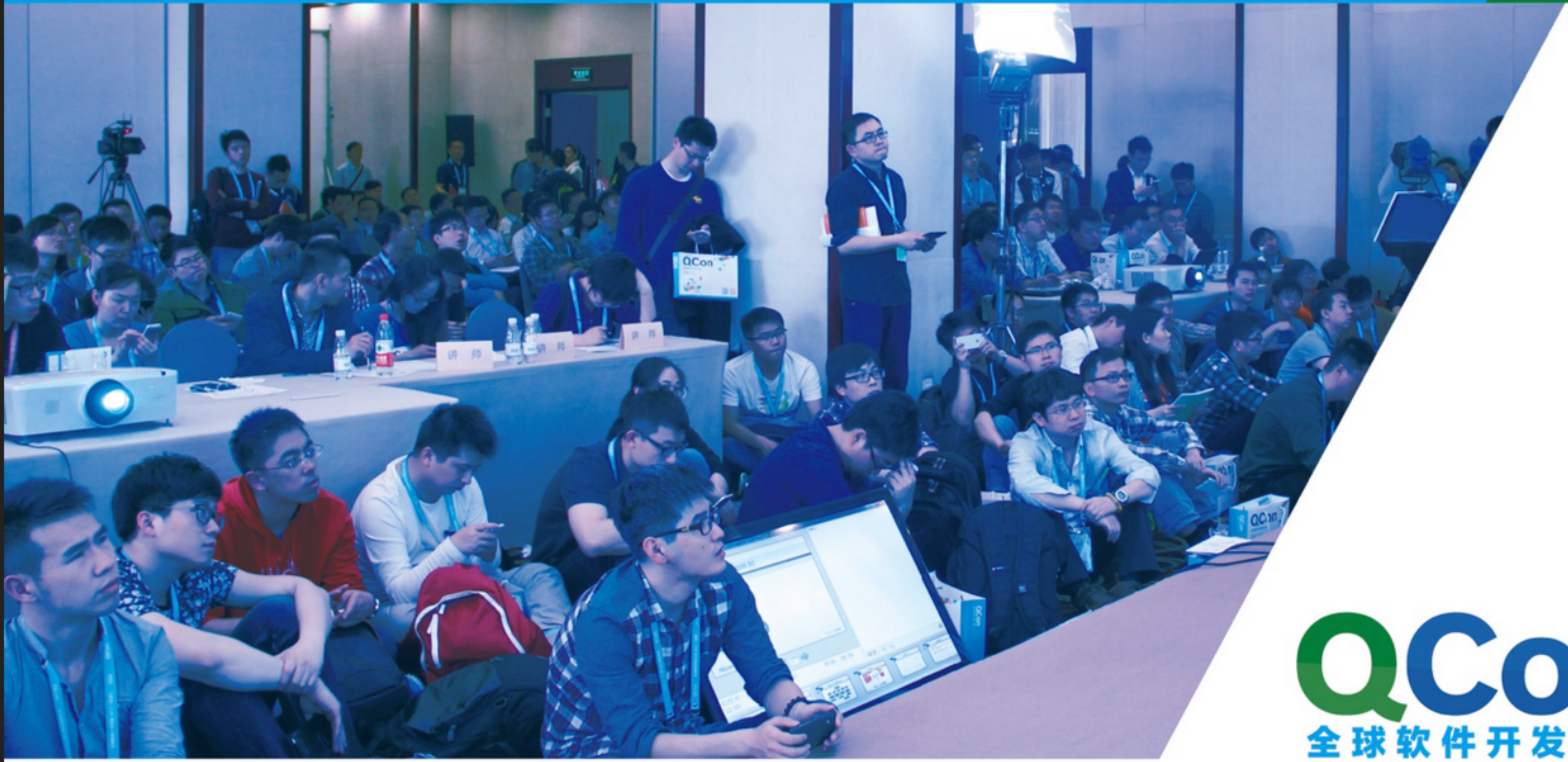
QCon全球软件开发大会

International Software Development Conference



QCon
全球软件开发大会

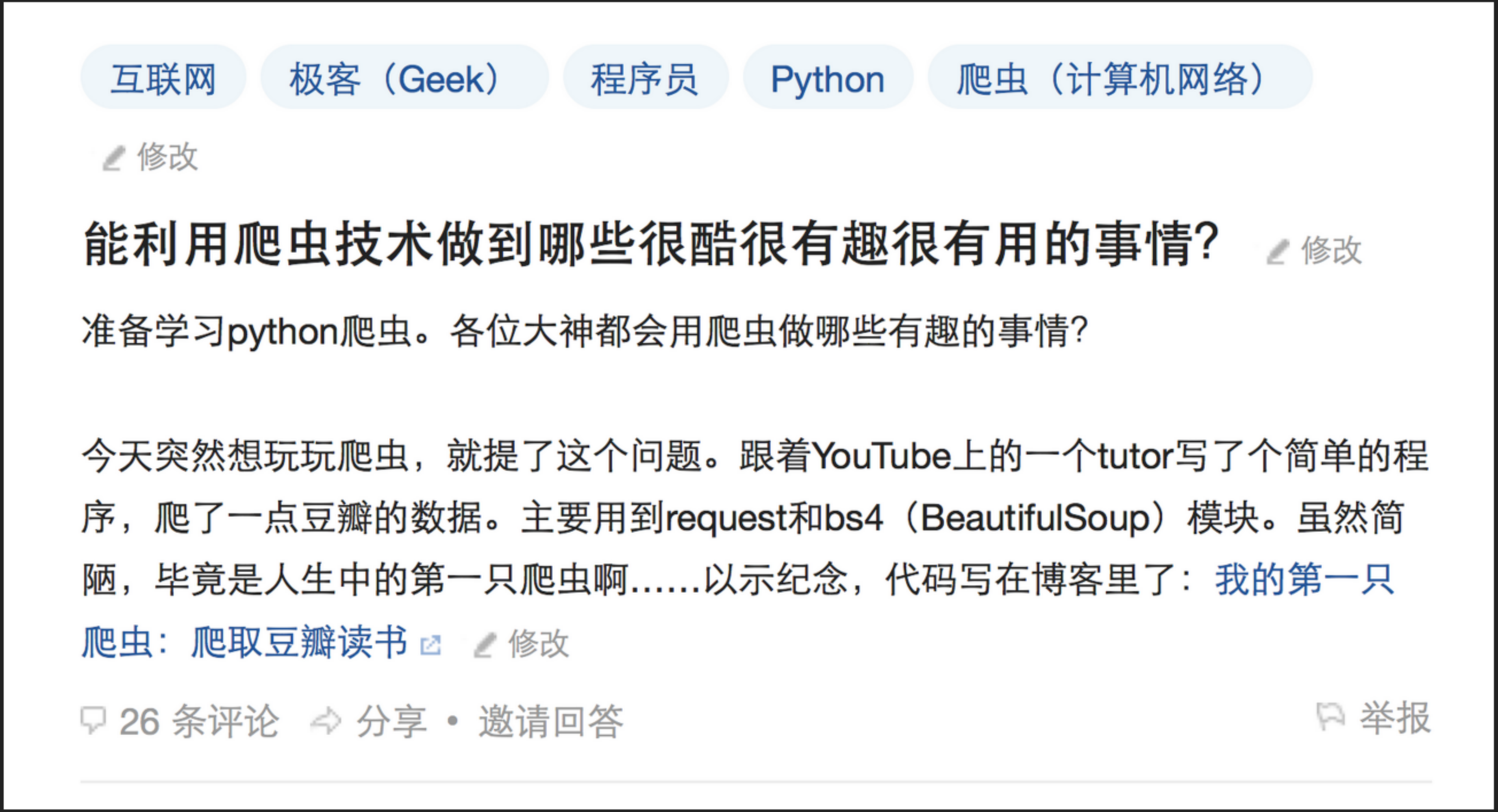
互联网信息获取技术实践

- 云端爬虫养成记

费良宏

AWS Technical Evangelist

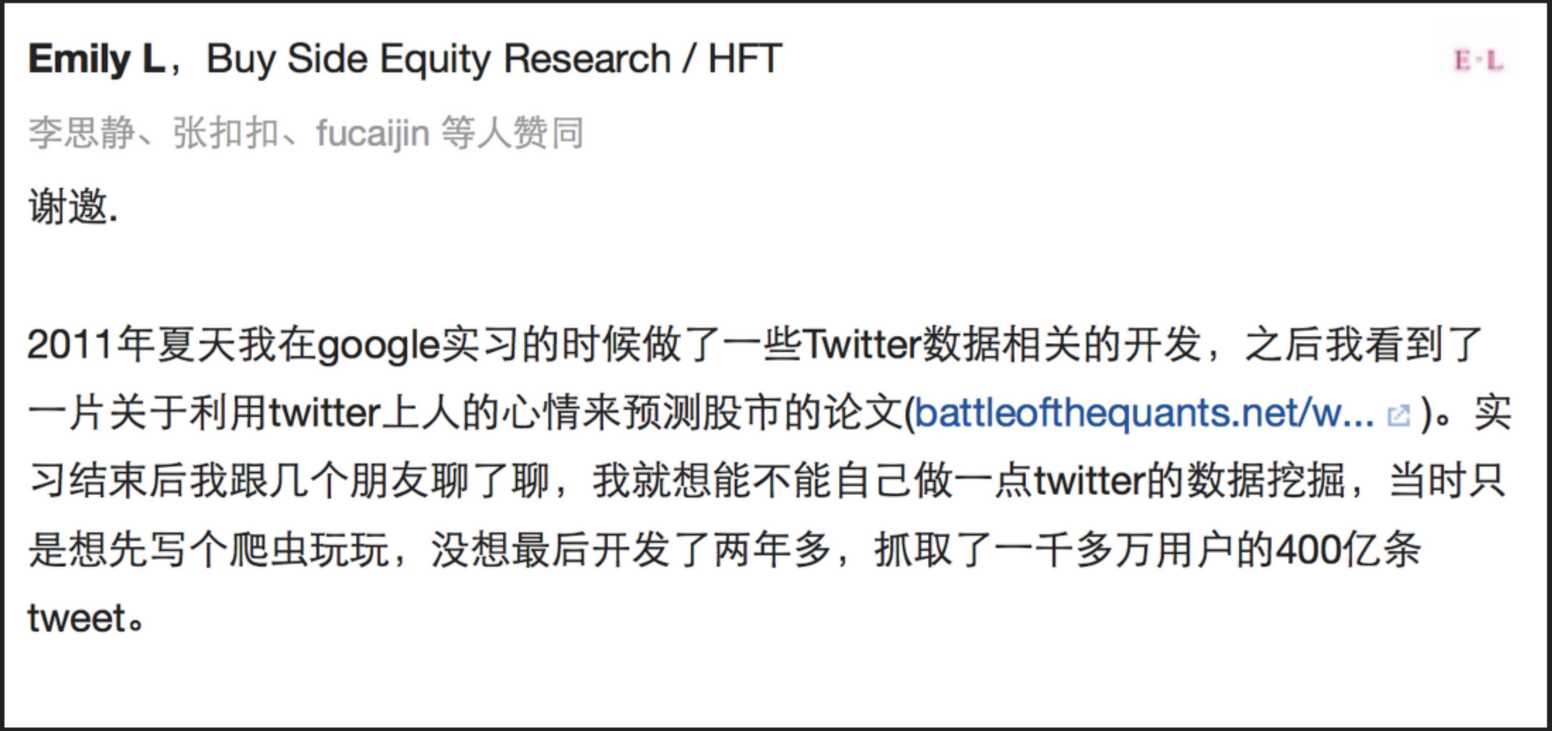
故事的开始



能利用爬虫技术做到哪些很酷很有趣很有用的事情?

准备学习python爬虫。各位大神都会用爬虫做哪些有趣的事情?

