Chapter 50

perlthrtut

Tutorial on threads in Perl

50.1 DESCRIPTION

NOTE: this tutorial describes the new Perl threading flavour introduced in Perl 5.6.0 called interpreter threads, or **ithreads** for short. In this model each thread runs in its own Perl interpreter, and any data sharing between threads must be explicit.

There is another older Perl threading flavour called the 5.005 model, unsurprisingly for 5.005 versions of Perl. The old model is known to have problems, deprecated, and will probably be removed around release 5.10. You are strongly encouraged to migrate any existing 5.005 threads code to the new model as soon as possible.

You can see which (or neither) threading flavour you have by running `perl -V` and looking at the `Platform` section. If you have `useithreads=define` you have ithreads, if you have `use5005threads=define` you have 5.005 threads. If you have neither, you don't have any thread support built in. If you have both, you are in trouble.

The user-level interface to the 5.005 threads was via the *Threads* class, while ithreads uses the *threads* class. Note the change in case.

50.2 Status

The ithreads code has been available since Perl 5.6.0, and is considered stable. The user-level interface to ithreads (the *threads* classes) appeared in the 5.8.0 release, and as of this time is considered stable although it should be treated with caution as with all new features.

50.3 What Is A Thread Anyway?

A thread is a flow of control through a program with a single execution point.

Sounds an awful lot like a process, doesn't it? Well, it should. Threads are one of the pieces of a process. Every process has at least one thread and, up until now, every process running Perl had only one thread. With 5.8, though, you can create extra threads. We're going to show you how, when, and why.

50.4 Threaded Program Models

There are three basic ways that you can structure a threaded program. Which model you choose depends on what you need your program to do. For many non-trivial threaded programs you'll need to choose different models for different pieces of your program.