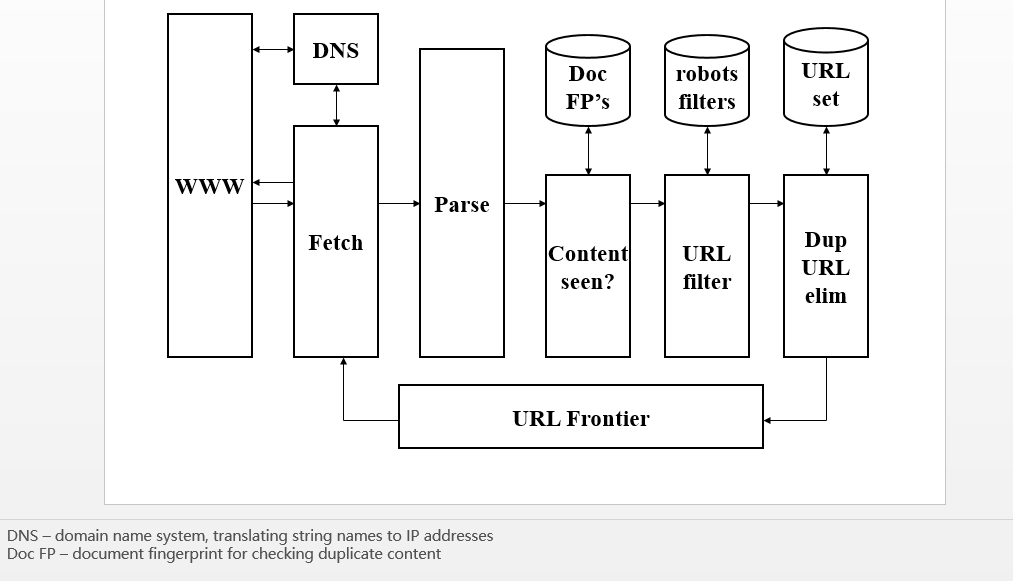
基本框架：



通过指纹来查重判断防止Spider trap：

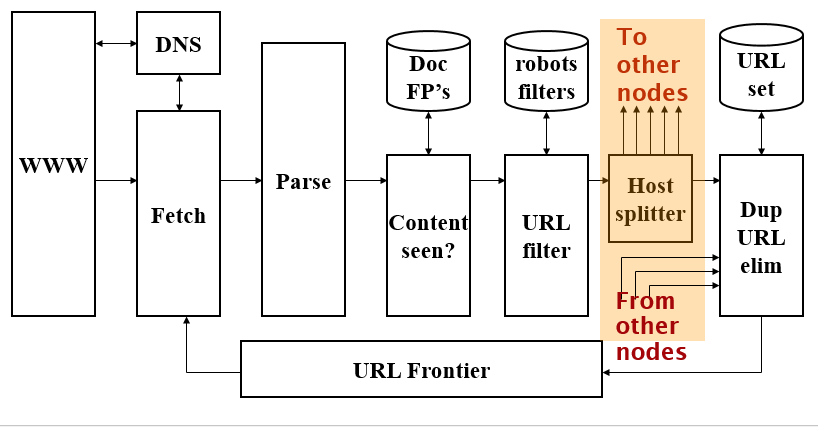
爬虫陷阱：

1.动态网页在url后面加上了一串码，每一次爬虫爬的时候这码都会不一样，这时候只要再将网页之间的关系弄成一个圈就GG

想办法创造出无限长度的URL：http://www.foo.com/bar/foo/bar/foo/bar/foo/bar/.....

