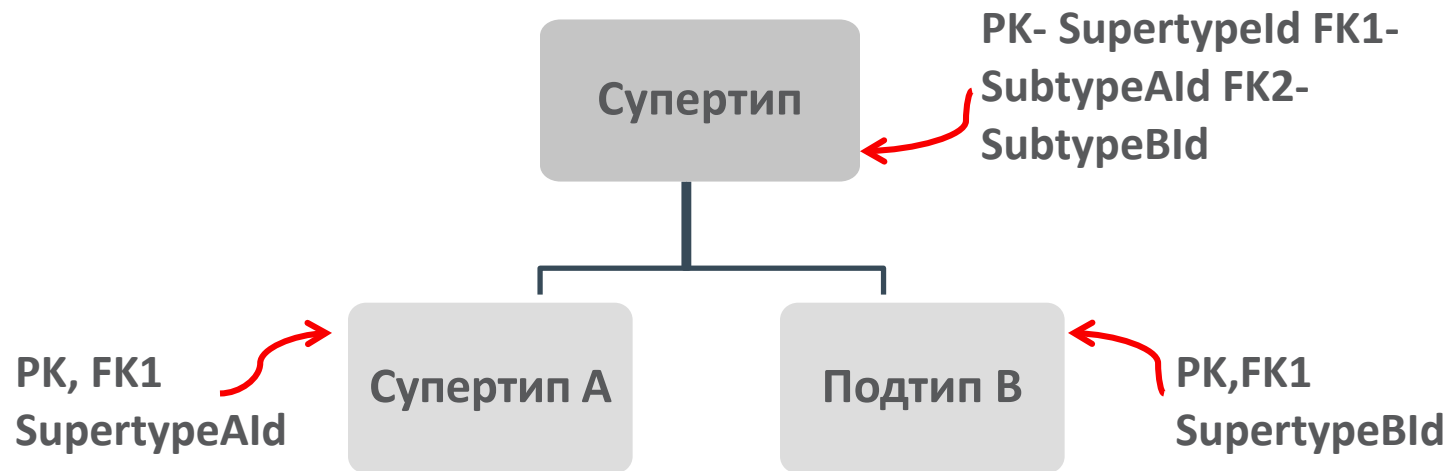
# Упражнение по проекту 2

## DFo\_3\_1\_2\_Project

- База данных магазина Oracle Baseball League определяет и иллюстрирует неперемещаемые связи

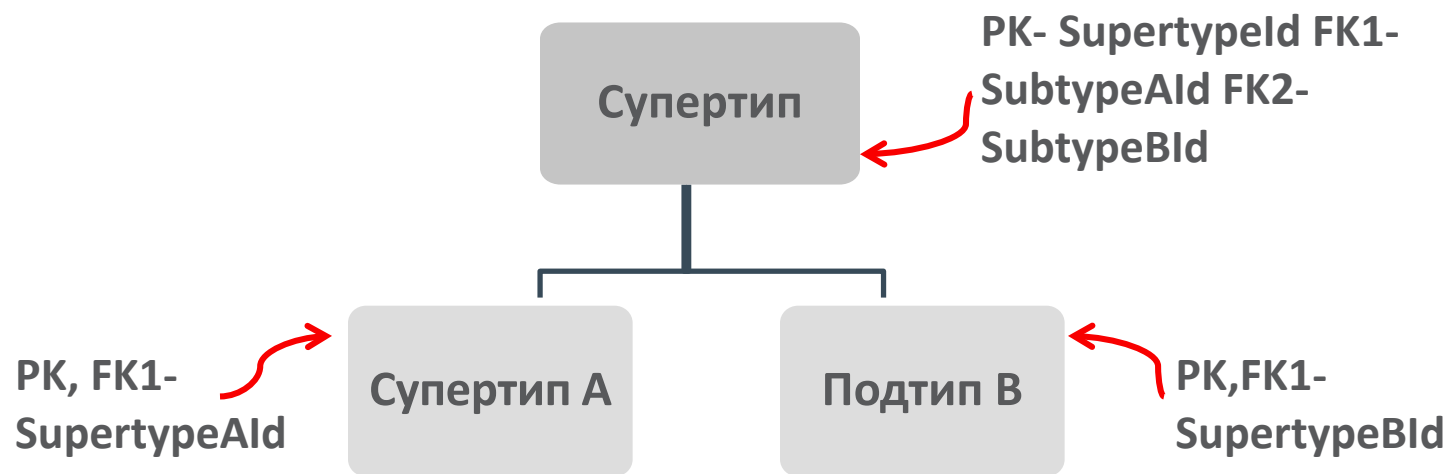
# Объекты-супертипы и объекты-подтипы

- Супертип имеет связь «родитель-потомок» с одним или несколькими подтипами.



