

视采采集器

# 需求分析说明书

日期	版本	作者	修改内容
2006-11-12	0.1	www.caijiqi.net	新版

## 目 录

- 1 概述
  - 1.1 目的
  - 1.2 需求概述
    - 1.2.1 网站采集
    - 1.2.2 信息采集
    - 1.2.3 数据结构化
  - 1.3 用户特点
- 2 系统需求
  - 2.1 多样化的采集目标
  - 2.2 多样化的数据格式
  - 2.3 分布式海量数据
  - 2.4 数据横向和纵向采集
  - 2.5 用户操作简单、快捷
- 3 功能定义
  - 3.1 插件管理
    - 3.1.1 功能说明
    - 3.1.2 界面元素
    - 3.1.3 功能需求
  - 3.2 输入输出插件定义
    - 3.3.1 功能说明
    - 3.3.2 功能需求
  - 3.3 内容过滤插件定义
    - 3.3.1 功能说明
    - 3.3.2 功能需求
  - 3.4 系统参数配置
    - 3.4.1 功能说明
    - 3.4.2 界面元素
    - 3.4.3 功能需求
  - 3.5 数据采集
    - 3.5.1 功能说明
    - 3.5.2 界面元素
    - 3.5.3 功能需求
  - 3.6 html 内容规则定义
    - 3.6.1 功能说明
    - 3.6.2 界面元素
    - 3.6.3 功能需求
  - 3.7 计划任务管理
    - 3.7.1 功能说明
    - 3.7.2 界面元素
    - 3.7.3 功能需求
- 4 支持信息

# 1 概述

## 1.1 目的

本文描述用户的需求特征，定义系统的功能结构以及面向用户的操作界面。详细说明功能的特征及行为。

本文档适合系统设计者，系统开发者，系统测试者阅读和参考。

## 1.2 需求概述

网站管理员最大的心愿是提供最丰富的网站内容，吸引更多访问量；市场营销人员每当通过蛛丝马迹而获取到隐藏的客户资源而兴奋不已；企业后勤人员做梦都想远离这些枯燥无味的文字录入。采集系统好比一双慧眼，让您看得更远，获得更多。

### 1.2.1 网站采集

网站管理员希望将别人的整站数据下载到自己的网站里或者将别人网站的一些内容保存到自己的服务器上。从内容中抽取相关的字段，发布到自己的网站系统中。有时需要将网页相关的文件也保存到本地，如图片、附件等。

网站管理员会定时从同一网站上采集内容，希望已经采集的内容不要再发布到网站系统中。对于一些网站，需要登陆才能获取页面。网站管理员希望能够通过一个内容列表页面获取所有的相关内容，包括内容列表的其它分页。当第二次采集相同网站时，希望不要再重复第一次的设定。

### 1.2.2 信息采集

网站管理员从互联网中收集各类图片、笑话、新闻、技术等各类信息，然后分类、编辑，发布到自己的网站系统中。网站管理员一般通过搜索引擎搜索各类关键字获取目标网址，然后再提取网页中的内容。关键字的组织决定获取内容的准确性和数量。由于内容来自不同的网站，所以提取内容的方法也各不相同。对于某一类的信息，发布到网站系统的数据结构是相同的。

网站管理员对站内进行搜索，将相关的内容在首页上进行编排和索引。

企业从互联网上搜索 email 和电话号码，并且能够查看该信息的相关信息，以便了解该对象的基本情况。企业希望能够搜索某一类别的客户信息，如这个客户属于女性，年龄为 20 到 30 岁等。并且能够将采集到的对象信息保存到企业内部的客户管理系统中。

企业需要了解某一产品的信息，希望得到该类产品的报价、厂商等，以及这些信息的对比情况。并且能够得到报价、厂商的进一步信息。这些信息希望能够保存到企业的内部的 ERP 系统或其它系统中。

