

ЛЕКЦИЯ 7

НОРМАЛИЗАЦИЯ ОТНОШЕНИЙ

- 1) Избыточность данных в БД
- 2) Первая нормальная форма
- 3) Вторая нормальная форма
- 4) Третья нормальная форма
- 5) Нормальная форма Бойса-Кодда
- 6) Четвертая нормальная форма
- 7) Пятая нормальная форма

ИЗБЫТОЧНОСТЬ ДАННЫХ В БД

Избыточность данных в БД относится к нежелательным явлениям, поскольку ведет к увеличению объема памяти, необходимого для физического хранения отношений. Избыточность вызывается, прежде всего, дублированием данных. Полностью избыточность устранять не требуется, нужно лишь ее минимизировать так, чтобы были устранены некоторые ее виды и осталась только доля избыточности, необходимая для удержания базы данных как единое целое.

<u>Ном_зач_кн</u>	ФИО_студента	Код_группы	ФИО_старосты	Куратор
20-Т-201	Иванов С.И.	20-Т-11	Рябов В.С	Доц. Сидоров И.И.
20-Т-215	Петров Я.Р.	20-Т-12	Михайлов М.М.	Доц. Видов С.С.
20-Т-217	Рябов В.С	20-Т-11	Рябов В.С	Доц. Сидоров И.И.
20-Т-211	Мехова А.Л.	20-Т-11	Рябов В.С	Доц. Сидоров И.И.

АНОМАЛИИ

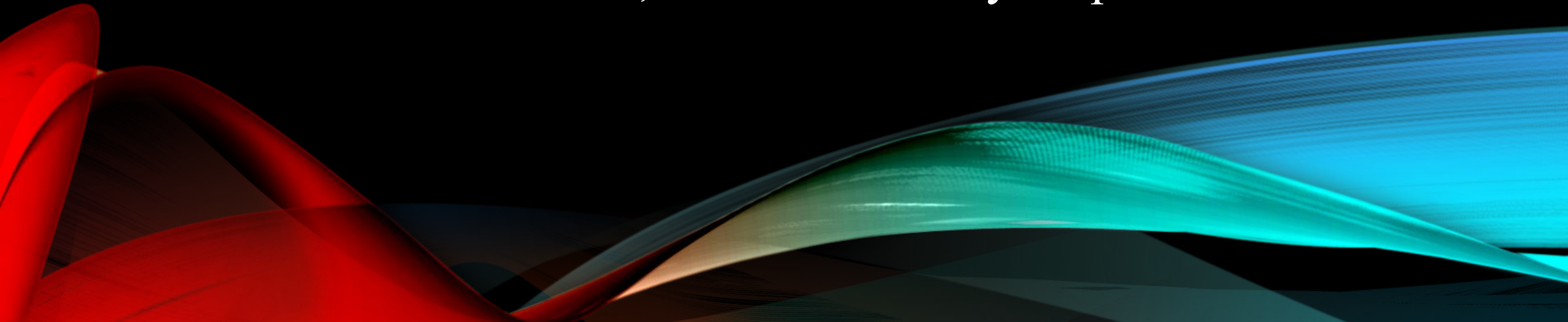
Процесс нормализации позволяет избавиться от присутствия в базе данных различного рода аномалий.

Различают три вида аномалий:

- × аномалии включения;
- × аномалии удаления;
- × аномалии модификации.

АНОМАЛИИ ВКЛЮЧЕНИЯ

Первый вид этой аномалии связан с присутствием избыточных данных и может возникнуть при вставке сведений о новом студенте некоторой группы. Поскольку кортеж, содержащий сведения об этом студенте, должен включать также информацию о старосте группы и о ее кураторе, то есть соответствовать подобным уже имеющимся в базе данных сведениям, то возникает дублирование данных.



Второй вид аномалии включения возникает при попытке создать новую группу и ввести ее в отношение при том условии, что еще в нее не зачислен ни один студент. Ввод такой информации в подобной ситуации требует присвоения значения NULL всем атрибутам описания студента, в том числе и атрибуту Ном_зач_кн, который является первичным ключом данного отношения. Но реализация такой попытки приведет к нарушению категорной целостности, а значит, система ее обязана отклонить.

Отношение СТУДЕНТ

<u>Ном_зач</u> <u>кн</u>	ФИО_ студента	Код_ группы
20-Т-201	Иванов С.И.	20-Т-11
20-Т-215	Петров Я.Р.	20-Т-12
20-Т-217	Рябов В.С	20-Т-11
20-Т-211	Мехова А.Л.	20-Т-11

Отношение ГРУППА

<u>Код_</u> <u>группы</u>	ФИО_старосты	Куратор
20-Т-11	Рябов В.С	Доц. Сидоров И.И.
20-Т-12	Михайлов М.М.	Доц. Видов С. С.
20-Т-13	NULL	NULL

АНОМАЛИЯ УДАЛЕНИЯ

Преобразования должны быть проведены точно такие же, какие были проведены для исключения аномалии включения.

АНОМАЛИЯ МОДИФИКАЦИИ

Помимо аномалий включения и удаления, присутствует еще одна аномалия, которая называется аномалией модификации. Такая аномалия возникает при попытке изменить что-либо касающееся сведений о группе обучения студента.



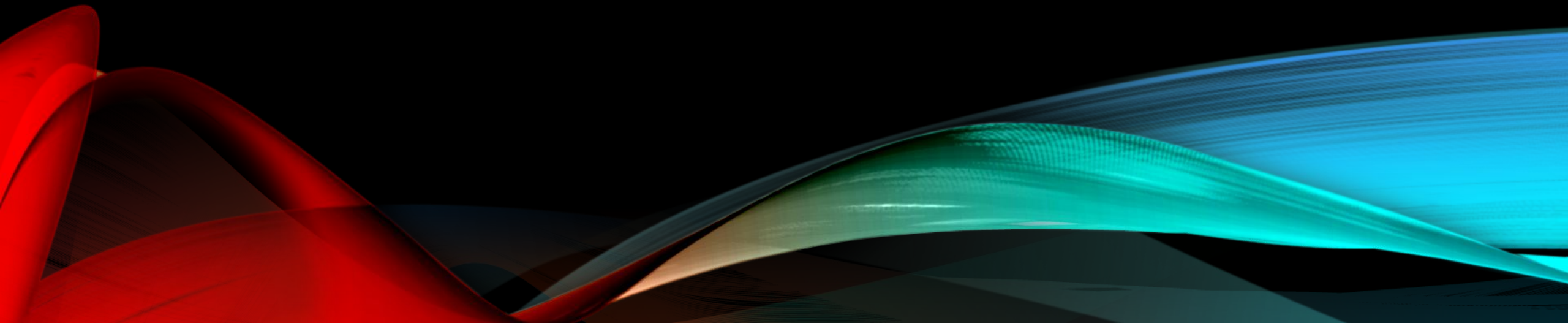
Задача группировки отношений может быть решена множеством способов, но требуется найти такой состав отношений, который удовлетворял бы следующим требованиям:

- × множество отношений должно обеспечивать минимизацию избыточности представления информации;
- × назначенные для отношения ключи должны быть минимальными;
- × при выполнении операций в БД не должно быть аномалий.

НОРМАЛИЗАЦИЯ

Чтобы исключить различного рода аномалии из отношения, его подвергают процессу декомпозиции. При решении задачи декомпозиции возникают две проблемы.

Первая проблема — это проблема обратимости, заключающаяся в возможности восстановления исходной схемы, а именно — восстановления любого кортежа исходного отношения, используя кортежи полученных в результате декомпозиции отношений.

