Subject: Re: PROPOSAL: Access to S_* Structure of TABLE crash Application.
Posted by Sender Ghost on Sat, 19 May 2012 01:12:10 GMT
View Forum Message <> Reply to Message

sergeynikitin wrote on Sat, 19 May 2012 02:02Talk about that (even in SQLite) you can execute the query

SELECT PLANT.NAME FROM PLANT
and it will be correctly understood by the SQL driver.
(By the way, - try make DUMP of NAME.Of(PLANT))

```
SQL * Select(PLANT(NAME)).From(PLANT);
LOG(SQL.ToString());
while (SQL.Fetch()) {
 DUMP(SQL.GetColumnInfo(0).name); // returns "NAME"
 DUMP(NAME.Of(PLANT)); // returns PLANT\nNAME
 DUMP(SQL[NAME]); // returns correct value, associated with "NAME" column
 DUMP(SQL[0]); // returns previous value
 DUMP(SQL[NAME.Of(PLANT)]); // "Assertion failed" here
}
```

With following output:

```
select PLANT.NAME from PLANT
SQL.GetColumnInfo(0).name = NAME
NAME.Of(PLANT) = PLANT
NAME
SQL[NAME] = First Plant
SQL[0] = First Plant
Assertion failed in C:\upp\uppsrc\Sql\Sql.cpp, line 339
0
```

sergeynikitin wrote on Sat, 19 May 2012 02:02I do not understand how the subject is connected with the function of sqlite3_column_name
The subject is connected with incorrect using of S_* structure, which throws "Assertion failed", because of requested empty column. The sqlite3_column_name function is related to SQLite database and how U++ wrapper uses it to get result set. Therefore, I said, that it is dependent from concrete database.

I think, you are free to propose your changes, if you want. So, U++ developers, which responsible to SQL parts of U++, might analyse it and return the answer to you.