Subject: MySqlSession Close and Reconnect crashes Application
Posted by hoelblin on Wed, 25 Nov 2020 16:17:33 GMT
View Forum Message <> Reply to Message

Hi,

I have an Application with a Connection to MySql. When the Application is Idle for
a long Period of Time the Connection is Closed with session.Close.

After that session.IsOpen returns false.

When I try to reconnect with session.Reconnect() the App Crashes with a Null Pointer dereference.
It seems like in Sql.Cancel cn->cancel is called while Member cn == NULL.

This behavior is on TheIDE Version 15512 on Windows with CLANG, on TheIDE 13664 with minGW this works as
expected without crashing.

To reproduce the Crash i have taken the SQL_MySql reference Example and make some minor changes.

```
#include <MySql/MySql.h>

// debian: sudo apt-get install libmysqlclient-dev

using namespace Upp;

#define SCHEMADIALECT <MySql/MySqlSchema.h>
#define MODEL <mysqlCrash/schema.sch>
#include <Sql/sch_header.h>
#include <Sql/sch_source.h>
#include <Sql/sch_schema.h>

CONSOLE_APP_MAIN
{
 MySqlSession session;
 // edit the connection parameters if necessary
 if(session.Connect("test_user", "test", "test")) {
  Cout() << "Connected\n";
  SQL = session;

  SqlSchema sch(MY_SQL);
  All_Tables(sch);
  // create the table if necessary
  SqlPerformScript(sch.Upgrade());
  SqlPerformScript(sch.Attributes());
```

```
SQL.ClearError();


try {
 // insert some random data
 SQL & Insert(TEST_TABLE)(VALUE, Uuid::Create().ToString());
 // fetch some data
 Sql sql;
 sql * Select(ID, VALUE).From(TEST_TABLE)
                         .OrderBy(Descending(ID))
                         .Limit(5);
 while(sql.Fetch())
  Cout() << AsString(sql[0]) << ": " << AsString(sql[VALUE]) << "\n";
}
catch(SqlExc &ex) {
 Cerr() << "ERROR: " << ex << "\n";
 SetExitCode(1);
}

session.Close();

Sleep(2000);

if(session.IsOpen()) {
 Cout() << "Connection closed FAILED!\n";
} else {
 Cout() << "Connection closed!\n";
}

Sleep(2000);

Cout() << "Try open Connection again\n";

if(session.Reconnect()) { // CRASH !!!!!!!!!
 try {
  Sql sql;
  sql * Select(ID, VALUE).From(TEST_TABLE)
                          .OrderBy(Descending(ID))
                          .Limit(3);
  while(sql.Fetch())
   Cout() << AsString(sql[0]) << ": " << AsString(sql[VALUE]) << "\n";
 }
 catch(SqlExc &ex) {
  Cerr() << "ERROR: " << ex << "\n";
  SetExitCode(1);
 }
} else {
 Cout() << "ERROR: " << session.GetLastError() << "\n";
```

