

Database - Bug #5289

Misc. problems with triggers

04/26/2021 05:05 PM - Igor Skornyakov

Status: WIP	Start date:
Priority: Normal	Due date:
Assignee: Ovidiu Maxiniuc	% Done: 0%
Category:	Estimated time: 0.00 hour
Target version:	case_num:
billable: No	
vendor_id: GCD	
Description	
Related issues:	
Related to Database - Feature #3814: more schema metadata	Closed

History

#1 - 04/26/2021 05:13 PM - Igor Skornyakov

I've found a number of issues with triggers' support in FWD:

- In 4GL if the CREATE trigger rejects the operation (hBuffer:BUFFER-CREATE() returns 'no') while in FWD hBuffer.unwrapBuffer().bufferCreate() returns true.
- In 4GL on FIND FIRST; GET CURRENT the FIND trigger is invoked once while if FWD it happens twice.
- In the presence of the FIND trigger we get NPE in PreselectQuery line 5630:

```
else
{
    Object[] data = available ? results.get() : null;
    coreFetch(data, lockType, errorIfNull, false);

    if (available)
    {
        accumulate();
    }
}
```

(results == null) on REPOSITION-FORWARD even if the trigger accepts the operation.

#2 - 04/26/2021 05:13 PM - Igor Skornyakov

#3 - 04/27/2021 07:20 AM - Igor Skorniyakov

How to reproduce:

Just run stat1.p with our standard "customer" table and two triggers (see @sftp://<userid>@xfer.goldencode.com/opt/testcases/meta).

TABLE-TRIGGER "CREATE" NO-OVERRIDE PROCEDURE "meta/create-trg.p"

TABLE-TRIGGER "FIND" NO-OVERRIDE PROCEDURE "meta/find-trg.p"

With current values create-trg-cnt-max=100 and find-trg-cnt-max=100 the triggers will accept all operations but you will see NPE close to the end of the test:

[04/26/2021 23:50:50 GMT+03:00] (com.goldencode.p2j.util.TransactionManager:SEVERE) Abnormal end; original error:

java.lang.NullPointerException

at com.goldencode.p2j.persist.PreselectQuery.fetch(PreselectQuery.java:5630)

at com.goldencode.p2j.persist.AdaptiveQuery.fetch(AdaptiveQuery.java:3762)

at com.goldencode.p2j.persist.AdaptiveQuery.next(AdaptiveQuery.java:2158)

at com.goldencode.p2j.persist.AdaptiveQuery.first(AdaptiveQuery.java:1680)

at com.goldencode.p2j.persist.PreselectQuery.first(PreselectQuery.java:2395)

at com.goldencode.p2j.persist.AbstractQuery.getFirst(AbstractQuery.java:2362)

at com.goldencode.p2j.persist.QueryWrapper.lambda\$getFirst\$4(QueryWrapper.java:4889)

at com.goldencode.p2j.persist.QueryWrapper.handleQueryOffEnd(QueryWrapper.java:6577)

at com.goldencode.p2j.persist.QueryWrapper.getFirst(QueryWrapper.java:4889)

at com.goldencode.testcases.meta.Stat1.lambda\$execute\$10(Stat1.java:151)

at com.goldencode.p2j.util.Block.body(Block.java:605)

at com.goldencode.p2j.util.BlockManager.processBody(BlockManager.java:8559)

at com.goldencode.p2j.util.BlockManager.topLevelBlock(BlockManager.java:8228)

at com.goldencode.p2j.util.BlockManager.externalProcedure(BlockManager.java:496)

at com.goldencode.p2j.util.BlockManager.externalProcedure(BlockManager.java:467)

at com.goldencode.testcases.meta.Stat1.execute(Stat1.java:79)

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)

at java.lang.reflect.Method.invoke(Method.java:498)

at com.goldencode.p2j.util.Utils.invoke(Utils.java:1558)

at com.goldencode.p2j.main.StandardServer\$MainInvoker.execute(StandardServer.java:2227)

at com.goldencode.p2j.main.StandardServer.invoke(StandardServer.java:1663)

at com.goldencode.p2j.main.StandardServer.standardEntry(StandardServer.java:582)

at com.goldencode.p2j.main.StandardServerMethodAccess.invoke(Unknown Source)

at com.goldencode.p2j.util.MethodInvoker.invoke(MethodInvoker.java:156)

at com.goldencode.p2j.net.Dispatcher.processInbound(Dispatcher.java:783)

at com.goldencode.p2j.net.Conversation.block(Conversation.java:422)

at com.goldencode.p2j.net.Conversation.run(Conversation.java:232)

at java.lang.Thread.run(Thread.java:748)

