项目简介-技术组

cpp-scutter系统简介

林行健

厦门大学　计算机系

2016年12月9日

目录

目录

需求分析

- 运行爬虫
- 统一调度多台服务器
- 自动化维护过程

目录

系统简介

- 爬虫组件：运行爬虫
  - 爬虫模块(Crawler)：运行爬虫
  - 代理模块(ProxiesPool)：设置HTTP请求头和代理
  - 日志生成模块(ScutterLog)：封装日志模块，便于调用
- 分布式组件：统一调度多台服务器
  - 机器调度模块(Work)：针对服务器扮演角色进行封装
  - 消息队列模块(RedisMQ)：以redis做中间件，维护消息队列
  - 脚本调用模块(ShellServer)：封装为shell脚本，用于linux环境下的调用
- 自动化运维组件：自动化维护过程
  - 配置模块(config.py+setting.py)：参数设置
  - 自动化运维模块(fabfile.py)：批量运维，一键对多台服务器进行调度

目录

爬虫模块(Crawler)

- 天猫爬虫(TmallScraper)
  - 天猫关键词列表(TmallCategories)：基于中国官方2011版CPI分类编制的商品关键词列表
  - 天猫搜索页爬虫(TmallPageScraper.py)：抓取天猫搜索页输入关键词后对应搜索结果
- 京东爬虫(JDScraper)
  - 京东关键词列表(JDCategories)：类似天猫关键词列表
  - 京东搜索页爬虫(JDPageScraper.py)：类似天猫搜索页爬爬虫
  - 京东详情页爬虫(JDScraperDetail.py)：抓取京东商品详情页商品ID对应数据

目录

代理模块(ProxiesPool)

- 请求头(headers.py)：构造HTTP请求报文的头部
- 代理IP(CheckedProxies)：储存的从代理IP网站抓取的代理IP池

目录

日志生成模块(ScutterLog)

- 日志生成模块(log.py): 重写Python自带的log模块, 使其更具有灵活性.

目录

master-slaver分布式系统

- 整个系统由一台master, 多台slaver组成
- master负责生成任务消息, 并将任务消息加入到中间件的消息队列中
- slaver从消息队列中抓取任务

机器调度模块(Work)

- master调度(master.py): 用于master机器上的调度, 包括生产者和管理者
- slaver调度(slaver.py): 用于slaver机器上的调度, 为消费者
- 爬虫任务分配(distributor.py): 用于分配节点任务
- 爬虫任务监控(procmon.py): 用于监控爬虫进程是否结束

目录

消息队列

- 原理：
  - 基于内存或磁盘的队列
  - 发送消息或者读出消息
  - 提供信息交换
- 功能：
  - 要求服务
  - 交换信息
  - 异步处理

生产者-消费者模型

- 生产者：生产数据，并扔给消息队列
- 消费者：从消息队列里取数据，并处理数据
- 消息队列：缓冲区，平衡生产者和消费者的处理能力

消息队列模块(RedisMQ)

- 建立redis连接(redispool.py)：连接redis
- 管理者(manager.py)：统计一天结束后的爬虫情况
- 消费者(consumer.py)：建立消费者模型，从redis的消息队列中取出url，进行爬取
- 生产者(producer.py)：建立生产者模型，生成url，将url加入redis的消息队列中

目录

脚本调用模块(ShellServer)

- addcron.sh: 添加定时任务脚本
- compress.sh: 压缩文件脚本
- init.sh: 初始化脚本
- kill.sh: 结束爬虫进程脚本
- manager.sh: 运行管理者脚本
- procmon.sh: 监控进程脚本
- repeat.sh: 多次运行爬虫脚本
- restart.sh: 重启爬虫脚本
- setlink.sh: 建立log软连接脚本
- slaver.sh: 运行爬虫脚本

目录

配置模块(config.py+setting.py)

- 不可变配置(config.py)：如日期格式, 凭证等
- 可设置配置(setting.py)：如redis地址, 服务器信息等

目录

自动化运维模块(fabfile.py)

- deploy: 批量部署指令
- restart: 爬虫由于意外中断重启
- repeat: 多次爬虫
- kill: 结束爬虫进程
- add_cron: 添加定时任务


CPP