南華大學
UNIVERSITY OF SOUTH CHINA

密级：普通

文档编号：20170630-SRS　　版次：2

分册名称：　　第 1 册/共 1 册

分布式爬虫

（20170630）

需求规格说明书

编写　：　刘宝玉

校对　：　王浩茂　杨奎

审核　：　刘立

批准　：　刘立

生效日期　：　2017 年 6 月 30 日

南华大学计算机学院制

2017 年 6 月

文件修改控制

修改记录编号	修改状态	修改页码及条款	修改人	审核人	批准人	修改日期
001	已修改	第 5 页参考资料和第 6 页术语	王浩茂	王浩茂	刘宝玉	2017/06/28

目录

1. 引言

1.1. 目的

为了保证项目团队按时保质地完成项目目标，便于项目团队成员更好地了解项目情况，使项目工作开展的各个过程合理有序，因此以文件化的形式，把开发过程中各项工作的人员、分工、经费、系统资源条件等问题的安排记录下来，作为项目团队成员以及项目干系人之间的共识与约定，项目团队开展和检查项目工作的依据，以便计划开展和确保项目开发成功。

1.2. 背景

互联网是企业进行发布信息的渠道，是个人共享和获取信息的工具，同时也为政府提供了大量有价值的信息，用于监管企业和个人。政府有效的利用互联网的信息，能发现舆论倾向，建立征信体系，发现犯罪行为等。

电商网站是个体户及企业进行网上销售的平台。电商网站中的数据具有重要的价值，能体现经济发展趋势，居民消费水平等。而电商网站具有以下特点：数据变化极快，时效性极高；不同网站数据组织不同，分类标签不同；网站的反爬虫机制较强；每个页面被多个页面链接，重复链接多

导致电商网站采集具有以下问题：

爬虫被反爬机制屏蔽；采集周期较长；需为不同的网站定制实现程序，进行结构化，人工成本较高；页面链接去重也影响采集效率

因此，对电商网站的高效的采集、并且能自动的（尽量减少人工的）提取网页中的数据，是具有价值和挑战性的。

此外，互联网上的数据内容丰富，组织形式也灵活多样。传统的爬虫系统，对所有的网页采用同样的办法处理，利用深度优先或广度优先的办法获取网页链接，下载网页，对网页中的所有的文本数据建立倒排索引。这种方式没有对网页数据的信息进行组织、归类。应大数据的需求，分布式爬虫系统是解决这一问题的方案。分布式爬虫，对同一个网站的同类数据，进行结构化。同时，能利用分布式的软件设计方法，实现爬虫的高效采集。

1.3. 参考资料

书目/文献名	作者	编号	出版时间	出版社
selenium webdriver 基于 Python 源码案例	葵花、上海-悠悠	/	/	百度阅读
廖雪峰的博客	廖雪峰	/	/	/

1.4. 术语

- 爬虫：一种自动获取网页内容的程序。
- 爬虫任务：可以理解为对一个或多个网站的一次采集过程。
- 分布式爬虫：将所有任务在多台机器上分布式执行。
- 分布式调度:将不同网站的 URL 混合后,分配到多台机器上执行。
- 网页自动结构化：对于电商类网页，能对同一个网站的数据进行自动结构化，生成不同的表，例如商品表、店铺表、评价表等；对于新闻博客类网页，能进行网页正文的自动抽取，对正文进行自动摘要和关键词分析。
- URL：统一资源定位符是对可以从互联网上得到的资源的位置和访问方法的一种简洁的表示，是互联网上标准资源的地址。互联网上的每个文件都有一个唯一的 URL，它包含的信息指出文件的位置以及浏览器应该怎么处理它。
- PID: PID=port ID，在 STP（生成树协议）中，若在端口收到的 BPDU 中 BID 和 path cost 相同时，则比较 PID 来选择阻塞端口。

2. 任务概述

2.1. 目的

分布式爬虫系统对同一个网站的同类数据，进行自动结构化。同时，能利用分布式的软件设计方法，实现爬虫的高效采集。并且保

证爬虫的下载快速和高效，解决爬虫面临的反爬虫问题。输入入口URL之后，自动分析网页的组织形态获取新的链接，进行下载。同时对URL进行去重，已经下载过的，没有进行数据更新的，不再进行下载，并将获取的信息返回给用户。