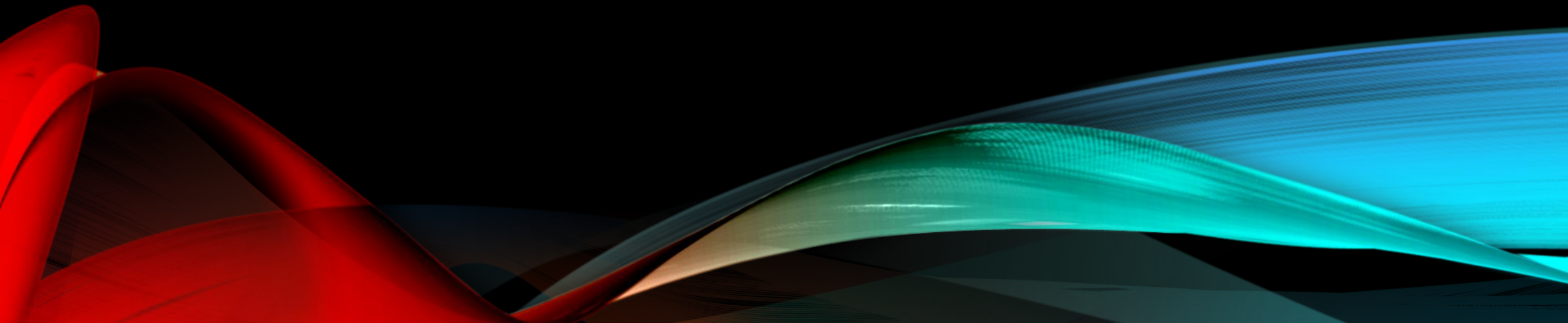




Вторая проблема связана с сохранением зависимостей. Проектирование базы данных включает в себя и определение ограничений, накладываемых на ее отношения. В процессе декомпозиции получаются новые отношения и ограничения, которые приписываются им, должны быть такими, чтобы были сохранены исходные ограничения.

Для устранения рассмотренных недостатков применяется процесс нормализация отношений. Данный процесс — это формальный метод анализа отношений на основе их первичных или потенциальных ключей и существующих функциональных зависимостей, являющийся одним из наиболее строгих способов улучшения характеристик БД. Он включает ряд формальных

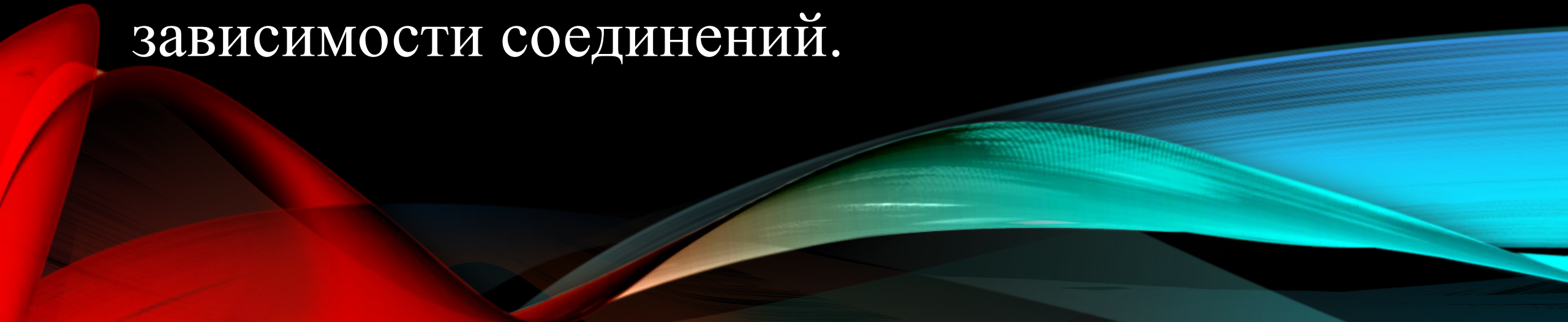
- × $1НФ$ — первая нормальную форму;
- × $2НФ$ — вторая нормальную форму;
- × $3НФ$ — третья нормальную форму;
- × $НФБК$ — нормальная форму Бойса-Кодда;
- × $4НФ$ — четвертая нормальную форму;
- × $5НФ$ — пятая нормальную форму.





Каждая нормальная форма налагает определенные ограничения на данные. Эти ограничения вводятся в каждом конкретном отношении, и соблюдение этих ограничений в отношении связано уже с наличием нормальной формы.

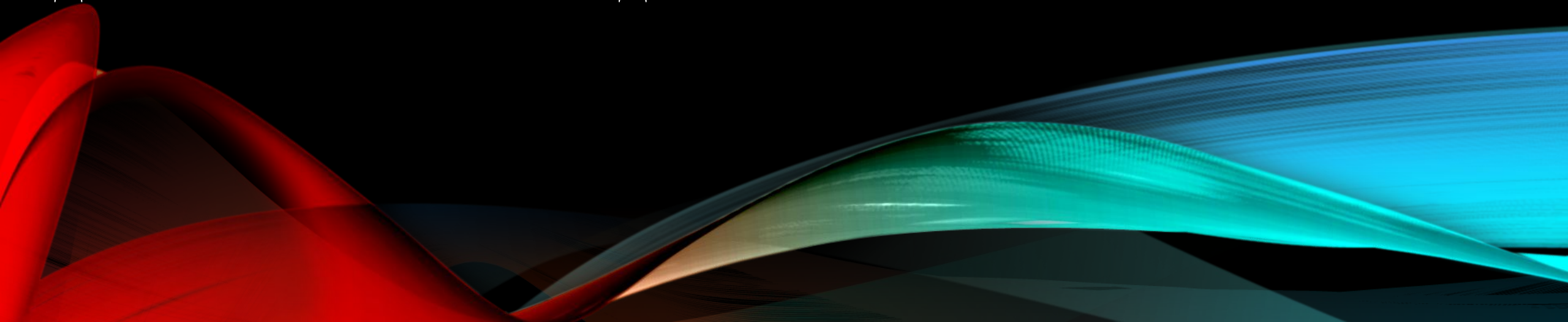
- × *1НФ, 2НФ, 3НФ* — ограничивают зависимость не первичных атрибутов от ключей.
- × *НФБК* — ограничивает зависимость первичных атрибутов.
- × *4НФ* — формулирует ограничения на виды многозначных зависимостей.
- × *5НФ* — вводит другие типы зависимостей: зависимости соединений.





Каждая нормальная форма более высокого уровня предполагает, что анализируемое отношение уже находится в нормальной форме на уровень ниже рассматриваемой. Чем выше уровень нормальной формы, тем жестче накладываемые на отношения ограничения. В ходе нормализации схема базы данных становится все более строгой, а ее отношения все менее подвержены различного рода аномалиям. Сам процесс перехода от нормальной формы более низкого уровня к нормальной форме более высокого уровня, и называется нормализацией отношений (НО).

Для реляционных баз данных необходимо, чтобы все отношения базы данных обязательно находились в 1НФ. Нормальные формы более высокого порядка могут использоваться разработчиками по своему усмотрению. Однако грамотный специалист стремится к тому, чтобы довести уровень нормализации базы данных хотя бы до 3НФ, тем самым, исключив из базы данных избыточность данных и аномалии обновления.