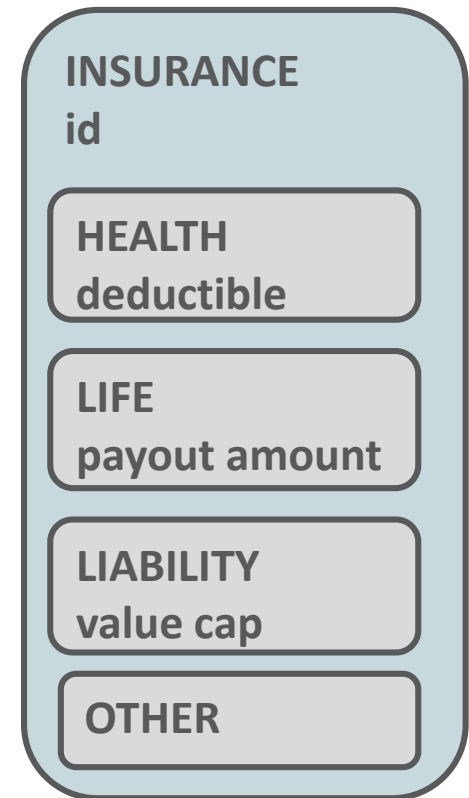
# Объекты-супертипы и объекты-подтипы

- Подтип — это подгруппировка объекта в типе объекта, имеющая атрибуты, которые отличаются от атрибутов в других подгруппировках.



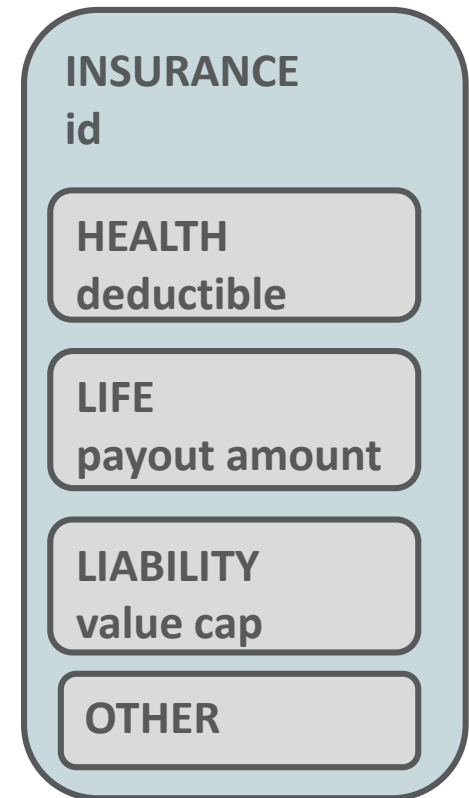
# Схематическое представление подтипа

- Каждый подтип является конкретизацией супертипа, и поэтому должен быть заключен внутри объекта.
- Общие атрибуты и связи для всех подтипов должны быть перечислены только в супертипе, но они наследуются в каждом подтипе.



# Схематическое представление подтипа

- Подтип может и, как правило, имеет собственные атрибуты и связи.
- Не может быть только одного подтипа; необходимо создать другой подтип, содержащий все остальное.



# Характеристики подтипа

## Подтип:

- Наследует все атрибуты супертипа.
- Наследует все связи супертипа.
- Обычно имеет собственные атрибуты или связи.
- Извлекается внутри супертипа.
- Никогда не существует один.
- Может иметь собственные подтипы.
- Имеет одинаковые первичные ключи супертипа и подтипа.

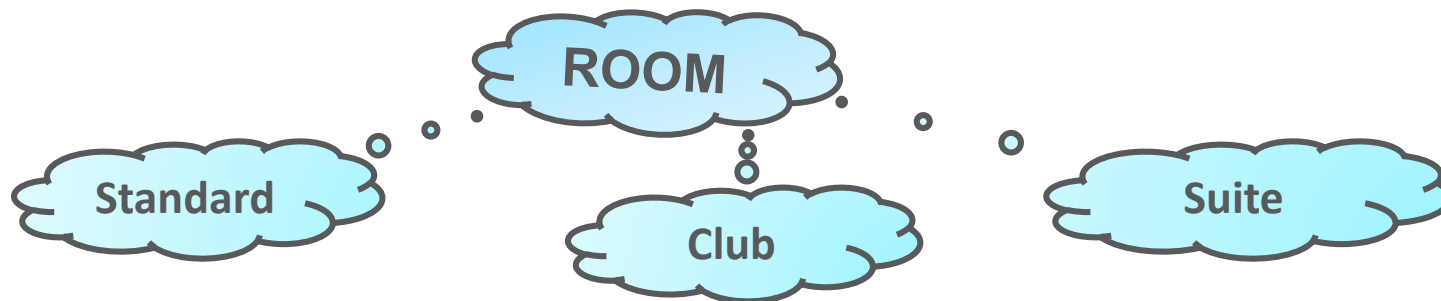
# Объекты-супертипы и объекты-подтипы: Пример



Шэрон хочет добавить типы категорий номеров для бронирования. Например:

