Database - Feature #1658

improve database trigger support

10/22/2012 12:06 AM - Eric Faulhaber

Status:	Closed		**Start date:**	06/19/2013
Priority:	Normal		**Due date:**	07/01/2013
Assignee:			**% Done:**	100%
Category:			**Estimated time:**	0.00 hour
Target version:	Runtime Support for Server Features			
billable:	No		**version:**	
vendor_id:	GCD			

Description

Subtasks:	
Feature # 1949: add conversion support for database triggers	**Closed**
Feature # 2024: implement runtime support for database triggers	**Closed**

History

#1 - 10/22/2012 12:08 AM - Eric Faulhaber

- runtime support for TRIGGER PROCEDURE (DELETE, WRITE, CREATE), DISABLE TRIGGERS, ON FIND OF table;
- need report changes to check if there are other ON statement cases in customer code, but it is likely that other ON statement usage is at least small in nature (and it would only be for DELETE, WRITE or CREATE, so the functionality must already be there);
- there will need to be work on the conversion side to emit procedures/trigger blocks (for ON statement) and register them.

#2 - 10/31/2012 04:04 PM - Greg Shah

- Target version set to Milestone 7

#3 - 04/10/2013 02:55 PM - Stanislav Lomany

What needs to be done:

1. Allow registration for WRITE and FIND schema triggers.
2. Fire WRITE and FIND triggers at proper places.
3. Handle triggers as arbitrary classes or DatabaseTrigger instances instead of classes that implement AssignTrigger/DeleteTrigger/etc. interface.
4. Implement execution of session and schema triggers (i.e. call TriggerClass.create/write/etc.).
5. Implement scoped registerDatabaseTrigger (with optional OVERRIDE) and deregisterDatabaseTrigger.
6. Make sure that triggers are properly handled as top level blocks.
7. Implement scoped disable*Triggers.

#4 - 04/10/2013 03:34 PM - Stanislav Lomany

My estimates:

1. 3 h - Allow registration for WRITE and FIND schema triggers.
2. 16 h, need to find these proper places and check all cases - Fire WRITE and FIND triggers at proper places.
3. 6 h - Handle triggers as arbitrary classes or DatabaseTrigger instances instead of classes that implement AssignTrigger/DeleteTrigger/etc. interface.
4. 6 h - Implement execution of session and schema triggers (i.e. call TriggerClass.create/write/etc.).

5. 16 h - Implement scoped registerDatabaseTrigger (with optional OVERRIDE) and deregisterDatabaseTrigger.
6. If everything is works as expected - 4 h - Make sure that triggers are properly handled as top level blocks.
7. 6 h - Implement scoped disable*Triggers.

#5 - 01/24/2014 12:33 PM - Eric Faulhaber

- Status changed from New to Closed

#6 - 11/16/2016 11:42 AM - Greg Shah

- Target version changed from Milestone 7 to Runtime Support for Server Features