Subject: Lowercase field name in table - error
Posted by mezise on Fri, 29 Jun 2012 22:51:18 GMT
View Forum Message <> Reply to Message

I try to use Upp with existing database where there are lowercase field names and I get "Assertion failed" error in NEVER() line:

C:\uppSVN5061\uppsrc\Sql\Sql.cpp, line 339

```
Value Sql::operator[](SqlId id) const {
 String s = ~id;
 for(int i = 0; i < cn->info.GetCount(); i++)
  if(cn->info[i].name == s)
   return operator[](i);
 NEVER(); // <---- here ----
 return Value();
}
```

To repeat error:
- revision: SVN 5061
- use package: reference/SQL_Sqlite3
- in simple.sch: change ID to id
- in simple.cpp: change all ID to id
- Execute

Subject: Re: Lowercase field name in table - error
Posted by mirek on Wed, 04 Jul 2012 13:25:22 GMT
View Forum Message <> Reply to Message

Should be now fixed. Anyway, the cause is that you are using lower-cased SqlIds, which is somewhat, well, unexpected; DB column names are uppercased.

Subject: Re: Lowercase field name in table - error
Posted by mezise on Thu, 05 Jul 2012 07:49:31 GMT
View Forum Message <> Reply to Message

Thank you for the answer.
Unfortunately the error still appears in SVN5110, and revealed a new one during compilation:

C:\uppSVN5110\reference\SQL_Sqlite3\simple.cpp(17) : error C2039: 'LogErrors' : is not a member of 'Upp::Sqli
 te3Session'

C:\uppSVN5110\uppsrc\plugin/sqlite3/Sqlite3.h(19) : see declaration of 'Upp::Sqlite3Session'

After commenting

sqlite3.LogErrors(true);

lowercase field name error still appears.

PS. In sqlite and mysql worlds there are a lot of tables with lowercase field names

---

Subject: Re: Lowercase field name in table - error
Posted by mirek on Thu, 05 Jul 2012 08:01:46 GMT
View Forum Message <> Reply to Message

...but it was fixed in 5113 (With LogErrors as well).

See http://www.ultimatepp.org/redmine/projects/upp/repository

Mirek

Subject: Re: Lowercase field name in table - error
Posted by mezise on Thu, 05 Jul 2012 09:01:50 GMT
View Forum Message <> Reply to Message

Ups, sorry, I thought that last builds are on
http://www.ultimatepp.org/www$uppweb$nightly$en-us.html

Subject: Re: Lowercase field name in table - error
Posted by mirek on Thu, 05 Jul 2012 09:11:53 GMT
View Forum Message <> Reply to Message

Last builds yes, but they get build in night, not the moment I commit the change. Plus, sometimes nightly building fails...

Subject: Re: Lowercase field name in table - error
Posted by mezise on Thu, 05 Jul 2012 09:57:45 GMT
View Forum Message <> Reply to Message

I checked out from repository. Works like a charm. Thank you.

Subject: Re: Lowercase field name in table - error
Posted by Sender Ghost on Thu, 05 Jul 2012 16:32:25 GMT
View Forum Message <> Reply to Message

Hello, Mirek.
mirek wrote on Wed, 04 July 2012 15:25Should be now fixed. Anyway, the cause is that you are using lower-cased SqlIds, which is somewhat, well, unexpected; DB column names are uppercased.
From the source code point of view, there is no problem to create case-sensitive column names, at least for SQLite. The code, which uppercase column names, is on U++ side:

```
for(int i = 0; i < cn->info.GetCount(); i++)
 cn->info[i].name = ToUpper(cn->info[i].name);
```

Commenting such part of code removes "case-insensitive restriction" of reading column names (but not the other restrictions with the same case-insensitive characters, inside other U++ source code, e.g. id, ID).
In opposite of uppercase, I read about lowercase identifiers on MySQL documentation.
So, in conclusion, I think, it is related to various database support on U++ side, but not a real restrictions of concrete database (e.g. SQLite) or even C++ identifiers, which are case-sensitive.
Therefore, "the fix" fixes the effects, but not the cause, which created by some purpose.

Subject: Re: Lowercase field name in table - error
Posted by mezise on Thu, 05 Jul 2012 17:52:17 GMT
View Forum Message <> Reply to Message

Even with the fix, code such:

```
ValueMap row;
sql.Execute("SELECT COUNT(*) cnt FROM info");
sql.Fetch(row);
DUMP(row);
```

gives in log file:

```
row = { CNT: 12}
```

instead of:

```
row = { cnt: 12}
```

Commenting the code mentioned by Sender Ghost in <the cause> link fixes this issue, but I guess there is a reason for this code.

Subject: Re: Lowercase field name in table - error
Posted by mirek on Fri, 06 Jul 2012 07:05:54 GMT
View Forum Message <> Reply to Message

Sender Ghost wrote on Thu, 05 July 2012 12:32
In opposite of uppercase, I read about lowercase identifiers on MySQL documentation.

Actually, if anything, I believe that above link is more or less supportive to make everything uppercase.

If you consider

sql.Execute("SELECT COUNT(*) cnt FROM info");

then it is not unlikely that some DB returns "CNT".

As case sensitivity itself is DB dependent issue, having everything insensitive and upper-case on U++ side seems like the most prudent option.

Mirek

Subject: Re: Lowercase field name in table - error
Posted by Sender Ghost on Fri, 06 Jul 2012 08:25:59 GMT
View Forum Message <> Reply to Message

In case of SQLite, it returns original column name(s), independent from the query case-sensitivity:

```
sqlite> .header on
sqlite> .mode column
sqlite> create table SIMPLE_TEST1 (id integer primary key, NAME text, LASTNAME text, BDATE
integer);
sqlite> insert into SIMPLE_TEST1(id, NAME, LASTNAME, BDATE) values (0, 'Joe', 'Smith',
20000101);
sqlite> insert into SIMPLE_TEST1(id, NAME, LASTNAME, BDATE) values (1, 'Mike', 'Smith',
20000102);
sqlite> insert into SIMPLE_TEST1(id, NAME, LASTNAME, BDATE) values (2, 'Jon', 'Goober',
20000103);
sqlite> select id, NAME, LASTNAME, BDATE from SIMPLE_TEST1;
id          NAME        LASTNAME    BDATE
----------  ----------  ----------  ----------
0           Joe         Smith       20000101
1           Mike        Smith       20000102
2           Jon         Goober      20000103
```

```
sqlite> select ID, NAME, LASTNAME, BDATE from SIMPLE_TEST1;
id          NAME        LASTNAME    BDATE
----------  ----------  ----------  ----------
0           Joe         Smith       20000101
1           Mike        Smith       20000102
2           Jon         Goober      20000103
```

But yes, I'm aware about possible ambiguities/differences between database engines and why uppercase used instead of lowercase (e.g. because of global SqlId(s) and possible lowercase local variables inside developer's source code). While this didn't change the conclusion, in my opinion.

Personally, I thought about access to column names with the case-sensitivity of constructed query. I already did something inside another topic.

Nevertheless, I agree, that in the current constraints, what you did is "the most prudent option".