今天突然想玩玩爬虫，就提了这个问题。跟着YouTube上的一个tutor写了个简单的程序，爬了一点豆瓣的数据。主要用到request和bs4（BeautifulSoup）模块。虽然简陋，毕竟是人生中的第一只爬虫啊......以示纪念，代码写在博客里了：我的第一只爬虫：爬取豆瓣读书



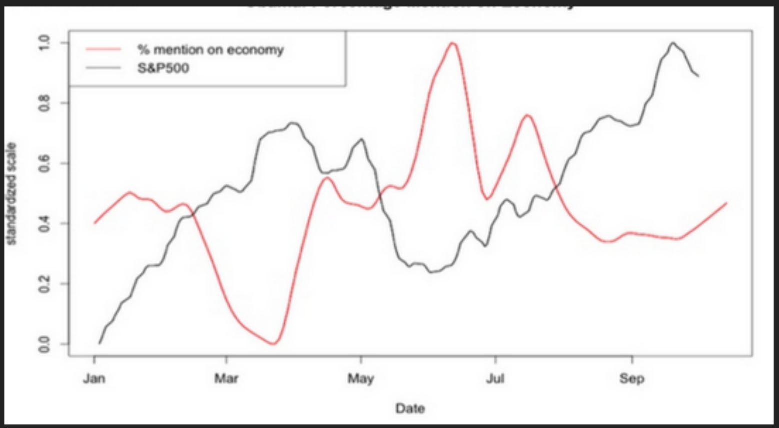
Emily L，Buy Side Equity Research / HFT

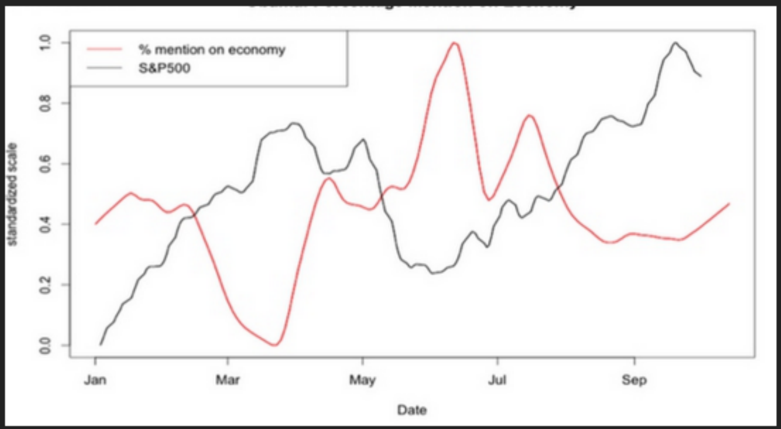
E·L

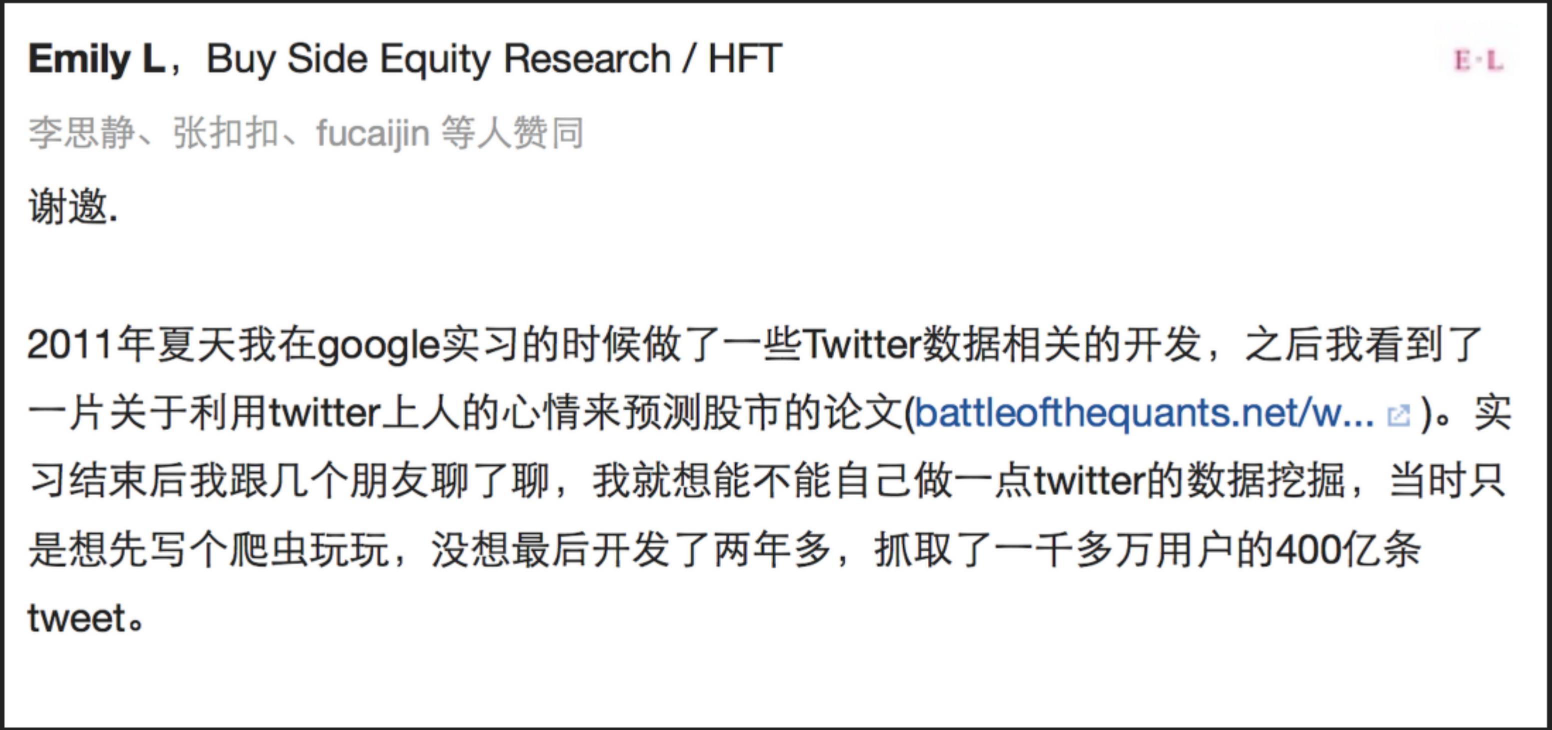
李思静、张扣扣、fucaijin 等人赞同

谢邀.

2011年夏天我在google实习的时候做了一些Twitter数据相关的开发，之后我看到了一片关于利用twitter上人的心情来预测股市的论文(battleofthequants.net/w...)。实习结束后我跟几个朋友聊了聊，我就想能不能自己做一点twitter的数据挖掘，当时只是想先写个爬虫玩玩，没想最后开发了两年多，抓取了一千多万用户的400亿条tweet。





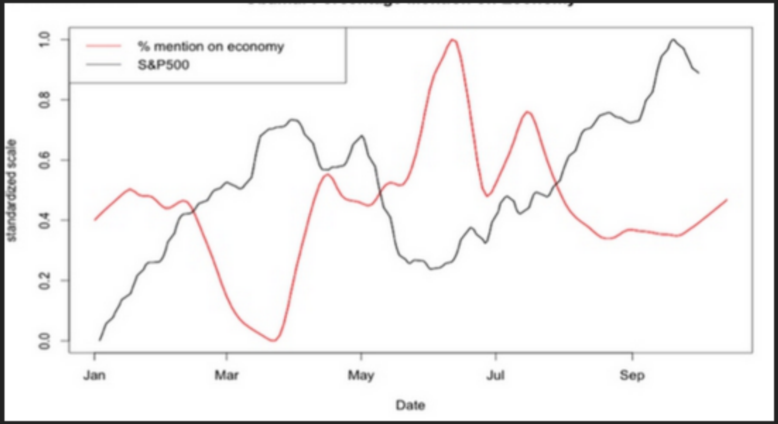


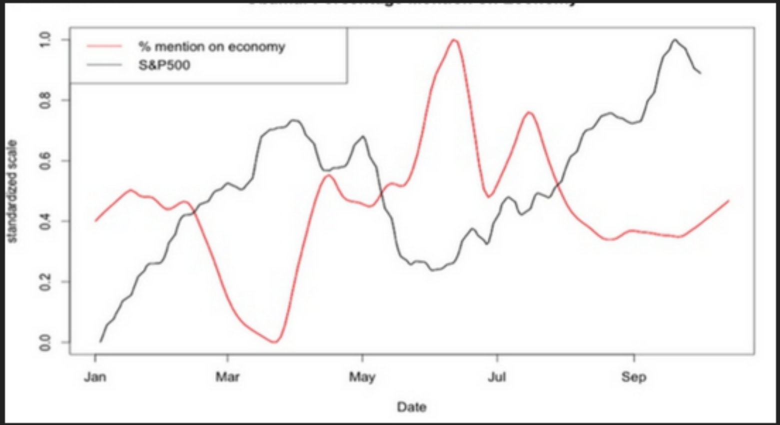
Emily L，Buy Side Equity Research / HFT

李思静、张扣扣、fucaijin 等人赞同

谢邀.

2011年夏天我在google实习的时候做了一些Twitter数据相关的开发，之后我看到了一片关于利用twitter上人的心情来预测股市的论文(battleofthequants.net/w...)。实习结束后我跟几个朋友聊了聊，我就想能不能自己做一点twitter的数据挖掘，当时只是想先写个爬虫玩玩，没想最后开发了两年多，抓取了一千多万用户的400亿条tweet。





为什么需要We 数据抓取?

没有在深夜痛哭不配谈人生，不拥有数据何谈什么大数据和算法!

