实 验 报 告

课程名称 信息检索与搜索引擎

实验项目 使用网络爬虫抓取网页

实验仪器 PC机

系　别 计算机学院

专　业 计算机科学与技术

班级/学号 计科1306 /2013011266

学生姓名 陈伟颖

实验日期 2016-4-16

成　绩

指导教师 陈若愚

实验1.1 爬虫开发环境的配置

1. 实验目的

- 了解Java语言开发环境的基本配置方法；
- 掌握在Eclipse集成开发环境中导入已有工程的方法；
- 掌握在Eclipse工程中导入第三方Jar包的方法；

2. 实验环境

- 操作系统：Windows / Mac OS X / Linux
- JDK版本：JDK 1.7及以上
- Eclipse：Eclipse

3. 实验步骤

1. 确认本机所装JDK版本。
2. 确认Eclipse的安装情况。
3. 在Eclipse中导入所提供的工程压缩包Crawler.zip。
4. 解决导入的工程中可能出现的乱码、Build Path异常等问题。
5. 运行cn.edu.bistu.cs.crawler.NgpodPageProcessor类【10分】

4. 实验结果

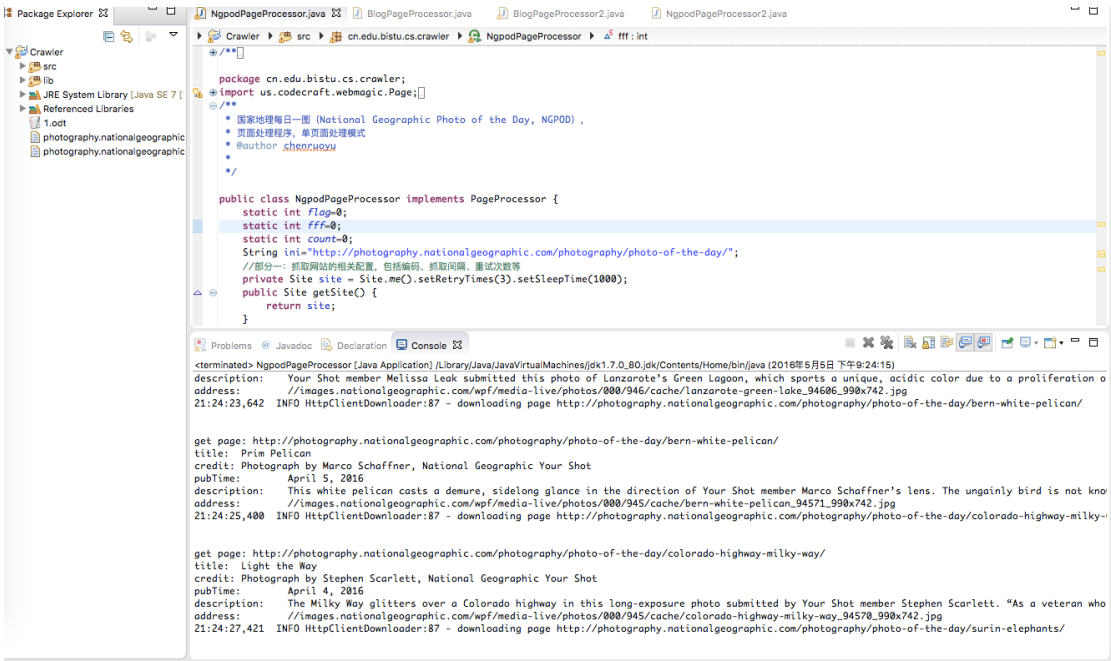

图1 一个简单爬虫的运行（放大可看清）

实验1.2 简单页面的抓取

1. 实验目的

- 掌握使用WebMagic爬虫工具编写爬虫程序的基本思路；
- 掌握抓取简单页面的方法

2. 实验要求

1. 以National Geographic Photo of the Day http://photography.nationalgeographic.com/photography/photo-of-the-day/，为目标设计爬虫

