CrawlerSYS分布式爬虫设计文档

### 一、题目分析及解题思路

**爬虫**

1. **爬虫策略**

下载：使用jsoup高效下载页面，并根据用户配置开启足够多的线程，或是自动识别博客、新闻的标题、正文信息。

反爬虫：用户需根据需要自行添加header和cookie信息，如未添加则使用默认的伪造window下浏览器信息。

1. **URL去重算法**

任何节点爬取到的需要爬取的网站URL经过筛选后将统一返回主节点，并直接存储至set中，再进行节点分配。有效避免了重复url的出现。

**分布式调度算法**

1. **爬虫任务**

将获取到的需要爬取的url根据用户设置的单次节点爬取量，分批发送给对应节点。即保证所有节点近似均匀分配，又兼顾了爬虫的高效和软件的轻量化。

1. **分布式调度策略**

本爬虫为用户提供了4种不同的调度方式，已满足不同的场景需求。例如，有的任务必须在特定日期执行可以使用fixTimeRun，有的任务需要在另一个任务之后执行则使用run。

**网页自动结构化**

对于新闻博客类网页，能进行网页正文的自动抽取，对标题和正文进行自动摘要。

### 二、软件支持

|  |  |  |  |
| --- | --- | --- | --- |
| 开发环境 | Window 10、MyEclipse 10、Tomcat 6、Java EE 6、MySQL 5.5 | | |
| 实验平台 | CentOS 6.5、Java JDK 1.8、Tomcat 8、MySQL 5.6 | | |
| 开发语言 | Java、JavaScript、HTML、CSS | | |
| 第三方代码 | Jsoup | 1.8.3 | https://github.com/jhy/jsoup |
| Xsoup | 0.3.1 | https://github.com/code4craft/xsoup |
| Log4j | 1.2.17 | https://github.com/apache/log4j |
| HikariCP | 3.16 | https://github.com/brettwooldridge/HikariCP |
| boilerpipe | 0.1.0 | https://github.com/kohlschutter/boilerpipe |
| Bootstrap | 3.3.7 | https://github.com/twbs/bootstrap |

### 三、软件结构

### 

**Crawler**

包括API接口和静态默认配置文件，详情请参考API文档。

**Entity**

是爬虫节点经过爬取后返回主节点的数据包实例，也是节点进行后续处理的单位，详情请参考API文档。

**Main**

包括主节点从初始化到向个节点发送请求并接收返回的数据包及数据相关处理的所有文件。

**Node**

依次包括：新闻博客网站的自动提取进程、爬虫节点的接收请求服务器、爬取后的数据处理接口、节点接收程序、请求分析及判断网站类型启动相应爬取程序、普通单独页面的数据爬取进程、节点服务器启动程序。单独页面爬取后将反射调用数据处理接口，进行处理及数据持久化操作，如未添加自定义处理类，则使用程序内置的数据持久化模块，详情请参考API文档。

**Utils**

所有静态工具类，依次包括：数据库访问工具、字符串处理工具、页面下载及处理工具。

**Servlet**

爬虫控制界面的响应服务器和爬虫状态监控长连接服务器。

**Test**

文件读取、分析、输出程序和测试类。

**Content**

爬虫控制界面的额外文件，依次包括：开关的样式、开关的动画及功能、输入框组的生成、主要脚本文件、主要样式文件、xpath自动提取工具。

**Page**

所有JSP文件，依次包括：页面预览浏览器界面、爬虫控制台界面、爬虫基本配置界面。