2.document哦中包含了大量的字符，从而crash爬虫使用的解析器

分布式爬虫（多线程与多进程）之间需要增加一个HOSTs模块进行沟通：



负载均衡，需要一个总控，他们或者旁观者至少有一个知道所有的负载情况，从而做出调整

**a continuous crawl (the same page may be crawled repeatedly)**

**a non-continuous (one-shot) crawl**

1.注意Spider trap

2.可能包含URL的HTML标签：

* Link tag: <a href=“URL” … > … </a>
* Option tag: <option value=“URL”…> … </option> 放在select标签当中作为选择标签。value被设置成URL并不会造成用户的跳转，value只会被进行提交。
* Map: <area href=“URL” …>

<area> 标签定义图像映射中的区域（注：图像映射指得是带有可点击区域的图像）。

* Frame: <frame src=“URL” …>

<iframe src="http://www.divcss5.com/" width="400" height="200" scrolling="yes" />Src值为被嵌入网页地址Scrolling 是否有滚动条

* Link to an image: <img src=“URL” …>
* Relative path vs. absolute path: <base href= …>
* 注释中的URL标签

3.对相对URL路径的恢复

4.注意网站的防爬策略与出于礼貌**do not hit a web server too frequently** 增加一层选择器。

4.1如何调度针对不同站点的 Web 请求以减轻对方服务器的负担

5.rank高的网页就应该经常刷新（不能同等对待）

6. 一个高性能的 Web Crawler 系统里，DNS 查询也会成为急需优化的瓶颈

8.动态设置User-Agent

9.禁用cookie（有些网站靠cookie识别爬虫信息）

10.延迟下载

11.网站设置了cookie和登录，但是访问一些访问只需要cookie不需要登录

正规爬虫

0. 爬虫的优势：

爬虫目前主要开发语言为java、python、c++

有些公司也用go语言（杭州某互联网金融公司）

对于一般的信息采集需要，各种语言差别不大。

c、c++

搜索引擎无一例外使用C\C++ 开发爬虫，猜想搜索引擎爬虫采集的网站数量巨大，对页面的解析要求不高，部分支持javascript

python

网络功能强大，模拟登陆、解析javascript，短处是网页解析

python写起程序来真的很便捷，著名的python爬虫有scrapy等

java

java有很多解析器，对网页的解析支持很好，缺点是网络部分

java开源爬虫非常多，著名的如 nutch 国内有webmagic

java优秀的解析器有htmlparser、jsoup

对于一般性的需求无论java还是python都可以胜任。

如需要模拟登陆、对抗防采集选择python更方便些，如果需要处理复杂的网页，解析网页内容生成结构化数据或者对网页内容精细的解析则可以选择java。

1.不能对网站造成DOS攻击

2.遵守robots.txt：

从上至下一条条下来

**User-agent: webcrawler**

**Disallow: # no restriction for webcrawler**

**User-agent: lycra**

**Disallow: / # no access for robot lycra**

**User-agent: \***

**Disallow: /tmp # all other robots can index**

**Disallow: /logs # docs not under /tmp, /logs**

3. 遇到过input的name是xxx，但是post的时候变成submitOrderParam.xxx=0

4. 一些post提交的数据，可以直接删除，但是更好一点呢就是给出类型接近的数据，比如说全是0。

5. rd是random的缩写

6. 通过程序得到的html显示 与 浏览器F12内容 可能相差还不是一点。浏览器会对img标签的onMouseOu属性小写变成onmouseout，同时会将一些没有的标签补上。

7. 百度的页面跳转从2015年6月开始就不再使用直接的URL链接跳转，同时取消了Referer关键字。<https://www.baidu.com/link?url=dEvj7DE-rHNYgoNrzCSrgBLAVrWns06Ub1h8DjGa5mv-gSonnZp9VCYaDBKRmvDV&wd=&eqid=c1a3fa880009ab000000000555fce97f> 通过这个中间页面的处理，如果浏览器支持Javascript，那么将通过其中的js函数跳转到目标页面，如果浏览器不支持Javascript，那么也能通过最后的META标签让浏览器实现跳转。