2. 示例工程中，爬虫cn.edu.bistu.cs.crawler.NgpodPageProcessor，已经能够抓取页面中图片标题、作者信息、发布日期三项数据。

3. 修改上述爬虫，要求除上述三项数据外，还至少能够抓取出页面中文字描述，图片地址两项数据【20分】

4. 上述爬虫有一个缺陷，如图2所示，“April 7, 2015”这一天的数据被重复抓取了两次，试着解释出现这一缺陷的原因，并尝试解决。【10分】

```
22:09:50,754  INFO Spider:307 - Spider photography.nationalgeographic.com started!
22:09:50,761  INFO HttpClientDownloader:87 - downloading page http://photography.nationalgeographic.com/photography/photo-of-the-day/
get page: http://photography.nationalgeographic.com/photography/photo-of-the-day/
title:  Grand Torino
credit: Photograph by Prandoni Livio, National Geographic Your Shot
pubTime:        April 7, 2015
22:09:53,947  INFO HttpClientDownloader:87 - downloading page http://photography.nationalgeographic.com/photography/photo-of-the-day/ice-iceland-jokulsarlon/
get page: http://photography.nationalgeographic.com/photography/photo-of-the-day/ice-iceland-jokulsarlon/
title:  Purple Haze
credit: Photograph by Szymon Bielikowski, National Geographic Your Shot
pubTime:        April 6, 2015
22:09:55,677  INFO HttpClientDownloader:87 - downloading page http://photography.nationalgeographic.com/photography/photo-of-the-day/bald-eagle-water-zoo/
22:09:55,677  INFO HttpClientDownloader:87 - downloading page http://photography.nationalgeographic.com/photography/photo-of-the-day/trolley-turin-italy/
get page: http://photography.nationalgeographic.com/photography/photo-of-the-day/trolley-turin-italy/
title:  Grand Torino
credit: Photograph by Prandoni Livio, National Geographic Your Shot
pubTime:        April 7, 2015
get page: http://photography.nationalgeographic.com/photography/photo-of-the-day/bald-eagle-water-zoo/
title:  Shake It Off
credit: Photograph by Michael Pachis, National Geographic Your Shot
pubTime:        April 5, 2015
```

图2 重复抓取缺陷

5. 上述爬虫使用了WebMagic默认提供的Scheduler来管理抓取URL队列，每次重新启动爬虫，都会将所有URL重新抓取一次。如何修改上述爬虫，使得关闭并重新启动爬虫后，能够从之前抓取到的URL继续抓取？（注：WebMagic中包含有多种Scheduler的实现，包括基于文件的，基于内存队列的和基于Redis数据库的）【10分】

4. 实验结果

1. 修改上述爬虫，要求除上述三项数据外，还至少能够抓取出页面中文字描述，图片地址两项数据【20分】

答：添加了如图3所示的Java语句，抓取出文字描述与图片地址如图4所示。

```
//部分二：定义如何抽取页面信息，并保存下来
//图片标题
String title = page.getHtml().xpath("//div[@id='caption']/h2/text()").toString();
page.putField("title", title);
//作者信息
String credit = page.getHtml().xpath("//div[@id='caption']/p[@class='credit']/allText()").toString();
page.putField("credit", credit);
//发布日期
String pubTime = page.getHtml().xpath("//div[@id='caption']/p[@class='publication_time']/text()").toString();
page.putField("pubTime", pubTime);
//中文字描述
String description=page.getHtml().xpath("//div[@id='caption']//p[3]/text()").toString();
page.putField("description", description);
//地址
String address=page.getHtml().xpath("//div[@id='content_top']/div[2]/a/img/@src").toString();
page.putField("address", address);
```

