



Universidade Federal de Sergipe
Departamento de Sistemas de Informação
Banco de Dados II



Gatilhos DDL

André Vinicius R. P. Nascimento
andreviniciusnascimento@gmail.com



Conteúdo

- Introdução
- Criando Triggers
- Informações sobre eventos
- Modificação de Triggers
- Remoção de Triggers
- Desabilitando Triggers
- Habilitando Triggers
- Obtendo informações



Introdução

- O que é um Gatilho ou Trigger DDL ?
 - É um tipo especial de gatilho que é disparado em resposta a um comando DDL.
 - Respondem a eventos do Banco de Dados ou do Servidor.
 - Também são conhecidos com gatilhos de sistema.



Introdução

- Cenários para uso de Triggers DDL:
 - Executar tarefas administrativas
 - Efetuar auditoria
 - Controlar alterações no banco de dados



Introdução

- Quando uma Trigger DDL é executada ?
 - Após a execução de um evento DDL.
 - Esse Evento pode ser um CREATE_TABLE, ALTER_TABLE, DROP_TABLE, CREATE_PROCEDURE, ALTER_PROCEDURE, CREATE_VIEW, etc.



Introdução

- Também é possível especificar grupo de eventos.
 - DDL_TABLE_EVENTS
(CREATE TABLE, DROP TABLE, ALTER TABLE)
 - DDL_VIEW_EVENTS
(CREATE VIEW, DROP VIEW, ALTER VIEW)



Criando Triggers DDL

```
CREATE TRIGGER trigger_name  
ON ALL SERVER | DATABASE  
FOR event_type | event_group  
AS sql_statement
```



Criando Triggers DDL

```
CREATE TRIGGER TG_DDL_ALTERACAO_BANCO
ON DATABASE -- Define o escopo
FOR DROP_TABLE -- Define o Evento
AS
    RAISERROR('Remoção de Tabelas não é
permitida',1,1)
ROLLBACK
```




Criando Triggers DDL

```
CREATE TRIGGER TG_DDL_ALTERACAO_BANCO
ON DATABASE
FOR DROP_TABLE, ALTER_TABLE
AS
    RAISERROR('Remoção e Alteração de
Tabelas não são permitidas',1,1)
ROLLBACK
```



Criando Triggers DDL

```
CREATE TRIGGER TG_DDL_ALTERACAO_BANCO  
ON DATABASE  
FOR DDL_TABLE_EVENTS  
AS
```

```
    RAISERROR('Criação, Remoção e Alteração  
de Tabelas não são permitidas',1,1)  
    ROLLBACK
```



Informações sobre eventos

- Triggers DDL não criam as tabelas INSERTED e DELETED.
- As informações sobre o evento que disparou a trigger podem ser capturadas através da função EVENTDATA.



Informações sobre eventos

- A função EVENTDATA retorna um valor xml.
- As informações no esquema xml incluem:
 - A hora do evento;
 - O id do processo da conexão;
 - O tipo do evento.



Informações sobre eventos

- A depender do tipo do evento, o esquema xml também pode incluir:
 - O banco de dados em que o evento ocorreu;
 - O objeto afetado pelo evento;
 - O comando transact-sql que representa o evento.



Informações sobre eventos

```
<EVENT_INSTANCE>
```

```
  <EventType>CREATE_TABLE</EventType>
```

```
  <ServerName>HOST_BD\SQLSERVER2008</ServerName>
```

```
  <LoginName>sa</LoginName>
```

```
  <UserName>dbo</UserName>
```

```
  <DatabaseName>BD_TOPICOS</DatabaseName>
```

```
  <SchemaName>dbo</SchemaName>
```

```
  <ObjectName>TB_ALUNO</ObjectName>
```

```
  <ObjectType>TABLE</ObjectType>
```

```
  <TSQLCommand>
```

```
    <CommandText>
```

```
      CREATE TABLE...
```

```
    </CommandText>
```

```
  </TSQLCommand>
```

```
</EVENT_INSTANCE>
```



Função EVENTDATA

- Considere a seguinte tabela

```
CREATE TABLE TB_LOG_ALTERACOES_ESQUEMA (  
    id_log      int not null primary key  
    identity(1,1),  
    dt_log      datetime,  
    nm_usuario  varchar(30),  
    evento      nvarchar(100),  
    comando     nvarchar(2000)  
)
```



Função EVENTDATA

```
CREATE TRIGGER TG_DDL_ALTERACOES_ESQUEMA
ON DATABASE
FOR DDL_TABLE_EVENTS
AS
DECLARE @evento XML
SET @evento = EVENTDATA()
INSERT INTO TB_LOG_ALTERACOES_ESQUEMA (dt_log,
    nm_usuario, evento, comando)
VALUES (GETDATE(), CURRENT_USER,
    @evento.value(' (/EVENT_INSTANCE/EventType) [1] ',
        'nvarchar(100) '),
    @evento.value(' (/EVENT_INSTANCE/TSQLCommand/CommandText) [1] ', 'nvarchar(2000) ')
)
```




Modificação de Triggers

- Através do comando ALTER TRIGGER

```
ALTER TRIGGER TG_DDL_ALTERACAO_BANCO  
ON DATABASE  
FOR DROP_TABLE  
AS
```

```
    RAISERROR('Remoção de Tabelas não são  
permitidas', 1, 1)  
    ROLLBACK
```



Remoção de Triggers

- Através do comando DROP TRIGGER

```
DROP TRIGGER TG_DDL_ALTERACAO_BANCO  
ON DATABASE
```

```
DROP TRIGGER TG_DDL_ALTERACAO_SERV  
ON ALL SERVER
```



Desabilitando Triggers

- Através do comando DISABLE TRIGGER

```
DISABLE TRIGGER TG_ALTERACAO_BANCO  
ON DATABASE
```

```
DISABLE TRIGGER TG_ALTERACAO_SERV  
ON ALL SERVER
```



Habilitando Triggers

- Através do comando ENABLE TRIGGER

```
ENABLE TRIGGER TG_ALTERACAO_BANCO  
ON DATABASE
```

```
ENABLE TRIGGER TG_ALTERACAO_SERV  
ON ALL SERVER
```



Obtendo Informações

- Triggers DDL com escopo de Banco

```
select * from sys.triggers
```

```
select * from sys.trigger_events
```

```
select * from sys.sql_modules
```



Obtendo Informações

- Triggers DDL com escopo de Servidor

```
select * from sys.server_triggers
```

```
select * from sys.server_trigger_events
```

```
select * from sys.server_sql_modules
```