

ЛЕКЦИЯ 8

ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ

- 1) Основные концепции.
- 2) Упрощение концептуальной модели данных.
- 3) Методика преобразования концептуальных структур данных в реляционные структуры.
- 4) Преобразование связи типа «суперкласс/подкласс».
- 5) Преобразование составных объектов.

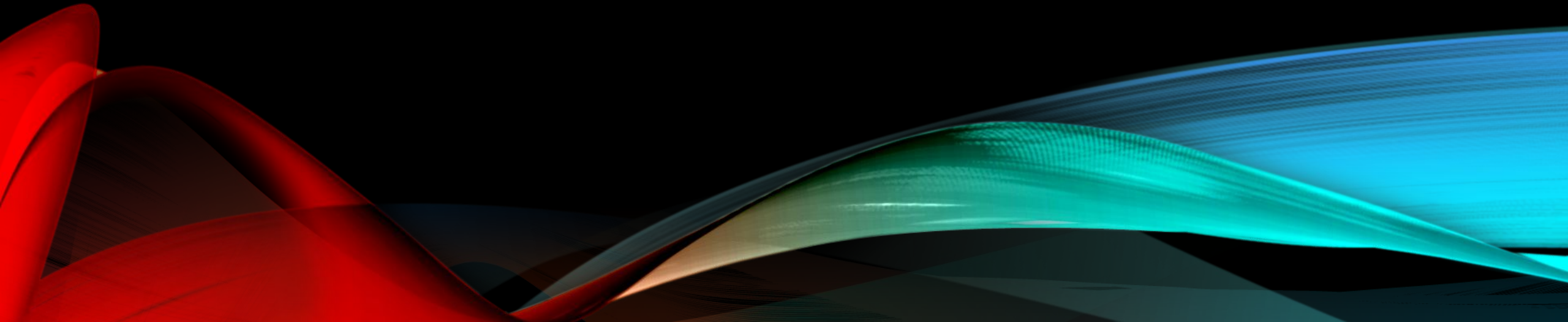
ОСНОВНЫЕ КОНЦЕПЦИИ

Фаза логического проектирования предполагает следующие действия:

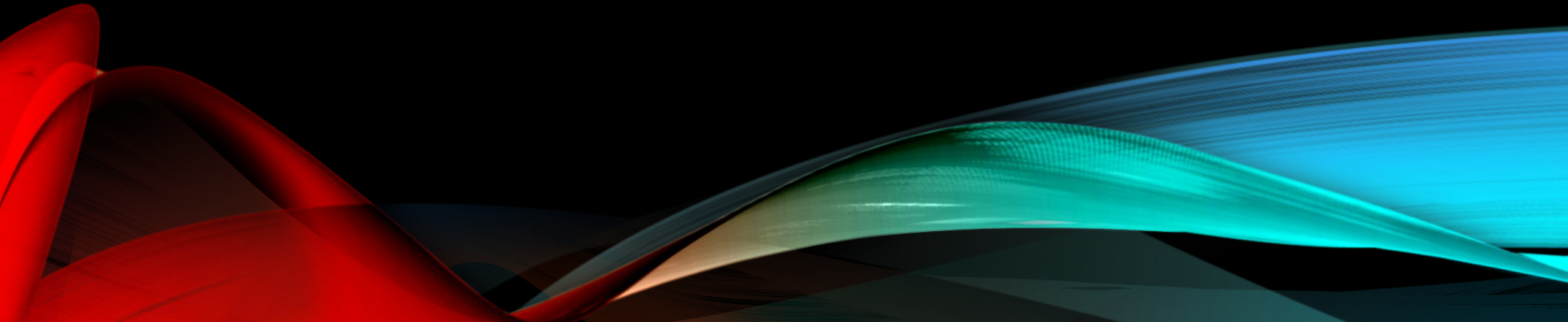
- × преобразование концептуальной модели данных в логическую модель, в результате которого будет определена схема реляционной модели данных;
- × проверка модели с помощью концепций последовательной нормализации;
- × проверка модели в отношении транзакций пользователей;
- × проверка поддержки целостности данных.