[04/26/2021 23:50:50 GMT+03:00] (StandardServer.invoke:SEVERE) {0000000E:00000022:bogus} Abnormal end!

java.lang.reflect.InvocationTargetException

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)

at java.lang.reflect.Method.invoke(Method.java:498)

at com.goldencode.p2j.util.Utils.invoke(Utils.java:1558)

at com.goldencode.p2j.main.StandardServer\$MainInvoker.execute(StandardServer.java:2227)

at com.goldencode.p2j.main.StandardServer.invoke(StandardServer.java:1663)

at com.goldencode.p2j.main.StandardServer.standardEntry(StandardServer.java:582)

at com.goldencode.p2j.main.StandardServerMethodAccess.invoke(Unknown Source)

at com.goldencode.p2j.util.MethodInvoker.invoke(MethodInvoker.java:156)

at com.goldencode.p2j.net.Dispatcher.processInbound(Dispatcher.java:783)

at com.goldencode.p2j.net.Conversation.block(Conversation.java:422)

at com.goldencode.p2j.net.Conversation.run(Conversation.java:232)

at java.lang.Thread.run(Thread.java:748)

Caused by: java.lang.NullPointerException

at com.goldencode.p2j.persist.PreselectQuery.fetch(PreselectQuery.java:5630)

at com.goldencode.p2j.persist.AdaptiveQuery.fetch(AdaptiveQuery.java:3762)

at com.goldencode.p2j.persist.AdaptiveQuery.next(AdaptiveQuery.java:2158)

at com.goldencode.p2j.persist.AdaptiveQuery.first(AdaptiveQuery.java:1680)

at com.goldencode.p2j.persist.PreselectQuery.first(PreselectQuery.java:2395)

at com.goldencode.p2j.persist.AbstractQuery.getFirst(AbstractQuery.java:2362)

at com.goldencode.p2j.persist.QueryWrapper.lambda\$getFirst\$4(QueryWrapper.java:4889)

at com.goldencode.p2j.persist.QueryWrapper.handleQueryOffEnd(QueryWrapper.java:6577)

at com.goldencode.p2j.persist.QueryWrapper.getFirst(QueryWrapper.java:4889)

at com.goldencode.testcases.meta.Stat1.lambda\$execute\$10(Stat1.java:151)

at com.goldencode.p2j.util.Block.body(Block.java:605)

at com.goldencode.p2j.util.BlockManager.processBody(BlockManager.java:8559)

at com.goldencode.p2j.util.BlockManager.topLevelBlock(BlockManager.java:8228)

at com.goldencode.p2j.util.BlockManager.externalProcedure(BlockManager.java:496)

at com.goldencode.p2j.util.BlockManager.externalProcedure(BlockManager.java:467)

at com.goldencode.testcases.meta.Stat1.execute(Stat1.java:79)

```
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at com.goldencode.p2j.util.Utills.invoke(Utills.java:1558)
at com.goldencode.p2j.main.StandardServer$MainInvoker.execute(StandardServer.java:2227)
at com.goldencode.p2j.main.StandardServer.invoke(StandardServer.java:1663)
at com.goldencode.p2j.main.StandardServer.standardEntry(StandardServer.java:582)
at com.goldencode.p2j.main.StandardServerMethodAccess.invoke(Unknown Source)
at com.goldencode.p2j.util.MethodInvoker.invoke(MethodInvoker.java:156)
at com.goldencode.p2j.net.Dispatcher.processInbound(Dispatcher.java:783)
at com.goldencode.p2j.net.Conversation.block(Conversation.java:422)
at com.goldencode.p2j.net.Conversation.run(Conversation.java:232)
at java.lang.Thread.run(Thread.java:748)
```

#4 - 04/27/2021 12:46 PM - Igor Skornyakov

In addition.