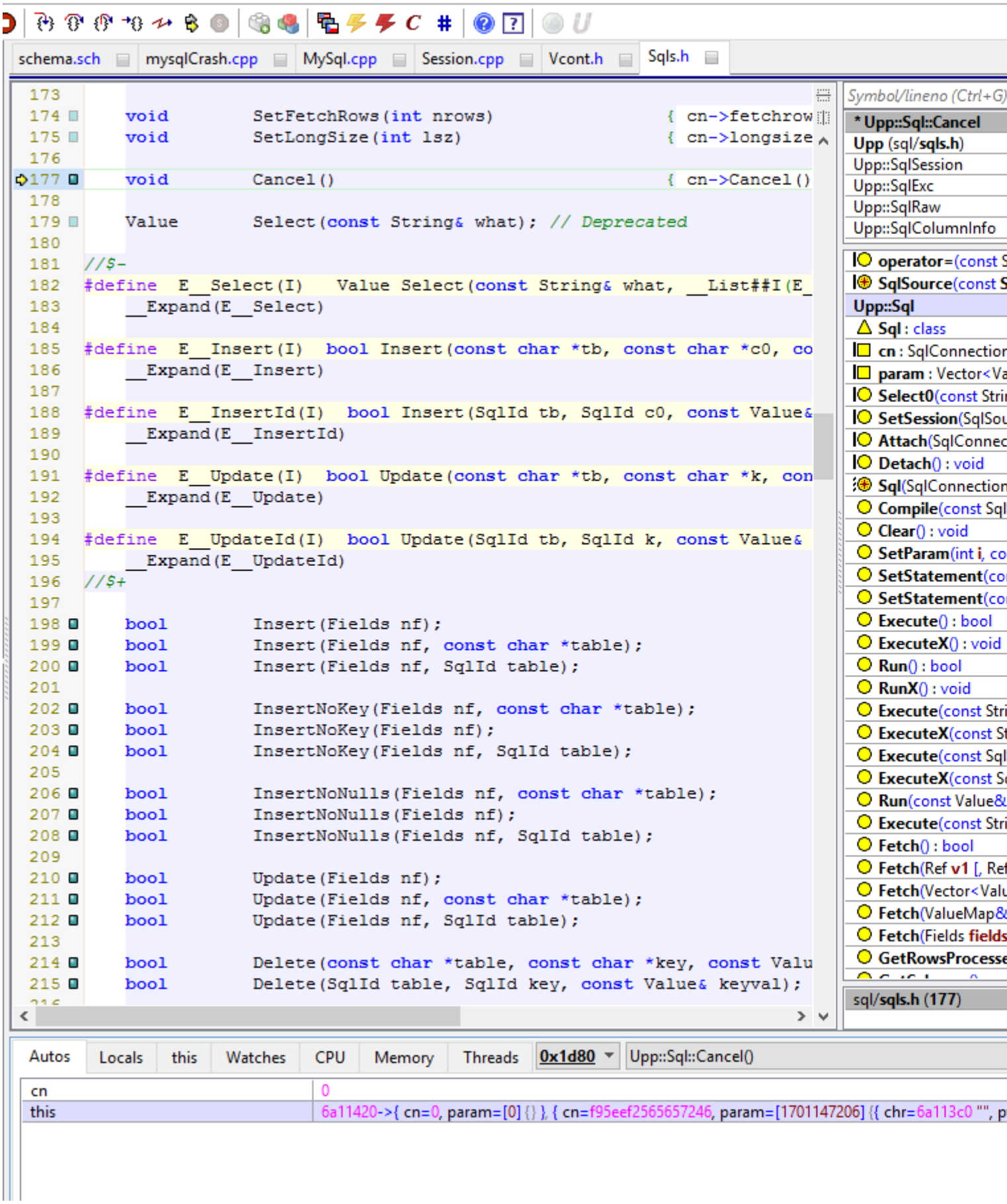
```
  }
 }
 else {
  Cerr() <<"ERROR: Unable to connect to database\n";
  SetExitCode(1);
 }
}
```

// Moderator: PLESE USE CODE TAG FOR THE NEXT TIME!

My Question is what is the right way to do a Reconnect after Close ?