ПЕРВАЯ НОРМАЛЬНАЯ ФОРМА

Отношение находится в первой нормальной форме, если все его атрибуты имеют простые (атомарные) значения. Значения в домене каждого атрибута отношения не являются ни списками, ни множествами простых или сложных значений. Т.е. в отношении не должно быть повторяющихся групп. В противном случае отношение считается ненормализованным и ему соответствует многоуровневая таблица (иерархия) в отличие от однородной табличной структуры нормализованного отношения.

Отношение рождение

Имя	Дата рождения
Аня	5 марта 1986
Саша	25 января 1987
Ольга	1 ноября 1987
Федор	14 сентября 1986

Отношение рождение

Имя	День рождения	Месяц рождения	Год рождения
Аня	5	март	1986
Саша	25	январь	1987
Ольга	1	ноябрь	1987
Федор	14	сентябрь	1986

Ненормализованная таблица ПОЛ

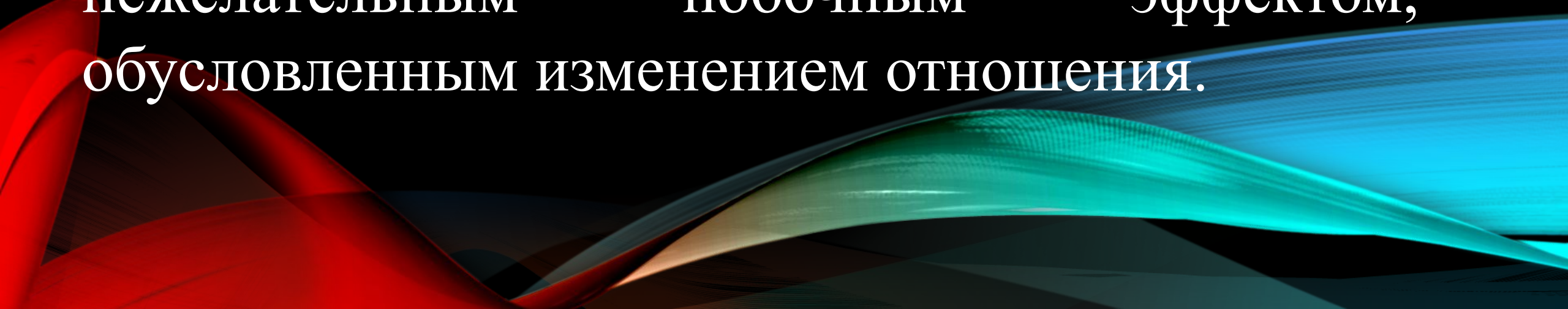
Имя	Пол
(Саша, Федор)	мужской
Ольга	женский

Имя	Пол
Саша	мужской
Федор	мужской
Ольга	женский

<u>Индекс_дисц</u>	Назв_предм	Кол_час	<u>Код_лек</u>	ФИО_лек	Должность
ЕНФ02	БСБД	54	001	Иванов И.И.	Доцент
ЕНФ02	БСБД	54	002	Петров Т.Т.	Старший преподаватель
ОПДФ08	ЭиС	102	003	Смирнов М.И.	Доцент
ОПДФ08	ЭиС	102	001	Иванов И.И.	Доцент
ОПДФ08	ЭиС	102	002	Петров Т.Т.	Старший преподаватель
СД03	ЯП	54	001	Иванов И.И.	Доцент
СД03	ЯП	54	002	Петров Т.Т.	Старший преподаватель

ВТОРАЯ НОРМАЛЬНАЯ ФОРМА

Вторая и третья нормальные формы возникли в результате стремления избежать аномалий обновления данных при работе с БД и избавиться от информационной избыточности в отношениях. Аномалии обновления являются нежелательным побочным эффектом, обусловленным изменением отношения.

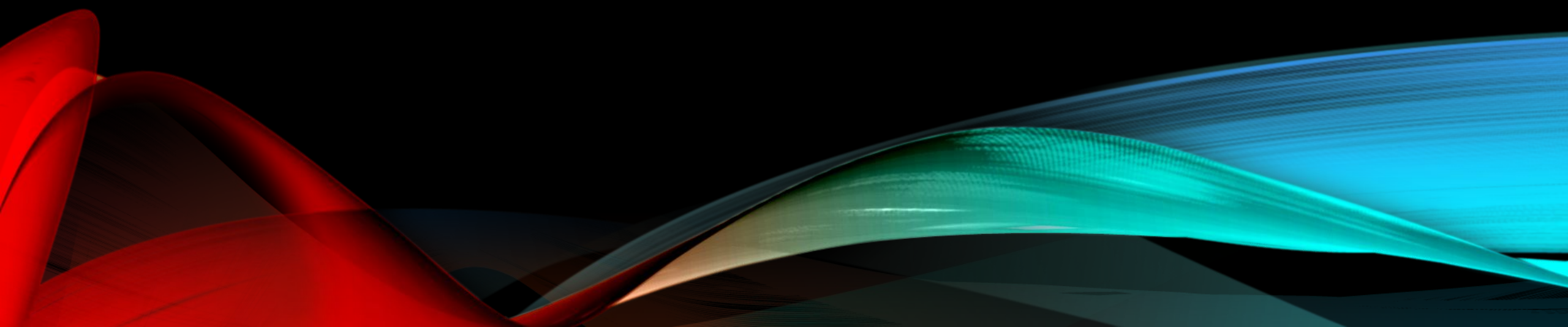


Вторая нормальная форма применяется к отношениям с составными ключами, т. е. к таким отношениям, первичный ключ которых состоит из двух или более атрибутов. Отношение, у которого первичный ключ включает только один атрибут, всегда находится во 2НФ.



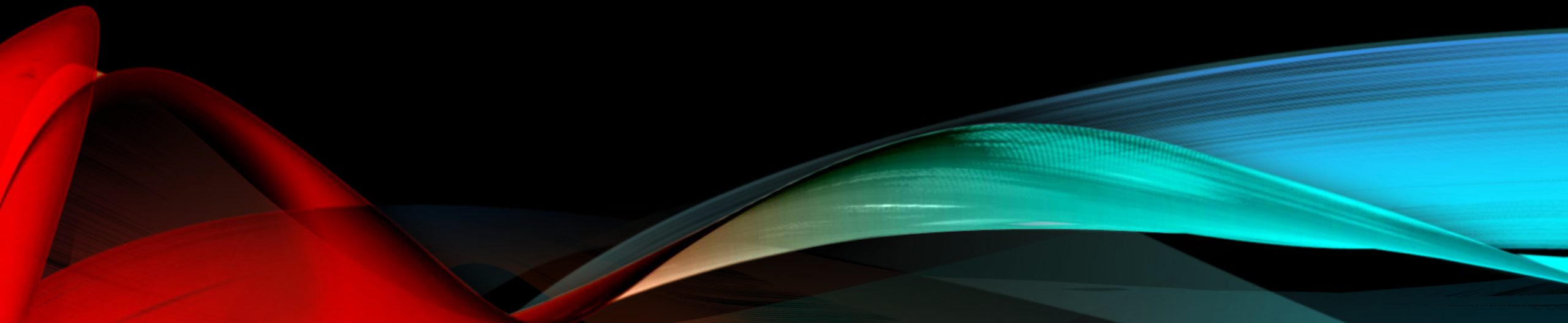
Рассмотрим отношение КОНСУЛЬТАЦИИ
_ДИПЛОМНИКОВ со схемой:

(Таб Ном преп, Ном зач кн, Дата, ФИО преп,
Должность, ФИО студ, Тема диплома, Время,
Аудитория, Вместимость).