图3 重复抓取缺陷

```
get page: http://photography.nationalgeographic.com/photography/photo-of-the-day/swallowtail-minnesota-spring/
title:  Springing Back to Life
credit: Photograph by Jim Brandenburg
pubTime:        April 7, 2016
description:    Over 93 days in 2014,  photographer Jim Brandenburg shot springtime images in his home state of Minnesota. This image of a swallowtail butte
address:        //images.nationalgeographic.com/wpf/media-live/photos/000/946/cache/swallowtail-minnesota-spring_94607_990x742.jpg
21:24:22,100  INFO HttpClientDownloader:87 - downloading page http://photography.nationalgeographic.com/photography/photo-of-the-day/lanzarote-green-lake/


get page: http://photography.nationalgeographic.com/photography/photo-of-the-day/lanzarote-green-lake/
title:  Curious Colors
credit: Photograph by Melissa Leak, National Geographic Your Shot
pubTime:        April 6, 2016
description:    Your Shot member Melissa Leak submitted this photo of Lanzarote's Green Lagoon, which sports a unique, acidic color due to a proliferation o
address:        //images.nationalgeographic.com/wpf/media-live/photos/000/946/cache/lanzarote-green-lake_94606_990x742.jpg
21:24:23,642  INFO HttpClientDownloader:87 - downloading page http://photography.nationalgeographic.com/photography/photo-of-the-day/bern-white-pelican/


get page: http://photography.nationalgeographic.com/photography/photo-of-the-day/bern-white-pelican/
title:  Prim Pelican
credit: Photograph by Marco Schaffner, National Geographic Your Shot
```

图4 抓取出的全部信息

2. 上述爬虫有一个缺陷，“April 7, 2015”这一天的数据被重复抓取了两次，试着解释出现这一缺陷的原因，并尝试解决。【10分】

答：这一天数据之所以重复，是因为有两个URL指向“今天”，一个URL为……/pthoto-of-the-day;　　另一个URL为……/photo-of-the-day/trolley-turin-italy。可以将首页设置为只分析、不抓取，来避免重复抓取两次。如图5所示：

```
public void process(Page page)
{
    String a=page.getUrl().toString();//获取当前url地址
    fff=0;

    //去除起始页
    if(a.equals("http://photography.nationalgeographic.com/photography/photo-of-the-day/"))
    {
        for(String str: page.getHtml().xpath("//div[@class='nav']/p/a/@href").all())
        {
            page.addTargetRequest(new Request(str));
        }
        return;
    }
    //不是起始页
    else {
        //部分二：定义如何抽取页面信息，并保存下来
        //图片标题
        String title = page.getHtml().xpath("//div[@id='caption']/h2/text()").toString();
        page.putField("title", title);
        //作者信息
        String credit = page.getHtml().xpath("//div[@id='caption']/p[@class='credit']/allText()").toString();
        page.putField("credit", credit);
```

图5 解决方案

3. 上述爬虫使用了WebMagic默认提供的Scheduler来管理抓取URL队列，每次重新启动爬虫，都会将所有URL重新抓取一次。如何修改上述爬虫，使得关闭并重新启动爬虫后，能够从之前抓取到的URL继续抓取？【10分】

答：可以通过FileCacheQueueScheduler来实现Scheduler的定制。该类会生成两个文件，一个代表指针，另一个代表URL列表。实现方式如图6所示：

```
Spider.create(new NgpodPageProcessor())
//设置起始URL
.addUrl("http://photography.nationalgeographic.com/photography/photo-of-the-day/")
//设置爬虫线程数
.thread(2)
//启动爬虫
.setScheduler(new FileCacheQueueScheduler("/Users/chanvain/Documents/Crawler/"))
.run();
```