- 精确获取、特定站点，不同于搜索引擎
- 无API 或标准接口／不开放的数据
- 计算机可处理，结构化的数据

我的第一个爬虫

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

import requests
r = requests.get('https://api.github.com/events')
print r.status_code
print r.text
```

HTML 解析器 vs. 正则表达式

我的结论

- 真实世界的 Web 页面复杂而缺少一致性
- LXML + XPath 是更好的选择
- 可读性、易维护性
- cssselector 是XPath 的替代方案

开源框架 vs. DIY

我的选择

- 一个功能良好的框架是复杂并且工作量很大
- 技术的发展与标准的改进的压力
- 专注与框架或是应用本身

- 我的选择是 – Scrapy

Scrapy 是什么鬼东西?

"An open source and collaborative framework for extracting the data you need from websites. In a fast, simple, yet extensible way."

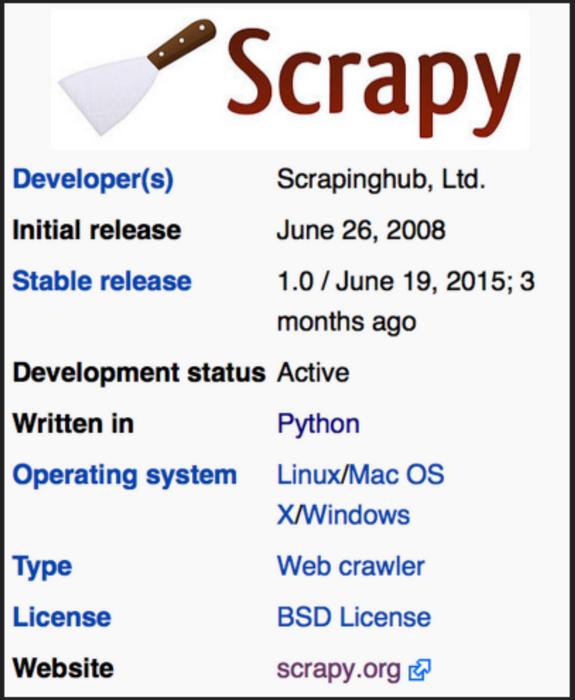
* 177 contributors

* 10k+ stars, 3k+ forks and 943 watchers on GitHub

* 2.2k followers on Twitter

* 4.1k questions on StackOverflow

* 2.5k members on mailing list



Developer(s)	Scrapinghub, Ltd.
Initial release	June 26, 2008
Stable release	1.0 / June 19, 2015; 3 months ago
Development status	Active
Written in	Python
Operating system	Linux/Mac OS X/Windows
Type	Web crawler
License	BSD License
Website	scrapy.org

Scrapy 是什么鬼东西?

"An open source and collaborative framework for extracting the data you need from websites. In a fast, simple, yet extensible way."

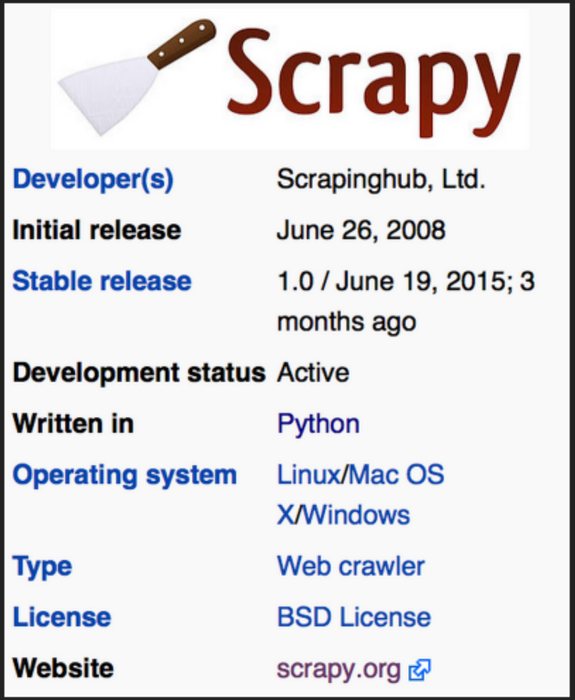
* 177 contributors

* 10k+ stars, 3k+ forks and 943 watchers on GitHub

* 2.2k followers on Twitter

* 4.1k questions on StackOverflow

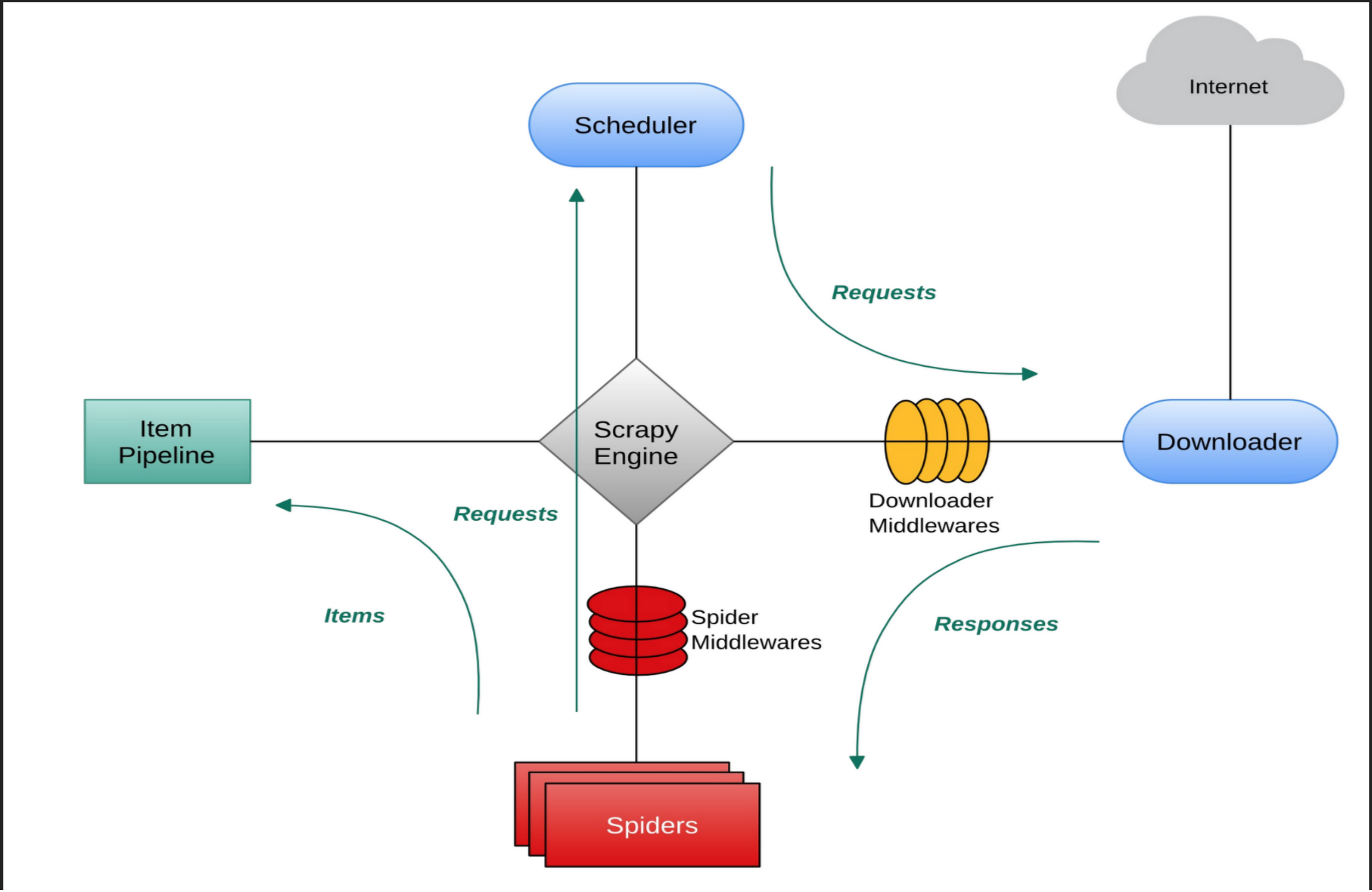
* 2.5k members on mailing list



Scrapy

Developer(s)	Scrapinghub, Ltd.
Initial release	June 26, 2008
Stable release	1.0 / June 19, 2015; 3 months ago
Development status	Active
Written in	Python
Operating system	Linux/Mac OS X/Windows
Type	Web crawler
License	BSD License
Website	scrapy.org

Scrapy 的架构


Internet
Scheduler
Requests
Item Pipeline
Scrapy Engine
Downloader
Downloader Middlewares
Requests
Items
Spider Middlewares
Responses
Spiders

我的第一个Spider

```
$ pip install scrapy
$ cat > myspider.py <<EOF
import scrapy

class BlogSpider(scrapy.Spider):
    name = 'blogspider'
    start_urls = ['http://blog.scrapinghub.com']

    def parse(self, response):
        for url in response.css('ul li a::attr("href")').re(r'.*/\d\d\d\d/\d\d/$'):
            yield scrapy.Request(response.urljoin(url), self.parse_titles)

    def parse_titles(self, response):
        for post_title in response.css('div.entries > ul > li a::text').extract():
            yield {'title': post_title}
EOF
$ scrapy runspider myspider.py
```

我的第一个Spider

```
$ pip install scrapy
$ cat > myspider.py <<EOF
import scrapy

class BlogSpider(scrapy.Spider):
    name = 'blogspider'
    start_urls = ['http://blog.scrapinghub.com']

    def parse(self, response):
        for url in response.css('ul li a::attr("href")').re(r'.*/\d\d\d\d/\d\d/$'):
            yield scrapy.Request(response.urljoin(url), self.parse_titles)

    def parse_titles(self, response):
        for post_title in response.css('div.entries > ul > li a::text').extract():
            yield {'title': post_title}
EOF
$ scrapy runspider myspider.py
```

我遇到的第一个麻烦 － 反爬虫

怎么办?

- Cookie ? COOKIES_ENABLED = False
- User Agent ? scrapy-fake-useragent
- 增加延迟 ? DOWNLOAD_DELAY = 0.25
- 减小并发 ? CONCURRENT_REQUESTS = 2
- IP 地址 ? 这个比较麻烦...

解决 IP 地址问题的尝试

解决的思路

- ProxyMesh？"贵"就一个字
- Free Proxy？挺不靠谱的
- Google cache？ 无法普遍适用
- Tor？这个有点意思

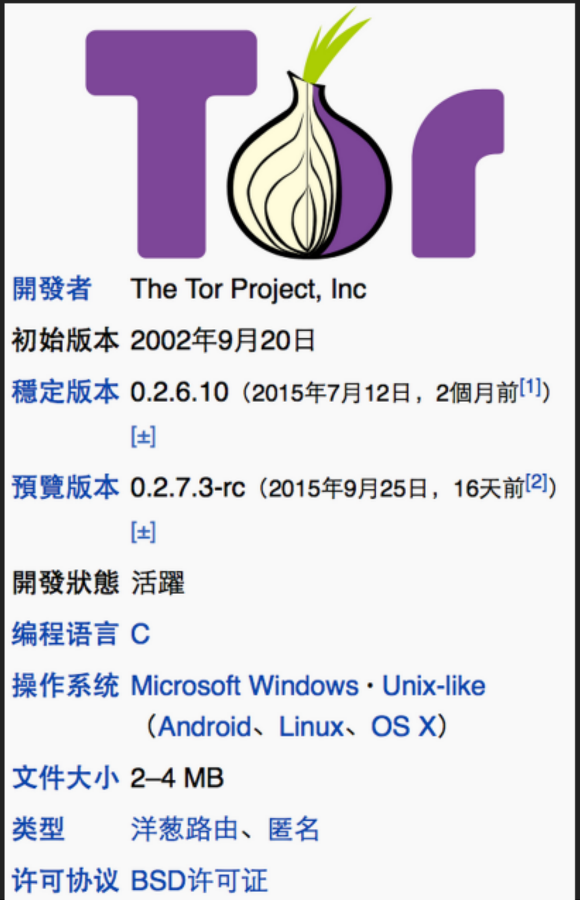
Tor 的前世今生

"Tor（The Onion Router，洋葱路由器）是实现匿名通信的自由软件。Tor是第二代洋葱路由的一种实现，用户通过Tor可以在互联网上进行匿名交流。最初该项目由美国海军研究实验室赞助。2004年后期，Tor成为电子前哨基金会（EFF）的一个项目。" -- Wikipedia

* 专门防范流量过虑、嗅探分析

* 可以匿名进行TCP传输

* 加密信息在路由器间层层传递，最后到达“出口节点”



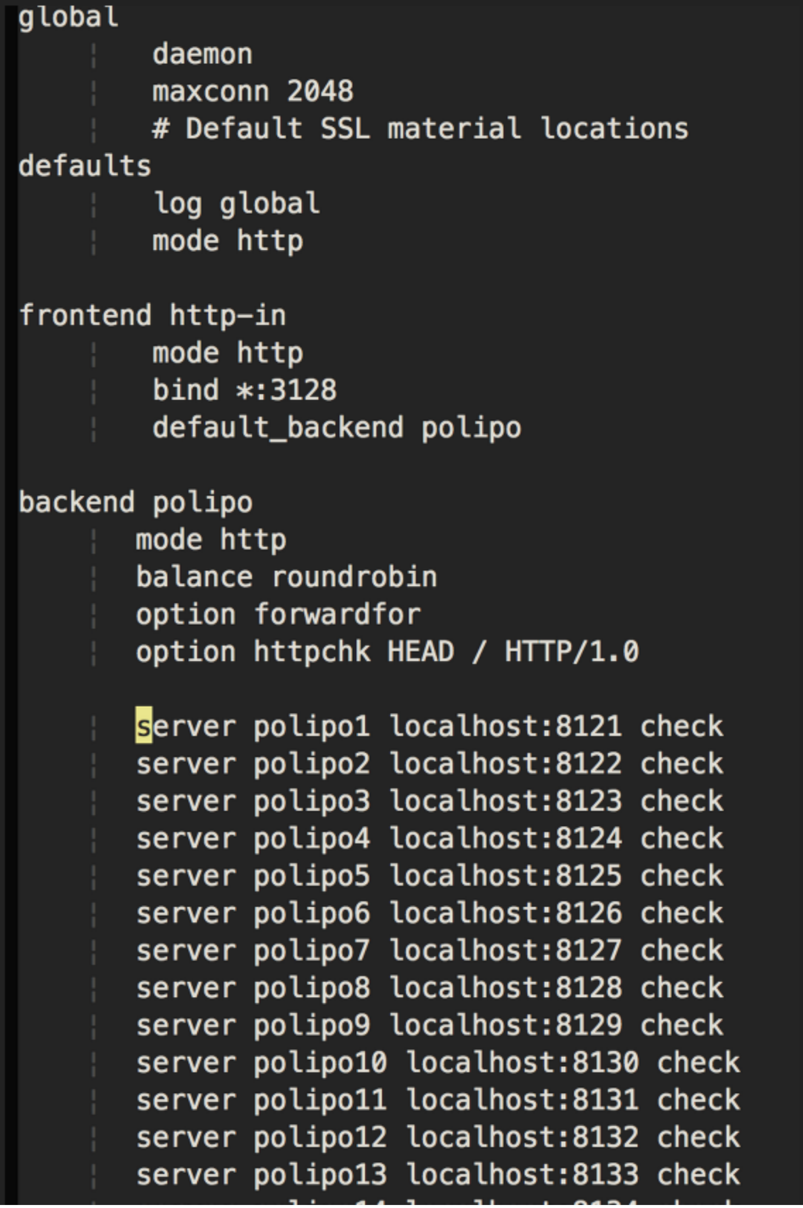
開發者	The Tor Project, Inc
初始版本	2002年9月20日
穩定版本	0.2.6.10（2015年7月12日，2個月前[1]） [±]
預覽版本	0.2.7.3-rc（2015年9月25日，16天前[2]） [±]
開發狀態	活躍
编程语言	C
操作系统	Microsoft Windows · Unix-like （Android、Linux、OS X）
文件大小	2–4 MB
类型	洋葱路由、匿名
许可协议	BSD许可证

Tor 与 Scrapy 的结合

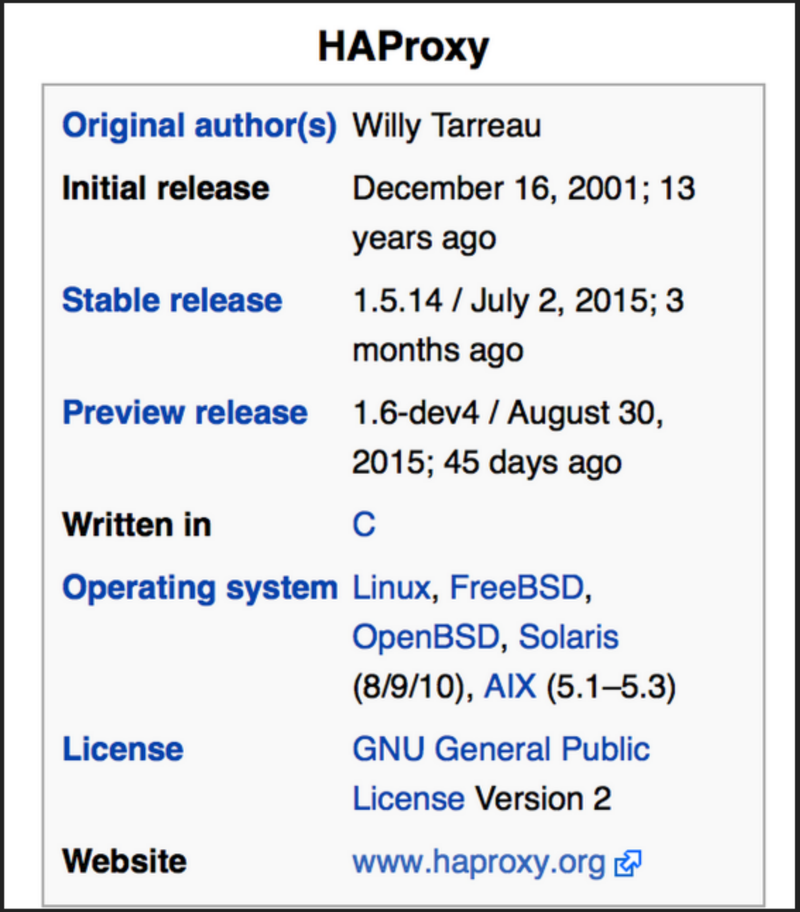
Scrapy ----- Haproxy ----- Polipo ----- Tor