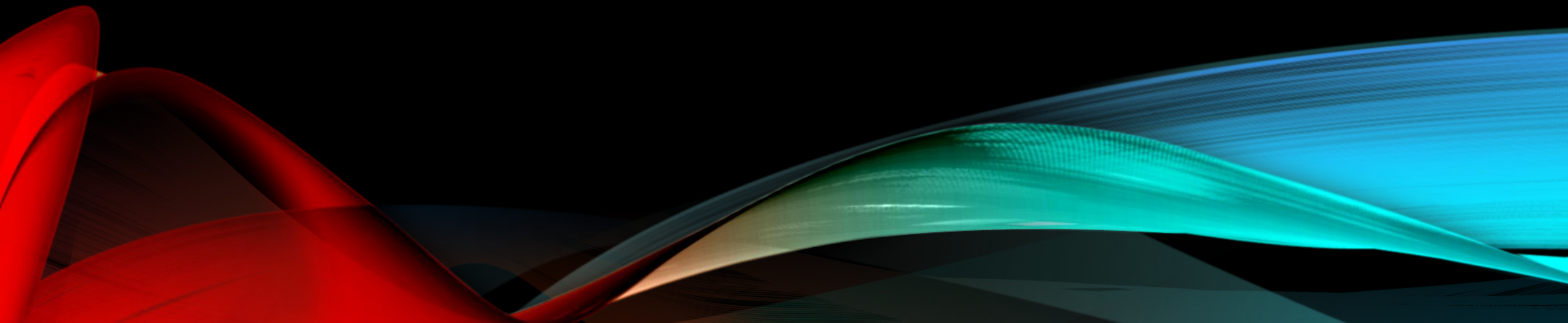
Первичный ключ данного отношения является составным и по определению однозначно идентифицирует каждый кортеж отношения:

(Таб Ном преп, Ном зач кн, Дата) → (ФИО_преп, Должность, ФИО_студ, Тема_диплома, Время, Аудитория, Вместимость).



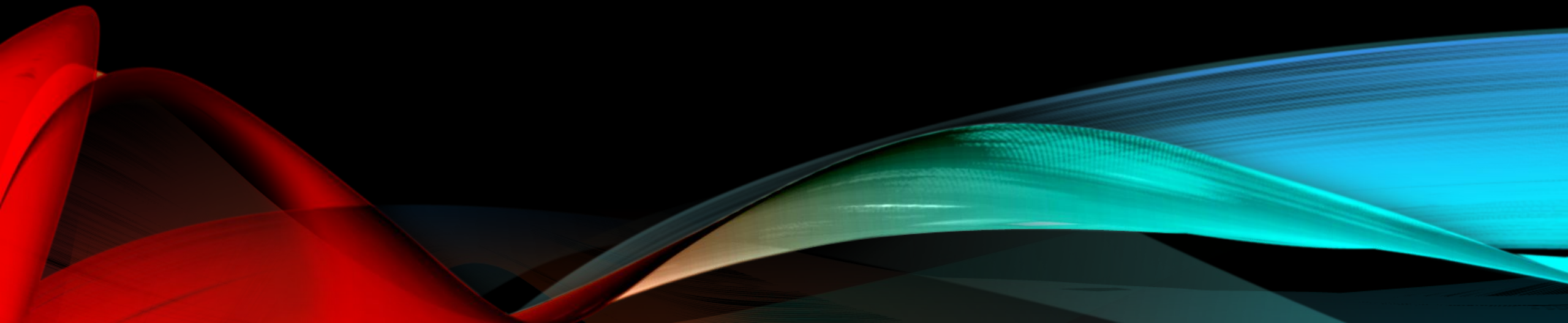
Описательные атрибуты преподавателя ФИО_преп, Должность зависят только от части первичного ключа. Данную ситуацию определяет зависимость:

Таб_Ном_преп → (ФИО_преп, Должность).



Описательные атрибуты студента
ФИО_студ, Тема_диплома также зависят
только от части первичного ключа и не зависят
от остальных атрибутов ключа, то есть
имеется зависимость вида:

Ном_зач_кн → (ФИО_студ, Тема_диплома).





Отсутствие полной функциональной зависимости каждого не первичного атрибута от первичного ключа, как и в других рассмотренных примерах, является источником аномалий обновления и вносит свою долю избыточности в БД. Устранение данных отрицательных явлений осуществляется путем декомпозиции исходного отношения на три со следующими схемами:

ПРЕПОДАВАТЕЛЬ (Таб Ном преп, ФИО_
преп, Должность);

СТУДЕНТ (Ном зач кн, ФИО_студ,
Тема_диплома);

КОНСУЛЬТАЦИИ (Таб Ном преп,
Ном зач кн, Дата, Время, Аудитория,
Вместимость).





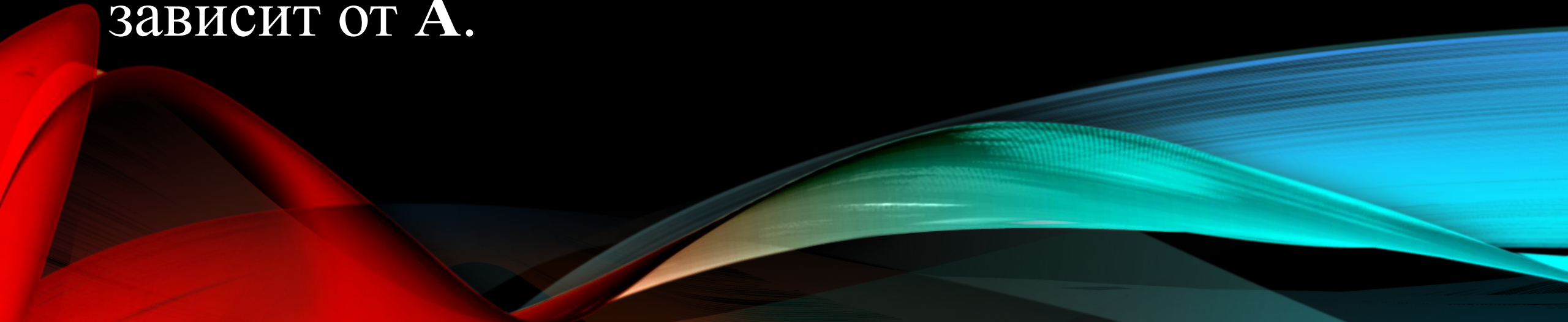
Полученная схема базы данных содержит три отношения: первое содержит сведения обо всех преподавателях, второе содержит сведения обо всех дипломниках, а последнее отношение КОНСУЛЬТАЦИИ связывает воедино базу данных, так как в него включены первичные атрибуты первых двух отношений. Поскольку в базе данных теперь отсутствуют повторяющиеся данные, то и отсутствует возможность привести ее в несогласованное состояние.

ТРЕТЬЯ НОРМАЛЬНАЯ ФОРМА

Рассмотрим транзитивную зависимость следующего типа:

если $A \rightarrow B$, $B \not\rightarrow A$ (B не является ключом), $B \rightarrow C$, то $A \rightarrow C$.

В этом случае считается, что C транзитивно зависит от A .



ПРЕПОДАВАТЕЛЬ (Таб_Ном_преп, ФИО_преп,
Должность);

СТУДЕНТ (Ном_зач_кн, ФИО_студ, Тема_диплома);

КОНСУЛЬТАЦИИ (Таб_Ном_преп, Ном_зач_кн,
Дата, Время, Аудитория, Вместимость).