File Attachments

1) mysqlCrash1.PNG, downloaded 1112 times


schema.sch
mysqlCrash.cpp
MySql.cpp
Session.cpp
Vcont.h
Sqls.h
void SetFetchRows(int nrows) { cn->fetchrow
void SetLongSize(int lsz) { cn->longsize
void Cancel() { cn->Cancel()
Value Select(const String& what); // Deprecated
//$-
#define E__Select(I) Value Select(const String& what, __List##I(E_
__Expand(E__Select)
#define E__Insert(I) bool Insert(const char *tb, const char *c0, co
__Expand(E__Insert)
#define E__InsertId(I) bool Insert(SqlId tb, SqlId c0, const Value&
__Expand(E__InsertId)
#define E__Update(I) bool Update(const char *tb, const char *k, con
__Expand(E__Update)
#define E__UpdateId(I) bool Update(SqlId tb, SqlId k, const Value&
__Expand(E__UpdateId)
//$+
bool Insert(Fields nf);
bool Insert(Fields nf, const char *table);
bool Insert(Fields nf, SqlId table);
bool InsertNoKey(Fields nf, const char *table);
bool InsertNoKey(Fields nf);
bool InsertNoKey(Fields nf, SqlId table);
bool InsertNoNulls(Fields nf, const char *table);
bool InsertNoNulls(Fields nf);
bool InsertNoNulls(Fields nf, SqlId table);
bool Update(Fields nf);
bool Update(Fields nf, const char *table);
bool Update(Fields nf, SqlId table);
bool Delete(const char *table, const char *key, const Valu
bool Delete(SqlId table, SqlId key, const Value& keyval);
Symbol/lineno (Ctrl+G)
* Upp::Sql::Cancel
Upp (sql/sqls.h)
Upp::SqlSession
Upp::SqlExc
Upp::SqlRaw
Upp::SqlColumnInfo
operator=(const S
SqlSource(const S
Upp::Sql
Sql : class
cn : SqlConnection
param : Vector<Va
Select0(const Stri
SetSession(SqlSou
Attach(SqlConnec
Detach() : void
Sql(SqlConnection
Compile(const Sql
Clear() : void
SetParam(int i, co
SetStatement(co
SetStatement(co
Execute() : bool
ExecuteX() : void
Run() : bool
RunX() : void
Execute(const Stri
ExecuteX(const St
Execute(const Sql
ExecuteX(const S
Run(const Value&
Execute(const Stri
Fetch() : bool
Fetch(Ref v1 [, Ref
Fetch(Vector<Valu
Fetch(ValueMap&
Fetch(Fields fields
GetRowsProcesse
sql/sqls.h (177)
Autos
Locals
this
Watches
CPU
Memory
Threads
0x1d80
Upp::Sql::Cancel()
cn
0
this
6a11420->{ cn=0, param=[0] {} }, { cn=f95eef2565657246, param=[1701147206] {{ chr=6a113c0 "", p

2) mysqlCrash2.PNG, downloaded 1100 times

```
Upp::Sql::Cancel() at C:\uppClang\upp\uppsrc\Sql\Sqls.h 177
Upp::SqlSession::SessionClose() at C:\uppClang\upp\uppsrc\Sql\Session.cpp 92
Upp::MySqlSession::Close() at C:\uppClang\upp\uppsrc\MySql\MySql.cpp 142
Upp::MySqlSession::Reconnect() at C:\uppClang\upp\uppsrc\MySql\MySql.cpp 79
ConsoleMainFn_() at C:\uppClang\upp\MyApps\mysqlCrash\mysqlCrash.cpp 51
Upp::AppExecute__(app=140002370) at C:\uppClang\upp\uppsrc\Core\App.cpp 442
main(argc=1, argv=18995d0->189f430 "C:\\uppClan") at C:\uppClang\upp\MyApps\mysqlCrash\mysqlCr
WinMainCRTStartup()
mainCRTStartup()
BaseThreadInitThunk()
RtlUserThreadStart()
```

Subject: Re: MySqlSession Close and Reconnect crashes Application
Posted by Alboni on Thu, 03 Dec 2020 18:55:45 GMT
View Forum Message <> Reply to Message

If you closed the session yourself you need to do a normal Connect() I think, not a Reconnect.

Reconnect is for when the session lost the connection from a network error etc. When that happens MySqlSession::WhenReconnect callback is activated.
The default behaviour is to call MySqlSession::Reconnect(), but you can set it to your own function from which you can call MySqlSession::Reconnect()

Subject: Re: MySqlSession Close and Reconnect crashes Application
Posted by mirek on Fri, 11 Dec 2020 10:11:55 GMT
View Forum Message <> Reply to Message

Alboni wrote on Thu, 03 December 2020 19:55If you closed the session yourself you need to do a normal Connect() I think, not a Reconnect.

Reconnect is for when the session lost the connection from a network error etc. When that happens MySqlSession::WhenReconnect callback is activated.
The default behaviour is to call MySqlSession::Reconnect(), but you can set it to your own function from which you can call MySqlSession::Reconnect()

That is correct, this was the intention of Reconnect. That said, expecting that you can call Reconnect after Close makes sense too, so I have plugged all holes (well, there was actually just one, in Cancel) to make possible, with one catch: Close unsets global SQL session (and that is probably correct behaviour), so you need to assign it again after Reconnect ("SQL = session").

Mirek

Subject: Re: MySqlSession Close and Reconnect crashes Application
Posted by hoelblin on Sun, 17 Jan 2021 20:30:53 GMT
View Forum Message <> Reply to Message

Sorry for my late Reply and thanks for all Answers !

I had now time for testing so I have written a small GUI Testapplication where I can CONNECT, DISCONNECT, RECONNECT, SELECT to a Mysql Database.

