

Базы данных

Тема 11

СУБД SQL Server 2012 – Триггеры.

Триггеры

- *триггером* называют хранимую процедуру особого типа, которая автоматически выполняется при возникновении языкового события;
- триггеры являются одним из двух механизмов реализации бизнес-правил и целостности данных (наряду с ограничениями), поддерживаемых SQL Server 2012;
- типы триггеров:
 - DML-триггер: действие, которое выполняется при наступлении события языка DML (инструкции UPDATE, INSERT и DELETE, выполняемые в таблице или представлении) на сервере базы данных - используются для расширения бизнес-правил при изменении данных, а также для расширения логики проверки целостности ограничений, правил и значений по умолчанию SQL Server 2012;
 - DDL-триггер: действие, которое выполняется при возникновении событий языка определения данных (DDL) на сервере баз данных;
- триггер и инструкция, при выполнении которой он срабатывает, считаются одной транзакцией, которую можно откатить назад внутри триггера;
- варианты реализации:
 - Transact-SQL,
 - CLR.

Триггеры DML: варианты использования

- для каскадных изменений в связанных таблицах базы данных; (данные изменения можно более эффективно выполнить при помощи каскадных ограничений ссылочной целостности);
- для предотвращения случайных или некорректных операций INSERT, UPDATE и DELETE;
- для реализации других более сложных ограничений, чем определенных при помощи ограничения CHECK (в отличие от ограничений CHECK, DML-триггеры могут ссылаться на столбцы других таблиц);
- для оценки состояния и анализа таблицы до и после изменения данных и выполнения действия на основе этого различия;
- несколько DML-триггеров одинакового типа (INSERT, UPDATE или DELETE) для таблицы позволяют предпринять несколько различных действий в ответ на одну инструкцию изменения данных;
- для реализации более сложных методов управления ошибками (по сравнению со стандартными системными сообщениями);
- для определения необходимых действий по обновлению представления (расширение типов обновлений, которые поддерживает представление).

Типы триггеров DML

- триггеры AFTER (или FOR):
 - могут применяться только в таблицах;
 - выполняются после выполнения инструкций INSERT, UPDATE или DELETE;
 - выполняются после проверки ограничений;
 - может быть более одного для каждого действия (UPDATE, DELETE или INSERT), можно задать выполнение в первую и в последнюю очередь;
 - не могут работать со столбцами типа `text`, `ntext` и `image`;
- триггеры INSTEAD OF:
 - используются в таблицах и представлениях (с одной или более базовыми таблицами);
 - выполняются вместо обычных действий, запускающих триггеры (например, дополнительная обработка может предотвращать нарушения ограничений), при повторном вызове действия триггер не срабатывает;
 - выполняются до проверки ограничений;
 - единственный триггер на каждое из запускающих триггеры действие (UPDATE, DELETE или INSERT);
 - не допускаются в таблицах, в которых используется каскадное обновление (для UPDATE и DELETE);
 - могут работать со столбцами типа `text`, `ntext` и `image`.

Специальные средства, доступные в триггерах

- таблицы **`inserted`** и **`deleted`**
(соответственно, вставленных и удаленных значений), формат таблиц соответствует формату таблицы или представления, для которых задан триггер;
- функция **`COLUMNS_UPDATED()`** возвращает битовую маску обновленных столбцов;
- функция **`UPDATE(column_name)`** возвращает **`TRUE`**, если столбец был обновлен.