Последнее отношение содержит транзитивную
зависимость:

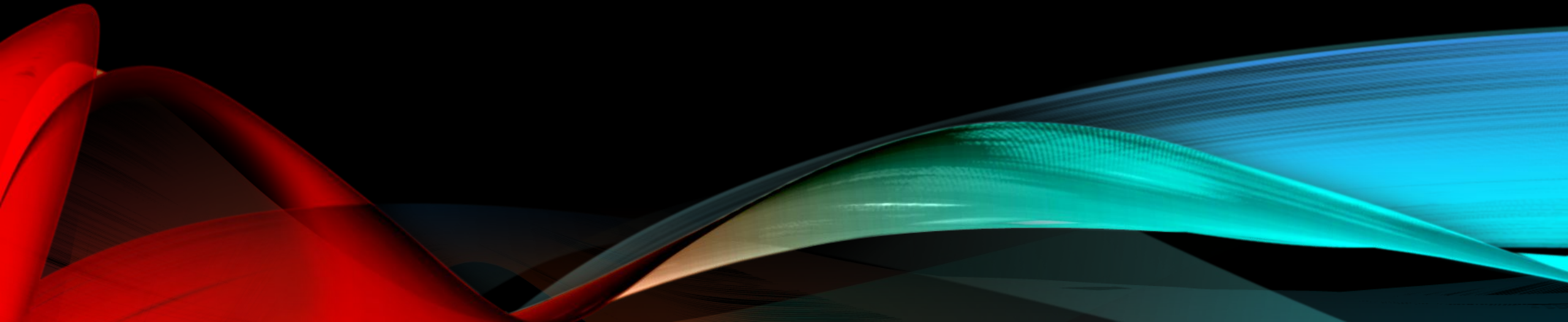
(Таб_Ном_преп, Ном_зач_кн, Дата) → Аудитория →
Вместимость.



Следовательно, это отношение не находится в 3НФ со всеми вытекающими из этого последствиями, и прежде всего следующими:

- ✗ если аудитория исключается из расписания консультаций, то о ней вообще теряются сведения;
- ✗ если аудитория перестроена и в результате изменилась ее вместимость, то придется просмотреть все corteжи и провести модификацию значений атрибута.

Для устранения транзитивной зависимости необходимо провести декомпозицию последнего отношения, удалив из него транзитивно- зависимый атрибут и поместив его в новое отношение вместе с копией того атрибута, от которого он зависит.



Таким образом, база данных этого примера, лишенная транзитивных зависимостей, в ЗНФ будет выглядеть так:

ПРЕПОДАВАТЕЛЬ (Таб_Ном_преп, ФИО_преп, Должность);

СТУДЕНТ (Ном_зач_кн, ФИО_студ, Тема_диплома);

КОНСУЛЬТАЦИИ (Таб_Ном_преп, Ном_зач_кн, Дата, Время, Аудитория);

АУДИТОРИЯ (Аудитория, Вместимость).

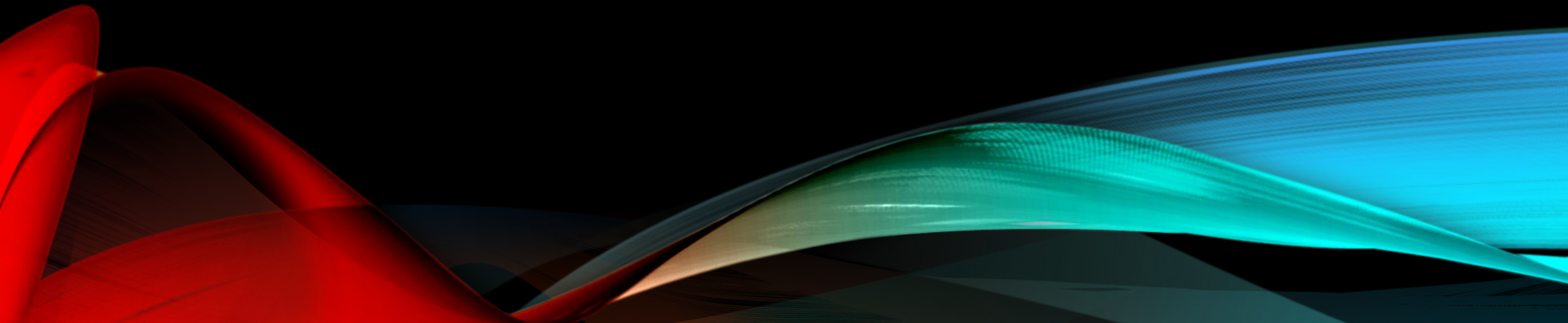
НОРМАЛИЗАЦИЯ НА ОСНОВЕ ДЕКОМПОЗИЦИИ

Схему отношения БД, не находящуюся в 3НФ необходимо к ней привести. Приведение осуществляется через разбиение схемы отношения на пару схем отношений R_1 и R_2 так, чтобы любое отношение $r(R)$, удовлетворяющее F , разлагалось на R_1 и R_2 . Если какое-то отношение из R_1 и R_2 не окажется в 3НФ, то процесс декомпозиции придется повторить. Этот процесс продолжается до тех пор, пока все полученные отношения не окажутся в 3НФ относительно F .

При нормализации схемы отношения посредством декомпозиции возникает еще ряд проблем, которые кратко можно обозначить следующим образом:

D1 — временная сложность процесса;

D2 — число порожденных процессом схем отношения может оказаться большим, чем в действительности необходимо для 3НФ;





D3 — при декомпозиции могут возникнуть частичные зависимости, которые также могут породить избыток схем;

D4 — для построенной схемы БД заданное множество F-зависимостей может оказаться не навязанным;

D5 — с помощью декомпозиции можно породить схемы со скрытыми транзитивными зависимостями.

НОРМАЛЬНАЯ ФОРМА БОЙСА-КОДДА

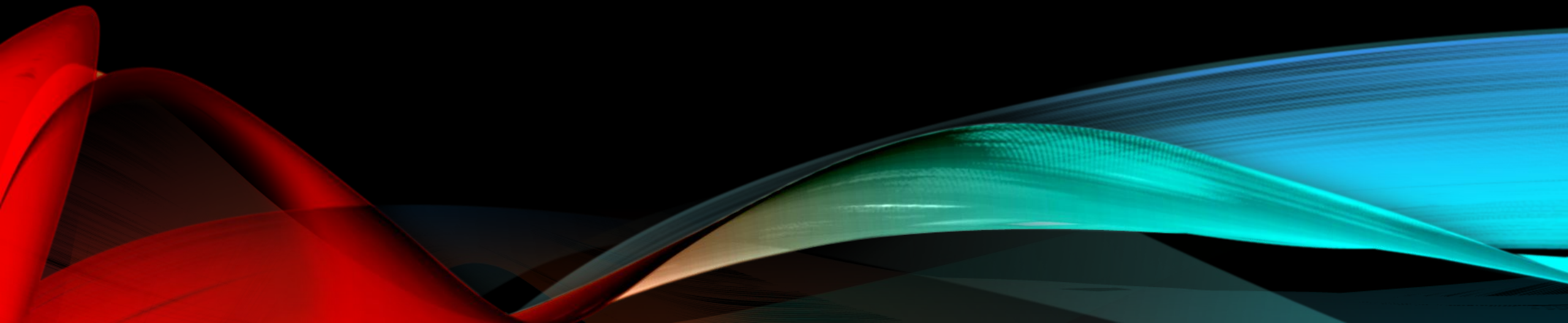
Определение для 3НФ подходит для ситуаций с упрощающим картину допущением того, что отношение имеет только один потенциальный ключ, который, естественно, и является первичным ключом. Далеко не все отношения могут быть уложены в данные довольно жесткие рамки.