УПРОЩЕНИЕ КОНЦЕПТУАЛЬНОЙ МОДЕЛИ ДАННЫХ

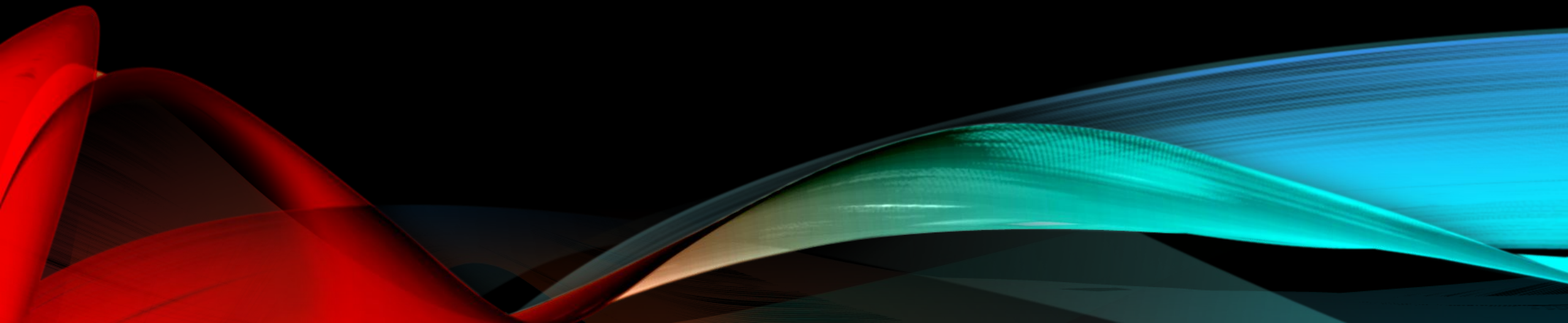
Преобразование концептуальной модели данных в логическую модель, в результате которого будет определена схема реляционной модели данных, может иметь два подхода.



1) Проектировщик работает с концептуальной моделью напрямую, не прибегая к ее предварительному преобразованию. В этом случае ему придется столкнуться с необходимостью преобразования разнообразных структур данных



2) Проектировщик, прежде чем приступить к процессу перехода от концептуальной модели к логической модели, стремится вначале данный переход максимально упростить, проводя предварительные преобразования концептуальной модели, преобразования некоторых ее, не подходящие для реляционных СУБД структур данных.



К таким структурам данных относятся:

- ✗ связи типа «многие ко многим»;
- ✗ сложные связи;
- ✗ рекурсивные связи;
- ✗ связи с атрибутами;
- ✗ множественные атрибуты;
- ✗ избыточные связи.

Если в модели присутствуют перечисленные нежелательные структуры, то они должны быть исключены путем тождественной их замены на структуры данных, допустимые для логической модели.

ИСКЛЮЧЕНИЕ СВЯЗИ ТИПА «МНОГИЕ КО МНОГИМ»

Преобразование связи типа «многие ко многим» осуществляется путем введения некоторого промежуточного объекта с заменой одной связи двумя связями типа «один ко многим» с вновь созданным объектом. При организации новых связей необходимо следить за тем, чтобы максимальная мощность связи «один» была направлена к исходному объекту, а максимальная мощность связи «много» — к вновь созданному объекту.

