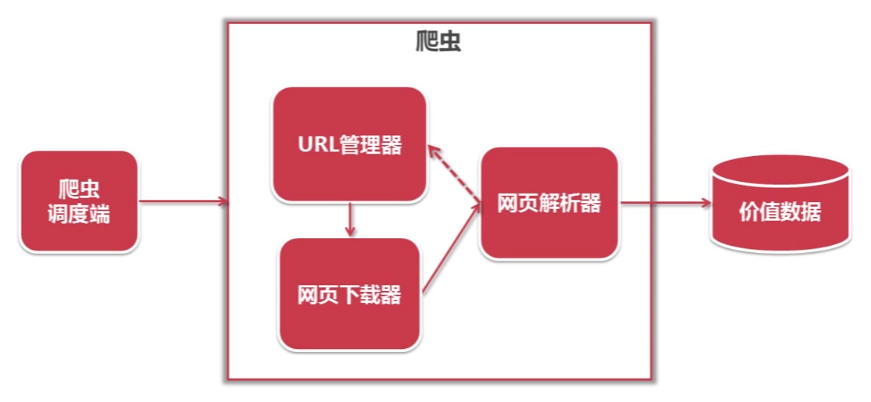
<https://www.imooc.com/learn/563>

## 爬虫简介

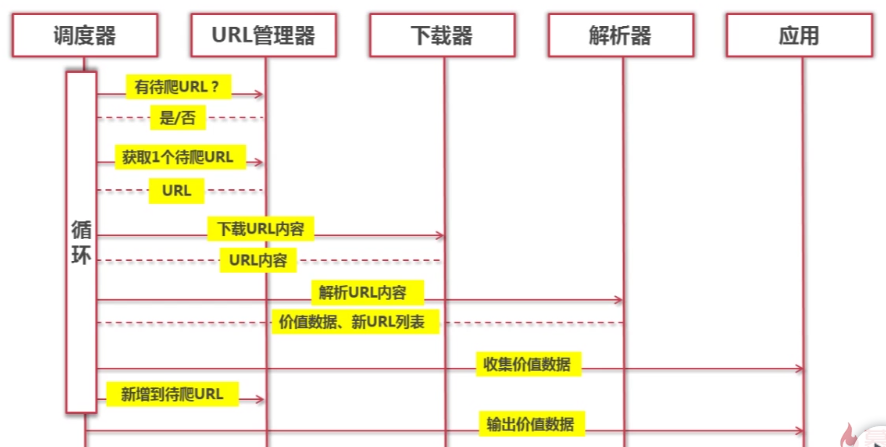
自动抓取互联网信息的程序。

## 爬虫架构

静态图

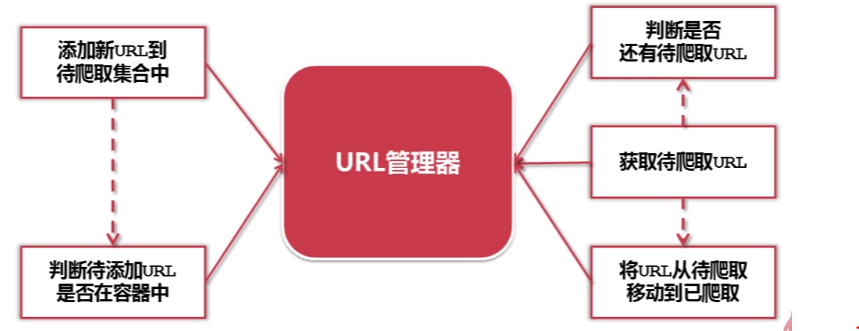


动态运行图：



## url管理器

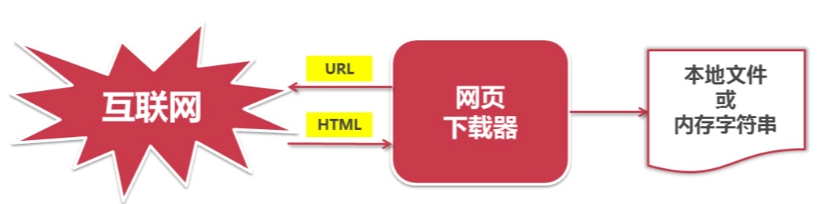
* 防止重复、循环抓取



实现方式：

内存、关系数据库、缓存数据库（redis）

## 下载器



urllib2（官方模块）、requests（第三方库）

urlib2使用：

方法1：最简洁方法。

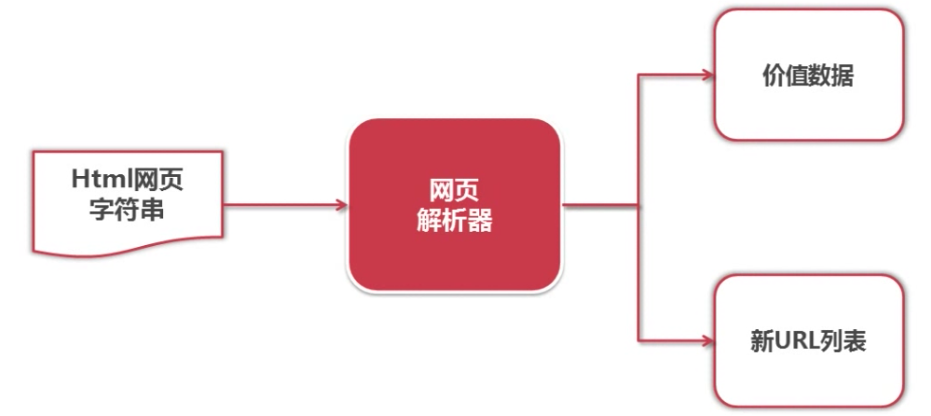
|  |
| --- |
| **import** urllib**.**request  **def** getCont**(**url**):**  response **=** urllib**.**request**.**urlopen**(**url**)**  **if** response**.**getcode**()** **!=** 200**:**  **print(**'获取%s失败'**%**url**)**  **return**  **print(**response**.**read**())**    url **=** 'http://www.baidu.com'  getCont**(**url**)**  ***注意***：urllib2在python3中不能使用，所有urllib2的地方改为urllib.request |

方法2：可以添加data和http header信息：

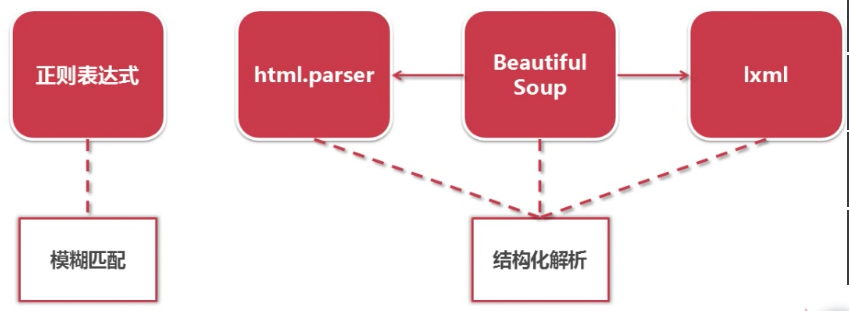
和python3兼容性不好

## 解析器

从网页中提取有价值信息的工具



python 的网页解析器：



beautiful soup4：

pip3 install beautifulsoup4

|  |
| --- |
| **from** bs4 **import** BeautifulSoup  **import** re  html\_doc **=** ''  **with** open**(**'./htmls/url.html'**,**'r'**)** **as** f**:**  html\_doc **=** f**.**read**()**  soup **=** BeautifulSoup**(**html\_doc**,**'html.parser'**,**from\_encoding **=** 'utf-8'**)**  **print(**'所有连接'**)**  links **=** soup**.**find\_all**(**'a'**,**attrs **=** **{**'href'**:True})**  **for** link **in** links**:**  **print(**link**.**name**,**link**[**'href'**],**link**.**get\_text**())**  **print(**'一个连接'**)**  link **=** soup**.**find**(**'a'**,**href **=** 'https://www.baidu.com/duty/'**)**  **print(**link**.**name**,**link**[**'href'**],**link**.**get\_text**())**  **print(**'正则表达式'**)**  link **=** soup**.**find**(**'a'**,**href **=** re**.**compile**(**r'duty'**))**  **print(**link**.**name**,**link**[**'href'**],**link**.**get\_text**())**  **print(**'获取id=wrapper'**)**  link **=** soup**.**find**(**'div'**,**id**=**'wrapper'**)**  **print(**link**.**name**,**link**.**get\_text**())** |