```
tor -f ./tor/tor1/torrc
tor -f ./tor/tor2/torrc
tor -f ./tor/tor3/torrc
tor -f ./tor/tor5/torrc
...

polipo -c ./polipo/polipo0/config
polipo -c ./polipo/polipo1/config
polipo -c ./polipo/polipo3/config
...

/usr/sbin/haproxy -f ./haproxy/haproxy.cfg
```

Haproxy - 代理服务器轮询



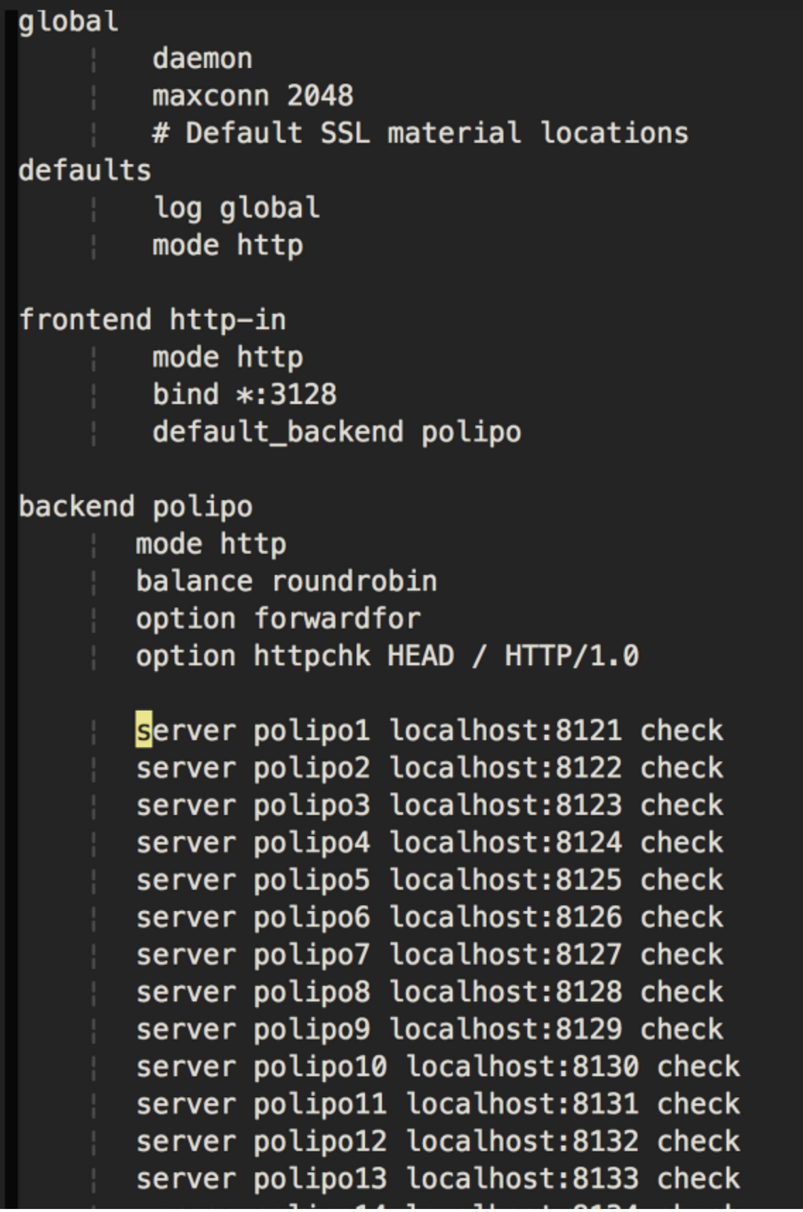
```
global
        daemon
        maxconn 2048
        # Default SSL material locations
defaults
        log global
        mode http

frontend http-in
        mode http
        bind *:3128
        default_backend polipo

backend polipo
       mode http
       balance roundrobin
       option forwardfor
       option httpchk HEAD / HTTP/1.0

       server polipo1 localhost:8121 check
       server polipo2 localhost:8122 check
       server polipo3 localhost:8123 check
       server polipo4 localhost:8124 check
       server polipo5 localhost:8125 check
       server polipo6 localhost:8126 check
       server polipo7 localhost:8127 check
       server polipo8 localhost:8128 check
       server polipo9 localhost:8129 check
       server polipo10 localhost:8130 check
       server polipo11 localhost:8131 check
       server polipo12 localhost:8132 check
       server polipo13 localhost:8133 check
```



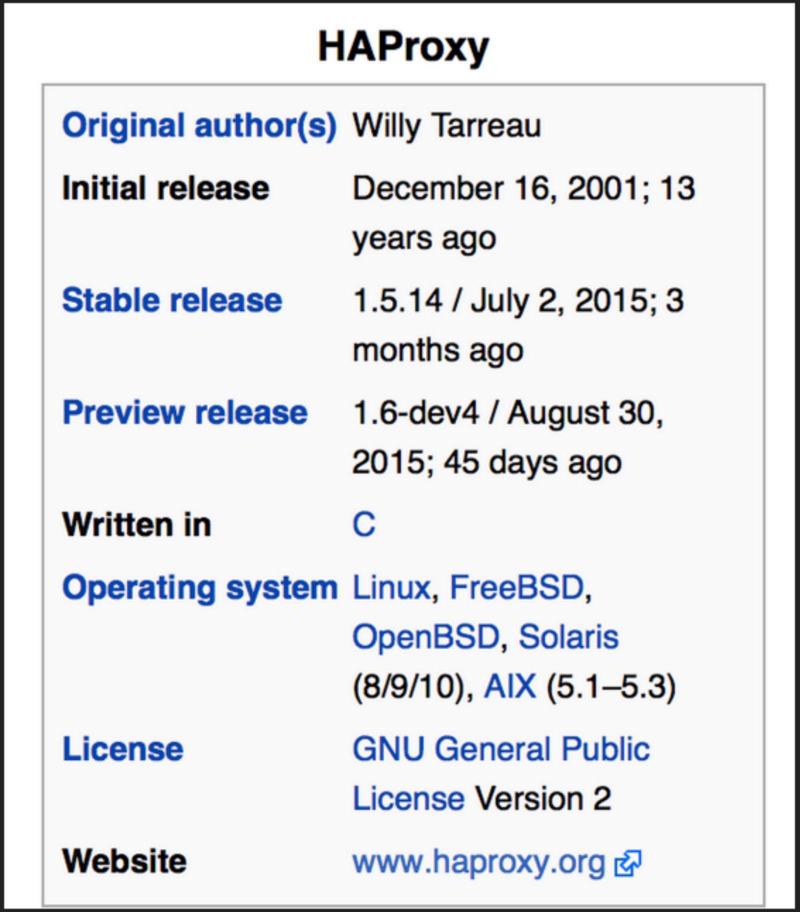
HAProxy

Original author(s)	Willy Tarreau
Initial release	December 16, 2001; 13 years ago
Stable release	1.5.14 / July 2, 2015; 3 months ago
Preview release	1.6-dev4 / August 30, 2015; 45 days ago
Written in	C
Operating system	Linux, FreeBSD, OpenBSD, Solaris (8/9/10), AIX (5.1–5.3)
License	GNU General Public License Version 2
Website	www.haproxy.org

Haproxy - 代理服务器轮询



```
global
        daemon
        maxconn 2048
        # Default SSL material locations
defaults
        log global
        mode http

frontend http-in
        mode http
        bind *:3128
        default_backend polipo

backend polipo
       mode http
       balance roundrobin
       option forwardfor
       option httpchk HEAD / HTTP/1.0