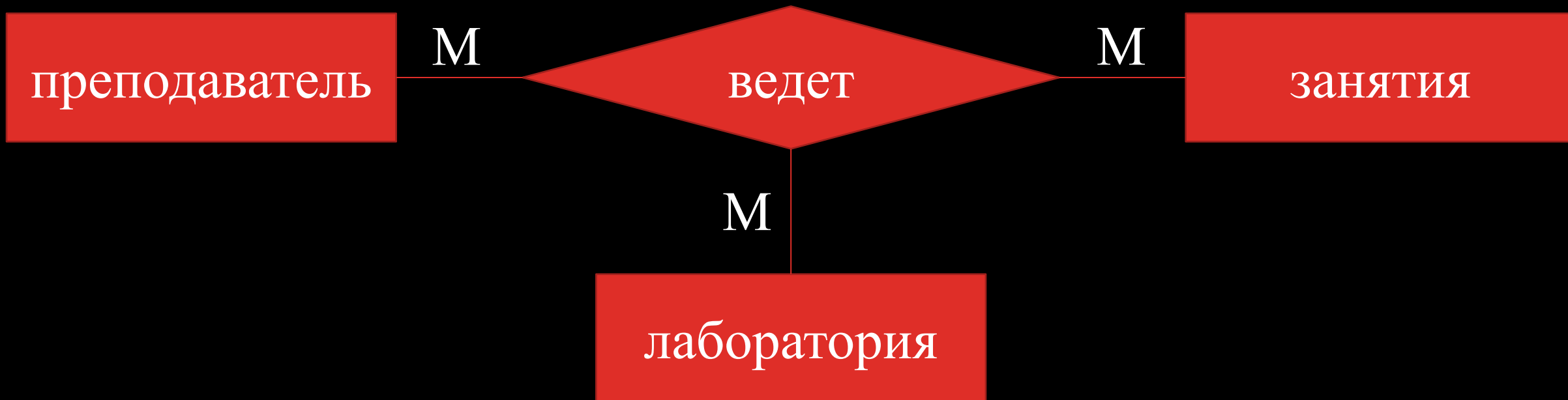


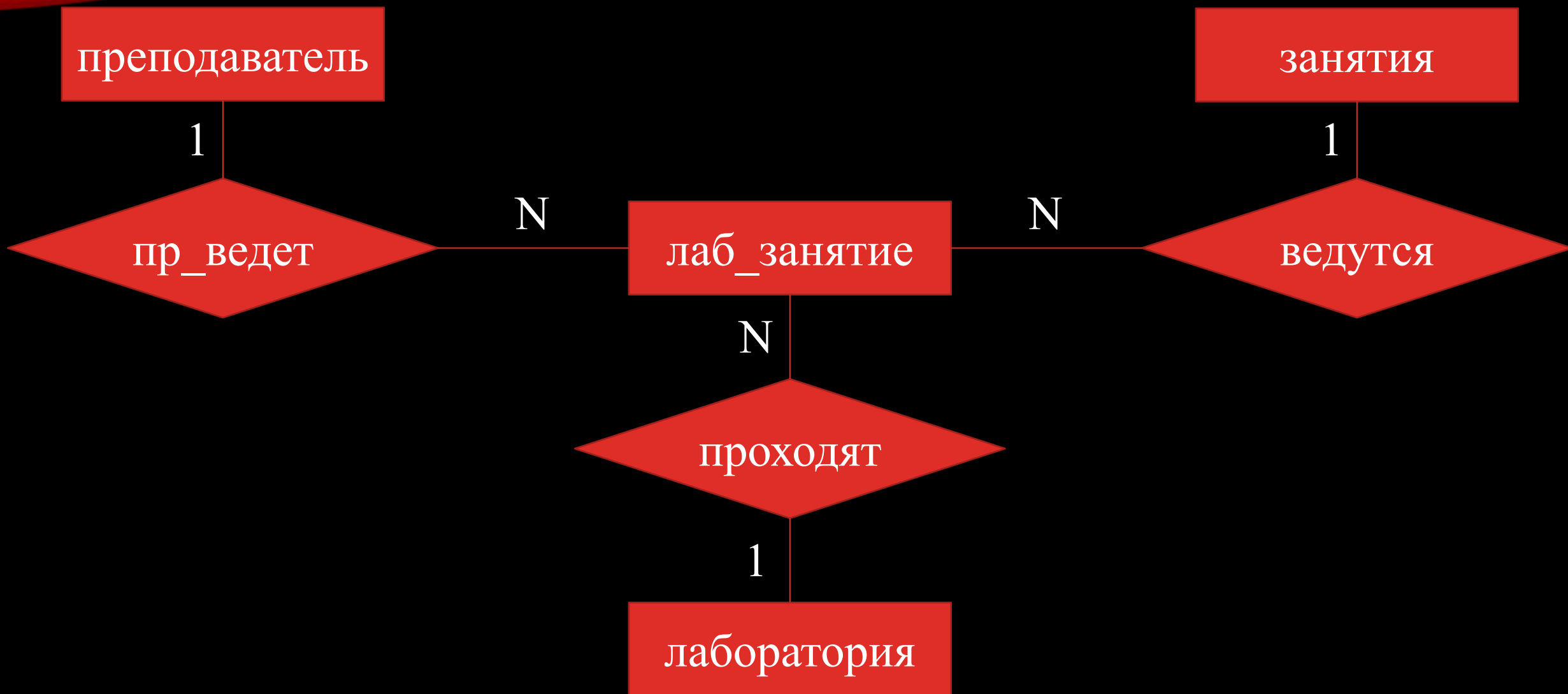


ПР_ЧИТАЕТ типа 1: N между объектами