Subject: Re: PROPOSAL: Access to S_* Structure of TABLE crash Application.
Posted by Sender Ghost on Fri, 18 May 2012 18:35:15 GMT
View Forum Message <> Reply to Message

sergeynikitin wrote on Thu, 17 May 2012 14:49Maybe enclose TABLE into separate class, and NAME clashes automatically gone away?
No. It depends on how concrete database works. In case of SQLite, the U++ wrapper uses sqlite3_column_name function to assign column names. From the documentation of sqlite3_column_name function we see:
Column Names In A Result Set[...]
These routines return the name assigned to a particular column in the result set of a SELECT statement.
[...]
The name of a result column is the value of the "AS" clause for that column, if there is an AS clause. If there is no AS clause then the name of the column is unspecified and may change from one release of SQLite to the next.
[...]

Therefore, to resolve column name clashes the user need to specify "AS" clause for clashing column names.
The correct example will look like follows:

```
LOG("Inserting values:");
SQL * Insert(WORKER)(ID, 0)(NAME, "Joe")(LASTNAME, "Smith")(PLANT_ID, 0);
LOG(SQL.ToString());
SQL * Insert(WORKER)(ID, 1)(NAME, "Mike")(LASTNAME, "Smith")(PLANT_ID, 0);
LOG(SQL.ToString());
SQL * Insert(WORKER)(ID, 2)(NAME, "Jon")(LASTNAME, "Goober")(PLANT_ID, 1);
LOG(SQL.ToString());

SQL * Insert(PLANT)(ID, 0)(NAME, "First Plant")(ADDRESS, "First st.");
LOG(SQL.ToString());
SQL * Insert(PLANT)(ID, 1)(NAME, "Second Plant")(ADDRESS, "Second st.");
LOG(SQL.ToString());

LOG("Selecting values:");
SQLID(PLANT_NAME);
SQL * Select(WORKER(NAME, LASTNAME), PLANT(NAME).As(PLANT_NAME),
PLANT(ADDRESS)).From(WORKER).LeftJoin(PLANT).On(WORKER(PLANT_ID) ==
PLANT(ID));
LOG(SQL.ToString());
while (SQL.Fetch()) {
 LOG("-----");
 DUMP(SQL[NAME]); DUMP(SQL[LASTNAME]);
 DUMP(SQL[PLANT_NAME]); DUMP(SQL[ADDRESS]);
}
```

With following output:

Inserting values:
insert into WORKER(ID, NAME, LASTNAME, PLANT_ID) values (0, 'Joe', 'Smith', 0)
insert into WORKER(ID, NAME, LASTNAME, PLANT_ID) values (1, 'Mike', 'Smith', 0)
insert into WORKER(ID, NAME, LASTNAME, PLANT_ID) values (2, 'Jon', 'Goober', 1)
insert into PLANT(ID, NAME, ADDRESS) values (0, 'First Plant', 'First st.')
insert into PLANT(ID, NAME, ADDRESS) values (1, 'Second Plant', 'Second st.')
Selecting values:
select WORKER.NAME, WORKER.LASTNAME, PLANT.NAME PLANT_NAME,
PLANT.ADDRESS from WORKER left outer join PLANT on WORKER.PLANT_ID = PLANT.ID

SQL[NAME] = Joe
SQL[LASTNAME] = Smith
SQL[PLANT_NAME] = First Plant
SQL[ADDRESS] = First st.

SQL[NAME] = Mike
SQL[LASTNAME] = Smith
SQL[PLANT_NAME] = First Plant
SQL[ADDRESS] = First st.

SQL[NAME] = Jon
SQL[LASTNAME] = Goober
SQL[PLANT_NAME] = Second Plant
SQL[ADDRESS] = Second st.