### 1.2.3 数据结构化

企业办公产生的电子文档，客户提交的客户资料等这些数据，一般需要大量的人力手工输入到企业的 ERP 系统或信息系统中，企业希望能够通过软件从这些文档中抽取相关的数据自动导入到系统中。这些数据一般都有固定的模板格式，并且同一类文档的模板格式是相同的。如客户的家庭信息，客户 1 和客户 2 的模板格式是一样的，只是内容不一样。

## 1.3 用户特点

系统的最终用户群包含网站的管理员们，对一些给目标地址做了隐藏的网站，可能会使他们操作失败，特别有些网站对网页内容作了扰乱处理，使得他们更难准确地定义规则。对于这些问题，系统提供一些范例和匹配通配符，告诉他们的应对方法。并且官方网站提供交流论坛，共享和学习相互之间的经验。

对于信息采集的用户，系统提供丰富的模板模式供用户使用，如 email 匹配模式、电话号码匹配模式等，只需要用户选择一个模板，就可以获得他们想要的信息。当然官方网站提供丰富模板资源供以下载。

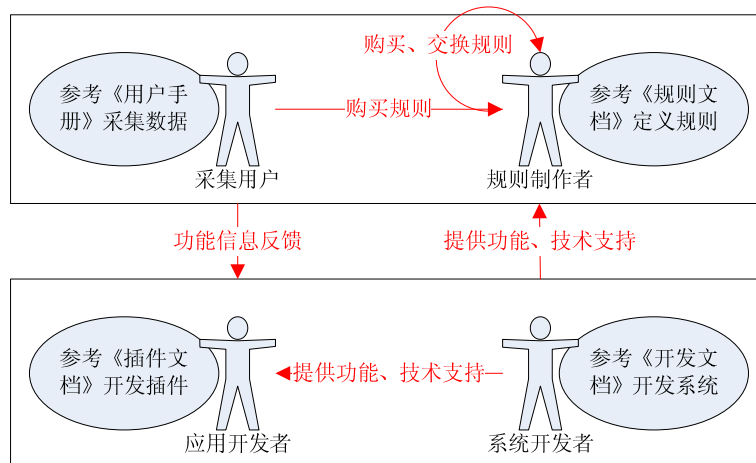
对于数据结构化的应用，会有第三方的技术人员提供支持。

系统预留了输入输出的编程接口，一些用户扩展这些接口，使系统应用到更多的场合下。针对这些用户，系统提供详细的接口说明，扩展示例代码。我们提供开发包，并描述每个类和每个方法的使用方法和功能。

还有一类用户属于商人的范畴，他们仅从事规则的制作，在网上交换或出售自己的规则。他们更关心网站的搜索和内容的质量，他们分两种类型，一种是猎人，他们能够发现各种各样的信息，能够满足各类网管的内容需求，他们从数量上获取大把金钱。当然，对于个别稀有的内容，价格就像黄金一样了。另一种属于黑客，他们精通 web 技术，机智并执着，在他们手里，都是一些很难发掘到的精品，当然价格都是高昂的。

由于采集系统属于开源软件，会有很多人去分析和使用采集系统中的组件，扩展和完善采集系统。他们使用的级别不仅是系统的界面上，而是深入到代码的内部上，他们需要参考采集系统的各类技术文档，所以系统除了用户手册，其它的开发文档也是必不可少的。