ПРЕПОДАВАТЕЛЬ и ЧТЕНИЕ;

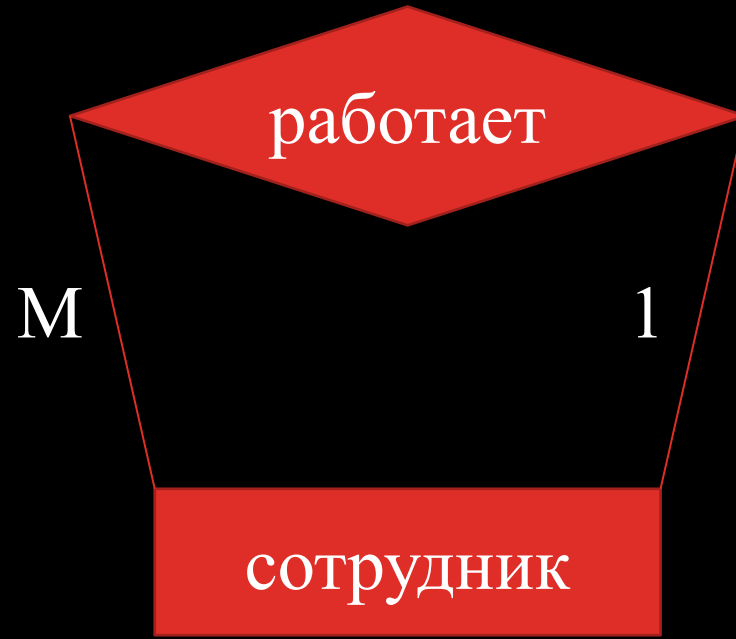
К_ЧИТАЕТСЯ типа 1: N между объектами КУРС и ЧТЕНИЕ.