- Standard
- Club
- Suite

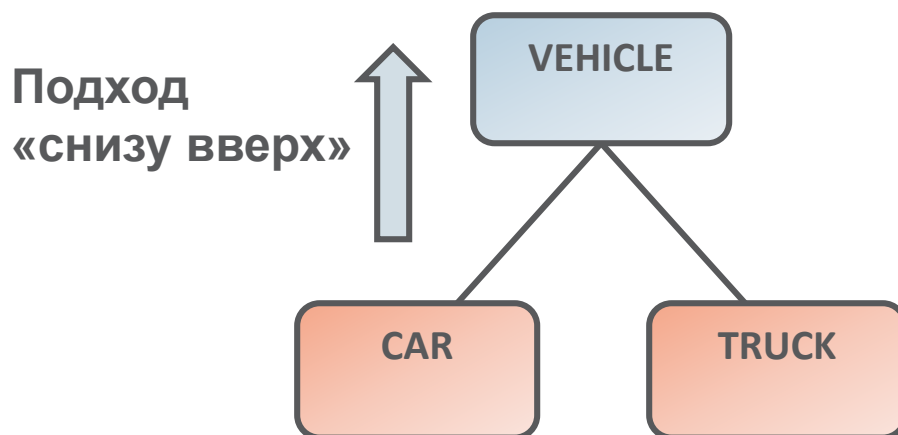
У нее уже есть объект ROOM со сведениями о номере. Это должен быть объект-супертип. Конкретные категории номеров унаследуют свойства объекта-супертипа в дополнение к собственным атрибутам. Конкретная категория номера будет объектом-подтипом.





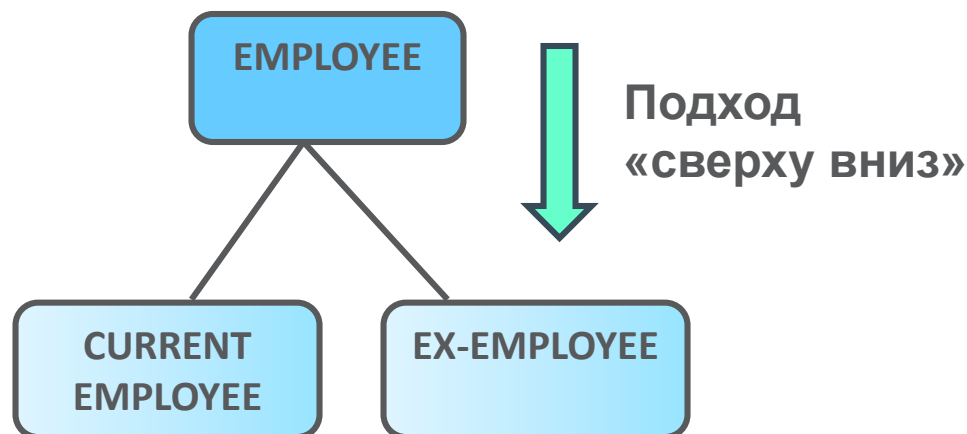
# Обобщение и конкретизация

- Обобщение — это подход «снизу вверх», при котором два или больше объектов нижнего уровня объединяются для формирования объекта более высокого уровня на основе общих черт.



# Обобщение и конкретизация

- Конкретизация — это подход «сверху вниз», при котором объект более высокого уровня разбивается на объекты нижнего уровня.



# Правила для объекта-подтипа

## Исчерпывающий:

- Все экземпляры супертипа также являются экземплярами одного из подтипов.
- Для категоризации экземпляров, которые не определяются одним из существующих подтипов необходимо добавить подтип OTHER.
- Пример: у сотрудника должна быть полная занятость, частичная занятость или другой тип занятости.

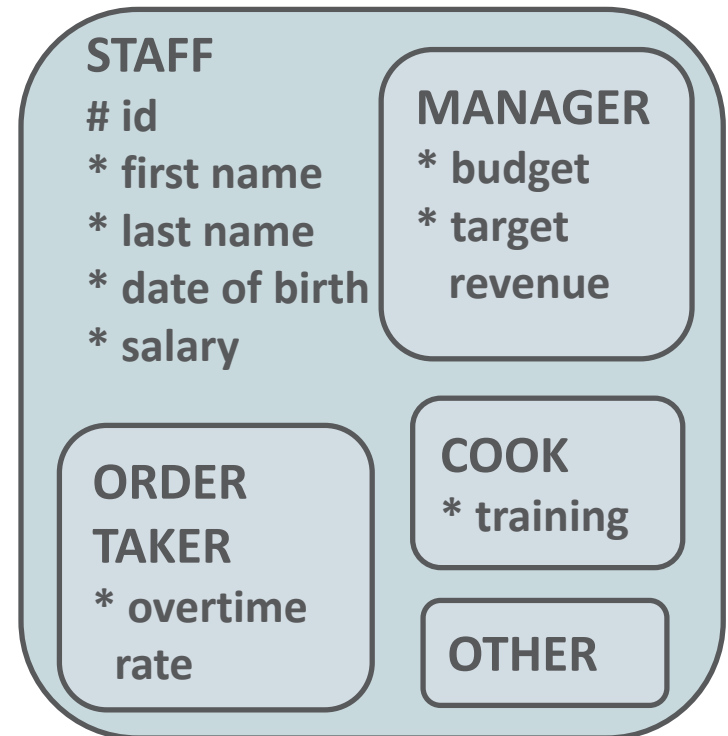
# Правила для объекта-подтипа

## Взаимоисключающий:

- Каждый экземпляр супертипа принадлежит только одному подтипу.
- Пример: у сотрудника не может быть одновременно и полная, и частичная занятость.