图6 Scheduler解决方案

但由于系统原本的class有两个缺陷：在终止程序时内存丢失；在超时终止时所记录的指针包括当前所遇到的不去重的文件。因此，采用了每次运行时读写着两个文件来解决，如图7所示：

```
public static void main(String[] args) {
    int cur=0;
    try {
        FileReader fr = new FileReader("/Users/chanvain/Documents/Crawler/photography.nationalgeographic.com.urls.txt");
        BufferedReader br = new BufferedReader(fr);
        String str = null;
        while ((str = br.readLine()) != null) {
            cur++;
        }
        br.close();
        fr.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
    cur--;
    FileOutputStream out = null;
    String a=Integer.toString(cur);
    try {
        java.io.File file=new java.io.File("/Users/chanvain/Documents/Crawler/photography.nationalgeographic.com.cursor.txt");
        if(file.exists())
        {
            java.util.Scanner input=new java.util.Scanner(file);
            int temp=input.nextInt();
            while(input.hasNext())
            {temp=input.nextInt();}
            input.close();
            PrintWriter output=new PrintWriter(file);
            output.println(cur);
            output.close();
        }
         }catch (IOException e){
             e.printStackTrace();
    }
```

图7 Scheduler解决方案

5. 实验总结

这个子实验在三个实验中花了最长时间，重复抓取缺陷想得过于复杂；而scheduler存储在运行后发现不能得出从上次停止的地方继续时又花了很长时间解决。但总的来说还是收获到了发现问题，从而解决问题的锻炼。

实验1.3 列表+详情的组合页面抓取

1. 实验目的

- 掌握使用WebMagic爬虫工具编写爬虫程序的基本思路；
- 掌握抓取列表+详情的组合页面的方法

2. 实验要求

1. 以“李开复的博客”：http://blog.sina.com.cn/kaifulee 为目标设计爬虫

2. 博文列表/目录页的地址格式为：
http://blog.sina.com.cn/s/articlelist_1197161814_0_1.html
其中，“_0_1”中的“1”是可变的，表示博文目录的分页，如图8所示：

图8 博文目录

博文详情页的地址格式为：
http://blog.sina.com.cn/s/blog_475b3d560102vhae.html
其中“475b3d560102vhae”部分是可变的。可以根据URL本身的特征对上述两种URL进行区分，具体如何实现？【10分】

3. cn.edu.bistu.cs.crawler.BlogPageProcessor是一个简单的，可以抓取博客标题以及发表日期的爬虫，修改这一爬虫，增加抓取“博文正文全文”以及“博文标签”的功能，如图9所示。【20分】

Google CFO 的辞职信 (2015-03-12 15:31:09) 转 载

标签： 杂谈

Google CFO 的辞职信

After nearly 7 years as CFO, I will be retiring from Google to spend more time with my family. Yeah, I know you've heard that line before. We give a lot to our jobs. I certainly did. And while I am not looking for sympathy, I want to share my thought process because so many people struggle to strike the right balance between work and personal life.

This story starts last fall. A very early morning last September, after a whole night of climbing, looking at the sunrise on top of Africa - Mt Kilimanjaro. Tamar (my wife) and I were not only enjoying the summit, but on such a clear day, we could see in the distance, the vast plain of the Serengeti at our feet, and with it the calling of all the potential adventures Africa has to offer. (see exhibit #1 - Tamar and I on Kili).

And Tamar out of the blue said "Hey, why don't we just keep on going". Let's explore Africa, and then turn east to make our way to India, it's just next door, and we're here already. Then, we keep going; the Himalayas, Everest, go to Bali, the Great Barrier Reef... Antarctica, let's go see Antarctica!?" Little did she know, she was tempting fate.

I remember telling Tamar a typical prudent CFO type response- I would love to keep going, but we have to go back. It's not time yet, There is still so much to do at Google, with my career, so many people counting on me/us - Boards, Non Profits, etc

图9 博文正文以及标签

4. 修改上述爬虫，使得它可以抓取其他博主的博客文章，如“冯志伟文化博客”：http://blog.sina.com.cn/zwfengde2011【10】

5. 对于无法用页面URL地址来做区分的列表+详情组合页，如NGPOD网站上动物类POD页面地址为：
http://photography.nationalgeographic.com/photography/photo-of-the-day/animals/