       server polipo1 localhost:8121 check
       server polipo2 localhost:8122 check
       server polipo3 localhost:8123 check
       server polipo4 localhost:8124 check
       server polipo5 localhost:8125 check
       server polipo6 localhost:8126 check
       server polipo7 localhost:8127 check
       server polipo8 localhost:8128 check
       server polipo9 localhost:8129 check
       server polipo10 localhost:8130 check
       server polipo11 localhost:8131 check
       server polipo12 localhost:8132 check
       server polipo13 localhost:8133 check
       server polipo14 localhost:8134 check
```

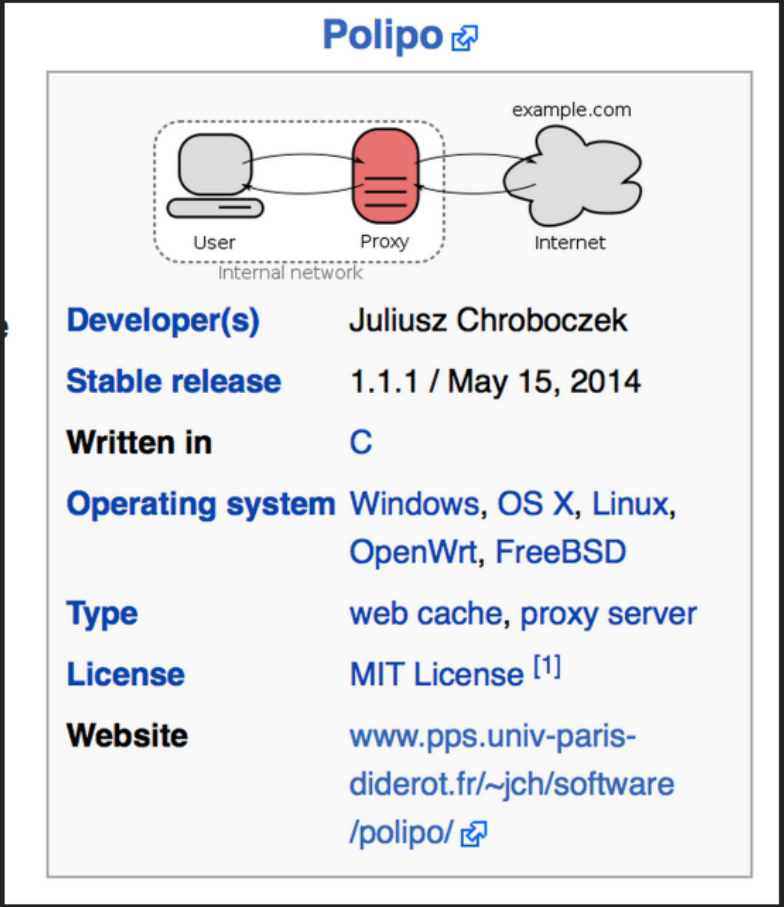


HAProxy

Original author(s)	Willy Tarreau
Initial release	December 16, 2001; 13 years ago
Stable release	1.5.14 / July 2, 2015; 3 months ago
Preview release	1.6-dev4 / August 30, 2015; 45 days ago
Written in	C
Operating system	Linux, FreeBSD, OpenBSD, Solaris (8/9/10), AIX (5.1–5.3)
License	GNU General Public License Version 2
Website	www.haproxy.org

Polipo - Sockets 5 Proxy 转换 HTTP Proxy

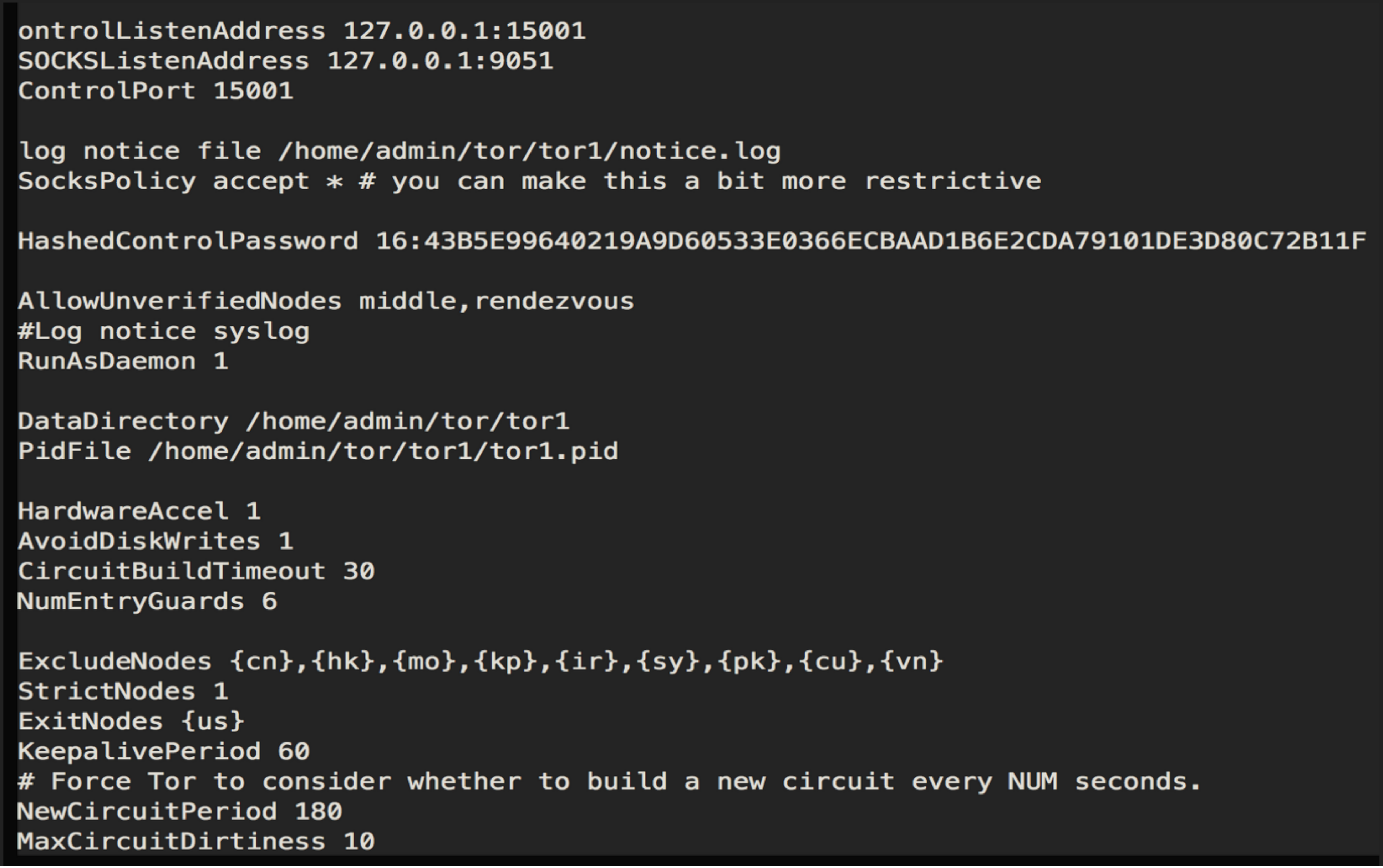
```
dnsQueryIPv6 = no
logLevel = 0xFF
logFile = /home/admin/polipo/polipo1/polipo1.log
pidFile = /home/admin/polipo/polipo1/polipo1.pid

daemonise = true

proxyPort = 8121
proxyAddress = "127.0.0.1"

socksProxyType = socks5
socksParentProxy = "127.0.0.1:9051"

diskCacheRoot = ""
disableLocalInterface = true

dnsNameServer = "8.8.8.8"
dnsUseGethostbyname = yes
```



Polipo

Developer(s)	Juliusz Chroboczek
Stable release	1.1.1 / May 15, 2014
Written in	C
Operating system	Windows, OS X, Linux, OpenWrt, FreeBSD
Type	web cache, proxy server
License	MIT License [1]
Website	www.pps.univ-paris-diderot.fr/~jch/software/polipo/

Tor - 匿名网络的配置



```
ontrolListenAddress 127.0.0.1:15001
SOCKSListenAddress 127.0.0.1:9051
ControlPort 15001

log notice file /home/admin/tor/tor1/notice.log
SocksPolicy accept * # you can make this a bit more restrictive

HashedControlPassword 16:43B5E99640219A9D60533E0366ECBAAD1B6E2CDA79101DE3D80C72B11F