2.2. 用户特点

本系统的用户针对所有人，系统可根据用户设置的从机，进行分布爬虫，快速高效的提供用户所需要的信息。

2.3. 假定和约束

- 开发人员为三人；
- 开发期限三个月。

3. 需求规定

3.1. 系统功能结构图

本系统的功能结构如下：

图 3.1-1 系统功能结构图模型

3.2. 用例图

本系统为分布式爬虫系统，参与者包括：普通用户、爬虫系统。其用例图如下：

- **普通用户**：登录、发布爬虫任务、查看爬虫任务、管理爬虫、从机管理、暂停爬虫、启动爬虫、停止爬虫、查看已爬取数据、监控爬虫、数据统计、正文测试、查看进程状态。
- **爬虫系统**：爬取数据、数据结构化、异常恢复、监听进程。

普通用户用例图：

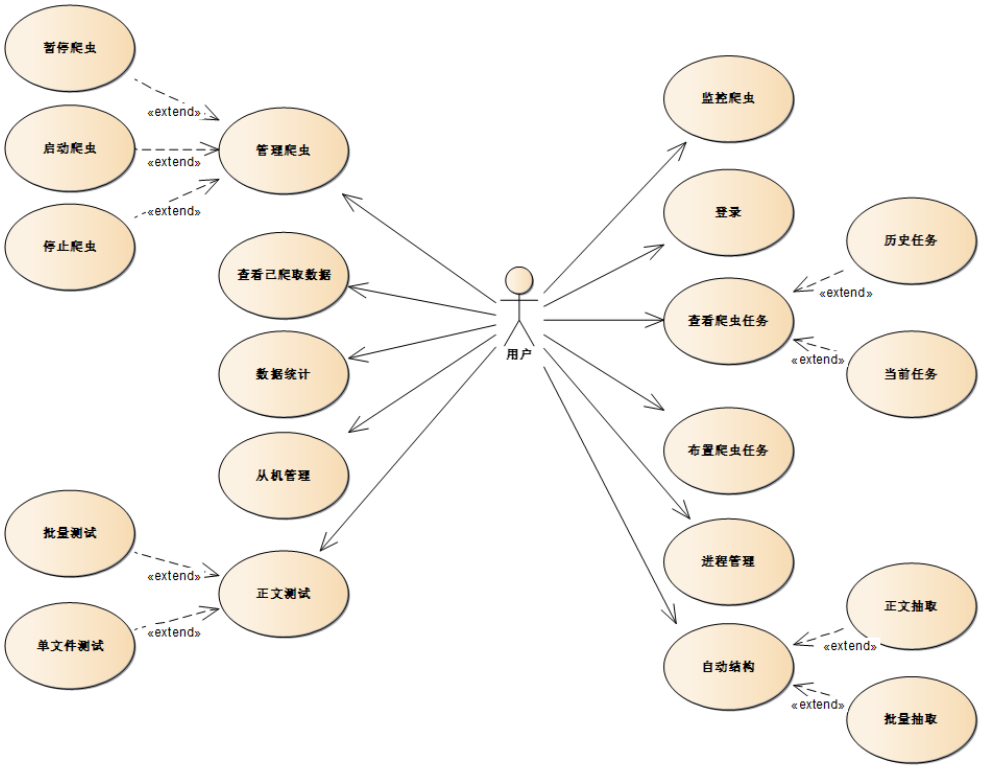

图 3.2-1　　普通用户用例图

爬虫系统用例图：

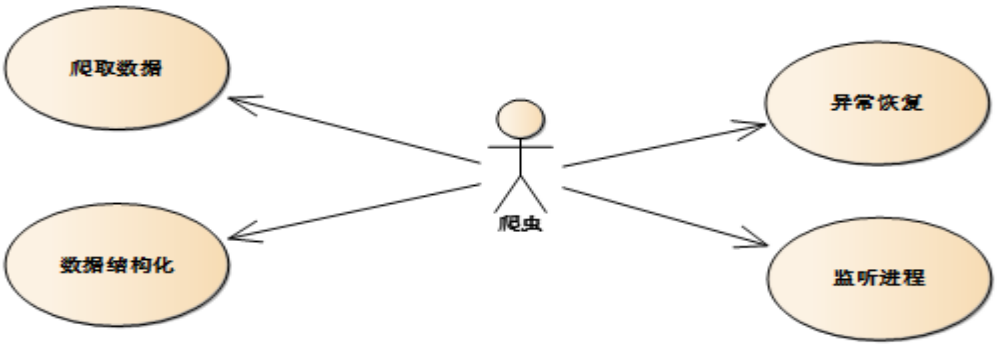

图 3.2-2　爬虫系统用例图

3.3.　用例概要描述

用例编号	用例名	用例概述	主要参与者
SRS_case1	管理爬虫	对已发布的爬虫任务进行相关管理操作	

SRS_case2	查看数据	查看已爬取的结构化数据	普通用户
SRS_case3	布置爬虫任务	根据用户的输入信息发布爬虫任务	
SRS_case4	查看爬虫任务	登录查看爬虫任务的详细情况	
SRS_case5	从机管理	查看或添加从机 IP	
SRS_case6	进程管理	查看或改变正在执行爬虫任务的主从机的进程状态	
SRS_case7	数据统计	对爬虫任务进行数据统计	
SRS_case8	监控爬虫	对定时爬虫任务进行监控	
SRS_case9	正文测试	对网页内容的正文进行抽取并测试	
SRS_case10	批量测试	测试大批量文件正文抽取的准确率	
SRS_case11	单例测试	对单个文件进行正文抽取并比较	