协作关系：



## 2 系统需求

### 2.1 多样化的采集目标

信息分布在各种信息存储系统中，各种存储系统有着各自的交互机制，需要采集系统提供多种并可扩展的连接模块。

### 2.2 多样化的数据格式

信息以多种形式存在，如网页、word 文档、pdf 等。这些不同的格式数据需要采用不同的采集机制。

### 2.3 分布式海量数据

由于网络通信的延时和网络带宽的限制，并发多线程通信能够有效地减低延时和抢夺资源。

### 2.4 数据横向和纵向采集

需要系统自动采集数据的下一页；自动采集数据的关联附件；自动根据当前采集结果采集下一数据。

### 2.5 用户操作简单、快捷

多样且复杂的数据格式增加用户的作业难度，用户希望所见及所得，及时提供相应的操作提示信息。

## 3 功能定义

系统提供可视化规则定义，支持多层次采集。功能包括：

1. 输入输出插件管理
2. 系统参数配置
3. 数据采集
4. 规则管理
5. 计划任务管理

### 3.1 插件管理

#### 3.1.1 功能说明

注册插件、查看插件、删除或屏蔽插件。

#### 3.1.2 界面元素

插件路径	<input type="text" value="文件选择控件"/>
<input type="button" value="安装"/>	

### 3.1.3 功能需求

用户上传插件包，安装到系统中。  
用户可以删除或屏蔽已安装的插件包。

插件包为 jar 包，包的目录结构如下：

```
\<pack-path>\<class>  
\resource\<file>  
\openwebant-plunin.xml
```

文件 openwebant -plunin.xml 为插件配置。参数如下：

```
<?xml version="1.0" encoding="utf-8" ?>  
<package>  
  <info>  
    包信息。信息包括名称、描述、作者、网站  
  </info>  
  <plunin type="openwebant-input" version="1.0" class="com.openwebant.httpInputPlunin">  
    <info>  
      插件信息，信息包括类型、描述  
    </info>  
    <mapping>  
      插件插入点  
    </mapping>  
  </plunin>  
  <plunin type="..." version="..." class="...">  
    ...  
  </plunin>  
</package>
```

当系统启动时，检查系统包路径下的所有的包，如果查找到 openwebant-plunin.xml，则注册该插件。

mapping 用来映射哪些目标由它来处理，支持正直表达式匹配。http://表示它可以处理以 http://打头的目标地址。text/html 表示它可以处理 html 网页。

## 3.2 输入输出插件定义

### 3.3.1 功能说明

定义各种通信协议下的输入输出插件

### 3.3.2 功能需求

插件参数在 `mapping` 中定义。

```
<plunin type="openwebant-input" version="1.0" class="com.openwebant.httpInputPlunin">
  <info>
    名称:http 输入插件
    描述:该插件通过 http 协议读取内容
    作者:openWebant
    网站:http://www.java51.com
  </info>
  <mapping>
    http://
  </mapping>
</plunin>
```

## 3.3 内容过滤插件定义

### 3.3.1 功能说明

定义各种内容过滤插件

### 3.3.2 功能需求

插件参数在 mapping 中定义。

```
<plunin type="openwebant-content" version="1.0" class="com.openwebant.htmlFilterPlunin">
  <info>
    名称:网页过滤插件
    描述:改插件以 html 结构进行结构化匹配
    作者:openWebant
    网站:http://www.java51.com
  </info>
  <mapping>
    text/html
  </mapping>
</plunin>
```

该插件包括采集规则，过滤引擎，用户界面。

## 3.4 系统参数配置

### 3.4.1 功能说明

设置系统参数。

### 3.4.2 界面元素

任务最大数	<input type="text"/>
线程最大数	<input type="text"/>
线程采集间隔	<input type="text"/>
报告刷新闻隔	<input type="text"/>
采集日记路径	<input type="text"/>
采集编码	<input type="text"/>
发布编码	<input type="text"/>
...	<input type="text"/>
<input type="button" value="保存"/>	

### 3.4.3 功能需求

系统参数改变后，系统地下一次行为要参照最新的参数，当前正在运行的任务可不参照参数的改变。

具体参数将根据系统详细设计确定。

## 3.5 数据采集

### 3.5.1 功能说明

通过采集规则将指定目标内容发布到数据库中。

### 3.5.2 界面元素

采集界面：

采集规则	美女网	选择	开始	新建	
停止采集					
管道窗口					
管道	输出数据				
url=titl[0]	百度裁员 官方回应 律师称违规 裁员录音曝光				
url=content[1]	大峡：一个“Spring 轮子”引发的血案 1 2 3 4 编辑空间：印度软件外包发展简记 外包频道 学习委托：函数指针的改头换面 实现机制				
线程窗口					
线程	采集目标	采集结果数	开始时间	结束时间	耗时
Titl[1]	http://www.csdn.net	100	12:00:00	12:00:20	20
Content[0]	http://www.csdn.net	2	12:00:01	12:00:10	9
Content[2]	http://www.blog.com	200	12:00:01	进行中	2