AllowUnverifiedNodes middle,rendezvous
#Log notice syslog
RunAsDaemon 1

DataDirectory /home/admin/tor/tor1
PidFile /home/admin/tor/tor1/tor1.pid

HardwareAccel 1
AvoidDiskWrites 1
CircuitBuildTimeout 30
NumEntryGuards 6

ExcludeNodes {cn},{hk},{mo},{kp},{ir},{sy},{pk},{cu},{vn}
StrictNodes 1
ExitNodes {us}
KeepalivePeriod 60
# Force Tor to consider whether to build a new circuit every NUM seconds.
NewCircuitPeriod 180
MaxCircuitDirtiness 10
```

Tor 是不是终极的方案?

问题依然存在

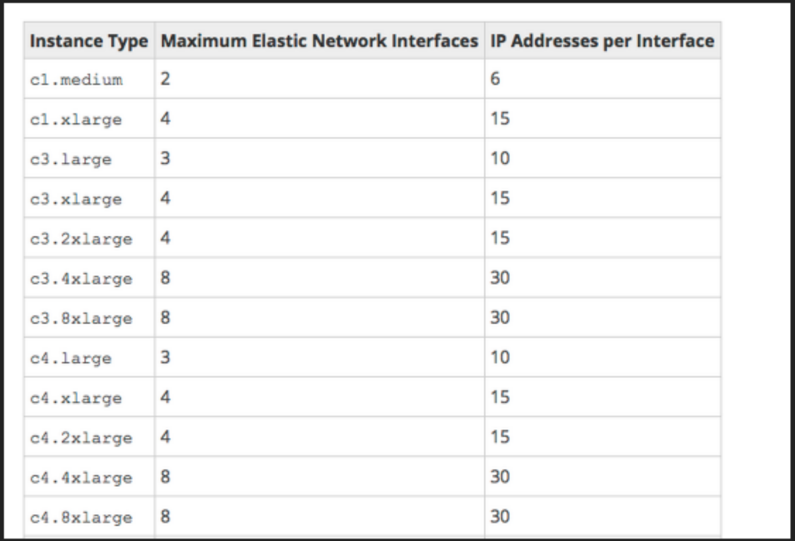
- 网络延迟较大，单条链路性能不高
- 稳定性无法保障，出口节点网络的限制
- Tor 控制协议需要二次开发
- 屏蔽Tor 的技术风险始终存在

IP 资源的难题的新解法

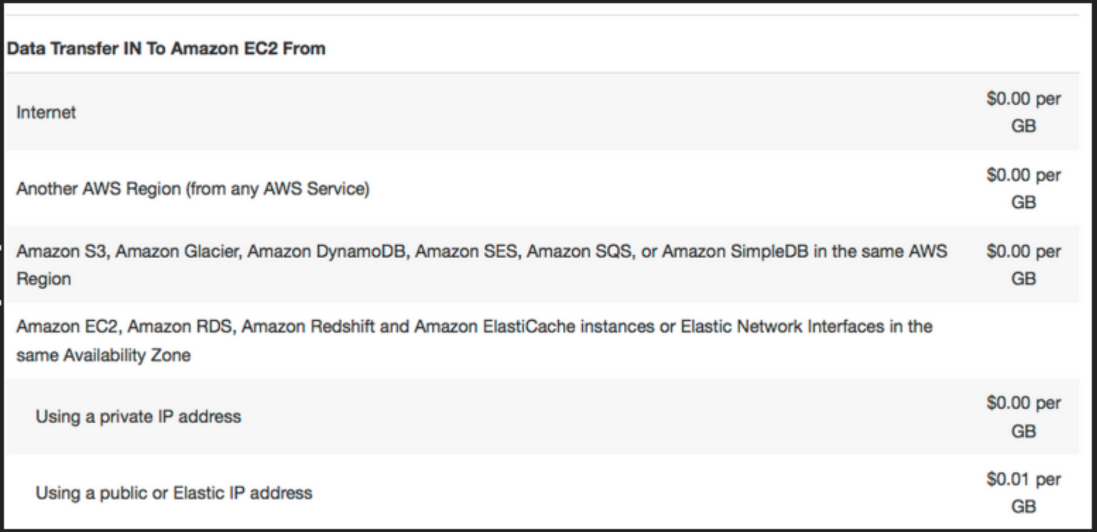
哪里有足够多的IP?

- 云计算
- Amazon Web Services
- EC2 + Elastic Network Interface(ENI) + Elastics IP(EIP)

AWS 上的IP资源

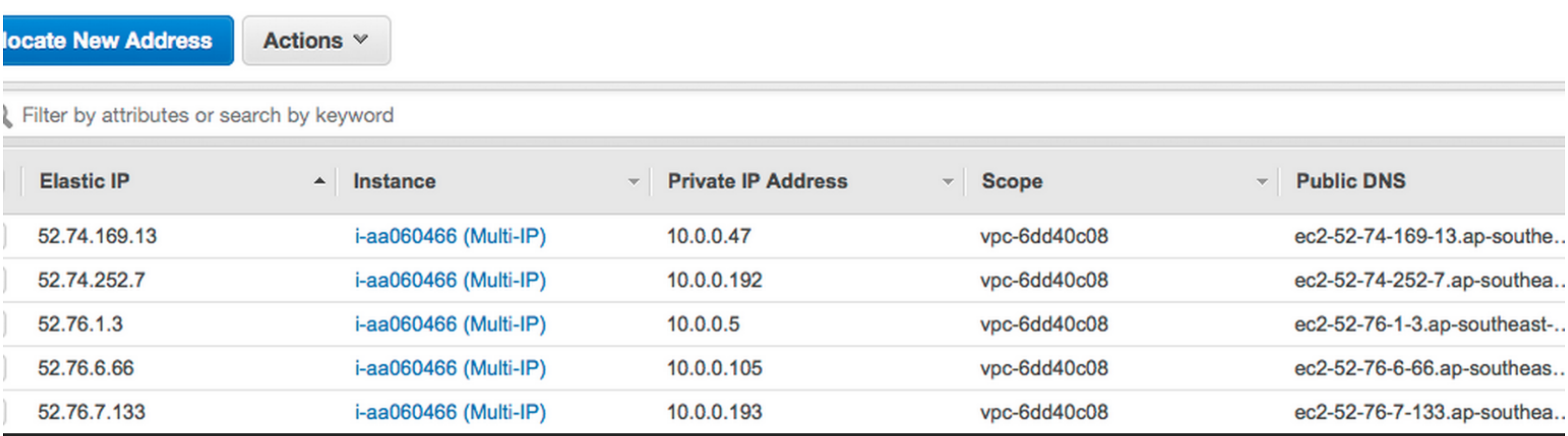


Instance Type	Maximum Elastic Network Interfaces	IP Addresses per Interface
c1.medium	2	6
c1.xlarge	4	15
c3.large	3	10
c3.xlarge	4	15
c3.2xlarge	4	15
c3.4xlarge	8	30
c3.8xlarge	8	30
c4.large	3	10
c4.xlarge	4	15
c4.2xlarge	4	15
c4.4xlarge	8	30
c4.8xlarge	8	30



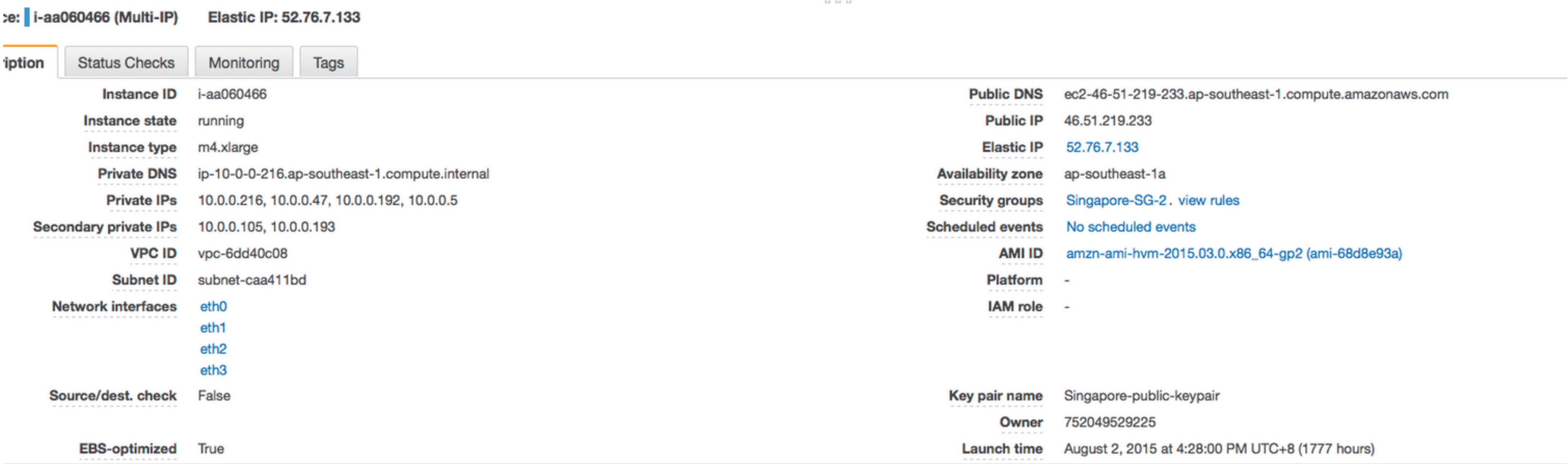
Data Transfer IN To Amazon EC2 From	
Internet	$0.00 per GB
Another AWS Region (from any AWS Service)	$0.00 per GB
Amazon S3, Amazon Glacier, Amazon DynamoDB, Amazon SES, Amazon SQS, or Amazon SimpleDB in the same AWS Region	$0.00 per GB
Amazon EC2, Amazon RDS, Amazon Redshift and Amazon ElastiCache instances or Elastic Network Interfaces in the same Availability Zone	
Using a private IP address	$0.00 per GB
Using a public or Elastic IP address	$0.01 per GB

AWS上的EIP



Elastic IP	Instance	Private IP Address	Scope	Public DNS
52.74.169.13	i-aa060466 (Multi-IP)	10.0.0.47	vpc-6dd40c08	ec2-52-74-169-13.ap-southe..
52.74.252.7	i-aa060466 (Multi-IP)	10.0.0.192	vpc-6dd40c08	ec2-52-74-252-7.ap-southea..
52.76.1.3	i-aa060466 (Multi-IP)	10.0.0.5	vpc-6dd40c08	ec2-52-76-1-3.ap-southeast-..
52.76.6.66	i-aa060466 (Multi-IP)	10.0.0.105	vpc-6dd40c08	ec2-52-76-6-66.ap-southeas..
52.76.7.133	i-aa060466 (Multi-IP)	10.0.0.193	vpc-6dd40c08	ec2-52-76-7-133.ap-southea..

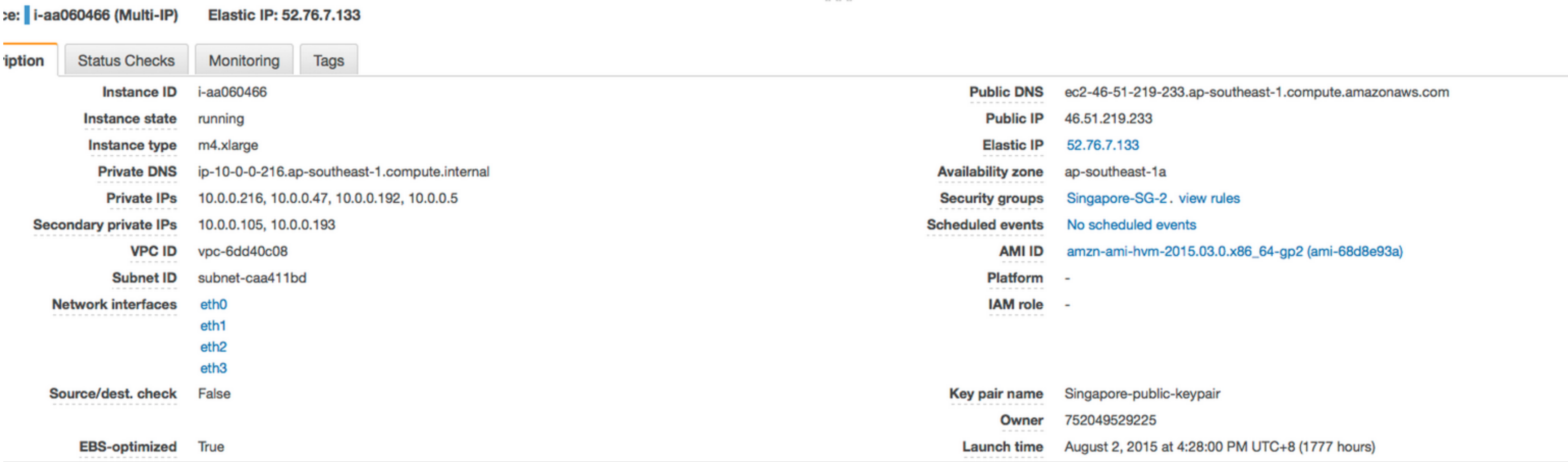
AWS上的实例



e: i-aa060466 (Multi-IP)   Elastic IP: 52.76.7.133

Instance ID	i-aa060466	Public DNS	ec2-46-51-219-233.ap-southeast-1.compute.amazonaws.com
Instance state	running	Public IP	46.51.219.233
Instance type	m4.xlarge	Elastic IP	52.76.7.133
Private DNS	ip-10-0-0-216.ap-southeast-1.compute.internal	Availability zone	ap-southeast-1a
Private IPs	10.0.0.216, 10.0.0.47, 10.0.0.192, 10.0.0.5	Security groups	Singapore-SG-2 . view rules
Secondary private IPs	10.0.0.105, 10.0.0.193	Scheduled events	No scheduled events
VPC ID	vpc-6dd40c08	AMI ID	amzn-ami-hvm-2015.03.0.x86_64-gp2 (ami-68d8e93a)
Subnet ID	subnet-caa411bd	Platform	-
Network interfaces	eth0 eth1 eth2 eth3	IAM role	-
Source/dest. check	False	Key pair name	Singapore-public-keypair
		Owner	752049529225
EBS-optimized	True	Launch time	August 2, 2015 at 4:28:00 PM UTC+8 (1777 hours)

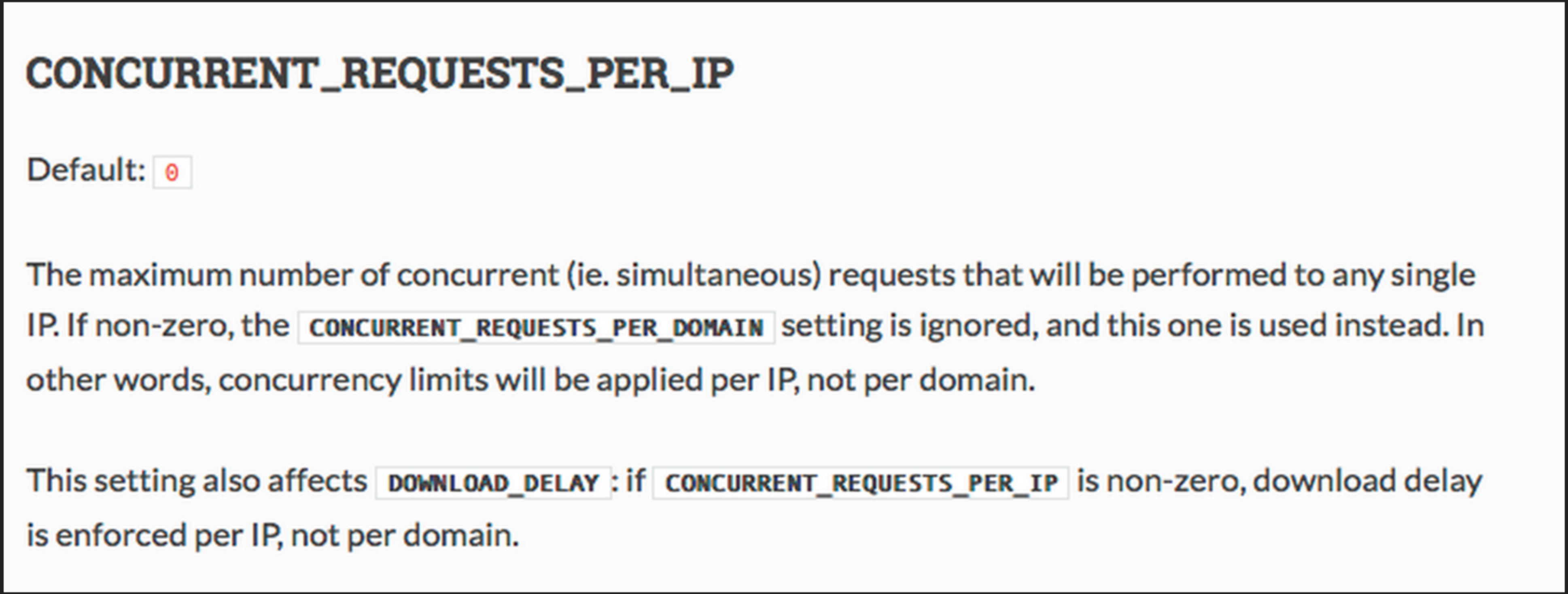
AWS上的实例



ce: i-aa060466 (Multi-IP) Elastic IP: 52.76.7.133

Instance ID	i-aa060466	Public DNS	ec2-46-51-219-233.ap-southeast-1.compute.amazonaws.com
Instance state	running	Public IP	46.51.219.233
Instance type	m4.xlarge	Elastic IP	52.76.7.133
Private DNS	ip-10-0-0-216.ap-southeast-1.compute.internal	Availability zone	ap-southeast-1a
Private IPs	10.0.0.216, 10.0.0.47, 10.0.0.192, 10.0.0.5	Security groups	Singapore-SG-2 . view rules
Secondary private IPs	10.0.0.105, 10.0.0.193	Scheduled events	No scheduled events
VPC ID	vpc-6dd40c08	AMI ID	amzn-ami-hvm-2015.03.0.x86_64-gp2 (ami-68d8e93a)
Subnet ID	subnet-caa411bd	Platform	-
Network interfaces	eth0 eth1 eth2 eth3	IAM role	-
Source/dest. check	False	Key pair name	Singapore-public-keypair
		Owner	752049529225
EBS-optimized	True	Launch time	August 2, 2015 at 4:28:00 PM UTC+8 (1777 hours)

Scrapy 上的多IP 配置



CONCURRENT_REQUESTS_PER_IP

Default: `0`

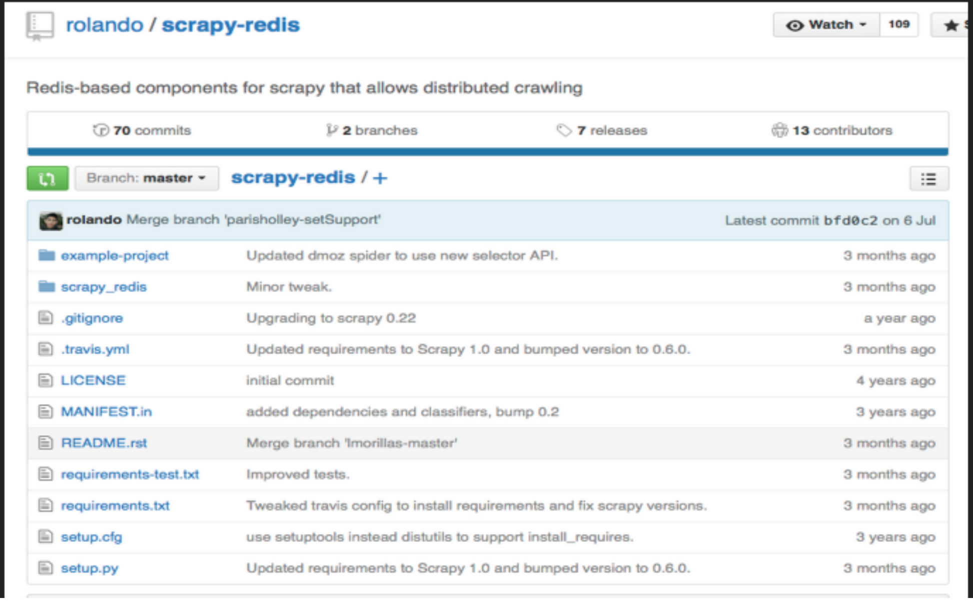
The maximum number of concurrent (ie. simultaneous) requests that will be performed to any single IP. If non-zero, the `CONCURRENT_REQUESTS_PER_DOMAIN` setting is ignored, and this one is used instead. In other words, concurrency limits will be applied per IP, not per domain.

This setting also affects `DOWNLOAD_DELAY`: if `CONCURRENT_REQUESTS_PER_IP` is non-zero, download delay is enforced per IP, not per domain.

Scrapy 多IP 网络性能的进一步优化

基于Redis 的分布式Scrapy 框架

- 分布式Spider集群
- 分布式的缓存队列
- Item Pipeline
- Duplication Filter



rolando / scrapy-redis

Redis-based components for scrapy that allows distributed crawling

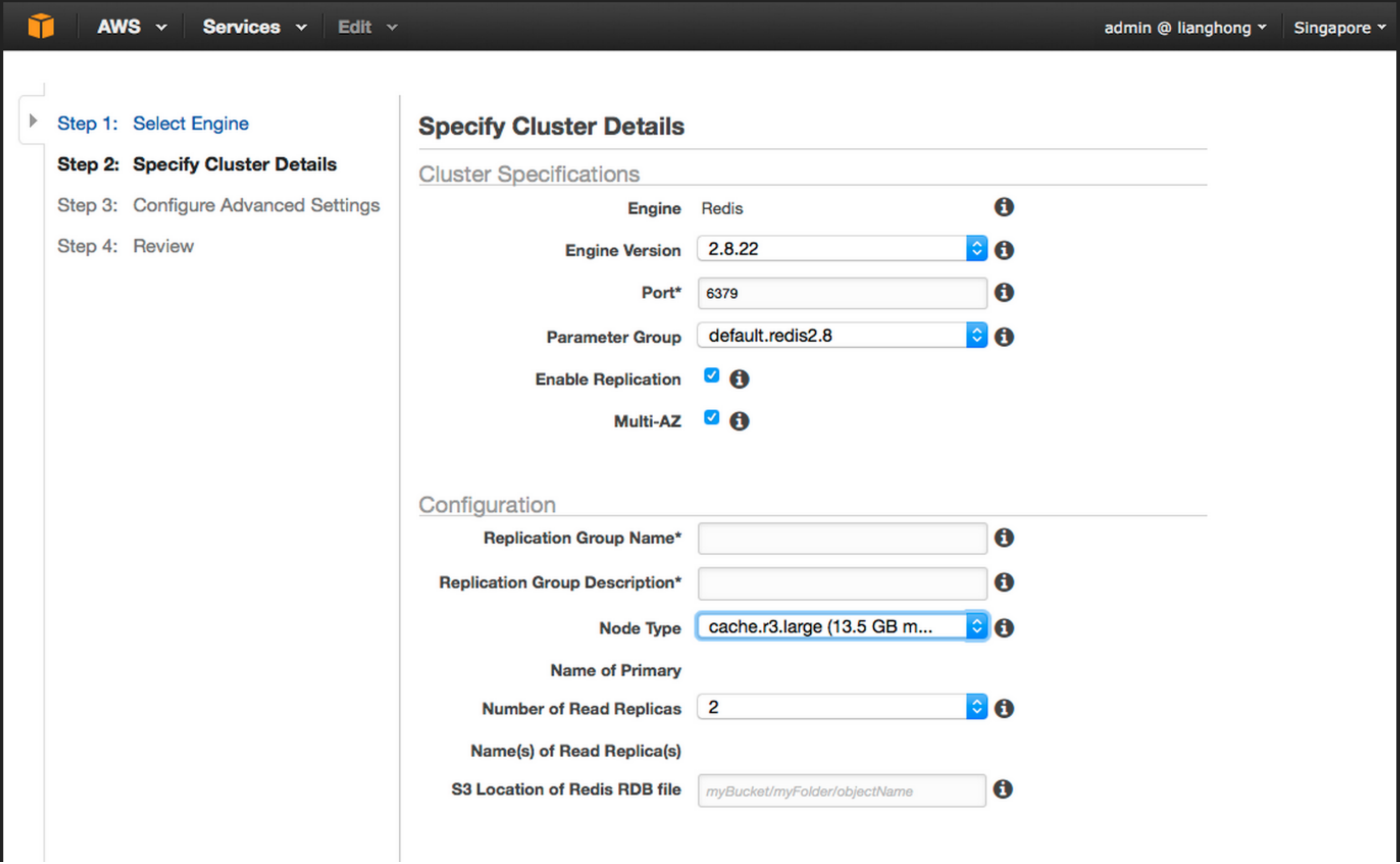
70 commits 2 branches 7 releases 13 contributors

Branch: master scrapy-redis / +

rolando Merge branch 'parisholley-setSupport' Latest commit bfd0c2 on 6 Jul

example-project	Updated dmoz spider to use new selector API.	3 months ago
scrapy_redis	Minor tweak.	3 months ago
