

Ограничения базы данных

Определение 12. Ограничения целостности базы данных представляют ограничения, накладываемые на значения двух или более связанных между собой отношений (в том числе отношение может быть связано само с собой).

Пример 13. Ограничение целостности ссылок, задаваемое внешним ключом отношения, является ограничением базы данных.

Пример 14. Ограничение на таблицы DEPART и PERSON из примера выше является отношением базы данных, т.к. оно связывает данные, размещенные в различных таблицах.

Проверка ограничения. К моменту проверки ограничения базы данных должны быть проверены ограничения целостности отношений.

Ограничение базы данных может быть как *немедленно проверяемым* ограничением, так и ограничением с *отложенной проверкой*.

Ограничение отношения, являющееся ограничением внешнего ключа может быть как немедленно проверяемым ограничением, так и отложенным ограничением. Действительно, в простейшем случае, если кортеж X отношения A должен ссылаться на кортеж Y отношения B , то проверку ограничения ссылочной целостности можно производить *сразу* после любой из операций вставки, обновления или удаления в любом из отношений A или B . В более сложном случае, предположим, что кортеж X отношения A должен ссылаться на кортеж Y отношения B , а кортеж Y отношения B должен в свою очередь ссылаться на кортеж X отношения A (циклическая ссылка). Очевидно, что сразу после вставки кортежа X отношение A ссылочная целостность *обязательно* нарушена, т.к. кортежа Y *еще нет* в отношении B . Проверку ссылочной целостности можно провести только посл завершения транзакции, состоящей из последовательности операций:

1. вставки кортежа X в отношение A с нулевой ссылкой на отношение B ,
2. вставки кортежа Y в отношение B со ссылкой на кортеж X отношения A ,
3. исправления ссылки в кортеже X с NULL на ссылку на кортеж Y .

Ограничение, приведенное в примере, может быть только ограничением с отложенной проверкой.

Реализация декларативных ограничений целостности средствами SQL

Общие принципы реализации ограничений средствами SQL

Стандарт SQL *не предусматривает* процедурных ограничений целостности, реализуемых при помощи триггеров и хранимых процедур. В стандарте SQL отсутствует понятие "триггер", хотя триггеры имеются во всех промышленных СУБД SQL-типа. Таким образом, реализация ограничений средствами конкретной СУБД обладает большей гибкостью, нежели с использованием исключительно стандартных средств SQL.

Стандарт SQL позволяет задавать декларативные ограничения следующими способами:

- Как ограничения домена.
- Как ограничения, входящие в определение таблицы.
- Как ограничения, хранящиеся в базе данных в виде независимых утверждений (assertion).

Допускаются как немедленно проверяемые, так и ограничения с отложенной проверкой. Режим проверки отложенных ограничений можно в любой момент изменить так, чтобы ограничение проверялось:

1. После исполнения *каждого оператора*, изменяющего содержимое таблицы, к которой относится данное ограничение.
2. При завершении *каждой транзакции*, включающей операторы, изменяющие содержимое таблиц, к которым относятся данное ограничение.
3. В *любой промежуточный момент*, если пользователь инициирует проверку.

При определении ограничения указывается тип проверки ограничения - является ли это ограничение *неоткладываемым* (**NOT DEFERRED**) или может быть *откладываемым* (**DEFERRED**). Во втором случае можно задать процедуру по умолчанию: *проверять немедленно* или *проверять по завершении транзакции*. Таким образом, можно определить

потенциально откладываемое ограничение, которое по умолчанию проверяется немедленно. В любой момент режим проверки такого ограничения можно изменить на отложенный и наоборот. Режим проверки может быть изменен для одного ограничения или сразу для всех потенциально откладываемых ограничений. Если ограничение определено как неоткладываемое, то тип такого ограничения *изменить нельзя* и ограничение всегда проверяется немедленно.

Элементы процедурности все же присутствуют в стандарте SQL в виде так называемых **действий, исполняемых по ссылке** (*referential triggered actions*). Эти действия определяют, что будет происходить при изменении значения родительского ключа, на который ссылается некоторый внешний ключ. Эти действия можно задавать независимо для операций обновления (ON UPDATE) или для операций удаления (ON DELETE) записей в родительском отношении. Стандартом SQL определяется 4 типа действий, исполняемых по ссылке:

- **CASCADE**. Изменения значения родительского ключа автоматически приводят к таким же изменениям связанного с ним значения внешнего ключа. Удаление кортежа в родительском отношении приводит к удалению связанных с ним кортежей в дочернем отношении.
- **SET NULL**. Все внешние ключи, которые ссылаются на обновленный или удаленный родительский ключ получают значения NULL.
- **SET DEFAULT**. Все внешние ключи, которые ссылаются на обновленный или удаленный родительский ключ получают значения, принятые по умолчанию для этих ключей.
- **NO ACTION**. Значения внешнего ключа не изменяются. Если операция приводит к нарушению ссылочной целостности (появляются "висящие" ссылки), то такая операция не выполняется.

