

# Can I make Perl ithreads in Windows run concurrently?

Asked 16 years, 3 months ago   Modified 16 years ago   Viewed 1k times



2



I have a Perl script that I'm attempting to set up using Perl Threads (use threads). When I run simple tests everything works, but when I do my actual script (which has the threads running multiple *SQLPlus sessions*), *each SQLPlus session* runs in order (i.e., thread 1's sqlplus runs steps 1-5, then thread 2's sqlplus runs steps 6-11, etc.).

I thought I understood that threads would do concurrent processing, but something's amiss. Any ideas, or should I be doing some other Perl magic?

perl

multithreading

concurrency

Share

Improve this question

Follow

edited Nov 28, 2008 at 18:31



brian d foy

132k ● 31 ● 211 ● 604

asked Sep 17, 2008 at 18:37



Milner

638 ● 7 ● 19

---

I, and would guess others, are curious to hear some follow-up on this question. Did you find a solution? Were the answers here helpful? – [Frosty](#) Sep 18, 2008 at 19:47

---

What Frosty said... What was the problem? – [Daren Thomas](#) Sep 19, 2008 at 6:51

---

### 3 Answers

Sorted by:

Highest score (default)



4



A few possible explanations:

1. Are you running this script on a multi-core processor or multi-processor machine? If you only have one CPU only one thread can use it at any time.
2. Are there transactions or locks involved with steps 1-6 that would prevent it from being done concurrently?
3. Are you certain you are using multiple connections to the database and not sharing a single one between threads?

Share Improve this answer

answered Sep 17, 2008 at 19:20

Follow



[Frosty](#)

6,443 ● 3 ● 26 ● 20

---

There was only one DB connection happening, which wasn't what I expected. Thanks for the idea (never thought to check that), now I have work to do... – [Milner](#) Sep 22, 2008 at 17:58

---



2

Actually, you have no way of guaranteeing in which order threads will execute. So the behavior (if not what you expect) is not really wrong.



I suspect you have some kind of synchronization going on here. Possibly SQL\*Plus only let's itself be called once? Some programs do that...



Other possibilities:

- thread creation and process creation (you *are* creating subprocesses for SQL\*Plus, aren't you?) take longer than running the thread, so thread 1 is finished before thread 2 even starts
- You are using transactions in your SQL scripts that force synchronization of database updates.

Share Improve this answer

answered Sep 17, 2008 at 18:45

Follow



[Daren Thomas](#)

70.2k ● 42 ● 155 ● 205



1

Check your database settings. You may find that it is set up in a conservative manner. That would cause even minor reads to block all access to that information.



You may also need to call `threads::yield`.



Share Improve this answer

edited Nov 29, 2008 at 16:55

Follow



answered Sep 18, 2008 at 4:02



**Brad Gilbert**

**34.1k** ● 11 ● 79 ● 130

---