规则选择窗口：

规则名	选择
美女网	选择
程序大本营	选择
新浪网	选择
网易	选择

### 3.5.3 功能需求

用户可以选择已经存在的任务文件进行采集。用户也可以新建采集任务。采集过程中，需要显示采集的状态，如线程列表，每个线程当前采集的目标，采集的结果等。用户可以终止采集任务。

用户选择采集规则，然后点击开始后，   按钮变灰失效，停止采集按钮有效。管道窗口报告当前采集到的数据。线程窗口显示当前系统正在运行的线程和已经结束的线程。

用户可以点击  按钮，系统弹出规则列表框，用户指定一个规则。

用户点击  按钮，系统进入规则定义页面。

## 3.6 html 内容规则定义

### 3.6.1 功能说明

定义采集规则,定义采集单元和数据表字段的联合。

系统分析 html 语法，构建节点树，根据节点属性进行匹配。

系统不同于一般的采集器根据关键字来匹配内容，而是根据节点属性和层次来匹配节点。

系统比采用内容匹配规则的采集器能更准确的匹配内容并可以有效地采集无规则的网页。

系统采用可视化的方法定义规则，并能在页面上实时显示匹配的单元结果集。

### 3.6.2 界面元素

第一层规则定义页面：

规则名称	<input type="text"/>			
目标网址	<input type="text"/>			<input type="button" value="获取内容"/>
<input type="checkbox"/> 块 1 <input type="checkbox"/> 块 2 <input type="checkbox"/> 块 3 <input type="checkbox"/> 块 4		<input type="button" value="新建块"/> <input type="button" value="删除块"/> <input type="button" value="预览"/>		
<可视化页面> 用户点击区域，系统在单元属性框里显示单元属性，用户指定哪些属性被采集。				
<input type="checkbox"/> <body> <input type="checkbox"/> <table> <input type="checkbox"/> <tr> <input type="checkbox"/> <td> <input type="checkbox"/> <a>				
单元名	<input type="text"/>	过滤	<input type="text"/>	
标签中的文本 <input type="text"/>				
属性	值	单元名	过滤	表达式
value	jsp 教程	<input type="text" value="标题"/>	<input type="text"/>	<input type="text"/>
href	http://www.java51.com	<input type="text" value="网址"/>	<input type="text"/>	<input type="text" value=".+\\.jsp?id=\\d"/>
				<input type="button" value="确定"/> <input type="button" value="删除"/>
<网页源码> 当前网页的源码。并选定当前单元所对应的代码。用户也可以在源码区里来定义不可视的单元格。				
		<input type="button" value="下一层"/> <input type="button" value="发布"/>		

第二层以下的规则定义页面：

<上一层页面，页面标识所有的带有网址的单元格，不可视单元格使用单元格名称表示>  
 用户选择单元格，系统请求网址，在下面显示该页面。

块 1
  块 2
  块 3
  块 4

<可视化页面>  
 用户点击区域，系统在单元属性框里显示单元属性，用户指定哪些属性被采集。

<body>
  <table>
  <tr>
  <td>
  <a>

单元名	<input type="text"/>	过滤	<input type="text"/>
标签中的文本			

属性	值	单元名	过滤	表达式
value	jsp 教程	标题	<input type="text"/>	<input type="text"/>
href	http://www.java51.com	网址	<input type="text"/>	.\.jsp?id=\d