Как видно, действия, исполняемые по ссылке, фактически являются встроенными в СУБД триггерами. Действия типа CASCADE, SET NULL и SET DEFAULT являются компенсирующими операциями, вызываемыми при попытке нарушить ссылочную целостность.

Синтаксис ограничений стандарта SQL

Понятие ограничения используется во многих операторах определения данных (DDL).

Ограничение *check*::=

CHECK Предикат

Ограничения таблицы ::=

[**CONSTRAINT** Имя ограничения]

{

{**PRIMARY KEY** (Имя столбца,...)}

| {**UNIQUE** (Имя столбца,...)}

| {**FOREIGN KEY** (Имя столбца,...) **REFERENCES** Имя таблицы [(Имя столбца,...)]

[Ссылочная спецификация]}

| { Ограничение *check* }

}

[Атрибуты ограничения]

Ограничения столбца::=
[CONSTRAINT Имя ограничения]
{
{NOT NULL}
| {PRIMARY KEY}
| {UNIQUE}
| {REFERENCES Имя таблицы [(Имя столбца)] [Ссылочная спецификация]}
| { Ограничение check }
}
[Атрибуты ограничения]

Ссылочная спецификация::=
[MATCH {FULL | PARTIAL}]
[ON UPDATE {CASCADE | SET NULL | SET DEFAULT | NO ACTION}]
[ON DELETE {CASCADE | SET NULL | SET DEFAULT | NO ACTION}]

Атрибуты ограничения::=
{DEFERRABLE [INITIALLY DEFERRED | INITIALLY IMMEDIATE]}
| {NOT DEFERRABLE}

Ограничение типа CHECK. Ограничение типа CHECK содержит предикат, могущий принимать значения TRUE, FALSE и UNKNOWN (NULL). Примеры предикатов различного вида приведены в главе 5. Ограничение типа CHECK может быть использовано как часть описания домена, таблицы, столбца таблицы или отдельного ограничения целостности - ASSERTION. Ограничение считается нарушенным, если предикат ограничения принимает значение FALSE.

Пример 15. Пример ограничения типа CHECK:

CHECK (Salespeople.Salary IS NOT NULL) OR (Salespeople.Commission IS NOT NULL)

Данное ограничение утверждает, что каждый продавец должен иметь либо ненулевую зарплату, либо ненулевые комиссионные.

Пример 16. Еще пример ограничения типа CHECK:

CHECK EXIST(SELECT * FROM Salespeople)

Данное ограничение утверждает, что список продавцов не может быть пустым.

Ограничения таблицы и ограничения столбца. Ограничения таблицы и ограничения столбца таблицы входят как часть описания соответственно таблицы или столбца таблицы. Ограничение таблицы может относиться к нескольким столбцам таблицы. Ограничение столбца относится только к одному столбцу таблицы. Любое ограничение столбца можно описать как ограничение таблицы, но не наоборот.

Ограничения таблицы или столбца могут иметь наименования, при помощи которого в дальнейшем можно отменять это ограничение или менять время его проверки.

Ограничение PRIMARY KEY. Ограничение PRIMARY KEY для таблицы или столбца означает, что группа из одного или нескольких столбцов образуют потенциальный ключ таблицы. Это означает, что комбинация значений в PRIMARY KEY должна быть

уникальной для каждой строки таблицы. Дублированные значения или значения, содержащие NULL, будут отвергнуты. Для одной таблицы может быть определено единственное ограничение PRIMARY KEY. В терминах стандарта SQL это называется первичным ключом таблицы.

Ограничение UNIQUE. Ограничение UNIQUE для таблицы или столбца означает, что группа из одного или нескольких столбцов образуют потенциальный ключ таблицы, в котором допускаются значения NULL. Это означает, что две строки, содержащие одинаковые и не равные NULL-значения, считаются нарушающими уникальность и не допускаются. Две строки, содержащие NULL-значения *считаются различными* и допускаются. Для одной таблицы может быть определено несколько ограничений UNIQUE.