如果存在这个字段，那么之后的网页通过Referer就能知道你之前访问了什么网站，也就是使用了什么关键词从百度跳转过来。因为加入一个一个中间连接，这样站长们就不知道自己的网站是用什么关键词调撞过来的。，

8.对自己指定的一个session进行login，就能够复活这个session。

9. 对选课课程的爬取，因为正常的源是不支持对每一页的数目支持，也就是说每一页显示的是固定的数目，所以要提高效率必须使用多线程异步爬虫。但是我换了一个思路，找到了第二个可以爬取当前学期课程的数据源。

但是也发现一个问题，如果想要一页返回所有的数据，也就是给一页设置一个很大很大的pageSise，服务器返回的时间可能会很长很长，本科系统返回一个学期开设的所有课程需要1分40秒，之后算上debug输出与记录到数据库，在本机又需要2分钟。如果使用多线程爬虫，那么就相当于是多个用户一起访问，服务器在瞬间按照道理来讲应该是支持这么大的数据量，所以推断是代码编写的问题。

10.如果网页是jsp编写，那么POST的参数可以变成GET那样，只要继续使用POST方法即可。这对我的QT编写很有帮助，在访问错误重新发包的时候。

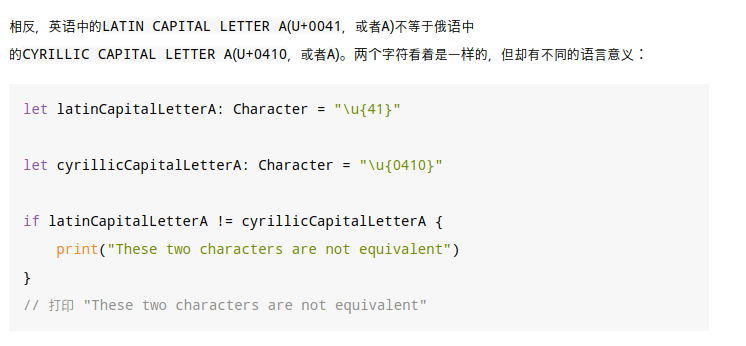
11.

对Unicode中文进行URL编码，每一个中文变成3个%xx

在线编码解码工具：<http://tool.chinaz.com/tools/urlencode.aspx>

对GBK中文进行URL编码，每一个中文变成2个

UTF-8,UTF-16,UTF-32是对Unicode的字符集编码。Unicode是编码字符集，一般是用十六进制表示数字，但是在计算机中一般用UTF-8来表示Unicode，汉字在Unicode是两个字节，但是用UTF-8表示就是三个字节。



12. 生成图片、Flash等，也有通过javascript加密邮箱、简单的加密等信息隐藏手段来防止爬虫中的匹配。

1. 增量爬取的思路：即保存上一次状态，本次抓取时与上次比对，如果不在上次的状态中，便视为增量，保存下来。
2. 断点重续, 就是重新下载的时候不要再下载已经下载的那部分

断点续传问题在批量下载的时候要考虑，已经下载的文件不要下载、下载到一半的文件尝试继续下载。

在爬虫批量下载一个网站上的资源,比如”飞火动态桌面”视频与图片下载,网站上不同index内容却会重复，所以说最好就是靠自己来实现去重，真的不值得相信,我这个任务中去重需要文件特征，这并不意味这需要将整个文件下载完，可以只是下载部分文件（刚好配合下载到一半的文件的断点续传），然后作为取样进行hash后判断 。

如果能够自己控制，那么估计要从socket开始编写了，除了能够实现上面的内容，还能实现最后多线程下载同一个文件。

1. 不阻塞下载，就意味着接受到的Response顺序可能是乱序的，所以这对文件命名是一件麻烦的事情。要么通过回调函数记住状态，要么通过URL来判断，比如url中的index。所以封装好的异步请求函数，都是一次性返回所有的请求数据，即便你请求的是一个视频文件。

16企业级爬虫居然也会使用docker去部署，好像是为了部署分布式爬虫。