The REPLICATION triggers are not invoked in FWD. See the same test as above. Just add the following triggers to the 'customer' table definition.

```
TABLE-TRIGGER "REPLICATION-CREATE" NO-OVERRIDE PROCEDURE "./meta/rep-create-trg.p" CRC "?"
TABLE-TRIGGER "REPLICATION-DELETE" NO-OVERRIDE PROCEDURE "./meta/rep-delete-trg.p" CRC "?"
TABLE-TRIGGER "REPLICATION-WRITE" NO-OVERRIDE PROCEDURE "./meta/rep-write-trg.p" CRC "?"
```

#5 - 05/26/2021 07:38 AM - Greg Shah

Ovidiu: From emails, I know you have some of these issues resolved. Please document the status here.

If we can get these changes added to 3821c, it would be good because we have a customer that needs the replication trigger fixes.

#6 - 05/26/2021 10:28 PM - Ovidiu Maxiniuc

Yes, the REPLICATION triggers work correctly now (pending commit).

I also added/fixed vetoing support for some triggers: if a trigger returns error, the respective operation which invoked it (ex: CREATE, DELETE) is aborted.

Still to do:

- vetoing support for ASSIGN triggers;
- items 2 & 3 from note-1.

#7 - 05/27/2021 07:59 AM - Greg Shah

- Start date deleted (04/26/2021)

Are those change in 3821c?

#8 - 05/27/2021 11:14 AM - Ovidiu Maxiniuc

- Status changed from New to WIP

No, not yet.

I am waiting to finish the ASSIGN triggers work, too, and commit all in a single unified commit. I estimate this is going to happen today.

#9 - 06/10/2021 10:59 PM - Ovidiu Maxiniuc

I committed the fix for REPLICATION and improved management of table triggers (including veto support) in r12468 and ASSIGN triggers a bit later, in r12504.

I could not reproduce the 2nd issue reported in first note. Is it possible it is specific to a different branch than we are working on? I will retest this, anyway, when I return for the 3rd issue.

#10 - 06/11/2021 03:46 AM - Eric Faulhaber

Is the generics regression in DatabaseTriggerManager which was causing converted code to not compile fixed in 3821c at this point? Constantin applied a fix or workaround to 5296a when I had trouble with this in one project, but I'm not sure if this was a permanent fix or a temporary workaround, nor what the status of 3821c is now.

#11 - 06/11/2021 05:06 PM - Ovidiu Maxiniuc

Yes, attempt to improve generics was rolled back in r12522. It was reported by other colleagues which compiled newly converted code. I failed to spot the issues because I used already compiled code which had the type already erased so no complains from Java runtime.

#12 - 06/14/2021 02:55 PM - Eric Faulhaber

Is bullet point 3 (NPE in PreselectQuery for FIND trigger) in [#5289-1](#) the only thing left in this task?

Were you ever able to recreate bullet point 2?

#13 - 06/14/2021 06:15 PM - Ovidiu Maxiniuc

Eric Faulhaber wrote:

Is bullet point 3 (NPE in PreselectQuery for FIND trigger) in [#5289-1](#) the only thing left in this task?

Actually I could not recreate this too. My debugger reaches the respective line (5614 in current revision of 3821c) but **no** NPE. So I do not have a testcase for it. Is there are special where predicate?

Were you ever able to recreate bullet point 2?

The bug report mixes two different kind of queries here: buffer based (FIND) and query based (GET). I tried again different combinations of

```
FIND FIRST trigger-test.  
FIND CURRENT trigger-test.
```

and

```
GET FIRST q1.  
GET CURRENT q1.
```

but the FIND trigger is called only once.

Is it possible that the observed event occurred in some more complex customer code and the second FIND trigger was caused by another FIND statement?

OTOH, I found a difference between 4GL and FWD. Using the following code:

```
DEFINE QUERY q1 FOR trigger-test.  
OPEN QUERY q1 PRESELECT EACH trigger-test.  
  
FIND FIRST trigger-test.  
GET CURRENT q1.
```

will cause No query record is available. (4114) to be raised in 4GL, while in FWD nothing happens. I have just encountered this and not investigated it yet.