
Subject: Re: Sqlite file locking - problems to handle
Posted by [mirek](#) on Sat, 16 Apr 2011 19:26:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

papascalientes wrote on Mon, 04 April 2011 08:18

1. in U++ I have a session object (z_sql3 in the example) and an Sql object (sqlz(z_sql3) in the example). In Sqlite documentation they are always referring to transactions. What corresponds to the transaction in U++, the session object or the sql object?

Session.

Quote:

1a) I can use Commit(), Begin(), Clear(), ... with the session object or with the SQL object. Is there any difference between using the methods with the session object and using the the methods with SQL object or is it the same?

No. Sql calls session Commit. Note: Not Clear, I believe you have meant "Rollback"

Quote:

2. (more Sqlite specific question/s): If I start a Sqlite transaction with Begin() in U++, does the transaction persist

- only when a Commit() failed because of a shared lock
- always if a Commit() failed

And related questions:

2a) If I Close() the session, what happens to the transaction that was started with Begin() and is closed after a successful Commit() but may still be open if the Commit() failed

2b) What happens to the Journal, if it is not possible to write to the database. When and how is it closed/destroyed?

This is very Sqlite specific. I think you should look into Sqlite docs..

Quote:

And of course I am also grateful for general tips how to handle file locking and concurrency database operations in U++.

Frankly, for concurrency, I would rather used different DB...