Более обобщающими являются случаи, когда в наличии имеются следующие условия:

1. отношение имеет два (или более) потенциальных ключа;
2. два потенциальных ключа являются составными;
3. два потенциальных ключа перекрываются, т.е. имеют, по крайней мере, один общий атрибут.

Допустим, что при проектировании базы данных
ПОСТАВКИ_ТОВАРОВ рассматривается
отношение:

ПОСТАВКА (Индекс_поставщ, Имя_поставщ,
Индекс_товара, Колич_товара).



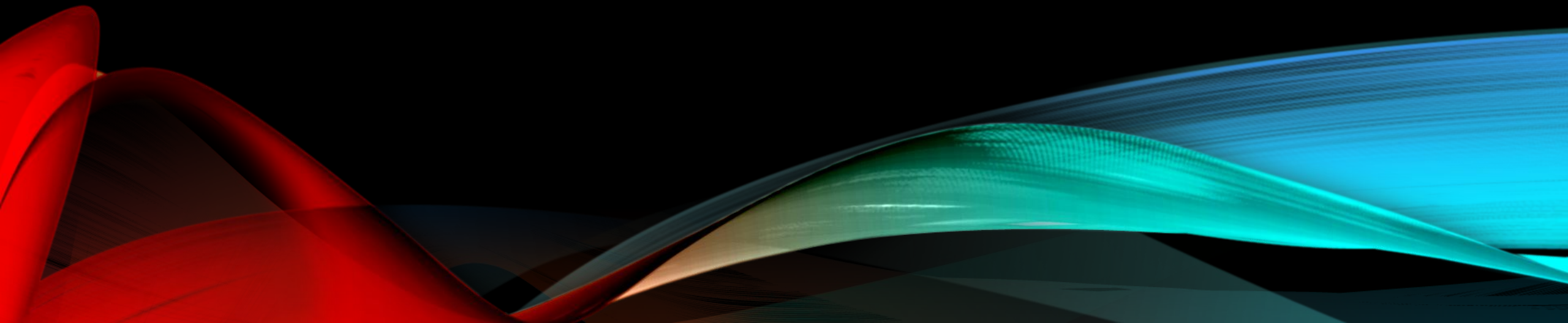


Допустим также, что значения атрибута **Имя_поставщ** уникальны и могут быть использованы наряду с атрибутом **Индекс_поставщ** для идентификации поставщика. В такой ситуации можно выделить два составных потенциальных ключа:

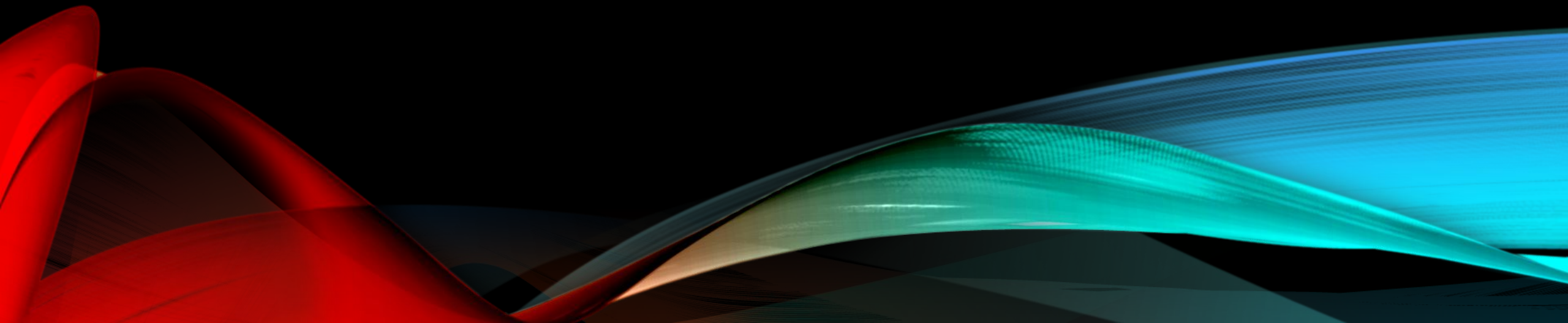
(**Индекс_поставщ, Индекс_товара**);

(**Имя_поставщ, Индекс_товара**).

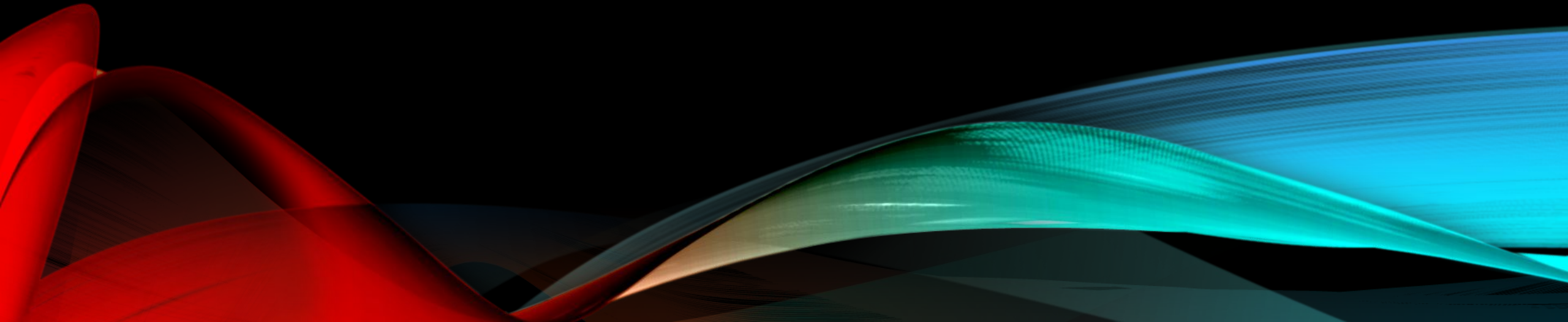
В рассматриваемом отношении есть два атрибута **Индекс_поставщ** и **Имя_поставщ**, которые идентифицируют один и тот же объект, а, значит, они определяют друг друга. В таком случае отношение содержит два детерминанта, но эти детерминанты не являются потенциальными ключами отношения.



Следовательно данное отношение не находится в НФБК со всеми вытекающими из этой ситуации негативными последствиями: избыточностью данных и аномалиями обновления.



Для рассматриваемого примера решение проблемы можно осуществить, разбив исходное отношение на два. Причем, поскольку два детерминанта **Индекс_поставщ** и **Имя_поставщ** определяют друг друга, то возможны два равносильных варианта декомпозиции, приводящей к НФБК.

