

Севастопольский государственный университет
Кафедра «Информационные системы»

Управление данными

курс лекций

лектор:
ст. преподаватель кафедры ИС Абрамович А.Ю.



Лекция 7

РЕЛЯЦИОННАЯ МОДЕЛЬ ДАННЫХ

Связывание сущностей. Ограничения
целостности

Концепция транзакций

СВЯЗЫВАНИЕ СУЩНОСТЕЙ

При проектировании реальных БД информацию обычно размещают в нескольких сущностях. Сущности при этом связаны семантикой информации. **В реляционных СУБД для указания связей сущностей производят операцию их связывания. Многие СУБД при связывании сущностей автоматически выполняют контроль целостности вводимых в базу данных в соответствии с установленными связями. Это повышает достоверность хранимой в БД информации.**

Установление связи между сущностями облегчает доступ к данным. Связывание сущностей при выполнении таких операций, как поиск, просмотр, редактирование, выборка, обычно **обеспечивает возможность обращения к произвольным полям связанных записей. Это уменьшает количество явных обращений к таблицам данных и число манипуляций в каждой из них.**

СВЯЗЫВАНИЕ СУЩНОСТЕЙ

При связывании двух таблиц выделяют зависимую и независимую сущность. Логическое связывание таблиц производится с помощью ключей.

Существует два типа сущностей:

- **зависимая сущность** (dependent entity) – для определения экземпляра такой сущности необходимо сослаться на экземпляр независимой сущности, с которой связана зависимая сущность;
- **независимая сущность** (independent entity) – для определения экземпляра сущности нет необходимости ссылаться на другие сущности.

Ключ связи состоит из одного или нескольких полей, которые в данном случае называют **полями связи (ПС)**.

Суть связи состоит в установлении соответствия полей связи независимой и зависимой сущности.

КЛАССЫ СВЯЗЕЙ МЕЖДУ СУЩНОСТЯМИ

Между двумя сущностями в общем случае могут устанавливаться следующие **четыре основных класса связей**:

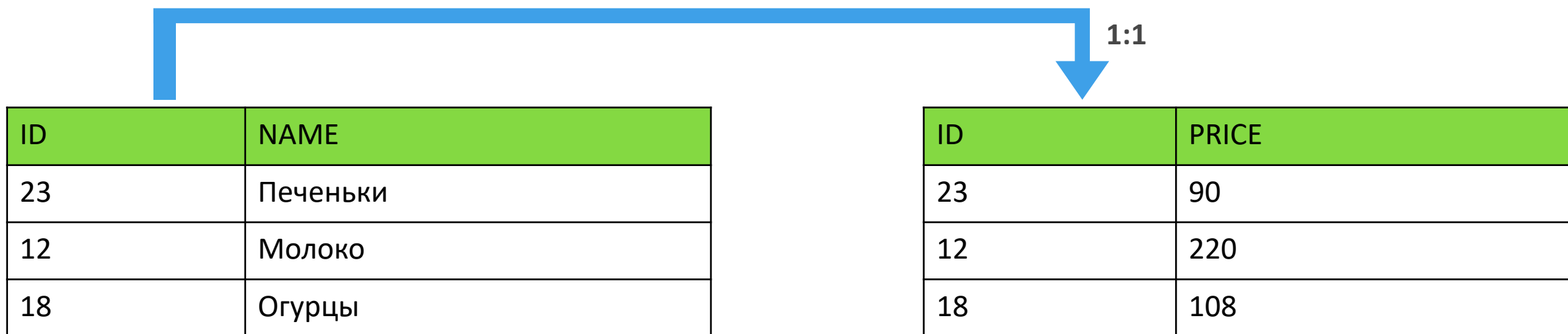
- один-к-одному (1:1);
- один-ко-многим (1:M);
- многие-к-одному (M:1);
- многие-ко-многим (M:M).

