Database - Bug #6868

MariaDB Dialect can't properly check if a sequence exists

10/20/2022 05:50 AM - Radu Apetrii

Status:	Feedback	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:			
billable:	No	**case_num:**	
vendor_id:	GCD	**version:**	

Description

History

#1 - 10/20/2022 05:57 AM - Radu Apetrii

I tried executing with MariaDB (10.3.34) database this example:

```
define variable seqvalue as int.
define variable seqvalue2 as int.
do transaction:
    seqvalue = current-value(TestSequence).
    seqvalue2 = next-value(TestSequence).
end.
message seqvalue seqvalue2.
```

TestSequence is a sequence inside the database.
I got the following error:

```
[10/20/2022 12:46:28 EEST] (org.mariadb.jdbc.util.log.Slf4JLogger:WARNING) Error: 1146-42S02: Table 'fwd.table
s' doesn't exist
[10/20/2022 12:46:28 EEST] (com.goldencode.p2j.persist.Persistence:SEVERE) Error executing SQL statement:  sel
ect TABLE_NAME from tables where table_type = 'SEQUENCE' and TABLE_SCHEMA = ?
com.goldencode.p2j.persist.PersistenceException: Error scrolling
Caused by: java.sql.SQLSyntaxErrorException: (conn=328) Table 'fwd.tables' doesn't exist
        at org.mariadb.jdbc.export.ExceptionFactory.createException(ExceptionFactory.java:280)
        at org.mariadb.jdbc.export.ExceptionFactory.create(ExceptionFactory.java:368)
        at org.mariadb.jdbc.message.ClientMessage.readPacket(ClientMessage.java:137)
        at org.mariadb.jdbc.client.impl.StandardClient.readPacket(StandardClient.java:833)
```

#2 - 10/20/2022 11:41 AM - Ovidiu Maxiniuc

- % Done changed from 0 to 100

- Status changed from New to Feedback

I assume you use 3821c.
This is an issue which was addressed in another branch. To allow you advance until the merge of the branches use the following patch:

```
--- 3821c/src/com/goldencode/p2j/persist/sequence/MariaDbSequenceHandler.java
+++ 6129b/src/com/goldencode/p2j/persist/sequence/MariaDbSequenceHandler.java
@@ -323,15 +339,19 @@
          Persistence persistence = ConnectionManager.getPersistence(ldbName);
          try
          {
+            String catalog = persistence.getSession().getConnection().getCatalog();
+
             ScrollableResults<String> rs = persistence.executeSQLQuery(
-                  "select TABLE_NAME from tables where table_type = 'SEQUENCE' and TABLE_SCHEMA = ?",
-                  new Object[] { ldbName });
+                  "select table_name from information_schema.tables " +
+                  "where table_type = 'SEQUENCE' and table_schema = ?",
+                  new Object[] { catalog });
             while (rs.next())
             {
                crtSeqList.add(rs.get(0, String.class));
             }
          }
-         catch (PersistenceException pe)
+         catch (PersistenceException |
+                SQLException pe)
          {
             if (LOG.isLoggable(Level.WARNING))
             {
```

#3 - 10/24/2022 06:23 AM - Radu Apetrii

As a side note, I noticed there was a slight difference in results when running the test from #6868-1 in both PostgreSQL and MariaDB.
With PostgreSQL the result is 100 101, while the result with MariaDB is ? 100.
Is this something you are aware of?

*As a matter of fact, I tried using current-value(TestSequence) = 102. In 4GL it works, but in FWD it doesn't. There is no error given, but the value of the sequence remains unchanged.

#4 - 10/24/2022 07:05 AM - Ovidiu Maxiniuc

Radu Apetrii wrote:

> As a side note, I noticed there was a slight difference in results when running the test from #6868-1 in both PostgreSQL and MariaDB.
> With PostgreSQL the result is 100 101, while the result with MariaDB is ? 100.
> Is this something you are aware of?
>
> *As a matter of fact, I tried using current-value(TestSequence) = 102. In 4GL it works, but in FWD it doesn't. There is no error given, but the value of the sequence remains unchanged.

You noticed correctly. MariaDB is a bit different from all other dialects (and 4GL). The current value is not available without executing a next first. Constantin tried to work around this by first invoking next and then rolling back one step. Unfortunately, this does not work as expected because ...once you have reach a certain value, it cannot set back to a previous one (see https://mariadb.com/kb/en/setval/). These are open issues.

The good part is that the latter seems to be fixable using ALTER SEQUENCE instead of setval(). This is in tests momentarily.