

Параллельные вычисления Многопоточное программирование, ответы на вопросы

Созыкин Андрей Владимирович

К.Т.Н.

Заведующий кафедрой высокопроизводительных компьютерных технологий Институт математики и компьютерных наук



Как выполняется «планирование» потоков

В С++11 нет платформонезависимого способа:

- Потоки управляются средствами ОС
- В Linux используется библиотека PThreads
- Начиная с Linux kernel 2.6 используется реализация Native POSIX Thread Library (NPTL)

u1213@asozykin:~\$ getconf GNU_LIBPTHREAD_VERSION
NPTL 2.19



NTPL

Task – единица, с которой работает планировщик

- Процесс
- Поток (легковесный процесс, lightweight process)

Процессы и потоки равнозначны при распределении ресурсов

- Процесс 1 1 поток
- Процесс 2 3 потока
- Одинаковый приоритет



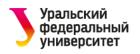
NTPL

Task – единица, с которой работает планировщик

- Процесс
- Поток (легковесный процесс, lightweight thread)

Процессы и потоки равнозначны при распределении ресурсов

- Процесс 1 1 поток: 25% процессорного времени
- Процесс 2 3 потока: 75% процессорного времени
- Одинаковый приоритет



Как посмотреть потоки в Linux

Команда ps, опция L (lightweight process)

```
u1213@asozykin:~$ ps -fLu mysql
                        LWP
UID
           PID
                PPID
                             C NLWP STIME TTY
                                                         TIME CMD
mysql
          1270
                    1
                       1270
                              0
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
          1270
mysql
                    1
                       1324
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
mysql
          1270
                    1
                       1325
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
          1270
                       1326
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
mysql
                    1
          1270
                       1327
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
mysql
                    1
          1270
                       1328
                                                     00:00:00 /usr/sbin/mysqld
mysql
                    1
                                  17 22:58 ?
          1270
                       1329
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
mysql
                    1
          1270
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
mysql
                       1330
                    1
mysql
          1270
                       1331
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
                    1
          1270
                       1332
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
mysql
                    1
                                  17 22:58 ?
mysql
          1270
                    1
                       1333
                                                     00:00:00 /usr/sbin/mysqld
mysql
          1270
                    1
                       1371
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
mysql
          1270
                       1372
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
                    1
          1270
                       1373
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
mysql
                    1
                       1374
                                  17 22:58 ?
mysql
          1270
                    1
                                                     00:00:00 /usr/sbin/mysqld
mysql
          1270
                    1
                       1402
                                  17 22:58 ?
                                                     00:00:00 /usr/sbin/mysqld
mysql
                                                     00:00:00 /usr/sbin/mysqld
          1270
                       1410
                                  17 22:58 ?
```



Как привязать потоки к процессорам

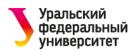
```
u1213@asozykin:~$ sudo taskset -p -c 0,1 1330 pid 1330's current affinity list: 0-7 pid 1330's new affinity list: 0,1
```



Как установить приоритет потока

В С++11 нет платформонезависимого способа Через native_handle (для pthread): std::thread t1(f, 1), t2(f, 2); sched param sch; int policy; pthread getschedparam(t1.native handle(), &policy, &sch); sch.sched priority = 20; if(pthread_setschedparam(t1.native_handle(), SCHED_FIFO, &sch)) { std::cout << "Failed to setschedparam: " <<</pre> std::strerror(errno) << '\n';</pre> t1.join(); t2.join();

http://en.cppreference.com/w/cpp/thread/thread/native_handle



Вопросы?