Характеристика видов связей таблиц

Характеристика полей связи	1:1	1:M	M:1	M:M
Поля связи независимой (основной) сущности	Являются ключом	Являются ключом	Не являются ключом	Не являются ключом
Поля связи зависимой (дочерней) сущности	Являются ключом	Не являются ключом	Являются ключом	Не являются ключом

СВЯЗЬ КЛАССА 1:1

Связь класса 1:1 образуется в случае, когда все поля связи основной и дочерней таблиц являются ключевыми.

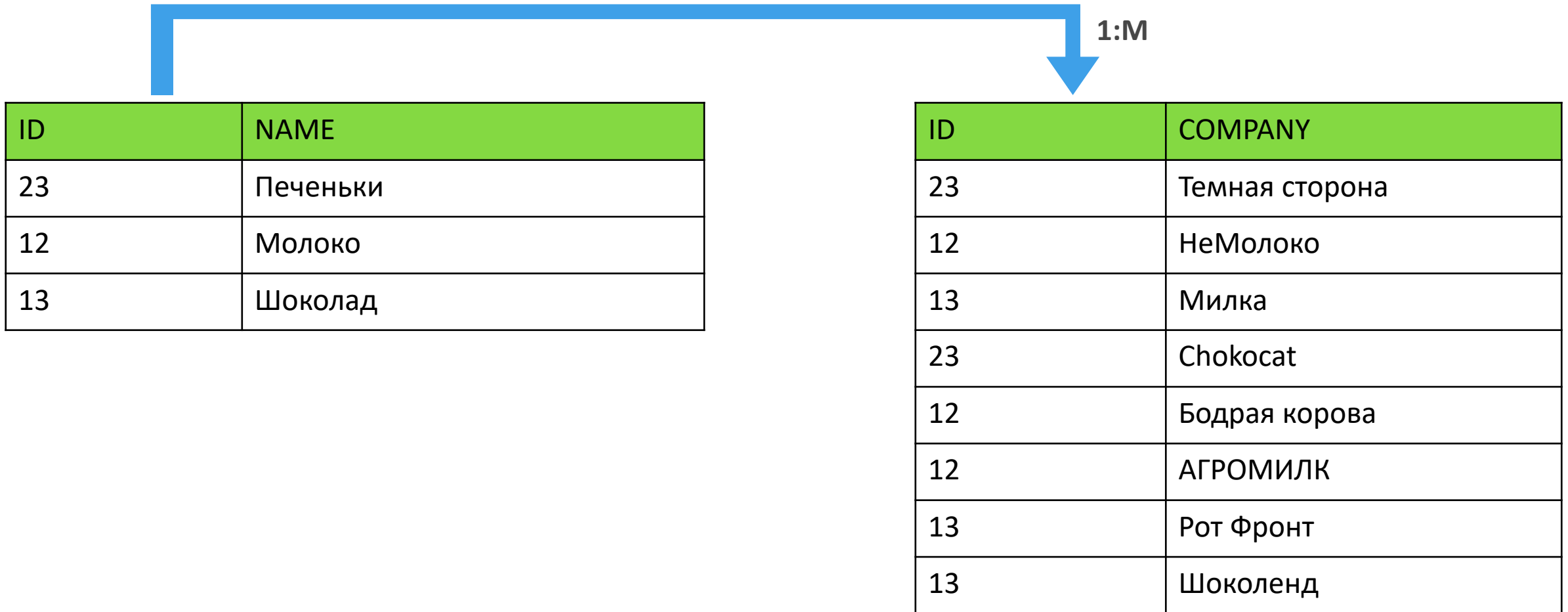


Поскольку значения в ключевых полях обеих таблиц не повторяются, **обеспечивается взаимнооднозначное соответствие записей из этих таблиц**. Сами таблицы равноправны.

На практике такая связь используется редко, так как хранимую в двух таблицах информацию легко объединить в одну таблицу, которая занимает гораздо меньше памяти.

СВЯЗЬ КЛАССА 1:M

Связь класса 1:M образуется в случае, когда одной записи основной таблицы соответствует несколько записей дочерней таблицы.