When I
 CONNECT
 SELECT
 .....
 .....
 DISCONNECT
 RECONNECT
 SELECT
 ....

all works well, but with
 CONNECT
 SELECT
 .....
 .....
 DISCONNECT
 CONNECT
 SELECT -> crashes in String0.isLarge

It is OK for me to use Reconnect but as Alboni and Mirek wrote Connect should also work.
So is still something wrong with my Connect Code ?

TheIDE 15596 from 2020-12-17 Windows with CLANG

Code of the Testapplication

```
mysqltest::mysqltest()
{
 CtrlLayout(*this, "Window title");

 acResult.AddColumn("Text");
 btConnect.WhenAction = THISBACK(mysqlConnect);
 btReConnect.WhenAction = THISBACK(mysqlReconnect);
 btDisconnect.WhenAction = THISBACK(mysqlDisconnect);
 btSelect.WhenAction = THISBACK(mysqlSelect);
```

```
 btConnect.Enable();
 btReConnect.Disable();
 btDisconnect.Disable();
 btSelect.Disable();
}

void mysqltest::mysqlConnect()
{
 if(session.Connect("user", "pw", "db", "localhost")) {
  SQL = session;

  btDisconnect.Enable();
  btSelect.Enable();
  btConnect.Disable();

  firstConnect = true;
 } else {
  Exclamation(session.GetLastError());
 }
}

void mysqltest::mysqlReconnect()
{
 if(!session.IsOpen()) {
  if(firstConnect) {
   if(session.Reconnect()) {
    SQL = session;

    btReConnect.Disable();
    btConnect.Disable();
    btSelect.Enable();
    btDisconnect.Enable();

    Exclamation("Reconnect succesfull!");
   } else {
    Exclamation(String() << "Reconnect failed with " << session.GetLastError());
   }
  } else {
   Exclamation("Use Connect first!");
  }
 }
}

void mysqltest::mysqlDisconnect()
{
 if(session.IsOpen()) {
  session.Close();
```

```
  acResult.Clear();

  if(!session.IsOpen()) {
   btConnect.Enable();
   btReConnect.Enable();
   btDisconnect.Disable();
   btSelect.Disable();

   Exclamation("Connection closed!");
  } else {
   Exclamation("Connection NOT closed!!!!!!!");
  }
 }
}

void mysqltest::mysqlSelect()
{
 if(session.IsOpen()) {

  try {
   Sql sql;
   sql.Execute("select name from names");

   if(sql.WasError()) {
    Exclamation(sql.GetLastError());
   } else {
    while(sql.Fetch()) {
     acResult.Add(sql[0]);
    }
   }
  } catch(SqlExc &ex) {
   Exclamation(ex);
  }
 } else {
  Exclamation("Database not opened!");
 }
}


GUI_APP_MAIN
{
 mysqltest().Run();
}
```

