Subject: SQLite and foreign keys
Posted by wimpie on Thu, 17 Oct 2013 00:28:35 GMT
View Forum Message <> Reply to Message

Hello,

I am using SQLite, with a Schema with a few tables with references and I noticed I could delete rows in one table which were used in another.
After some digging I noticed in Sqlite3Schema.h the REFERENCES*** macro's are not defined.

First I wondered why, but after reading Sqlite doc's I think it's because it doesn't allow adding constraints to an existing table. Am I right?

But, it does allow it when creating the column, so I changed this in Sqlite3Schema.h:

#define REFERENCES(x) INLINE_ATTRIBUTE("references " #x)

And as far as I can see, this works. I think it can also work with CASCADE ("references " #x " on delete cascade") but I did not test that.

But it seems not to be the perfect solution because I can't reference TABLE.COLUMN; Sqlite gives error on the ".". it needs TABLE(COLUMN) to work.
This can be put in the schema also like
TABLE (TEST)
 INT (ID) PRIMARY_KEY
 INT (KEY1) REFERENCES(SIMPLE_TEST1(ID))
END_TABLE
but I guess this is not portable with other SQL engines?

Can someone please comment on this?

Subject: Re: SQLite and foreign keys
Posted by chickenk on Thu, 17 Oct 2013 08:13:34 GMT
View Forum Message <> Reply to Message

it seems that REFERENCES TABLE(COLUMN) is a portable construct:
http://www.w3schools.com/sql/sql_foreignkey.asp

Subject: Re: SQLite and foreign keys
Posted by chickenk on Thu, 17 Oct 2013 08:20:21 GMT
View Forum Message <> Reply to Message

Another note: foreign keys in sqlite3 are deactivated by default.

To have them enabled by default, one should define the CPP macro SQLITE_DEFAULT_FOREIGN_KEYS, see line 112284 of uppsrc/plugin/sqlite3/lib/sqlite3.c.

I use sqlite3 at work outside of U++, and my own solution is the following code (because I use a precompiled sqlite3 library), but I suppose defining SQLITE_DEFAULT_FOREIGN_KEYS is equivalent:

```
   int fkey_support;
   int sqlRet = sqlite3_open(path, &DB);
   /* [...] */
   if (sqlite3_db_config(DB, SQLITE_DBCONFIG_ENABLE_FKEY, 1, &fkey_support) !=
SQLITE_OK)
   {
      /* handle error */
   }
   printf("Foreign keys support is %s.\n", fkey_support?"ON":"OFF");
```

Subject: Re: SQLite and foreign keys
Posted by wimpie on Thu, 17 Oct 2013 08:33:54 GMT
View Forum Message <> Reply to Message

chickenk wrote on Thu, 17 October 2013 10:20Another note: foreign keys in sqlite3 are deactivated by default.

You are right; I forget to mention, I use "PRAGMA foreign_keys = ON" in my program.

Subject: Re: SQLite and foreign keys
Posted by mirek on Mon, 25 Nov 2013 19:05:22 GMT
View Forum Message <> Reply to Message

wimpie wrote on Wed, 16 October 2013 20:28Hello,

I am using SQLite, with a Schema with a few tables with references and I noticed I could delete rows in one table which were used in another.
After some digging I noticed in Sqlite3Schema.h the REFERENCES*** macro's are not defined.

First I wondered why, but after reading Sqlite doc's I think it's because it doesn't allow adding constraints to an existing table. Am I right?

Actually, it is rather because nobody bothered so far to implement it

In fact, the issue is in RM for 2 years already:

http://www.ultimatepp.org/redmine/issues/172

I did not know there is problem with adding references later, if that is true, it really is kind of problem...

Mirek