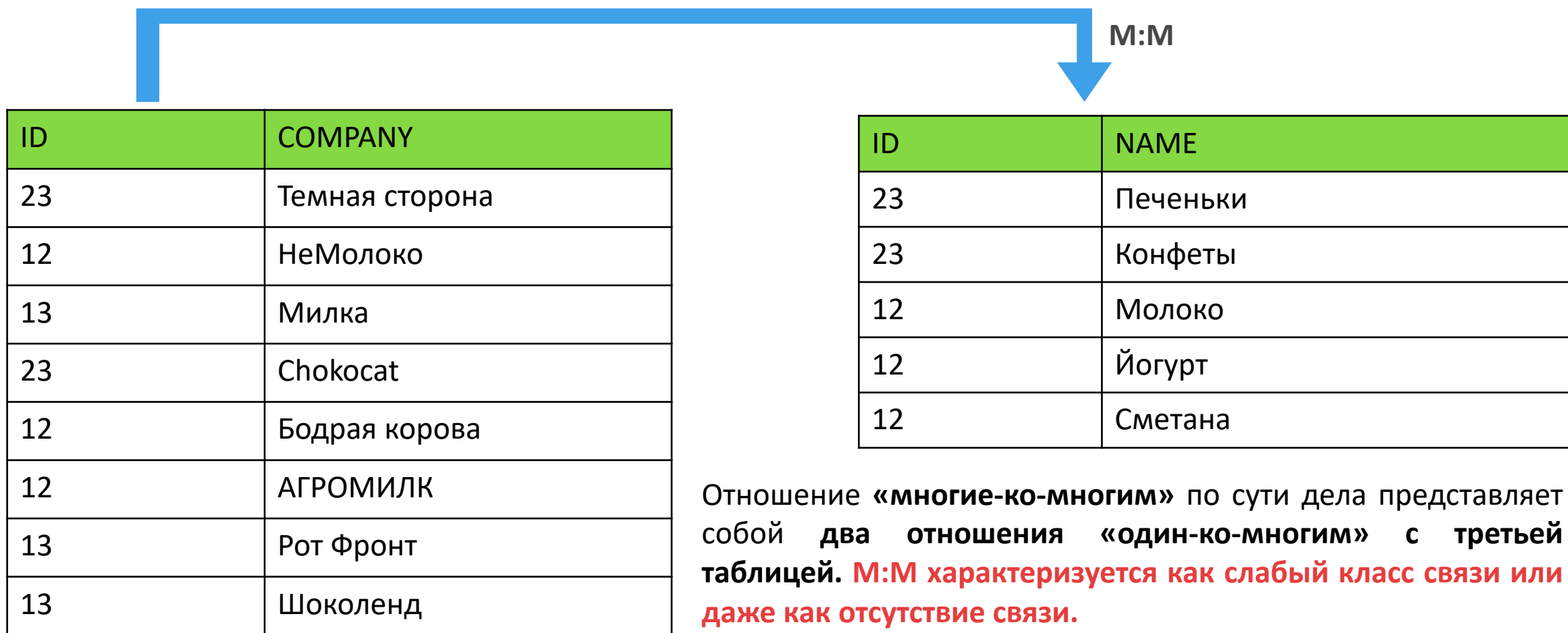
СВЯЗЬ КЛАССА M:1

Связь класса M:1 образуется в случае, когда одной или несколькими записями основной таблицы ставится в соответствие одна запись дополнительной таблицы.



СВЯЗЬ КЛАССА М:М

Связь класса М:М образуется в случае, когда нескольким записям основной таблицы соответствует несколько записей дополнительной таблицы.



СПОСОБ ОПРЕДЕЛЕНИЯ СВЯЗЕЙ МЕЖДУ ТАБЛИЦАМИ

Тип отношения в создаваемой связи зависит от способа определения связываемых полей:

- отношение «**один-ко-многим**» создается в том случае, когда **только одно из полей является полем первичного ключа или уникального индекса;**
- отношение «**один-к-одному**» создается в том случае, когда **оба связываемых поля являются ключевыми или имеют уникальные индексы;**
- отношение «**многие-ко-многим**» фактически является **двумя отношениями «один-ко-многим» с третьей таблицей**, первичный ключ которой состоит из полей внешнего ключа двух других таблиц.