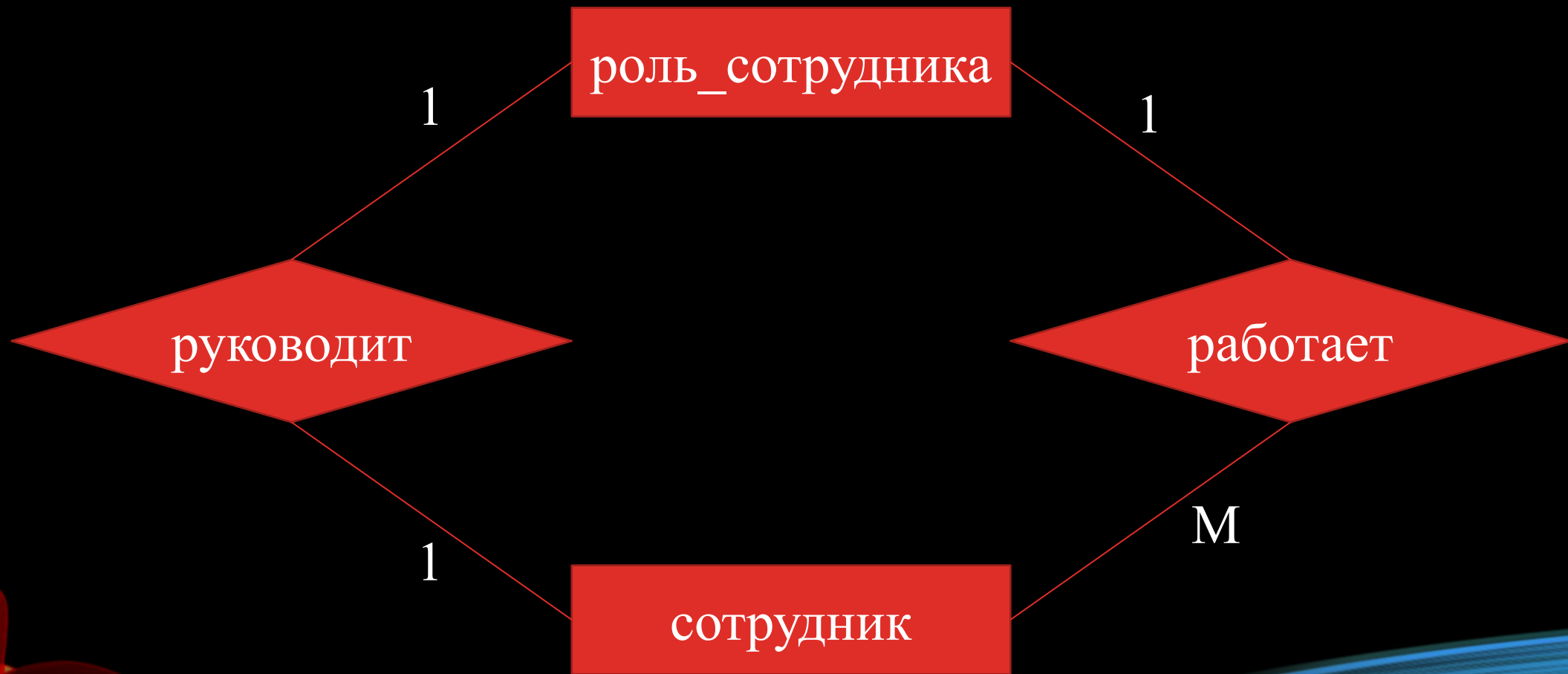
ИСКЛЮЧЕНИЕ СЛОЖНЫХ СВЯЗЕЙ





ИСКЛЮЧЕНИЕ РЕКУРСИВНЫХ СВЯЗЕЙ





ПРЕОБРАЗОВАНИЕ ОБЪЕКТОВ И АТТРИБУТОВ

Общий подход к преобразованию объектов концептуальной модели ПрО в отношения реляционной базы данных состоит в следующем:

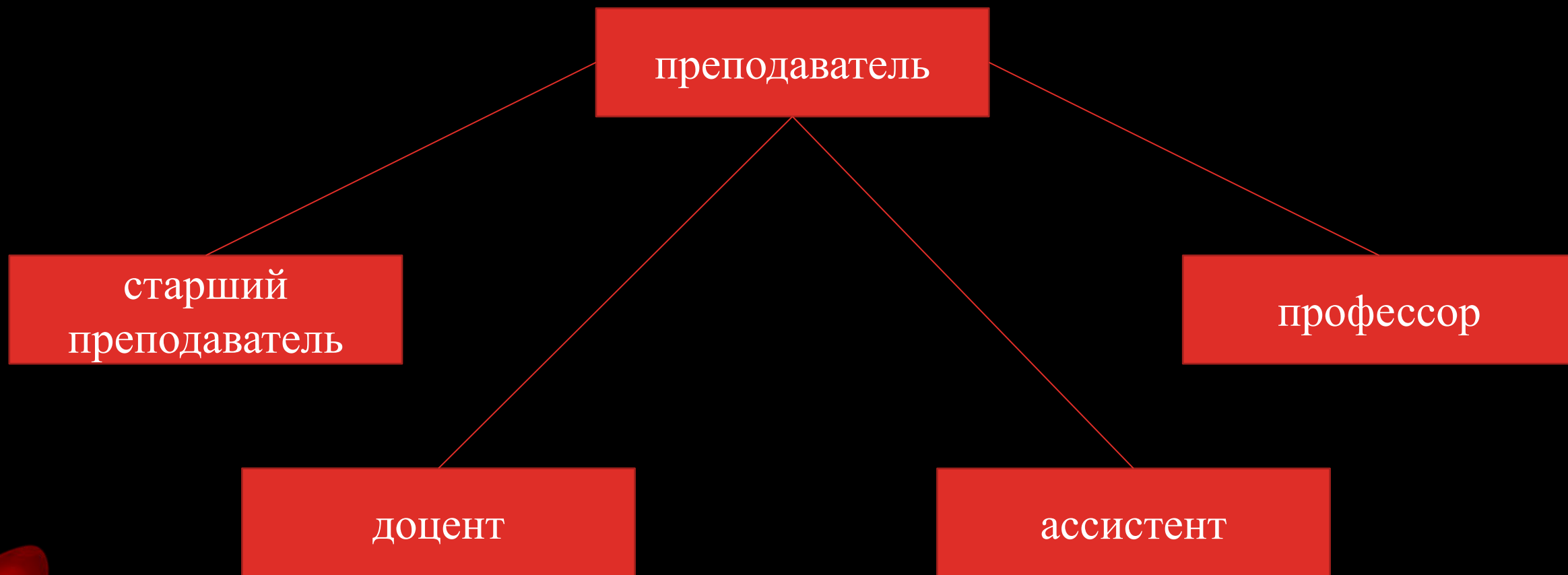
- ✗ построить набор предварительных отношений и указать первичные ключи для каждого отношения;

✗ подготовить список всех представляющих интерес атрибутов (тех из них, которые не были перечислены в диаграмме в качестве первичных ключей объектов) и назначить каждый из этих атрибутов одному из предварительных отношений с тем условием, чтобы эти отношения находились в НФБК. Если полученные отношения в итоге не находятся в НФБК, то в этих случаях диаграммы необходимо пересмотреть.

ПРЕОБРАЗОВАНИЕ СВЯЗИ ТИПА СУПЕРКЛАСС/ПОДКЛАСС

Для каждой присутствующей в логической модели данных связи типа «суперкласс/подкласс» объект суперкласса необходимо определить как родительский, а объект подкласса — как дочерний.

Если суперкласс с его подклассами имеет непересекающиеся связи, где каждый экземпляр суперкласса обязательно должен быть членом одного подкласса, то решением является представление каждого из подклассов в виде отдельного отношения



Подобная диаграмма преобразуется в следующую
реляционную схему отношений:

ПРЕПОДАВАТЕЛЬ (Табельный номер, ФИО, Адрес,
Педагог_стаж);