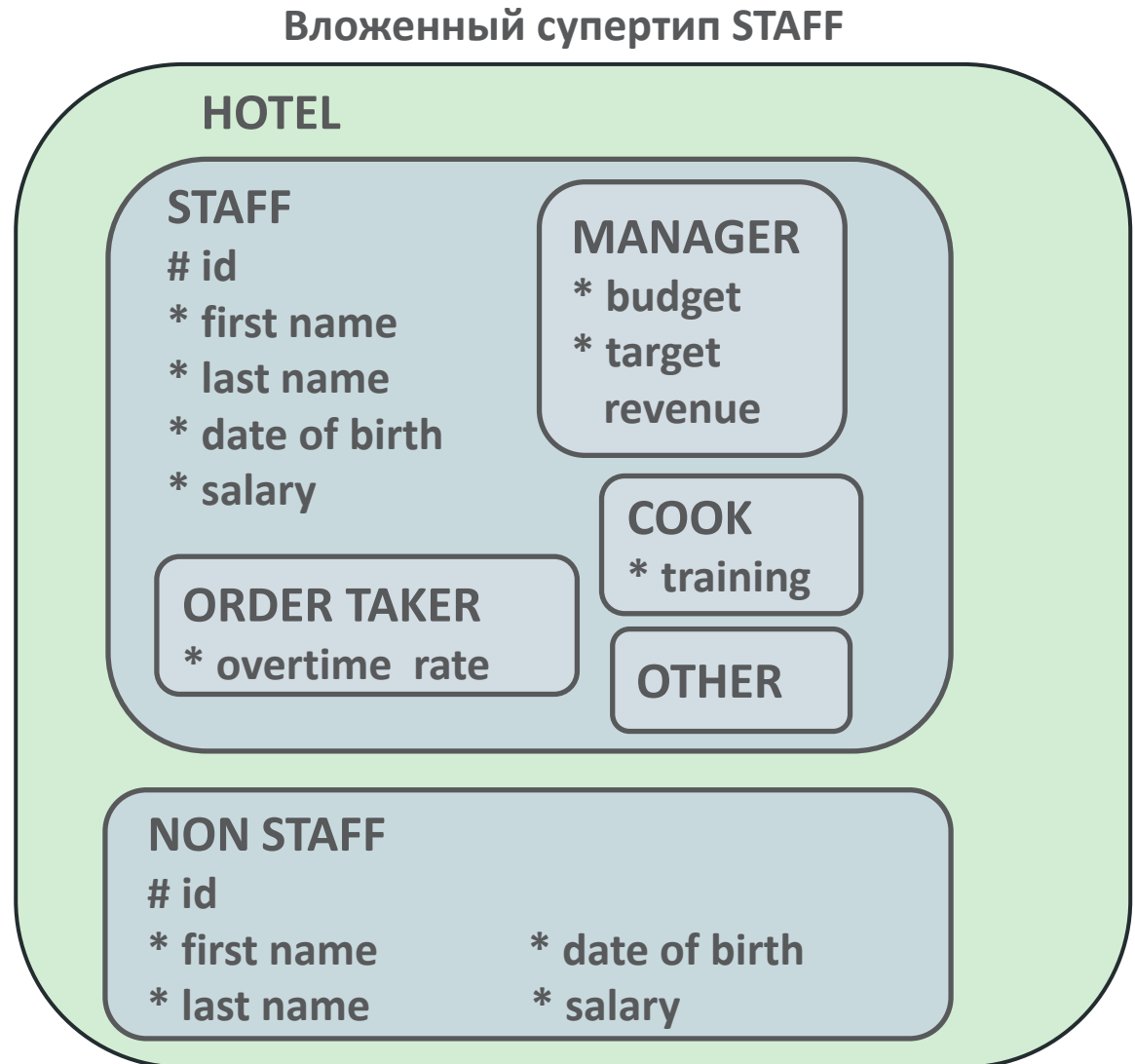
# Правильное определение подтипов

- Является ли этот подтип разновидностью супертипа?
- Учтены все возможные случаи? (исчерпывающий)
- Каждый экземпляр соответствует только одному подтипу? (взаимоисключающий)



# Вложенные подтипы

- Можно создавать вложенные подтипы.