.gitignore	Upgrading to scrapy 0.22	a year ago
.travis.yml	Updated requirements to Scrapy 1.0 and bumped version to 0.6.0.	3 months ago
LICENSE	initial commit	4 years ago
MANIFEST.in	added dependencies and classifiers, bump 0.2	3 years ago
README.rst	Merge branch 'lmorillas-master'	3 months ago
requirements-test.txt	Improved tests.	3 months ago
requirements.txt	Tweaked travis config to install requirements and fix scrapy versions.	3 months ago
setup.cfg	use setuptools instead distutils to support install_requires.	3 years ago
setup.py	Updated requirements to Scrapy 1.0 and bumped version to 0.6.0.	3 months ago

AWS 提供的Redis －Elastics Cache



Step 1: Select Engine

Step 2: Specify Cluster Details

Step 3: Configure Advanced Settings

Step 4: Review

Specify Cluster Details

Cluster Specifications

Engine	Redis
Engine Version	2.8.22
Port*	6379
Parameter Group	default.redis2.8
Enable Replication	☑
Multi-AZ	☑

Configuration

Replication Group Name*	
Replication Group Description*	
Node Type	cache.r3.large (13.5 GB m...
Name of Primary	
Number of Read Replicas	2
Name(s) of Read Replica(s)	
S3 Location of Redis RDB file	myBucket/myFolder/objectName

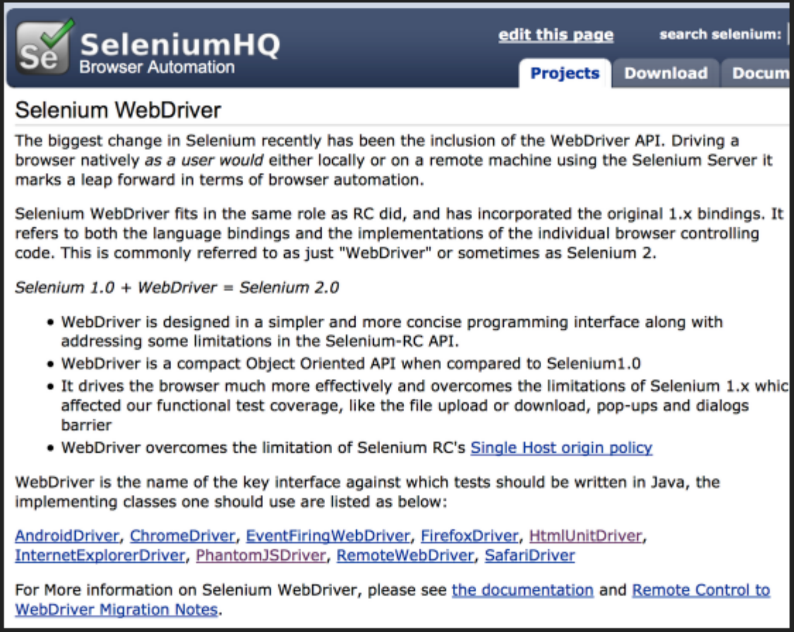
总结：我的分布式爬虫的方案

- 运行环境 AWS EC2
- 多IP环境 AWS Elastics IP
- 爬虫框架 Scrapy
- 开发语言 Python 2.7
- 数据队列 AWS Elactic Cache
- 还有什么需要进一步改进?

总结：我的分布式爬虫的方案

- 运行环境 AWS EC2
- 多IP环境 AWS Elastics IP
- 爬虫框架 Scrapy
- 开发语言 Python 2.7
- 数据队列 AWS Elactic Cache
- 还有什么需要进一步改进?

改进之一：支持Ajax/Javascript



Selenium WebDriver

The biggest change in Selenium recently has been the inclusion of the WebDriver API. Driving a browser natively *as a user would* either locally or on a remote machine using the Selenium Server it marks a leap forward in terms of browser automation.

Selenium WebDriver fits in the same role as RC did, and has incorporated the original 1.x bindings. It refers to both the language bindings and the implementations of the individual browser controlling code. This is commonly referred to as just "WebDriver" or sometimes as Selenium 2.

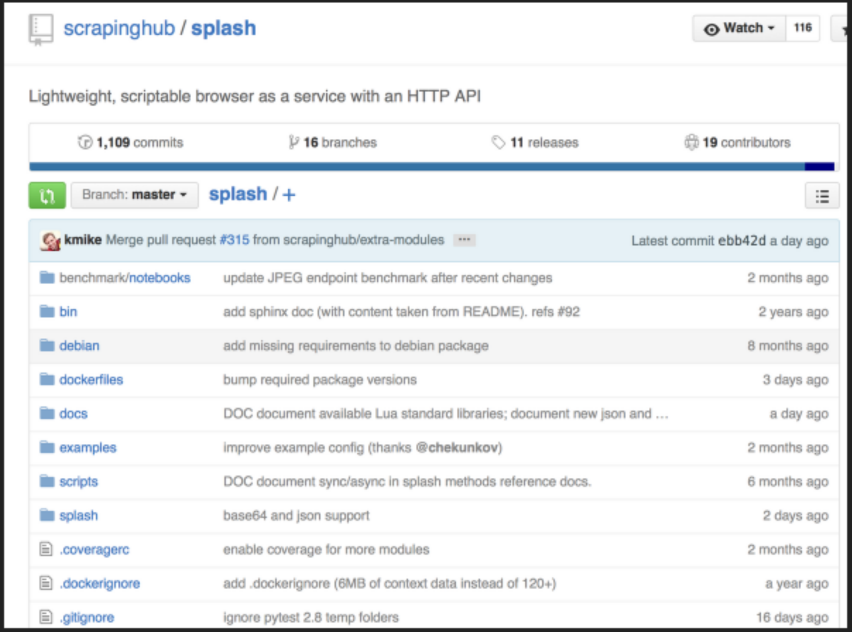
Selenium 1.0 + WebDriver = Selenium 2.0

- WebDriver is designed in a simpler and more concise programming interface along with addressing some limitations in the Selenium-RC API.
- WebDriver is a compact Object Oriented API when compared to Selenium1.0
- It drives the browser much more effectively and overcomes the limitations of Selenium 1.x whic affected our functional test coverage, like the file upload or download, pop-ups and dialogs barrier
- WebDriver overcomes the limitation of Selenium RC's Single Host origin policy

WebDriver is the name of the key interface against which tests should be written in Java, the implementing classes one should use are listed as below:

AndroidDriver, ChromeDriver, EventFiringWebDriver, FirefoxDriver, HtmlUnitDriver, InternetExplorerDriver, PhantomJSDriver, RemoteWebDriver, SafariDriver

For More information on Selenium WebDriver, please see the documentation and Remote Control to WebDriver Migration Notes.



Lightweight, scriptable browser as a service with an HTTP API

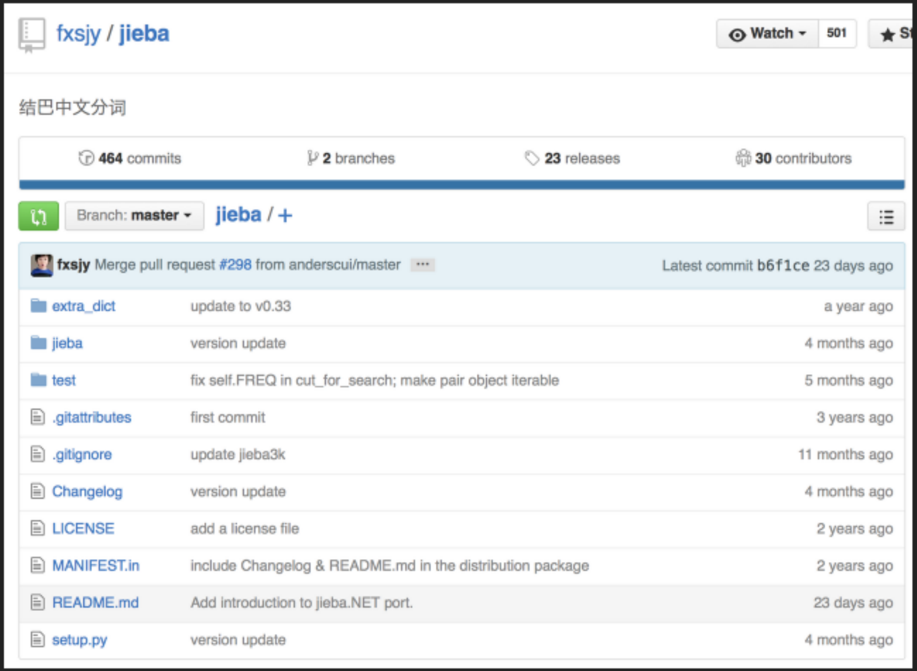
1,109 commits · 16 branches · 11 releases · 19 contributors

Branch: master · splash / +

kmike Merge pull request #315 from scrapinghub/extra-modules ··· Latest commit ebb42d a day ago

benchmark/notebooks	update JPEG endpoint benchmark after recent changes	2 months ago
bin	add sphinx doc (with content taken from README). refs #92	2 years ago
debian	add missing requirements to debian package	8 months ago
dockerfiles	bump required package versions	3 days ago
docs	DOC document available Lua standard libraries; document new json and …	a day ago
examples	improve example config (thanks **@chekunkov**)	2 months ago
scripts	DOC document sync/async in splash methods reference docs.	6 months ago
splash	base64 and json support	2 days ago
.coveragerc	enable coverage for more modules	2 months ago
.dockerignore	add .dockerignore (6MB of context data instead of 120+)	a year ago
.gitignore	ignore pytest 2.8 temp folders	16 days ago

改进之二：中文分词



fxsjy / jieba

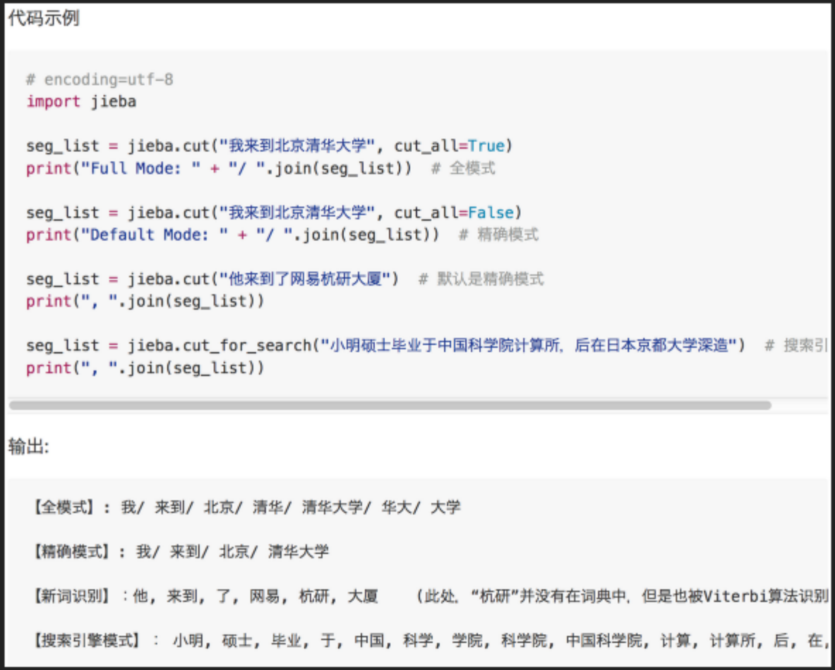
结巴中文分词

464 commits · 2 branches · 23 releases · 30 contributors

Branch: master ▾ jieba / +

fxsjy Merge pull request #298 from anderscui/master … Latest commit b6f1ce 23 days ago

extra_dict	update to v0.33	a year ago
jieba	version update	4 months ago
test	fix self.FREQ in cut_for_search; make pair object iterable	5 months ago
.gitattributes	first commit	3 years ago
.gitignore	update jieba3k	11 months ago
Changelog	version update	4 months ago
LICENSE	add a license file	2 years ago
MANIFEST.in	include Changelog & README.md in the distribution package	2 years ago
README.md	Add introduction to jieba.NET port.	23 days ago
setup.py	version update	4 months ago



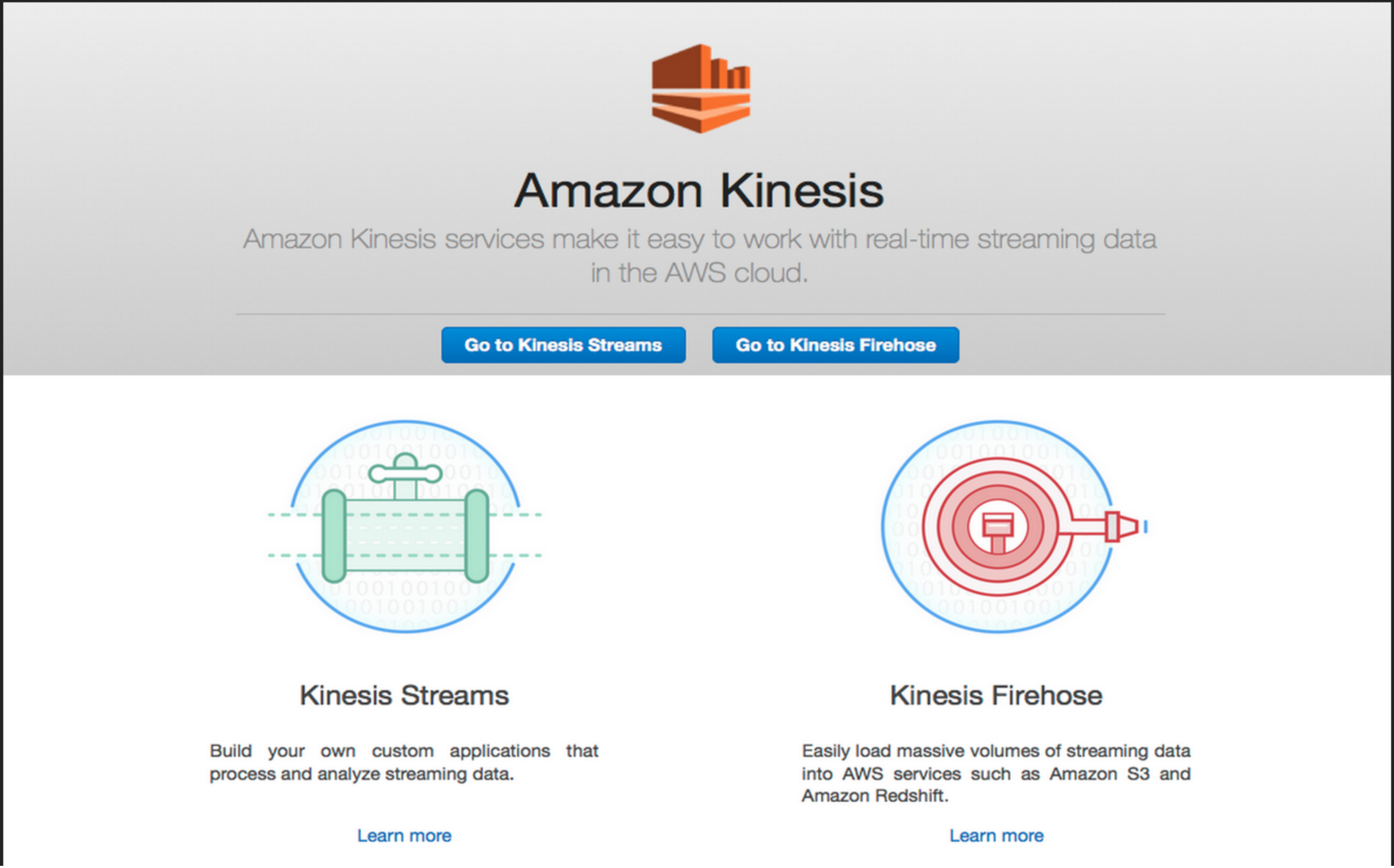
代码示例