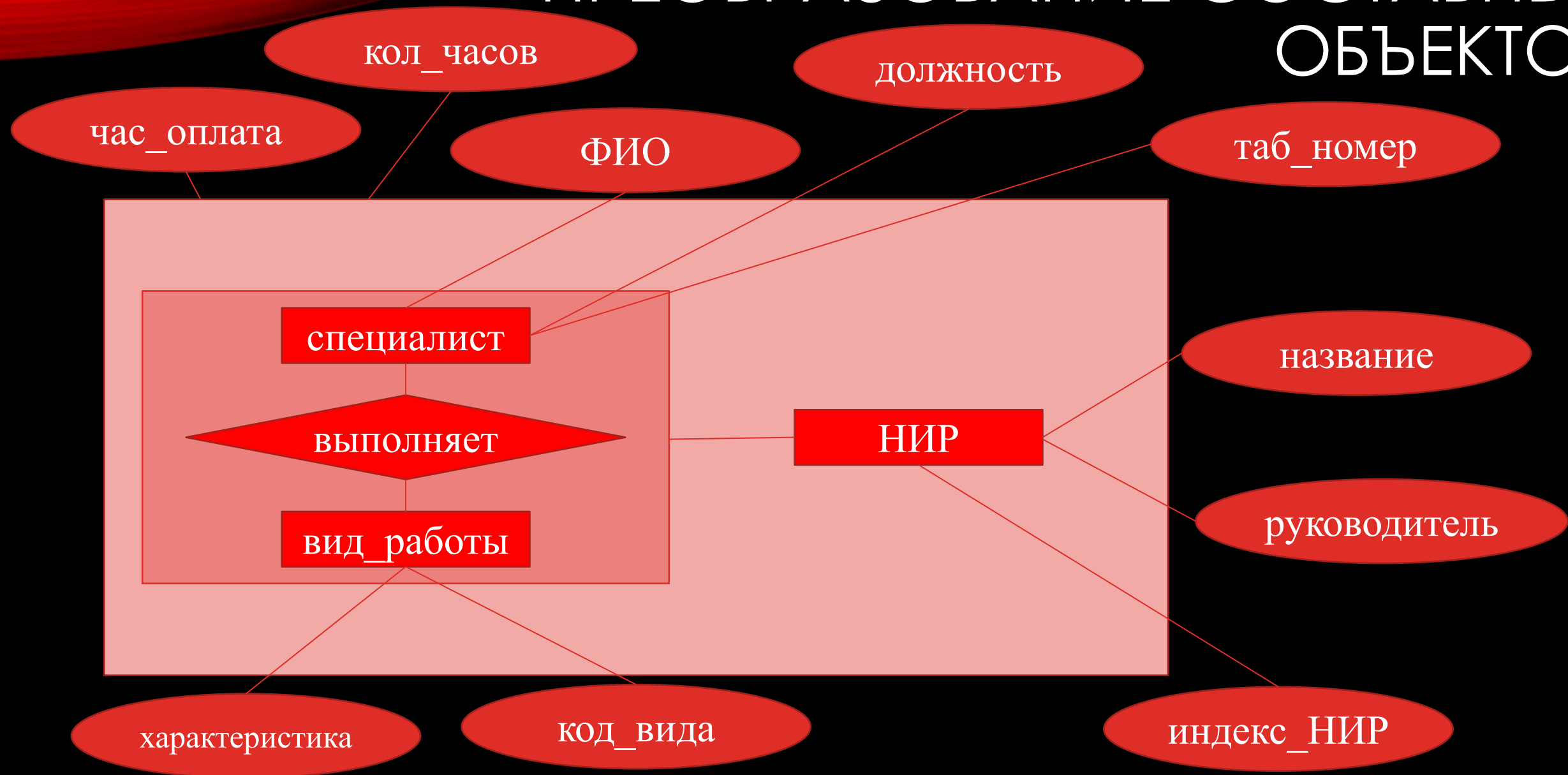
ПРОФЕССОР (Табельный номер,
Номер_диплома_профессора);

ДОЦЕНТ (Табельный номер, Номер_диплома_доцента);

СТАРШИЙ_ПРЕПОДАВАТЕЛЬ (Табельный номер);

АССИСТЕНТ (Табельный номер).

ПРЕОБРАЗОВАНИЕ СОСТАВНЫХ ОБЪЕКТОВ



Конечный результат преобразования выглядит так:

СПЕЦИАЛИСТ (Таб_номер, Ф_И_О,
Должность);

ВИД_РАБОТЫ (Код_вида, Характеристика);

НИР (Индекс_Нир, Название, Руководитель);

ПО (Таб_номер, Код_вида, Индекс_Нир,
Час_оплата, Кол_часов).