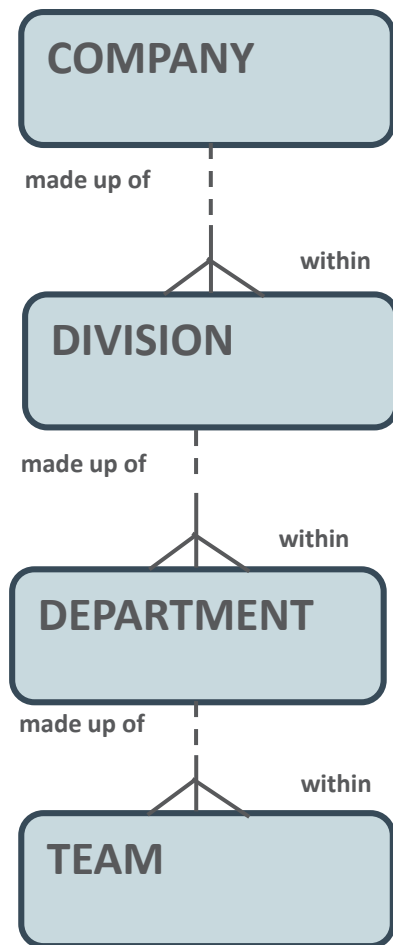


# Упражнение по проекту 3

## DFo\_3\_1\_3\_Project

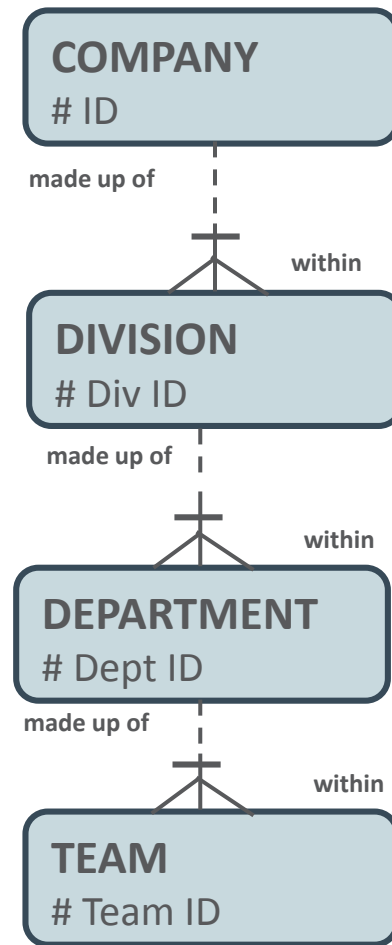
- База данных магазина Oracle Baseball League определяет и создает объекты супертипа и подтипа

# Моделирование иерархических данных



Представление иерархических данных в виде набора связей 1:M.

UID для набора иерархических объектов можно подставить через несколько ассоциативных связей.

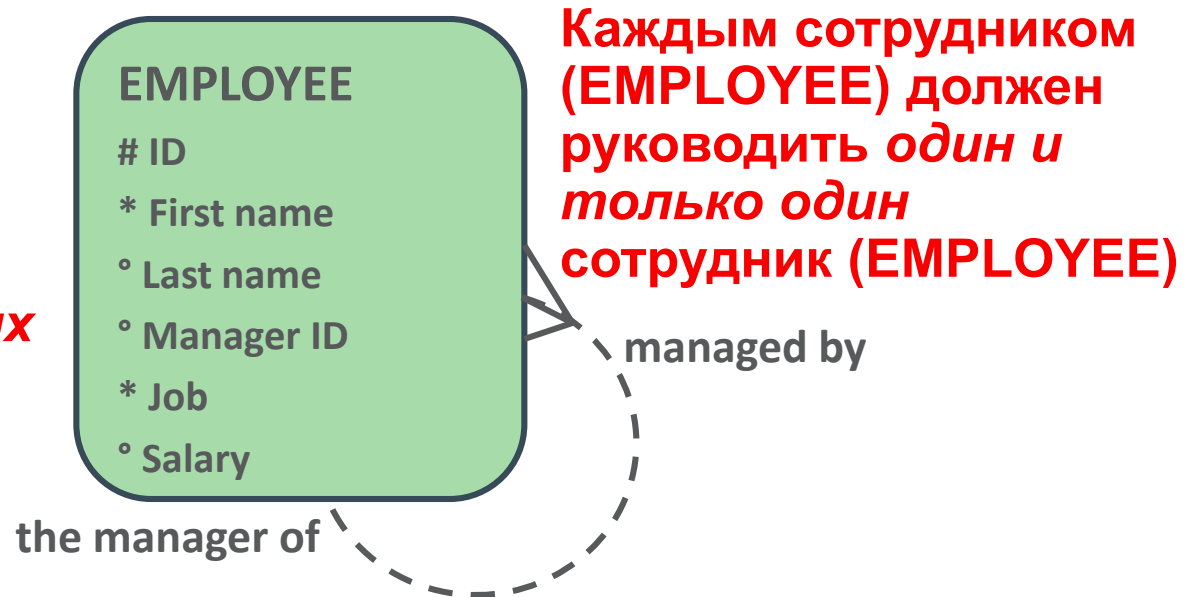




# Рекурсивные связи

- Рекурсивная связь — это связь, при которой экземпляр объекта связан с другим экземпляром в этом же объекте.

