Atitit node.js多线程解决方案

目录

[1. 多线程 通过制定线程数和线程id来自动分配多想出现 1](#_Toc16962)

[2. Cluseer模式使用了，但是不起作用 2](#_Toc22672)

# 多线程 通过制定线程数和线程id来自动分配多想出现

// node  D:\prj\spdJs\geneListPageFiles.js 2 0

var threadCount =parseInt( process.argv[2] )

var threaid = parseInt (process.argv[3] )

  var taskNum = 0;

  for (catid of arr) {

    nm++;

    // if(nm<paramStartIdex)

    //   continue

    for (i = 500; i <= 700; i++) {

      //task palam lib

      if (threadCount > 1)  //multi thread

      {

        taskNum++;

        var modRzt = taskNum % threadCount;

        if (modRzt != threaid)

          continue;

      }

      //task palam lib  end

# Cluseer模式使用了，但是不起作用

(async () => {

  await appMain();

  // await clusefun();

})();

async function clusefun() {

  var cluster = require('cluster');

  var numCPUs = require('os').cpus().length; // 获取CPU的个数

  console.log("numCPUs:" + numCPUs)

  if (cluster.isMaster) {

    console.log(`main主进程 ${process.pid} 正在运行`);

    for (var i = 0; i < 2; i++) {

      cluster.fork();

    }

    cluster.on('exit', function (worker, code, signal) {

      console.log('worker ' + worker.process.pid + ' died');

    });

  } else {

    // sub process

    //appMain()

    console.log(`工作进程 ${process.pid} start  启动`);

    //require("./geneListPageFiles.js");

    await appMain();

    console.log(`工作进程 ${process.pid} 已启动`);

  }

  console.log(`main主进程 ${process.pid} 正在运行 sec`);

}

# 线程池好像没有现成的组件，要自己写，pass

## Cluseer 取代process child功能