ОГРАНИЧЕНИЯ ЦЕЛОСТНОСТИ

Обеспечение целостности данных **гарантирует их системную полноту, адекватность модели данных моделируемой предметной области.**

Ограничения целостности представляют собой условия, которые определяют допустимые отношения между элементами структуры данных. Для конкретной модели данных эти условия выполняются или не выполняются. **Средством спецификации ограничений целостности является язык математической логики.**

По происхождению ограничения целостности, принято различать:

- **внутренние**, обусловленные особенностью типа структуры данных, в частности отношения;
- **семантические (явные)**, обусловленные смыслом, значением взаимосвязанных данных конкретной предметной области.

По способу контроля целостности данных, который осуществляет СУБД ограничения целостности, принято различать:

- **безотлагательные**, проверка которых осуществляется непосредственно **в процессе манипулирования данными;**
- **отложенные**, проверка которых совершается **по завершении всех манипуляций** со связанными таблицами.

ВНУТРЕННИЕ ОГРАНИЧЕНИЯ ЦЕЛОСТНОСТИ

Внутренние ограничения целостности данных реляционной модели **обусловлены свойствами отношения** по определению как множества:

- у отношения не может быть одинаковых строк-кортежей;
- порядок следования кортежей значения не имеет;
- порядок следования атрибутов значения не имеет;
- все значения атрибутов атомарны (неделимы).

Внутренние ограничения целостности являются безотлагательными. Их **проверка** реляционной СУБД **выполняется автоматически.**

Из **определения внутренних ограничений** целостности следует, что **две и более таблиц могут считаться эквивалентными** при выполнении следующих условий:

- таблицы имеют **одинаковое количество столбцов**;
- таблицы содержат **столбцы с одинаковыми наименованиями**;
- **столбцы с одинаковыми наименованиями** содержат **данные из одних и тех же доменов**;
- **таблицы имеют одинаковые строки** с учетом того, что порядок столбцов может различаться.

СЕМАНТИЧЕСКИЕ ОГРАНИЧЕНИЯ ЦЕЛОСТНОСТИ

Семантические ограничения целостности (или семантические условия) присущи **самой предметной области** и их учет основан на **понимании смыслового содержания данных**. Это свойства данных, которые выполняются или не выполняются для рассматриваемого отношения элементов данных. Семантические ограничения целостности **для каждой предметной области будут свои**, что представляет трудности для аналитика.

Семантические ограничения целостности различают:

- ограничения объектов;
- ограничения на ссылки (связи) между объектами.

В реляционном представлении **ограничения целостности объектов** реализуются в виде ограничений **целостности доменов, атрибутов и отношений**.

Ограничения **ссылочной целостности или ограничения**, основанные на связях между отношениями-объектами, проявляются **как ограничения целостности реляционной базы данных в целом**.

Ограничения целостности домена (атрибута)

Каждый атрибут определяется на своем домене, или наоборот: домен атрибута задает множество значений, которые может принимать атрибут.

Сами домены никогда не обновляются в процессе функционирования базы данных, поэтому ограничения целостности доменов не проверяются. Обновляются значения атрибутов, определенных на доменах. Поэтому проверяются ограничения целостности атрибутов на предмет соответствия типу данных или множеству допустимых значений, причем проверяются всегда и немедленно.

Ограничения целостности отношения

Ограничения целостности отношения базируются на понятиях функциональных зависимостей атрибутов, потенциальных ключей, неопределенных Null-значениях атрибутов и др.

Правило целостности отношений: атрибуты, входящие в состав некоторого потенциального ключа, не могут принимать Null-значений. Ограничения целостности отношений, как и ограничения целостности атрибутов, являются безотлагательными.

Ограничения ссылочной целостности

Бизнес-правила использования данных конкретной предметной области выражаются ограничениями ссылочной целостности. Ограничения **ссылочной целостности** — это правила, **контролирующие корректность связей между отношениями.**

Контроль целостности связей обычно означает анализ содержимого двух таблиц на соблюдение следующих правил:

- каждой записи основного отношения соответствует NULL или более записей дочернего отношения;
- в дочернем отношении нет записей, которые не имеют родительских записей в основном отношении;
- каждая запись дочернего отношения имеет только одну родительскую запись основного отношения.