页面内容如图10所示：

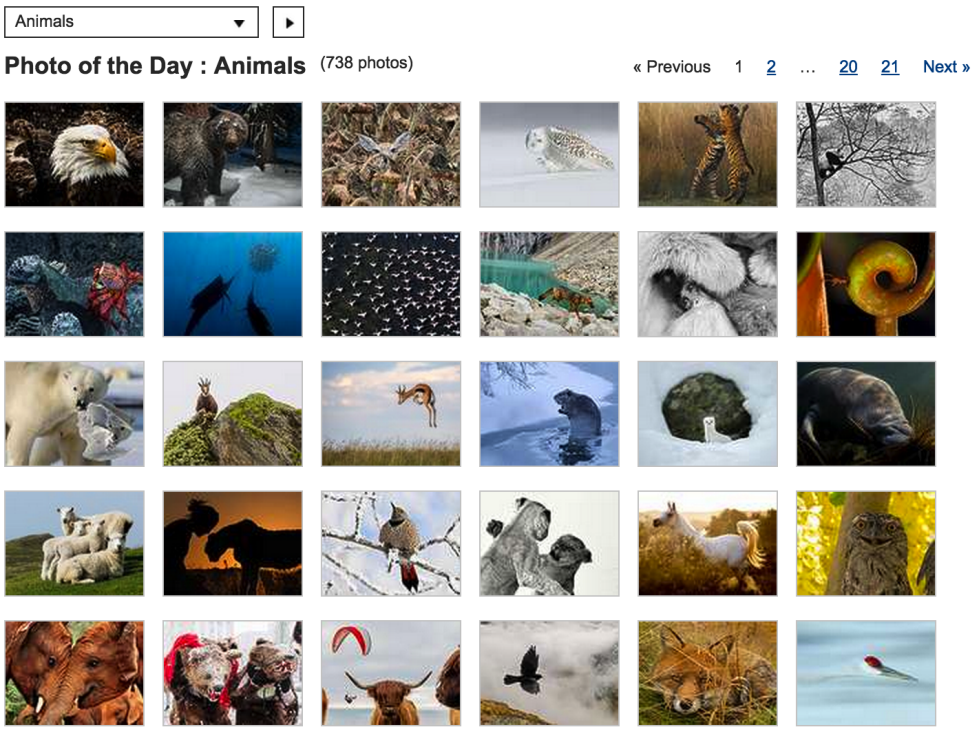

图10 NGPOD动物类列表页

而动物类中某一个详情页的URL地址为：

http://photography.nationalgeographic.com/photography/photo-of-the-day/bald-eagle-water-zoo/ 与上述列表页URL没有本质区别。**如何修改上述爬虫，使得程序能够区分上述两种页面，并抓取出详情页的内容？【选做，10分】**

4. 实验结果

1. 可以根据URL本身的特征对上述两种URL进行区分，具体如何实现？【10分】

答：一类页面是的结构是“http://blog.sina.com.cn/s/articlelist + ……”，而另一类则是无规律的，可以通过匹配区分这两种URL，如图所示：

```
//列表页
if(page.getUrl().toString().startsWith("http://blog.sina.com.cn/s/articlelist")){

}else if(page.getUrl().toString().startsWith("http://blog.sina.com.cn/s/blog")){

}
```

图11 匹配方法

2. cn.edu.bistu.cs.crawler.BlogPageProcessor是一个简单的，可以抓取博客标题以及发表日期的爬虫，修改这一爬虫，增加抓取“博文正文全文”以及“博文标签”的功能，如图5所示。【20分】

