MINISTRY FOR EDUCATION AND SCIENCE OF RUSSIA

SAINT PETERSBURG ELECTROTECHNICAL UNIVERSITY
«LETI»

CONSTRUCTION AND OPTIMISATION THE ALGORITHME

REPORT ABOUT MASTER/SLAVE PATTERN IN PARALLEL COMPUTER

3 pages / страниц
объём документа

St. Petersburg / Санкт-Петербург
2021

CONTENT

I. INTRODUCTION

The master-slave paradigm involves two sets of processors. The master processors that are responsible for pre- and post-processing of work orders, and the slave processors that are responsible for the actual execution of the orders." Application of this include parallel computing, semiconductor testing and problems in transportation as will be described shortly".

In the master/slave paradigm, one master node generates and allocates work, usually called tasks (pre or post processing task that must be executed in this order), to N nodes(is a basic unit of a data structure, such as a linked list or tree data structure).

-Explanation

Each node receive tasks, computes them, and sends the result back. Thus, master node is responsible for receiving and analyzing the computed results by slave nodes. This paradigm is generally suitable for shared-memory or message-passing platforms, since the interaction is naturally two-way.
One of the main performance issues in parallel programming is the choice between many small tasks (fine grain) or few large tasks (coarse grain
First we give a brief description of the model under consideration. A set of jobs is to be processed by a system of master and slave processor. Each job has three tasks associated with it.

The first is a preprocessing task, the second is a slave task, and the third a post processing task. The tasks of each job are to be performed in the order: preprocessing, slave, post processing.

- problem

The main problem with M/S paradigm is that the master may become a bottleneck.
This may happen if the tasks are too small or if there are too many slaves connected to one master. The choice of the amount of work in each is called granularity. Between many small tasks or few large tasks. According to how the master distributes tasks and collects results from the slaves, we present two different kinds of master/slave implementations: programs with synchronous or asynchronous interaction. A program is said to have synchronous interaction when the tasks need to be performed in phases, i.e., all tasks in each phase must finish before the distribution of next phase tasks. In opposition, the interaction between the master and the slaves is called asynchronous when the master distributes new tasks every time a slave finishes its computation. Such interaction may reduce waiting time considering a nondeterministic distribution of slave requests (tasks have an unknown computational cost). In this context, the master node usually implements awaiting buffer in order to deal with simultaneous arrivals of results from slaves.

- pseudo code

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/type>
#include <unistd.h>

int main(int, char const *[])
{
  int master, slave;
  char name[1024];
```

```
  char mode[] = "0777"; //I know this isn't good, it is for testing at the
moment
  int access;

  int e = openpty(&master, &slave, &name[0], 0, 0);
  if(0 > e) {
    std::printf("Error: %s\n", strerror(errno));
    return -1;
  }

  if( 0 != unlockpt(slave) )
  {
      perror("Slave Error");
  }

  access = strtol(mode, 0, 8);

  if( 0 > chmod(name, access) )
  {
      perror("Permission Error");
  }

  //std::cout << "Master: " << master << std::endl;
  std::printf("Slave PTY: %s\n", name);

  int r;
  prompt = "login: ";

  while(true)
  {
    std::cout << prompt << std::flush;
    r = read(master, &name[0], sizeof(name)-1);
    checkInput(name);
    name[r] = '\0';
    std::printf("%s", &name[0]);
    std::printf("\n");
  }

  close(slave);
  close(master);

  return 0;
}
```

II. CONCLUSION

Master/slave is a model of asymmetric communication or control where one device or process (the "master") controls one or more other devices or processes (the "slaves") and serves as their communication hub. In some systems, a master is selected from a group of eligible devices, with the other devices acting in the role of slaves. Historically, the master/slave terminology has existed for decades, although in the 21st century it has been a subject of controversy due to its association with slavery, and some organizations and products have since replaced it with alternative terms.