Базы данных

13 | Триггеры

План

- Типы триггеров
- Триггеры входа
- DDL-триггеры
- DML-триггеры
- Соглашения в DML-триггерах
- Вложенные и рекурсивные триггеры

Триггеры и типы триггеров

Триггер – это хранимая процедура, автоматически выполняемая при наступлении некоторого события.

В SQL Server Trigger есть следующие типы триггеров:

- **DML**
  - INSERT, UPDATE и DELETE
- **DDL**
  - При создании / удалении объектов
- **Logon**
  - При входе пользователя

Триггеры входа

Триггеры входа вызывают срабатывание хранимых процедур в ответ на событие LOGON (при начале пользовательской сессии). Триггеры входа срабатывают после завершения этапа проверки подлинности при входе, но перед тем, как пользовательский сеанс реально устанавливается. Если проверка подлинности завершается сбоем, триггеры входа не срабатывают.

Можно использовать триггеры входа для проверки и управления сеансами сервера.

Например, в следующем коде триггер входа запрещает попытки входа на SQL Server, инициированные под именем входа login_test, если уже созданы три пользовательских сеанса под этим именем входа:

```
CREATE TRIGGER connection_limit_trigger
ON ALL SERVER WITH EXECUTE AS 'login_test'
FOR LOGON
AS
BEGIN
IF ORIGINAL_LOGIN()= 'login_test' AND
   (SELECT COUNT(*) FROM sys.dm_exec_sessions
         WHERE is_user_process = 1 AND
            original_login_name = 'login_test') > 3
   ROLLBACK;
END;
```

https://docs.microsoft.com/ru-ru/sql/relational-databases/triggers/logon-triggers?view=sql-server-ver15

DDL-триггеры

DDL-триггеры используются для административных задач (аудит и управление структурой БД).

Используйте DDL-триггеры, если хотите сделать следующее:

- предотвращать внесение определенных изменений в схему базы данных;
- настроить выполнение в базе данных некоторых действий в ответ на изменения в схеме БД;
- записывать изменения или события схемы базы данных.

DDL-триггеры срабатывают **после** выполнения DDL-операторов, повлекших запуск триггера. DDL-триггеры нельзя использовать в качестве INSTEAD OF-триггеров.

Пример. Следующий код предотвращает изменение структуры и удаление таблиц:

```
CREATE TRIGGER safety
ON DATABASE
FOR DROP_TABLE, ALTER_TABLE
AS
   PRINT ‘Отключите триггер "safety" для изм./удаления таблиц!'
   ROLLBACK;
```

https://docs.microsoft.com/ru-ru/sql/relational-databases/triggers/ddl-triggers?view=sql-server-ver15

DML-триггеры

- For = After (после)
  - “стандартный” триггер – работает в конце транзакции (дополнительно к операции изменения данных)
- Instead Of (вместо)
  - Работает вместо команды изменения данных

Базовый синтаксис:

```
CREATE TRIGGER [ имя_схемы . ] имя_триггера
ON { таблица | представление }
[ WITH <dml_trigger_option> [ ,...n ] ]
{ FOR | AFTER | INSTEAD OF }
{ [ INSERT ] [ , ] [ UPDATE ] [ , ] [ DELETE ] }
AS { операторы_sql  [ ; ] [ ,...n ]
        | EXTERNAL NAME <method specifier [ ; ] > }
```

```
<dml_trigger_option> ::=
[ ENCRYPTION ] [ EXECUTE AS Clause ]

<method_specifier> ::=
assembly_name.class_name.
method_name
```

https://docs.microsoft.com/ru-ru/sql/relational-databases/triggers/dml-triggers?view=sql-server-ver15

Использование DML-триггеров

- Не требуется изменение кода клиентского ПО для:
  - Автоматизации
  - Уведомлений
  - Журналирования
  - Поддержки актуальности денормализованных данных

Определение изменившихся данных в DML-триггере

- UPDATE()

Возвращает логическое значение, указывающее, была ли сделана попытка INSERT или UPDATE в указанном столбце таблицы или представления. UPDATE () используется в любом месте тела триггера Transact-SQL INSERT или UPDATE для проверки того, должен ли триггер выполнять определенные действия.

```
CREATE TRIGGER reminder
ON SalesLT.Product
AFTER UPDATE
AS
IF ( UPDATE (ProductNumber) OR UPDATE (rowguid) )
BEGIN
RAISERROR (50009, 16, 10)
END;
```

- COLUMNS_UPDATED()

Возвращает битовую маску (varbinary), указывающую на столбцы в таблице или представлении, которые были обновлены. COLUMNS_UPDATED используется в любом месте тела триггера Transact-SQL INSERT или UPDATE для проверки того, должен ли триггер выполнять определенные действия.

```
CREATE TRIGGER dbo.updEmployeeData
ON dbo.employeeData
AFTER UPDATE AS
/* Проверяем были ли столбцы 2, 3 или 4 обновлены.*/
  IF (COLUMNS_UPDATED() & 14) > 0
            …
```

https://docs.microsoft.com/ru-ru/sql/t-sql/functions/columns-updated-transact-sql?view=sql-server-ver15

Соглашения в DML-триггерах

- Таблица “**inserted**” (используется в Insert и Update);
- Таблица “**deleted**” (используется в Delete и Update);
- After-триггер участвует в транзакции:
  - Может откатить изменения в данных (при необходимости);
- Пользователю нужны права доступа к объектам, использующимся в триггере;
- Можно задавать порядок выполнения триггеров через **sp_settriggerorder** (@order = 'first' | 'last' | 'none’).

```
EXEC sp_settriggerorder @triggername = ‘myLovelyTrigger',
@order = 'first', @stmttype='update'
```

https://docs.microsoft.com/ru-ru/sql/relational-databases/triggers/use-the-inserted-and-deleted-tables?view=sql-server-ver15

Вложенные (каскадные) триггеры

- Вложенный триггер – это триггер, выполняемый из-за действий в другом триггере (вложенность до 32 уровней);
- Можно разрешать или запрещать вложенность (каскадность) триггеров AFTER с помощью параметра конфигурации сервера **nested triggers:** 1 – разрешает (by def.), 0 – запрещает);
- **@@NESTLEVEL** – текущий уровень вложенности хр.проц. (0, 1, …)

```
EXEC sp_configure 'show advanced options', 1;
GO
RECONFIGURE ;
GO
EXEC sp_configure 'nested triggers', 0 ;
GO
RECONFIGURE;
```

https://docs.microsoft.com/ru-ru/sql/relational-databases/triggers/create-nested-triggers?view=sql-server-ver15

Рекурсивные триггеры

- Прямая рекурсия – триггер выполняет действие, вызывающее повторное срабатывание того же триггера. По умолчанию включена, управляется параметром 'server trigger recursion’.

```
EXEC sp_configure 'show advanced options', 1;
GO
RECONFIGURE ;
GO
EXEC sp_configure 'server trigger recursion', 0 ;
GO
RECONFIGURE;
```

- Косвенная рекурсия – триггер срабатывает и выполняет действие, которое вызывает срабатывание другого триггера того же типа (AFTER или INSTEAD OF). Отключается через nested triggers = 0 (см. выше).

DEMO

Работа с триггерами

Изучено

- Типы триггеров
- Триггеры входа
- DDL-триггеры
- DML-триггеры
- Соглашения в DML-триггерах
- Вложенные и рекурсивные триггеры