**Каждый сотрудник (EMPLOYEE) может быть руководителем одного или нескольких сотрудников (EMPLOYEE).**



# Рекурсивные связи

- Рекурсивная связь всегда моделируется циклом.

Каждый сотрудник (EMPLOYEE) может быть руководителем *одного или нескольких* сотрудников (EMPLOYEE).



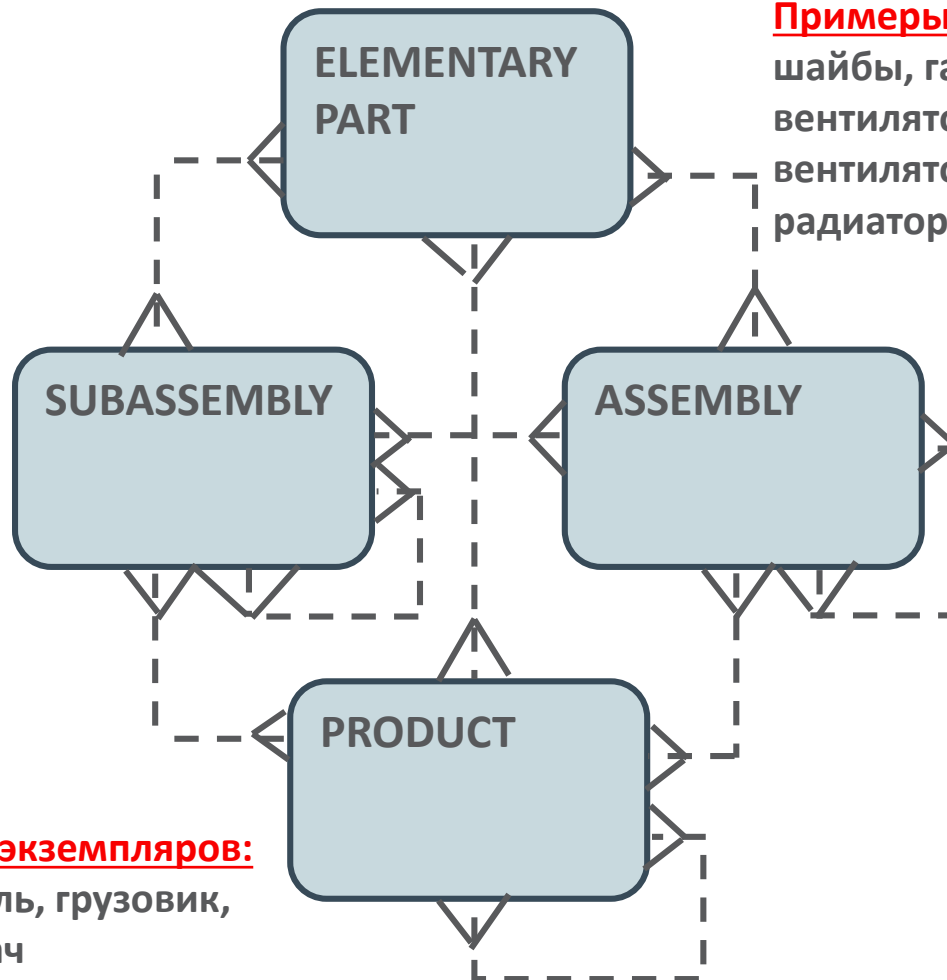
# Изучение рекурсивных связей

## Примеры экземпляров:

вентилятор,  
радиатор, модуль  
зажигания,  
карбюратор,  
автоматическая  
заслонка

## Примеры экземпляров:

автомобиль, грузовик,  
такси, тягач



## Примеры экземпляров:

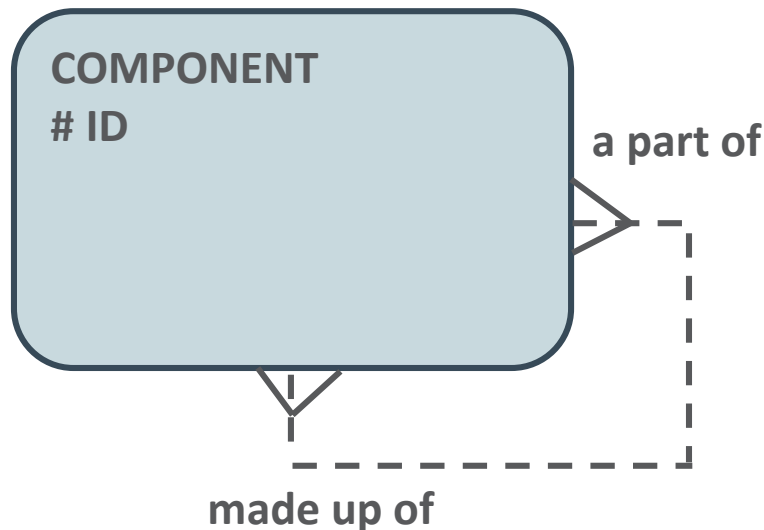
шайбы, гайки, винты, лопасти  
вентилятора, ремень  
вентилятора, пробка  
радиатора, шланги, термостат