SRS_case12	历史任务	查看已完成的爬虫	

		任务的详情	
SRS_case13	当前任务	查看当前正在进行的爬虫任务的详情	
SRS_case14	暂停爬虫	暂停正在进行的爬虫任务	
SRS_case15	启动爬虫	启动已暂停的爬虫任务	
SRS_case16	停止爬虫	停止正在进行的爬虫任务	
SRS_case17	自动结构化	根据输入的 URL 将数据结构化	
SRS_case18	正文抽取	根据输入的正文 URL，将正文信息结构化	
SRS_case19	批量抽取	根据输入的多个 URL 将网页上的数据动结构化	
SRS_case20	爬取数据	根据发布的爬虫任务爬取数据	爬虫系统
SRS_case21	数据结构化	将爬取的数据信息结构化	

SRS_case22	异常恢复	爬虫程序遇到异常时断点恢复	
SRS_case23	监听进程	对正在执行爬虫任务的进程进行监听	

4. 详细功能需求

4.1. 管理爬虫（SRS_case1）

对正在进行的的爬虫任务进行以下管理操作。

4.1.1. 暂停爬虫（SRS_case14）

爬虫启动后，可以根据用户需要保存当前爬虫状态，暂停爬虫任务。

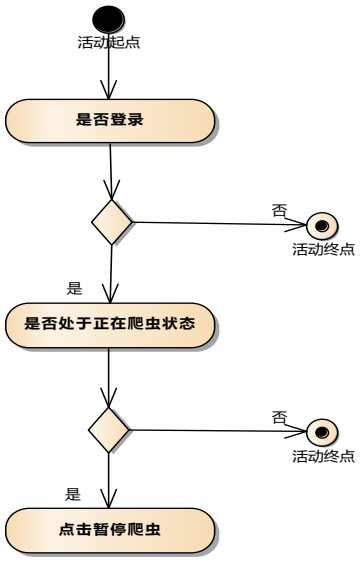

图 4.1-1 暂停爬虫活动图

4.1.2. 启动爬虫（SRS_case15）

重新启动已暂停的爬虫，爬虫从 URL 链接节点暂停处重新开始启动。

4.1.3. 停止爬虫（SRS_case16）

杀死爬虫，清空 URL 队列。

4.2. 查看数据（SRS_case2）

将所有爬虫任务爬取到的数据依次排列在界面上供用户查看。

4.3. 布置爬虫任务（SRS_case3）

发布爬虫任务要求用户设定任务名称、起始 URL、网站类型，也可以进行高级参数设置（始末时间、从机 IP），系统将根据这些参数设置爬虫属性进行爬取。在电商类网站爬虫任务时可添加关键字信息。

