**Perfect job scheduler Algorithm**

**Rui Yan 44486375**

# Introduction

Stage 3 is to design and implement a new scheduling algorithm to "optimize" one or more goals. It balances more indicators and arranges more work effectively. A good algorithm is to make the application better so stage 3 work optimization stage 2 algorithm. I will call it perfect fit (PF).

# Problem definition

Load balancing is based on the existing network structure, which provides a cheap, effective and transparent way to expand the bandwidth of network equipment and servers, increase throughput, strengthen the capacity of network data processing, and improve the flexibility and availability of the network.

The meaning of load balance is to allocate to multiple operation units for execution, such as web server, FTP server, enterprise key application server and other key task servers, to complete the work together

# Algorithm Description

## Definition

Load balance means to balance the load (work task) and allocate it to multiple operation units for operation, such as FTP server, web server, enterprise core application server and other main task servers, so as to complete the work tasks together.

Load balancing is built on the original network structure, which provides a transparent, cheap and effective way to expand the bandwidth of servers and network equipment, strengthen the network data processing capacity, increase throughput, improve the availability and flexibility of the network.

**3.2 Algorithm example:**

FF is an algorithm of scheduling server tasks according to priority. The disadvantage of FF is that it is the first one to obtain according to the priority of resources. Such an algorithm cannot guarantee the optimal utilization of system and server resources, BF algorithm is based on the strategy of optimizing resource utilization, but only two factors are considered, so this algorithm is not a comprehensive resource optimization algorithm, so we need to find more optimized algorithm to improve server utilization and I optimize algorithm by combining FF and BF.

# Implementation details

The algorithm optimization part is analyzed in detail. First of all, if different types of skips are used, if they are of the same type, the FF algorithm is the same. Select one. Then if multiple factor comparisons are of high resource utilization, update the selected server.

PFClient.java

private void cntj(Job job) throws Exception {

      String send, recv;

      Boolean find = false;

      int bestFit = Integer.MAX\_VALUE;

      int minAvail = Integer.MAX\_VALUE;

      int wstFit = Integer.MAX\_VALUE;

      int bestMemoryFit = Integer.MAX\_VALUE;

      Server select = null;

    Server selectW = null;

    for(Type type: this.system.getTypeList()) {

      for(Server server: type.getServers()) {

        send = server.getCNTJ();

        this.sendMsg(send);

        recv = this.recvMsg();

        int state = server.getState();

        int fit = server.fit(job);

        int fitMemory = server.fitMemory(job);

        int fitDisk = server.fitDisk(job);

        if(server.available(job)) {

          int availTime = server.getAvailTime();

            if ((fit < bestFit)||(availTime < minAvail)||(fitMemory < bestMemoryFit)) {

            bestFit = fit;

            minAvail = availTime;

            bestMemoryFit = fitMemory;

            select = server;

            find = true;

          }

        } else if (type.available(job)) {

            fit = type.computeFitness(job);

            if (fit < wstFit) {

                wstFit = fit;

                selectW = server;

            }

        }

      }

    }

      if(find) {

        send = String.format("SCHD %d %s", job.getID(), select.show());

        this.sendMsg(send);

        this.recvMsg("OK");

      } else {

        send = String.format("SCHD %d %s", job.getID(), selectW.show());

        this.sendMsg(send);

        this.recvMsg("OK");

      }

    }

1. **User Guide**
2. make
3. ./test.sh TestPF.class pf

# Summary

The algorithm world is only better, there is no best, for the resource scheduling problem is the same. According to the actual situation, the optimization algorithm is also a process of gradual improvement, which cannot be modified without theoretical basis.