Замечание. С точки зрения реляционной модели данных (правила целостности сущностей и внешних ключей), ограничение типа UNIQUE не определяет потенциальный ключ, т.к. потенциальный ключ не должен содержать NULL-значений.

Ограничения FOREIGN KEY и REFERENCES. Ограничение FOREIGN KEY... REFERENCES... для таблицы и ограничение REFERENCES... для столбца определяют внешний ключ таблицы. Ограничение REFERENCES... для столбца определяет простой внешний ключ, т.е. ключ, состоящий из одной колонки. Ограничение FOREIGN KEY... REFERENCES... для таблицы может определять как простой, так и сложный внешний ключ, т.е. ключ, состоящий из нескольких колонок таблицы. Столбец или группа столбцов таблицы, на которую ссылается внешний ключ, должна иметь ограничения PRIMARY KEY или UNIQUE. Столбцы, на которые ссылается внешний ключ, должны иметь тот же тип данных, что и столбцы, входящие в состав внешнего ключа. Таблица может иметь ссылку на себя. Ограничение внешнего ключа нарушается, если значения, присутствующие во внешнем ключе, не совпадают со значениями соответствующего ключа родительской таблицы ни для одной строки из родительской таблицы. Операции, приводящие к нарушению ограничения внешнего ключа, отвергаются. Как должны совпадать значения внешнего ключа и ключа родительской таблицы, а также, какие действия необходимо выполнить при изменениях ключей в родительской таблице, описаны ниже в ссылочной спецификации.

Ограничение NOT NULL. Ограничение NOT NULL столбца не допускает появления в столбце NULL-значений.

Ссылочная спецификация. Ссылочная спецификация определяет характеристики внешнего ключа таблицы.

Предложение MATCH {FULL / PARTIAL}. Предложение MATCH FULL требует полного совпадения значений внешнего и первичного ключей. Предложение MATCH PARTIAL допускает частичное совпадение значений внешнего и первичного ключей. Предложение MATCH может быть также пропущенным. Для случая MATCH PARTIAL в дочерней таблице могут появиться строки, имеющие значения внешнего ключа, *неуникально* совпадающие со значениями родительского ключа. Т.е. одна строка дочерней таблицы может иметь неуникальные ссылки на несколько строк родительской таблицы. Это очень сильно отличается от реляционной модели данных, и это отличие провоцируется допущением NULL-значений. Чтобы рассмотреть различные варианты совпадений внешнего и родительского ключей, рассмотрим следующий пример.

Предложения ON UPDATE и ON DELETE. Предложения ON UPDATE и ON DELETE определяют действия, исполняемые по ссылке. Действия, исполняемые по ссылке, в основном описаны выше в этой главе. Сложности в понимании того, как выполняются эти действия, возникают если установлено MATCH PARTIAL и колонки, входящие в состав внешнего ключа, допускают NULL-значения.

Атрибуты ограничения. Атрибуты ограничения определяют, в какой момент проверяются ограничения. Ограничение может быть определено как NOT DEFERRABLE (неоткладываемое) или DEFERRABLE (откладываемое). Если атрибуты ограничения не указаны, то по умолчанию принимается NOT DEFERRABLE.

Если ограничение определено как NOT DEFERRABLE (*неоткладываемое*), то ограничение всегда проверяется *сразу после выполнения каждого оператора* INSERT, UPDATE или DELETE, которые могут привести к нарушению ограничения.

Если ограничение определено как DEFERRABLE (*откладываемое*), то ограничение *может иметь два режима проверки* - немедленно после выполнения операции или в конце транзакции. Режим проверки может быть изменен в любой момент внутри транзакции командой SET CONSTRAINTS. При определении ограничения можно указать начальный режим проверки INITIALLY DEFERRED (начально отложенное) или INITIALLY IMMEDIATE (начально немедленно проверяемое).

Синтаксис операторов SQL, использующих ограничения

Стандарт SQL описывает следующие операторы, в которых может быть использованы ограничения:

- CREATE DOMAIN - создать домен
- ALTER DOMAIN - изменить домен
- DROP DOMAIN - удалить домен
- CREATE TABLE - создать таблицу
- ALTER TABLE - изменить таблицу
- DROP TABLE - удалить таблицу
- CREATE ASSERTION - создать утверждение
- DROP ASSERTION - удалить утверждение
- COMMIT WORK - зафиксировать транзакцию
- SET CONSTRAINTS - установить момент проверки ограничений