<网页源码>  
 当前网页的源码。并选定当前单元所对应的代码。用户也可以在源码区里来定义不可视的单元格。

单元格发布定义页面：

数据库地址: <input type="text"/>			<input type="button" value="连接"/>
数据库表: article_content article_user article_mark article_type article_template  用户输入数据库 url, 连接数据库, 系统显示数据库表。用户选择一个表, 列出字段。	表字段: article_id article_parent_id article_title article_body article_autor article_type_id article_post_date  用户选择一个字段, 选择一个单元格, 点击 <input type="button" value="联合"/> 按钮。	单元格列表: └列表标题 └列表标题链接   └文章标题   └文章内容   └文章作者   └文章评论   └评论标题   └评论内容   └评论日期 └点击数	自变量: <input type="text"/> seq,filed,filed[index],v1:v2 seq:序列号,filed:字段,index:索引
	<input type="button" value="联合"/> <input type="button" value="移除"/>		
	article_content.article_id    <---> seq article_content.article_title    <---> 列表标题链接.文章标题 article_content.article_body    <---> 列表标题链接.文章内容		
	<input type="button" value="保存"/> <input type="button" value="采集"/>		

### 3.6.3 功能需求

用户输入目标网址, 获取页面和源文件, 在页面上点击一区域, 系统自动显示该单元的属性, 源码区里显示网页代码, 在这里可以定义不可视的单元。修正表示系统在 **html** 标签树中最多向左向右或向上向下多少个节点来查找这个单元。系统根据标签类型向外在最小的层次里匹配, 并将结果立即在页面上标示。范围匹配设定可以让用户指定在什么范围内匹配节点。用户定义好某一单元格, 页面上立即显示所有与之匹配的单元格。点击  按钮, 页面显示上一层的单元格列表。用户选择一个单元格, 系统自动获取页面和源文件, 定义单元格。重复以上过程, 直到采集深度达到要求为止。点击  按钮, 进入发布设置页面。

在页面上以深度结构显示单元格的树型列表。用户输入数据库的 url, 显示数据库表, 用户选择一张表, 显示表的字段。用户将单元格和字段关联起来。点击  保存采集规则, 以后可以在采集页面上选择该规则进行采集。点击  按钮, 系统保存采集规则, 并立即采集。

单元格发布向导定义数据库和采集的单元格的关联关系, 系统提供预定义的自变量: seq: 序列号, 值自动增长, 可以用来做字段 id 的值。

filed: 表字段或单元格

index: 表字段或单元格的索引, 适用于多条数据。

v1:v2: 当 v1 可用的时候, 取 v1 的值, 否则取 v2 的值。

规则配置文件采用 xml 格式定义。

规则配置文件用来保存规则定义, 系统可以通过规则配置文件, 手动或自动采集。根据 html 采集机制, 规则配置文件保存目标节点的前三个节点名和后三个节点名。

```
<?xml version="1.0" encoding="utf-8" ?>
<rule>
  <page url='http://www.sina.com.cn?page=${count}' count='10'>
    <block>
      <tag xmlId='2' >
        <cell property='text' label='标题'/>
        <cell property='href' label='网址'/>
      </tag>
    </block>
  </page>
  <page url='${网址}'>
    <block>
      <tag xmlId='4' >
        <cell property='text' label='正文' filter='br*\n'/>
      </tag>
    </block>
  </page>
  <database url='xxxxx' driver='xxx' user='user' password='pas'>
    <table name='article_content'>
      <relation filed='article_title' cell='标题'/>
      <relation filed='article_body' cell='正文'/>
    </table >
  </database>
</rule>
```

## 3.7 计划任务管理

### 3.7.1 功能说明

让任务在指定的时间里自动执行。

### 3.7.2 界面元素

计划任务设置页面：

采集规则	<input type="text" value="美女网"/>	<input type="button" value="选择"/>	<input type="button" value="新建"/>
启动时间	<input type="text" value="12-16 12:23"/>		
采集结束后 自动地关机	<input checked="" type="checkbox"/>		
<input type="button" value="保存"/>			

### 3.7.3 功能需求

用户点击  按钮，弹出规则列表，选择一个规则，关闭列表窗口。指定启动时间，时间格式为 月-日 时:分。如果用户选择自动关机项，系统采集结束后自动关闭计算机。

## 4 支持信息

资源	说明
<a href="http://www.caijiqi.net">www.caijiqi.net</a>	项目官方网站，发布项目文档，提供系统下载。
QQ:107175884	
Mail:hotheartboy@gmail.com	