The Backtrace from the Debugger

Upp::String0::IsLarge() at C:\upp\uppsrc\Core\String.h 214

Upp::String0::Free() at C:\upp\uppsrc\Core\String.h 256
Upp::String0::Assign(s=) at C:\upp\uppsrc\Core\String.h 273
Upp::String::operator=(s=) at C:\upp\uppsrc\Core\String.h 376
Upp::Sql::SetStatement(s=) at C:\upp\uppsrc\Sql\Sql.cpp 105
Upp::Sql::Execute(s=) at C:\upp\uppsrc\Sql\Sql.cpp 166
mysqltest::mysqlSelect() at C:\upp\MyApps\mysqltest\main.cpp 83
Upp::callback<mysqltest,mysqltest>::<unnamed-tag>::operator()() at
C:\upp\uppsrc\Core\CallbackNP.i 38
Upp::Function<void ()>::Wrapper<`lambda at C:\upp/uppsrc\Core/CallbackNP.i:38:17'>::Execute()
at C:\upp\uppsrc\Core\Function.h 17
Upp::Function<void ()>::operator()() at C:\upp\uppsrc\Core\Function.h 76
Upp::Ctrl::Action() at C:\upp\uppsrc\CtrlCore\Ctrl.cpp 474
Upp::Pusher::PerformAction() at C:\upp\uppsrc\CtrlLib\Button.cpp 17
Upp::Pusher::FinishPush() at C:\upp\uppsrc\CtrlLib\Button.cpp 101
Upp::Pusher::LeftUp() at C:\upp\uppsrc\CtrlLib\Button.cpp 58
Upp::Ctrl::MouseEvent(event=145, p=, zdelta=0, keyflags=0) at
C:\upp\uppsrc\CtrlCore\CtrlMouse.cpp 169
Upp::Ctrl::MouseEvent0(event=145, p=, zdelta=0, keyflags=0) at
C:\upp\uppsrc\CtrlCore\CtrlMouse.cpp 92
Upp::Ctrl::MouseEventH(event=145, p=, zdelta=0, keyflags=0) at
C:\upp\uppsrc\CtrlCore\CtrlMouse.cpp 110
Upp::Ctrl::MEvent0(e=145, p=, zd=0) at C:\upp\uppsrc\CtrlCore\CtrlMouse.cpp 337
Upp::Ctrl::DispatchMouseEvent(e=145, p=, zd=0) at C:\upp\uppsrc\CtrlCore\CtrlMouse.cpp 598
Upp::Ctrl::DispatchMouse(e=145, p=, zd=0) at C:\upp\uppsrc\CtrlCore\CtrlMouse.cpp 588
Upp::Ctrl::DoMouse(e=145, p=, zd=0) at C:\upp\uppsrc\CtrlCore\Win32Wnd.cpp 571
Upp::Ctrl::WindowProc(message=514, wParam=0, lParam=5046318) at
C:\upp\uppsrc\CtrlCore\Win32Proc.cpp 136
Upp::TopWindow::WindowProc(message=514, wParam=0, lParam=5046318) at
C:\upp\uppsrc\CtrlCore\TopWin32.cpp 70
Upp::Ctrl::WndProc(hWnd=302bc, message=514, wParam=0, lParam=5046318) at
C:\upp\uppsrc\CtrlCore\Win32Wnd.cpp 674
CallWindowProcW()
DispatchMessageW()
Upp::Ctrl::sProcessMSG(msg=) at C:\upp\uppsrc\CtrlCore\Win32Wnd.cpp 770
Upp::Ctrl::ProcessEvent(quit=171e47f) at C:\upp\uppsrc\CtrlCore\Win32Wnd.cpp 794
Upp::Ctrl::ProcessEvents(quit=171e47f) at C:\upp\uppsrc\CtrlCore\Win32Wnd.cpp 808
Upp::Ctrl::EventLoop(ctrl=171e5e8) at C:\upp\uppsrc\CtrlCore\Win32Wnd.cpp 840
Upp::TopWindow::Run(appmodal=0) at C:\upp\uppsrc\CtrlCore\TopWindow.cpp 324
GuiMainFn_() at C:\upp\MyApps\mysqltest\main.cpp 103
Upp::AppExecute__(app=1400022e0) at C:\upp\uppsrc\Core\App.cpp 442
WinMain(hInstance=140000000, lpCmdLine=17cf286 "", nCmdShow=10) at
C:\upp\MyApps\mysqltest\main.cpp 101
WinMainCRTStartup()
WinMainCRTStartup()
BaseThreadInitThunk()
RtlUserThreadStart()

Subject: Re: MySqlSession Close and Reconnect crashes Application
Posted by mirek on Tue, 19 Jan 2021 08:28:42 GMT
View Forum Message <> Reply to Message

Could you please .zip the whole package and post here so that I do not need to invent the rest of the code?

Mirek

Subject: Re: MySqlSession Close and Reconnect crashes Application
Posted by hoelblin on Tue, 19 Jan 2021 21:13:36 GMT
View Forum Message <> Reply to Message

Here is the zipped Package.

hoelblin

File Attachments
1) mysqltest.7z, downloaded 354 times

Subject: Re: MySqlSession Close and Reconnect crashes Application
Posted by mirek on Thu, 21 Jan 2021 14:19:27 GMT
View Forum Message <> Reply to Message

Thank you. Should be now fixed, can you please check (trunk or nightly tomorrow)?

Mirek

Subject: Re: MySqlSession Close and Reconnect crashes Application
Posted by hoelblin on Sat, 23 Jan 2021 14:40:53 GMT
View Forum Message <> Reply to Message

Tested with nightly 15689 and all works fine.
Thank you very much