## Примеры экземпляров:

система  
охлаждения,  
система  
зажигания,  
система подачи  
топлива,  
двигатель

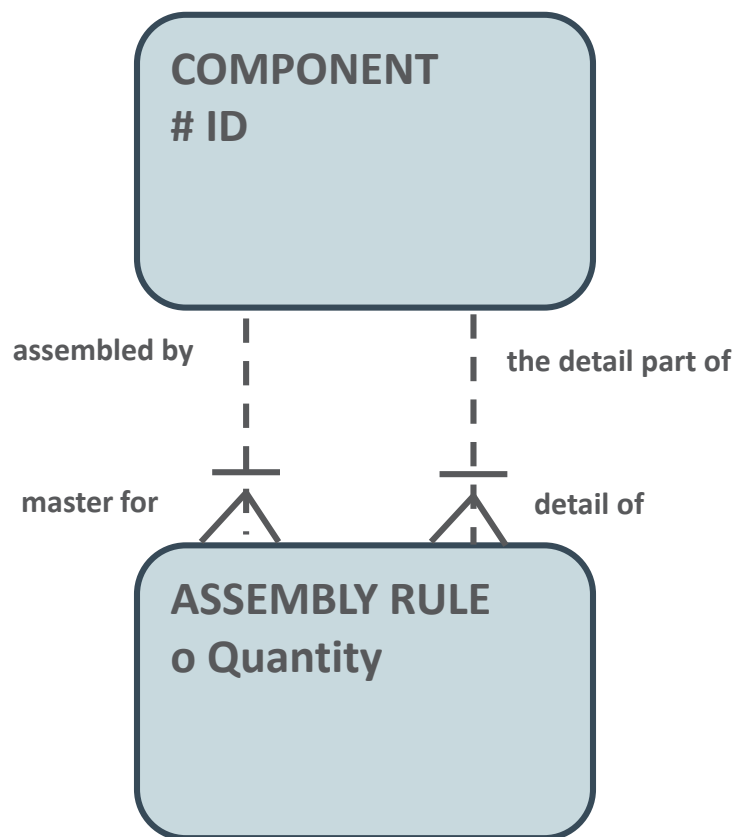
# Общее моделирование

- Еще один способ моделировать рекурсивную связь спецификации заключается в создании общего объекта COMPONENT.
- Это позволит создать рекурсивную связь М:М.



# Решение рекурсивной связи М:М

- Решите рекурсивную связь М:М с помощью объекта пересечения.



# Дуговые связи

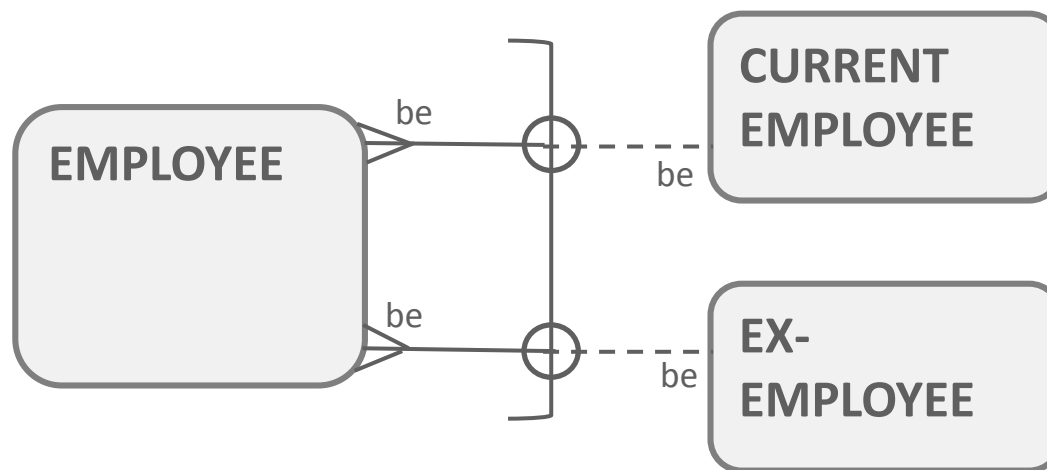
- Дугой обводится группа эксклюзивных связей — в ней может существовать только одна из связей для любого экземпляра объекта.
- Дуга, нарисованная между двумя связями, соединяет их и демонстрирует взаимоисключение.
- Такая связь предполагает наличие условия «или».
- Дуга означает, что любой экземпляр этого объекта в заданный момент времени может иметь только одну действительную связь внутри дуги.

# Характеристики дуговой связи

- Связи в дуге часто имеют одинаковое название.
- Связи в дуге должны быть либо все обязательные, либо все необязательные, и должны иметь одинаковую кардинальность.
- Дуга принадлежит одному объекту и должна содержать только те связи, которые исходят из него.
- Объект может иметь несколько дуг, но каждая связь может быть частью только одной дуги.
- Дуговая связь представлена дугообразной линией, соединяющей две или более линий связи.
  - Связи, включенные в дугу, обозначены кружком на линии дуги связи.