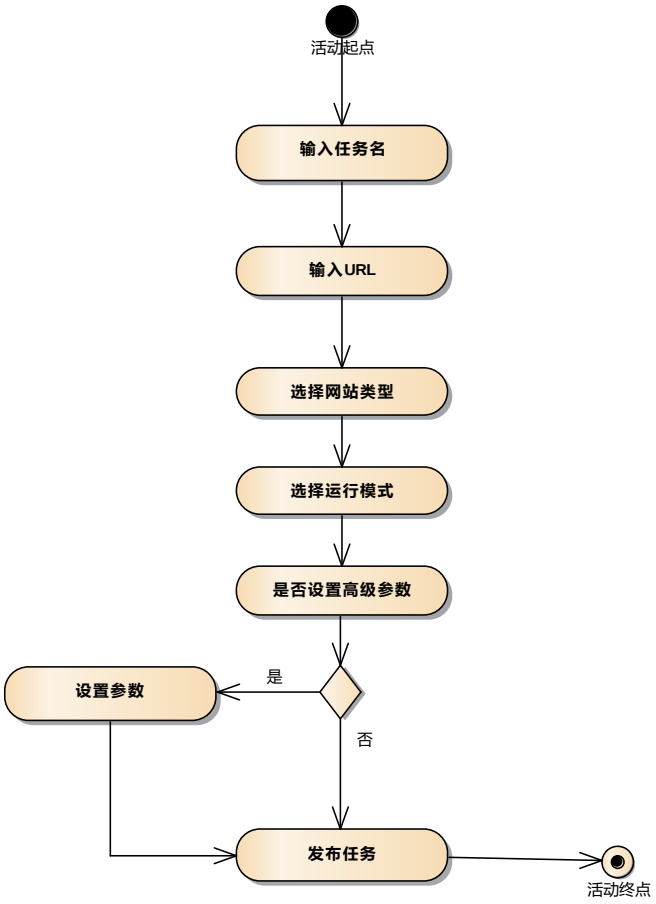

图 4.3-1　发布爬虫任务活动图

4.4.　查看爬虫任务（SRS_case4）

查看已发布的所有爬虫任务的详细信息。爬虫任务的状态包括：运行、暂停、停止、故障。在查看任务详细信息时，可以改变爬虫的状态。

4.4.1.　历史任务（SRS_case12）

查看已经完成的爬虫任务的详细信息，包括 ID、任务名等。

4.4.2. 当前任务（SRS_case13）

查看当前正在进行的爬虫任务的详细信息，包括其当前状态、ID、任务名等。

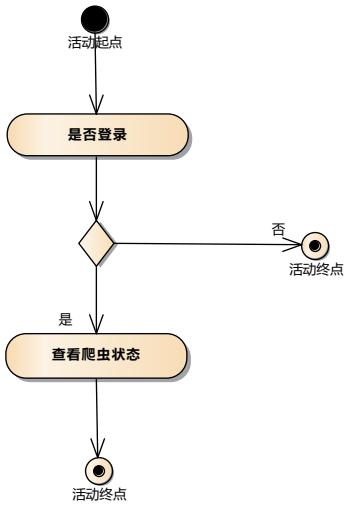

图 4.4-1 查看爬虫任务活动图

4.5. 爬取数据（SRS_case20）

从输入的起始 URL 进入，识别该页面中的满足需求的 URL 加入 redis 中的 URL 队列中，分析页面结构抽取出有价值的内容插入 mongodb 数据库。

4.6. 数据结构化 （SRS_case21）

- 电商类：

1. 从电商首页中获取导航栏中的商品分类。
2. 从商品分类进入商品列表页，获取每页的商品信息并自动翻页。
3. 从商品信息进入商品详情页，获取更详细的商品信息。