К нарушению ссылочной целостности могут привести:

- обновление кортежа в родительском отношении;
- удаление кортежа в родительском отношении;
- вставка кортежа в дочернее отношение;
- обновление кортежа в дочернем отношении.

Ограничения ссылочной целостности

При обновлении кортежа в родительском отношении

запрет — не разрешается обновление первичного ключа в кортеже, если имеется хотя бы один кортеж в отношении-потомке, ссылающийся на обновляемый кортеж;

каскадирование — разрешается изменять значение первичного ключа в кортеже родительского отношения, если в отношении-потомке есть ссылающиеся на него кортежи. При этом **будут изменены «по каскаду» значения внешних ключей во всех кортежах отношений-потомков**, ссылающихся на обновляемый кортеж;

установка в Null — актуальна для неидентифицирующих связей, в случае обновления первичного ключа в кортеже родительского отношения внешний ключ кортежей отношения-потомка может быть установлен в Null.

При удалении кортежа в родительском отношении

запрет — не разрешается удаление кортежа, если имеется хотя бы один кортеж в отношении-потомке, ссылающийся на удаляемый кортеж;

каскадирование — разрешается удаление кортежа родительского отношения, если в отношении-потомке есть ссылающиеся на него кортежи. При этом **будут удалены «по каскаду» значения внешних ключей во всех кортежах отношений-потомков**, ссылающихся на обновляемый кортеж;

установка в Null — актуальна для неидентифицирующих связей, в этом случае разрешается удаление кортежа родительского отношения, но во всех кортежах отношения-потомка, ссылающихся на удаляемый кортеж, внешний ключ может быть установлен в Null.

Ограничения ссылочной целостности

При вставке
кортежа в
отношение-
потомок

запрет — не разрешается вставка, если внешний ключ во вставляемом кортеже не соответствует ни одному значению первичного ключа родительского отношения;

установка в Null — актуальна для неидентифицирующих связей, в этом случае разрешается вставка кортежа, но в качестве значения внешнего ключа отношения-потомка следует установить Null-значение.

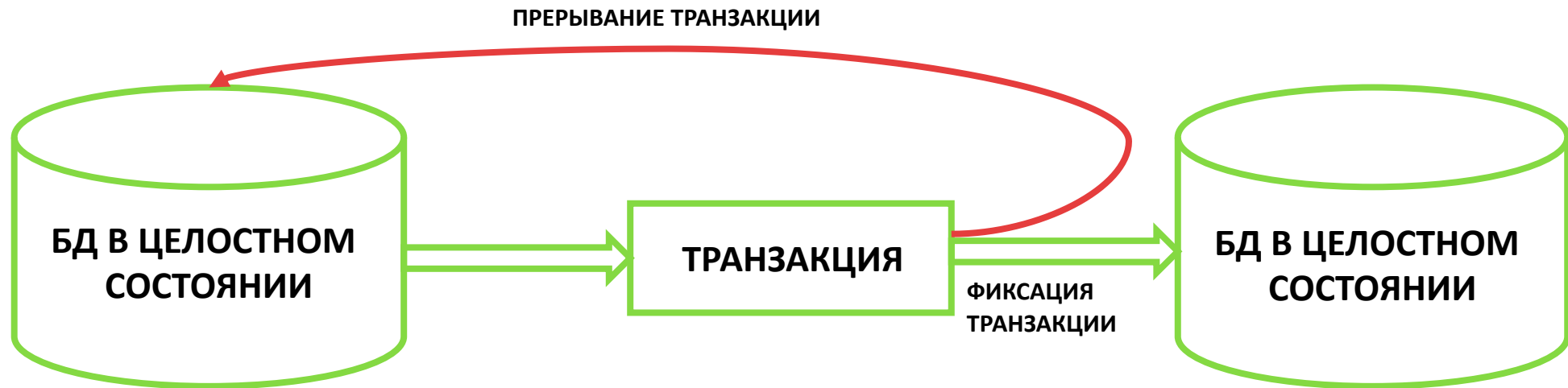
При
обновлении
кортежа в
отношении-
потомок

