Subject: Re: How does All_Tables determine if a table exists?
Posted by mirek on Fri, 21 Aug 2020 14:39:35 GMT
View Forum Message <> Reply to Message

Lance wrote on Wed, 19 August 2020 18:55I encountered this problem in PostgreSQL, but it likely is not limited to PG.

To reproduce the problem(you need a PostgreSQL server):

1. Open the SQL_PostgreSql example in the reference assembly. Modify the connection string to a valid one in Postgres.cpp
!m_session.Open("host=localhost dbname=test user=test password=yourpassword")

2. F5 to run the program.

3. Drop the test database (eg., in pgAdmin4) and recreate it either with the same name of change to a new name, eg, test1

4. modify the connection string like you did in step 1, with the name you changed to in step 3.

5. F5, the program will fail. As All_Tables(sch) now doesn't do anything as somehow it thinks all tables exsit.

Actually, those mechanisms never ever read anything from the database, so they do not even know that tables exist.

What is the real culprit there:


```
SqlSchema sch(PGSQL);
All_Tables(sch);
if(sch.ScriptChanged(SqlSchema::UPGRADE))
 SqlPerformScript(sch.Upgrade(), p);
if(sch.ScriptChanged(SqlSchema::ATTRIBUTES)) {
 SqlPerformScript(sch.Attributes(), p);
}
if(sch.ScriptChanged(SqlSchema::CONFIG)) {
 SqlPerformScript(sch.ConfigDrop(), p);
 SqlPerformScript(sch.Config(), p);
}
sch.SaveNormal();
```


is these ScriptChanged calls.

Thing is, All_Tables just generates a couple of sql scripts, into Strings. SaveNormal saves these scripts to files. ScriptChanged compares String with the file to detect that script has changed and

only runs it when it did here.

All this is to save the time when developing so that those scripts are not performed at each application start (it can be lengthy). If you need to completely update the database, the easy way is to simply delete those files (they are normally in .exe dir in win32 and config dir in linux).

Alternatively, you can reorganize the code and e.g. run scripts without checking. One popular method is to use main configuration, e.g. add to have flag NOSCHEMA

```
 SqlSchema sch(PGSQL);
 All_Tables(sch);
#ifndef flagNOSCHEMA
 SqlPerformScript(sch.Upgrade(), p);
 SqlPerformScript(sch.Attributes(), p);
 SqlPerformScript(sch.Config(), p);
#endif
```

Mirek