➢ 新闻博客类：

从新闻或博客网页中解析所有 URL,从中筛选出正文详情页的 URL 并获取标题、关键词、正文、时间等。

4.7. 从机管理（SRS_case5）

对从机 IP 进行增删查改。

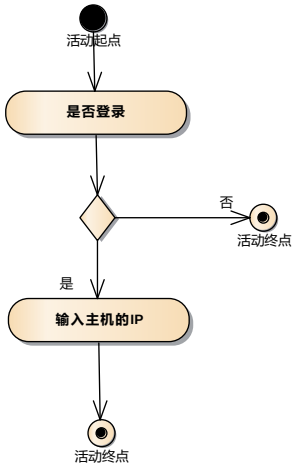

图 4.8-1 添加主机活动图

4.8. 进程管理（SRS_case6）

查看或改变各从机的执行爬虫任务的进程状态，分为执行和挂起状态。

4.9. 数据统计（SRS_case7）

对已经发布的爬虫任务的状态和类别进行相关数量统计，并生成柱形图和饼图，从而直观地展现当前的所有爬虫任务的统计情况。

4.10. 正文测试（SRS_case9）

对网页文件进行正文的抽取，可分为以下两类：

4.10.1. 批量测试（SRS_case10）

将一个存储网页文件的文件夹里的所有文件进行正文的抽取，并存放到一个新的文件夹中，然后和标准文件比较计算准确率。

4.10.2. 单文件测试（SRS_case11）

对单个网页文件进行正文抽取并比较。

4.11. 异常恢复（SRS_case22）

当某台正在执行爬虫任务的从机因为某种因素而导致异常时，该从机将清除之前的进程信息，重新启动该从机上的所有爬虫任务，爬虫任务将会从其断点处继续执行。

4.12. 监控爬虫（SRS_case8）

对爬虫任务进行监控，并对其性能进行分析，将其可视化。即利用 graphite 监控 scrapy。

4.13. 监听进程（SRS_case23）

监听定时爬虫任务的结束时间，当时间到达时终止该爬虫，清除该爬虫任务的 URL 队列和进程信息。

4.14. 自动结构化（SRS_case17）

根据用户输入的网页 URL 将该网页上的数据结构化。

4.14.1. 正文抽取

根据用户输入的正文网页 URL 将网页上的正文自动结构化，从而将已经结构化的正文信息显示出来。

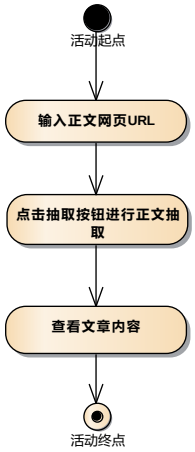

图 4.14-1 正文抽取活动图

4.14.2. 批量抽取

根据用户输入的多个网页 URL 将网页上的数据自动结构化，从而将已经结构化的网页信息显示出来，并将显示结构化的信息链接，根据链接可查看详细内容。

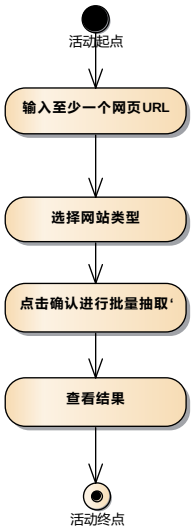

图 4.14-1 批量抽取活动图

5. 性能需求

5.1. 易用性需求

- 软件功能界面友好，方便用户使用。
- 可视化效果展现直接丰富。

5.2. 可靠性和可用性需求

- 软件可以每天使用 24 小时，每年使用 365 天。
- 系统中软件故障小。

5.3. 容错性需求

- 在网络断开时，对本地数据的操作可以正常进行。
- 如果存在对数据的范围约束，应保证录入的数据在约束范围内。

5.4. 容量需求

- 排除硬件和网络的因数，分布式爬虫系统的最大并发人数为1000 人。

5.5. 时间特性需求

- 最大系统响应时间：最大系统响应时间少于 5 秒，2 秒以内为最佳。

5.6. 可拓展性需求

- 可与其他爬虫技术拓展。

5.7. 其他专门要求

- 能够保证数据的独立性。数据和程序相互独立有利于加快软件开发速度，节省开发费用；
- 冗余数据少，数据共享程度高；
- 系统的用户接口简单，用户容易掌握，使用方便；
- 能够确保系统运行可靠，出现故障时能迅速排除；能够保护数据不受非受权者访问或破坏；能够防止错误数据的产生，一旦产生也能及时发现；
- 有重新组织数据的能力，能改变数据的存储结构或数据存储位置，以适应用户操作特性的变化，改善由于频繁插入、删除操作造成的数据组织零乱和时空性能变坏的状况；

- 具有可修改性和可扩充性；
- 能够充分描述数据间的内在联系。

6. 运行环境规定

6.1. 设备及分布

- 服务器类型：Ubuntu Server
- 网络类型：通用局域网或广域网
- 存储容量：内存容量： 不少于 2GB 外存容量： 30GB 以上

6.2. 支撑软件

- 操作系统：Ubuntu
- 数据库：mongodb、redis
- web 浏览器：Chrome、Firefox、IE、Opera 等主流浏览器

7. 附录

7.1. 用户输入参数列表

参数类型	输入参数	单位及要求
	任务名	必填
	起始 URL	必填（可填多个）

发布爬虫任务输入	网站类型	必填
	关键字	选填
	描述	选填
	运行的时间区间	选填
	主机 IP	选填（多选）
	进程数量	选填

参数类型	输入参数	单位及要求
自动结构化输入	正文网页 URL	必填（仅一个）
	起始 URL	必填（可填多个）
	网站类型	必填

参数类型	输入参数	单位及要求
从机管理输入	IP	必填（仅一个）

7.2. 用户信息

- 账号
- 密码

7.3. 记录爬虫结果信息项

- ID
- 标题

- URL
- 关键词
- 时间

7.4. 记录爬虫任务信息项

- ID
- 启动时间
- 任务名
- 网站类型
- 状态
- 操作

7.5. 记录进程状态信息项

- ID
- 主机
- PID
- 任务 ID
- 任务名
- 爬虫状态

7.6. 记录从机信息项

- ID

- IP
- 操作