запрет — не разрешается обновление, если внешний ключ в обновляемом кортеже становится не соответствующим ни одному значению первичного ключа родительского отношения;

установка в Null — актуальна для неидентифицирующих связей, в этом случае разрешается обновить кортеж, но в качестве значения внешнего ключа отношенияпотомка следует установить Null-значение.

КОНЦЕПЦИЯ ТРАНЗАКЦИЙ

Под **транзакцией** понимается неделимая с точки зрения воздействия на БД последовательность операторов манипулирования данными (чтения, удаления, вставки, модификации), приводящая к одному из двух возможных результатов: либо последовательность выполняется, если все операторы правильные, либо вся транзакция прерывается, если хотя бы один оператор не может быть успешно выполнен. Обработка транзакций гарантирует целостность информации в базе данных. **Транзакция переводит базу данных из одного целостного состояния в другое.**



КОНЦЕПЦИЯ ТРАНЗАКЦИЙ

Корректное поддержание транзакций является основой обеспечения целостности БД. Транзакции также составляют основу изолированности в многопользовательских системах, где с одной БД параллельно могут работать несколько пользователей или прикладных программ. Такую задачу СУБД принято называть **параллелизмом транзакций**.

Большинство выполняемых действий производится в теле транзакций. По умолчанию **каждая команда выполняется как самостоятельная транзакция**. При необходимости пользователь может явно указать ее начало и конец, чтобы иметь возможность включить в нее несколько команд.

При выполнении транзакции система управления базами данных должна придерживаться определенных правил обработки набора команд, входящих в транзакцию.

ACID-свойства транзакций

Разработано четыре правила, известные как требования **ACID** (**A**tomicity, **C**onsistency, **I**solation, **D**urability – **неделимость, согласованность, изолированность, устойчивость**), они гарантируют **правильность и надежность работы системы.**

Транзакция неделима
в том смысле, что
представляет собой
единое целое.

Все ее компоненты
либо имеют место,
либо нет.

Не бывает частичной
транзакции.

Если может быть
выполнена лишь часть
транзакции, она
отклоняется.

Транзакция является
согласованной, потому
что не нарушает
бизнес-логику и
отношения между
элементами данных.

Если хотя бы одна
транзакция нарушит
целостность данных,
то все остальные могут
выдать неверные
результаты.

Транзакция всегда
изолирована,
поскольку ее
результаты
самодостаточны.

Они не зависят от
предыдущих или
последующих
транзакций.

Это свойство
называется
сериализуемостью –
транзакции в
последовательности
независимы.

Транзакция устойчива.

После своего
завершения она
сохраняется в системе,
которую ничто не
может вернуть в
исходное состояние,
т.е. **происходит**
фиксация транзакции,
означающая, что ее
действие постоянно
даже при сбое
системы.

УПРАВЛЕНИЕ ТРАНЗАКЦИЯМИ

Под **управлением транзакциями** понимается способность управлять различными операциями над данными, которые выполняются внутри реляционной СУБД. Прежде всего, имеется в виду **выполнение операторов *INSERT, UPDATE и DELETE***.

После успешного выполнения команд, заключенных в тело одной транзакции, немедленного изменения данных не происходит. Для окончательного завершения транзакции существуют команды управления транзакциями, с помощью которых можно либо сохранить в базе данных все изменения, произошедшие в ходе ее выполнения, либо полностью их отменить (***COMMIT, ROLLBACK, SAVEPOINT***).

После завершения транзакции вся информация о произведенных изменениях хранится либо в специально выделенной оперативной памяти, либо во временной области отката в самой базе данных до тех пор, пока не будет выполнена одна из команд управления транзакциями. Затем все изменения или фиксируются в базе данных, или отбрасываются, а временная область отката освобождается.