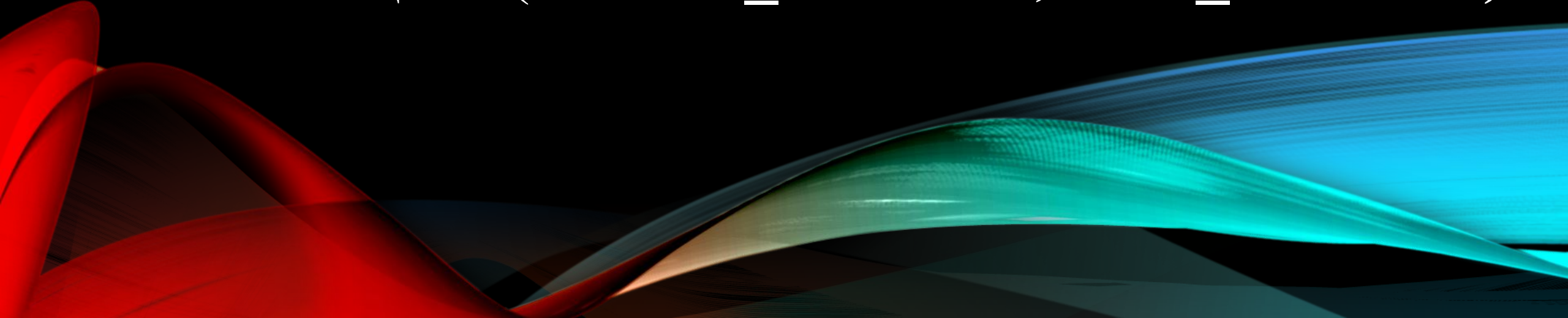


Первый вариант получается, если учитывается зависимость

Индекс_поставщ → Имя_поставщ, в результате чего имеем следующих два отношения:

ПОСТАВКА (Индекс_поставщ, Индекс_товара, Колич_товара);

ПОСТАВЩИК (Индекс_поставщ, Имя_поставщ).

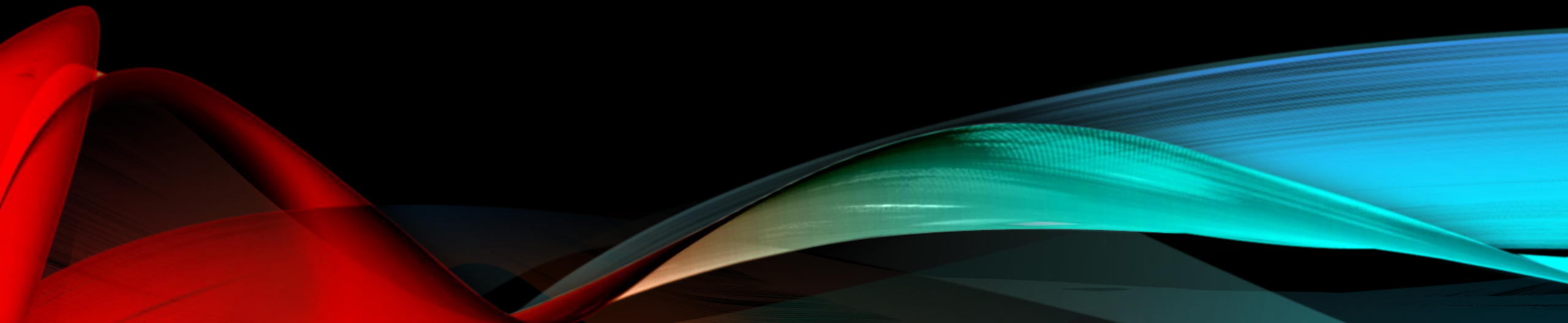


Второй вариант исходит из зависимости

Имя_поставщ → Индекс_поставщ, в результате чего получаем альтернативную группу отношений:

ПОСТАВКА (Имя_поставщ, Индекс_товара, Колич_товара);

ПОСТАВЩИК (Индекс_поставщ, Имя_поставщ).

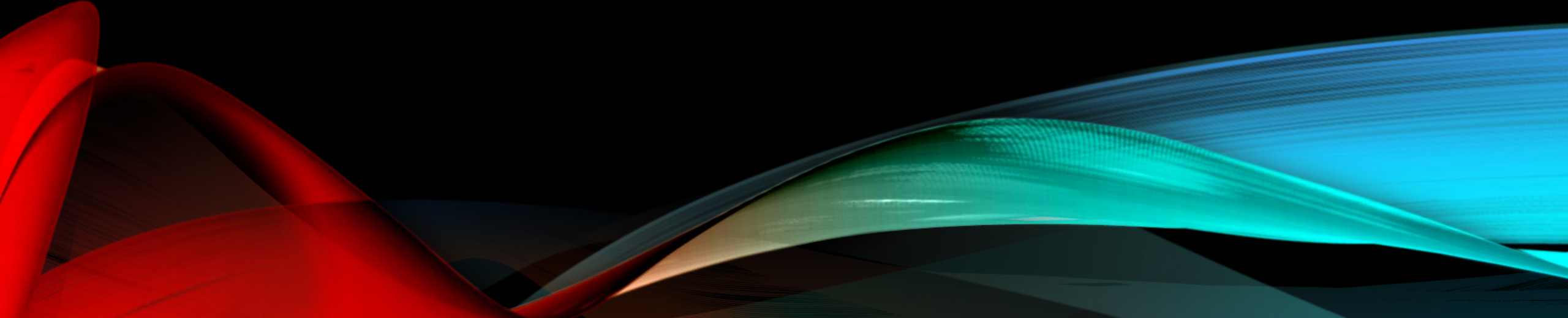


ЧЕТВЕРТАЯ НОРМАЛЬНАЯ ФОРМА

Еще один тип зависимостей — многозначная зависимость. Возможность существования в отношении многозначных зависимостей возникает в процессе приведения исходных таблиц к 1НФ. Первая нормальная форма запрещает отношениям иметь не атомарные, многозначные атрибуты. Однако существует множество ситуаций моделирования, требующих многозначных атрибутов.

Многозначность присутствует в тех отношениях, где моделируется связи типа 1:M. Если в отношении моделируется одна связь такого рода, то такая многозначность называется тривиальной.

Иначе, если в отношении представлены две связи типа 1:M. Рассмотрим следующую схему отношения: НИР (Номер_НИР, Сотр, Задан_НИР).

An abstract graphic at the bottom of the slide featuring flowing, wavy shapes in red and blue against a dark background.



Отношение НИР содержит номера тем научно-исследовательских работ, для каждой темы список сотрудников, которые могут выполнять работы по теме, и список заданий темы. Сотрудники могут участвовать в нескольких темах, и разные темы могут включать одинаковые задания. В такой ситуации единственным возможным ключом отношения является составной атрибут:

(Номер НИР, Сотр, Задан НИР).

В отношении НИР существуют следующие две многозначные зависимости:

Номер_НИР $\rightarrow$ > Сотр;

Номер_НИР $\rightarrow$ $\rightarrow$ Задан_НИР.

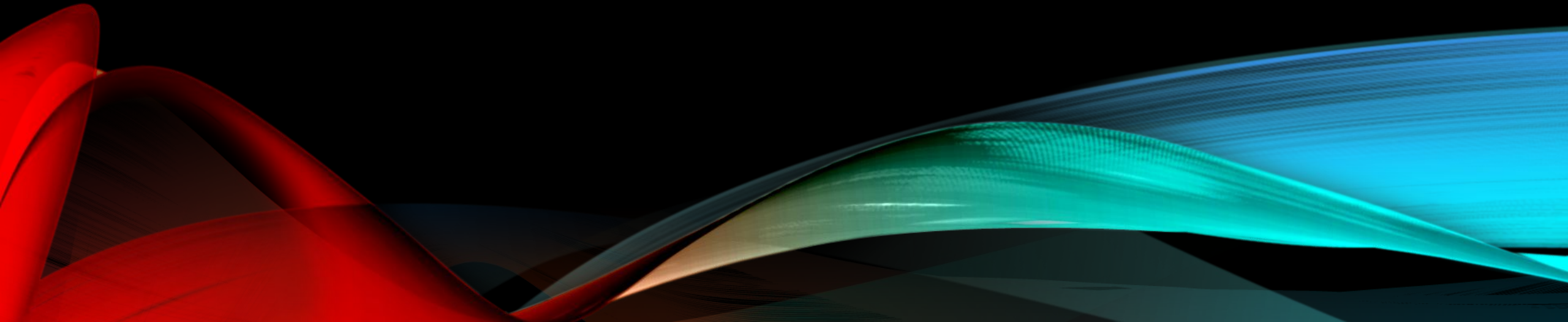
Условие, обеспечивающее независимость атрибутов путем обязательного повторения значений, называется многозначной зависимостью.