Создание триггера DML

```
CREATE TRIGGER trigger_name
  ON { table | view }
  { FOR | AFTER | INSTEAD OF }
  { [ DELETE ] [ , ] [ INSERT ] [ , ] [ UPDATE ] }
AS
  sql_statement [ ...n ]
```

```
CREATE TRIGGER TableAInsertTrig ON TableA
  INSTEAD OF INSERT AS ...
```

```
CREATE TRIGGER TableBDeleteTrig ON TableB
  AFTER DELETE AS ...
```

-- uses the FOR keyword to generate an AFTER trigger.

```
CREATE TRIGGER TableCUpdateTrig ON TableC
  FOR UPDATE AS ...
```

Использование триггеров DML: пример

```
CREATE TABLE my_table (a int NULL, b int NULL)
```

```
GO
```

```
CREATE TRIGGER my_trig
```

```
    ON my_table
```

```
    FOR INSERT
```

```
    AS
```

```
    IF UPDATE (b)
```

```
        PRINT 'Column b Modified'
```

```
GO
```

```
CREATE TRIGGER my_trig2
```

```
    ON my_table
```

```
    FOR INSERT
```

```
    AS
```

```
    IF ( COLUMNS_UPDATED() & 2 = 2 )
```

```
        PRINT 'Column b Modified'
```

```
GO
```

Триггеры DML: работа с одной и несколькими строками

```
CREATE TRIGGER NewPODetail
ON Purchasing.PurchaseOrderDetail
AFTER INSERT
AS
```

```
    UPDATE PurchaseOrderHeader
    SET SubTotal = SubTotal + LineTotal
    FROM inserted
    WHERE PurchaseOrderHeader.PurchaseOrderID =
    inserted.PurchaseOrderID;
```

```
CREATE TRIGGER NewPODetail2
ON Purchasing.PurchaseOrderDetail
AFTER INSERT
AS
```

```
UPDATE PurchaseOrderHeader SET SubTotal = SubTotal +
    (SELECT SUM(LineTotal) FROM inserted WHERE
    PurchaseOrderHeader.PurchaseOrderID = inserted.PurchaseOrderID)
WHERE PurchaseOrderHeader.PurchaseOrderID
IN (SELECT PurchaseOrderID FROM inserted);
```


Триггеры DDL

- триггеры DDL - особый вид триггеров, которые срабатывают при выполнении инструкций языка DDL (CREATE, ALTER, DROP и т.п.)
- триггеры DDL могут применяться при выполнении административных задач, например, для аудита и регулирования операций в базе данных:
 - для предотвращения внесений определенных изменений в схему базы данных;
 - для выполнения в базе данных некоторых действий в ответ на изменения в схеме базы данных;
 - для записи изменений или событий схемы базы данных;
- триггеры DDL срабатывают только после выполнения соответствующих инструкций DDL;
- триггер DDL и инструкция, приводящая к его срабатыванию, выполняются в одной транзакции;
- для одной инструкции Transact-SQL можно создать несколько триггеров DDL.

Триггеры DDL: пример

```
CREATE TRIGGER safety
ON DATABASE
FOR DROP_TABLE, ALTER_TABLE
AS
PRINT 'You must disable Trigger "safety" to drop or alter tables!'
ROLLBACK ;
```

```
CREATE TRIGGER ddl_trig_login
ON ALL SERVER
FOR DDL_LOGIN_EVENTS
AS
PRINT 'Login Event Issued.'
    SELECT
    EVENTDATA().value(' (/EVENT_INSTANCE/TSQLCommand/CommandText) [1] '
    , 'nvarchar(max) ' )
GO
DROP TRIGGER ddl_trig_login
ON ALL SERVER
GO;
```

Лабораторная работа №9.

Триггеры DML.

1. Для одной из таблиц пункта 2 задания 7 создать триггеры на вставку, удаление и добавление, при выполнении заданных условий один из триггеров должен инициировать возникновение ошибки (RAISERROR / THROW).
2. Для представления пункта 2 задания 7 создать триггеры на вставку, удаление и добавление, обеспечивающие возможность выполнения операций с данными непосредственно через представление.

Вопросы к экзамену

- Триггеры: назначение, типы. Триггеры DDL.
- Триггеры DML: триггеры типа `INSTEAD OF`.
- Триггеры DML: триггеры типа `AFTER`.
- Триггеры DML: сравнение триггеров типа `INSTEAD OF` и `AFTER`. Дополнительные средства управления данными, доступные в триггерах.