# Дуговые связи

- Объект-супертип и его подтипы можно моделировать как дуговую связь.
- Пример: объект EMPLOYEE является либо CURRENT EMPLOYEE, либо EX-EMPLOYEE, но не может быть и тем, и другим одновременно (это также можно смоделировать как супертип/подтип).





# Практический сценарий: Дуговые связи



Преподаватель

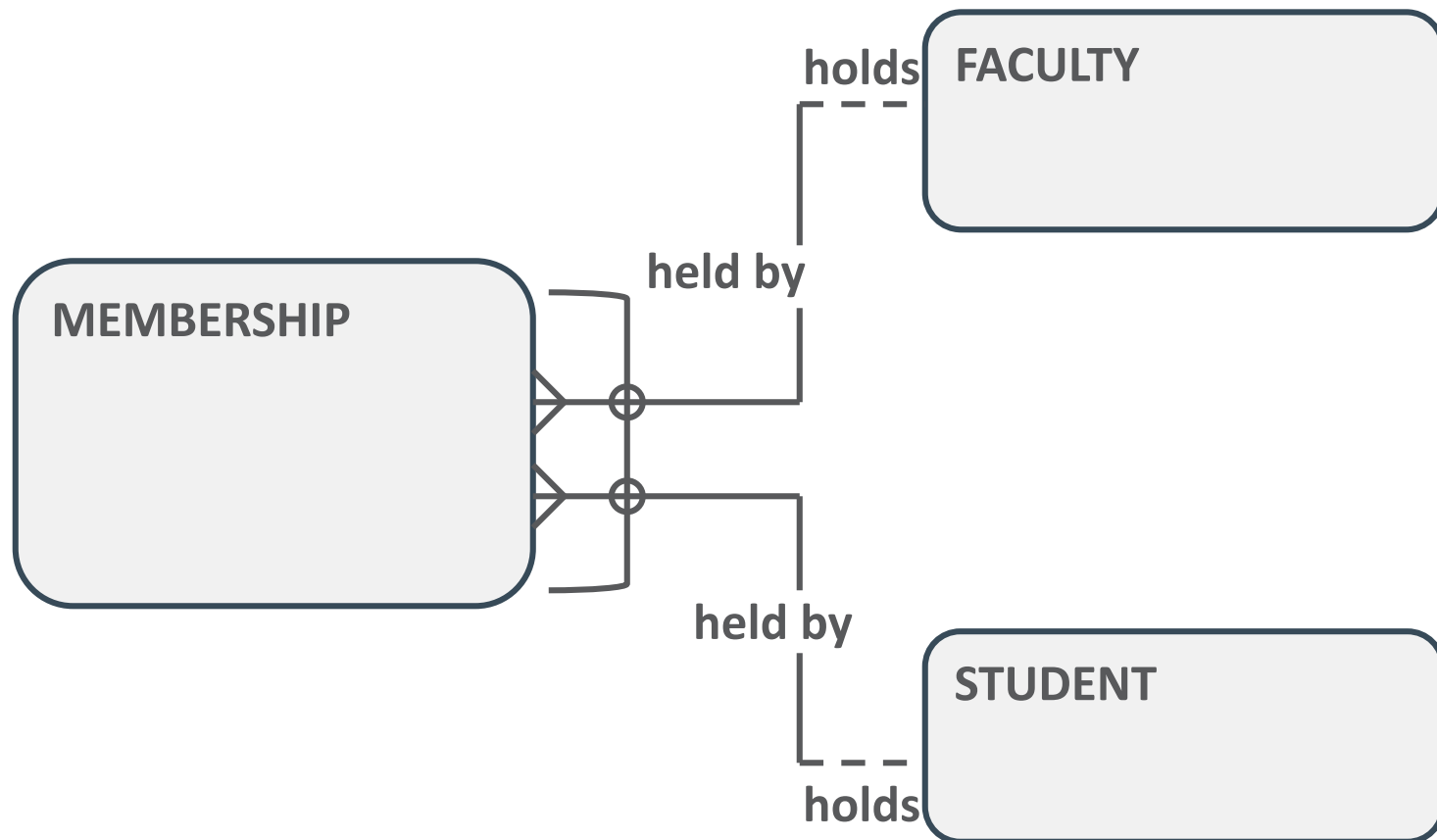
**Мэтт, ты можешь создать объект, имеющий дуговую связь?**

**Я могу создать общий объект MEMBERSHIP, содержащий сведения об участии, общие для всех категорий участия.**



Мэтт

# Практический сценарий: Создание общего объекта



# Упражнение по проекту 4

## DFo\_3\_1\_4\_Project

- База данных магазина Oracle Baseball League определяет иерархические, рекурсивные и дуговые связи

# Сводка

В этом уроке вы узнали следующее:

- Определение ассоциативных связей
- Решение связей типа M:M
- Определение и примеры непередаваемых связей
- Определение и построение объектов с супертипом и подтипом
- Определение иерархических, рекурсивных и дуговых связей