答：可以添加如下语句：

```
public void process(Page page) {
    //列表页
    if (page.getUrl().regex(URL_LIST).match()) {
        page.addTargetRequests(page.getHtml().xpath("//div[@class=\"articleList\"]").links().regex(URL_POST).all());
        page.addTargetRequests(page.getHtml().links().regex(URL_LIST).all());
        //文章页
    } else {
        page.putField("title", page.getHtml().xpath("//div[@class='articalTitle']/h2/tidyText()").toString().trim());
        page.putField("date",
        page.getHtml().xpath("//div[@id='articlebody']//span[@class='time SG_txtc']").regex("\\((.*)\\)"));
        //博文正文全文
        String passage=page.getHtml().xpath("//div[@id='sina_keyword_ad_area2']/allText()").toString();
        page.putField("passage", passage);
        //博文标签 //*[@id='sina_keyword_ad_area']/table/tbody/tr/td[1]/h3/a
        String tag=page.getHtml().xpath("//div[@id='sina_keyword_ad_area']/table/tbody/tr/td[1]/h3/a/text()").toString().trim();
        page.putField("tag", tag);
        System.out.println("\n");
    }
}
```

图12 抓取正文全文及标签

效果如图13所示：

```
get page: http://blog.sina.com.cn/s/blog_475b3d560102wdi4.html
title:  患上癌症后如何抗癌？我的抗癌心得
date:   2016-05-04 10:46:01
passage:        在生病期间以及康复后，有不少同样不幸患上癌症的病友或者家属留言问我，患上癌症后如何抗癌？需要注意哪些饮食？在化疗过程中是如何康复过来的？  复旦大学一位罹患乳腺癌的教师于娟，自知时日
tag:    杂谈
22:01:53,688  INFO HttpClientDownloader:87 - downloading page http://blog.sina.com.cn/s/blog_475b3d560102wc9p.html


get page: http://blog.sina.com.cn/s/blog_475b3d560102wc9p.html
title:  李开复给创业者的建议：打造百亿美元独角兽，必须要做三件事
date:   2016-04-16 15:24:43
passage:        本文为前不久我在“群英会”第二期内部培训中的演讲，群英会报道组根据现场录音整理，以下为全文。 去年我们去了一趟硅谷，去了10天，拜见了30位大佬。今天我可以用40分钟，把里面最精华的东西
tag:    杂谈
```

图13 抓取正文全文及标签

3. 修改上述爬虫，使得它可以抓取其他博主的博客文章，如“冯志伟文化博客”：http://blog.sina.com.cn/zwfengde2011【10】

答：只需要找到其他博主相应的文章起始页，然后修改起始页和相应的匹配前缀即可，具体如图14所示：

```
//被修改内容
public static final String URL_LIST = "http://blog\\.sina\\.com\\.cn/s/articlelist_1926267847_0_\\d+\\.html";


//1926267847
    public static void main(String[] args) {
        Spider.create(new BlogPageProcessor2()).addUrl("http://blog.sina.com.cn/s/articlelist_1926267847_0_1.html")
                .run();
    }

}
```

图14 修改内容

4. 对于无法用页面URL地址来做区分的列表+详情组合页，如NGPOD网站上动物类POD页面地址为：http://photography.nationalgeographic.com/photography/photo-of-the-day/animals/而动物类中某一个详情页的URL地址为：http://photography.nationalgeographic.com/photography/photo-of-the-day/bald-eagle-water-zoo/ 与上述列表页URL没有本质区别。如何修改上述爬虫，使得程序能够区分上述两种页面，并抓取出详情页的内容？【选做，10分】

答：可以寻找同样xpath路径下的元素，进行比较，从而区分两种页面，具体如图15所示：

```
String cmp=page.getHtml().xpath( "//div[@id='page_head'/h1/text()").toString();
if(cmp.equals("See All Photos")){
}
else{

}
```

图15 区分页面

对于这个个例而言，也可以判别是否起始页来区分，如图16所示：

```
String a=page.getUrl().toString();//获取当前url地址
fff=0;

if(a.equals("http://photography.nationalgeographic.com/photography/photo-of-the-day/animals"))
{

    for(String str: page.getHtml().xpath("//div[@id='search_results']").links().all())
    {
        page.addTargetRequest(new Request(str));
    }

}
else {
```

图16 区分页面

5. 实验总结

这个实验用的时间比较少，在经过了第二个子实验后，爬虫相应的用法已经基本清晰，所以实现起来很快，对于字符的匹配又学到了新的方法。