Отношение находится в четвертой нормальной форме (4НФ) тогда и только тогда, когда существуют такие подмножества A и B атрибутов отношения R , что выполняется нетривиальная многозначная зависимость $A \twoheadrightarrow B$. Тогда все атрибуты отношения R также функционально зависят от атрибута A .

В рассматриваемом примере можно произвести декомпозицию отношения НИР в два отношения НИР-СОТРУДНИКИ и НИР-ЗАДАНИЯ:

НИР-СОТРУДНИКИ (Номер_НИР, Сотр);

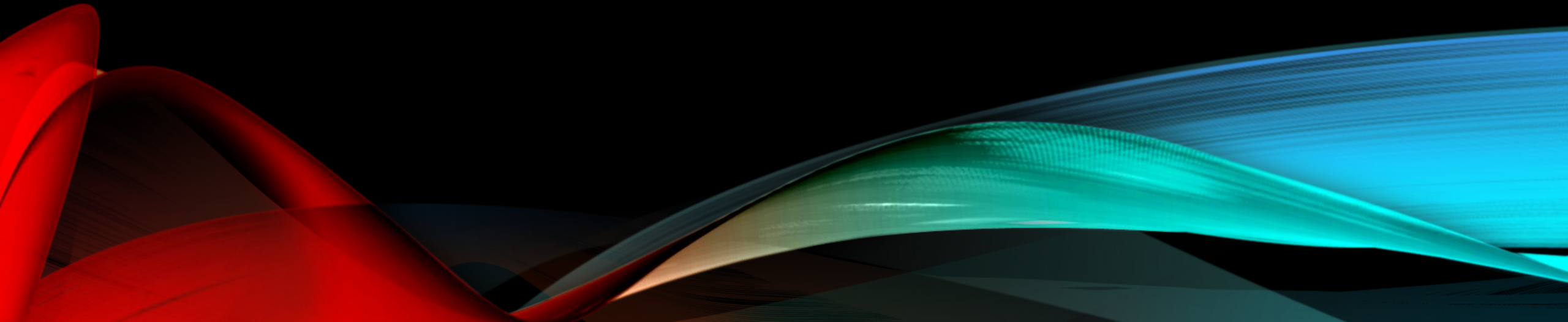
НИР-ЗАДАНИЯ (Номер_НИР, Задан_НИР).



ПЯТАЯ НОРМАЛЬНАЯ ФОРМА

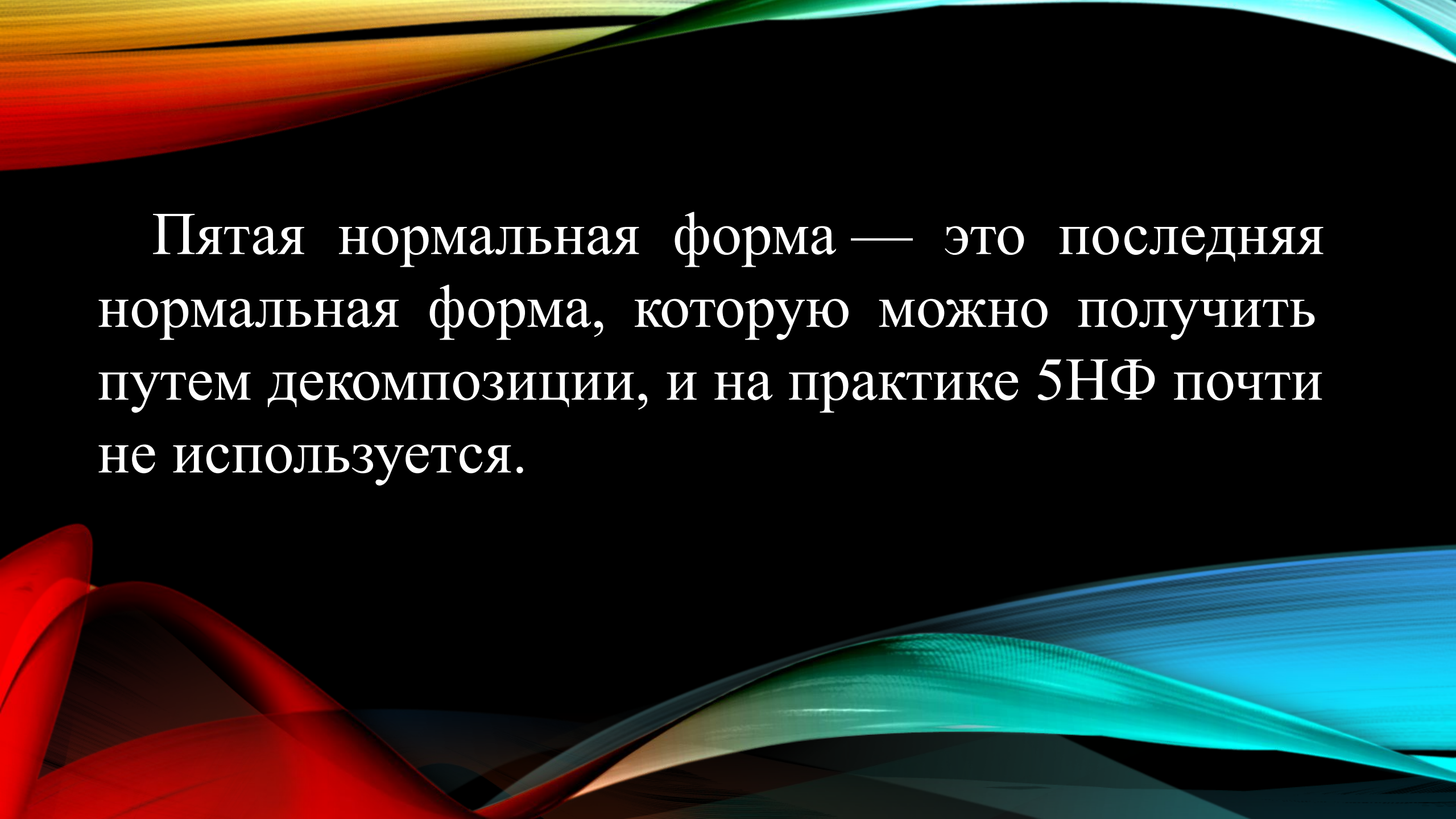
Во всех рассмотренных до этого момента ситуациях нормализация отношений производилась декомпозицией одного отношения на два. Иногда нормализовать отношение путем декомпозиции на два отношения без потерь не удастся, но просматривается возможность декомпозиции исходного отношения без потерь на большее число отношений, каждое из которых обладает лучшими свойствами.

Если в процессе естественного соединения декомпозированных отношений в сравнении с первоначальным отношением генерируются ложные кортежи, то такая декомпозиция характеризуется зависимостью соединения.



В отношении $R(X, Y, \dots, Z)$ отсутствует зависимость соединения $(X, Y, \dots, Z)$ в том и только в том случае, когда R восстанавливается без потерь путем соединения своих проекций на $X, Y, \dots, Z$.*

Отношение находится в пятой нормальной форме, если оно не содержит зависимостей соединения.



Пятая нормальная форма — это последняя нормальная форма, которую можно получить путем декомпозиции, и на практике 5НФ почти не используется.