Subject: How SQL deal with multiple threads?
Posted by akebee on Mon, 10 Jul 2017 09:18:42 GMT
View Forum Message <> Reply to Message

The tutorial "Skylark09" shows "session per-thread" mode to deal with multiple threads,

So, eath thread each sql session, but the SQL is a global main database, i think is a global session, how it deal with multiple threads? :blush:

Subject: Re: How SQL deal with multiple threads?
Posted by mr_ped on Mon, 10 Jul 2017 09:28:48 GMT
View Forum Message <> Reply to Message

replying just from memory, without checking anything, so I may be wrong...

IIRC "SQL" is macro hiding the default SqlSession instance, so it's probably not supposed to be used in multi-thread source, you should use the underlaying SqlSession instances themselves. But if that example is doing it, then it's probably ok, and the SQL definition accounts for it? But I don't think so.

Here's the docs about SqlSession for further reading (I'm not going to, sorry :)):
http://www.ultimatepp.org/src$Sql$SqlSession$en-us.html

Subject: Re: How SQL deal with multiple threads?
Posted by mirek on Mon, 10 Jul 2017 09:59:32 GMT
View Forum Message <> Reply to Message

Check docs for void SqlSession::PerThread(bool b = true)

means you can choose:

- either you have single global SQL object (and need to serialize access to it)
- or each thread has its own one (but of course, then each thread needs to create and open its DB connection)

Subject: Re: How SQL deal with multiple threads?
Posted by akebee on Mon, 10 Jul 2017 10:00:41 GMT
View Forum Message <> Reply to Message

if the SQL not support multi-thread the SqlExp would be have no use in multi-thread? :?

Subject: Re: How SQL deal with multiple threads?
Posted by akebee on Mon, 10 Jul 2017 10:26:56 GMT
View Forum Message <> Reply to Message

yeah, i understand a little
the SqlSession::PerThread set sPerThread

```
thread__ SqlSession *sThreadSession;
...
 #ifdef _MULTITHREADED
  if(sPerThread)
   sThreadSession = &s;
  else
 #endif
   sGlobalSession = &s;
  return;
```

i don`t see how each thread has a sThreadSession ?

Subject: Re: How SQL deal with multiple threads?
Posted by mirek on Mon, 10 Jul 2017 10:28:53 GMT
View Forum Message <> Reply to Message

akebee wrote on Mon, 10 July 2017 12:00if the SQL not support multi-thread

I am not sure where you got that idea...

Sql fully supports MT.

Only thing for you to decide is whether 'default' SQL session is really 'global' (single one in the process), or you have an option to have SQL per-thread. By SQL in this paragraph I mean 'SQL default session object'.

Note that even if you opt for single global SQL, multithreading is still a go. You can either open other session in thread, or you can even use that single global SQL, but need to mutex it.

Subject: Re: How SQL deal with multiple threads?
Posted by mirek on Mon, 10 Jul 2017 10:30:23 GMT
View Forum Message <> Reply to Message

akebee wrote on Mon, 10 July 2017 12:26yeah, i understand a little
the SqlSession::PerThread set sPerThread

```
thread__ SqlSession *sThreadSession;
```

```
...
 #ifdef _MULTITHREADED
  if(sPerThread)
   sThreadSession = &s;
  else
 #endif
   sGlobalSession = &s;
  return;
```

i don`t see how each thread has a sThreadSession ?

Because 'thread__' ?

Subject: Re: How SQL deal with multiple threads?
Posted by akebee on Mon, 10 Jul 2017 11:26:15 GMT

View Forum Message <> Reply to Message

oh yes, it is!!

```
#ifdef DEPRECATED
#define thread__ thread_local
#endif
```