```
# encoding=utf-8
import jieba

seg_list = jieba.cut("我来到北京清华大学", cut_all=True)
print("Full Mode: " + "/ ".join(seg_list))  # 全模式

seg_list = jieba.cut("我来到北京清华大学", cut_all=False)
print("Default Mode: " + "/ ".join(seg_list))  # 精确模式

seg_list = jieba.cut("他来到了网易杭研大厦")  # 默认是精确模式
print(", ".join(seg_list))

seg_list = jieba.cut_for_search("小明硕士毕业于中国科学院计算所，后在日本京都大学深造")  # 搜索引
print(", ".join(seg_list))
```

输出:

```
【全模式】: 我/ 来到/ 北京/ 清华/ 清华大学/ 华大/ 大学

【精确模式】: 我/ 来到/ 北京/ 清华大学

【新词识别】：他, 来到, 了, 网易, 杭研, 大厦    (此处，“杭研”并没有在词典中，但是也被Viterbi算法识别

【搜索引擎模式】： 小明, 硕士, 毕业, 于, 中国, 科学, 学院, 科学院, 中国科学院, 计算, 计算所, 后, 在,
```

改进之三：数据流(Strem) 处理



Amazon Kinesis

Amazon Kinesis services make it easy to work with real-time streaming data in the AWS cloud.

Kinesis Streams

Build your own custom applications that process and analyze streaming data.

Kinesis Firehose

Easily load massive volumes of streaming data into AWS services such as Amazon S3 and Amazon Redshift.

你们听烦了，我也讲累了，那就到这里吧 :)

谢谢！

- Powered by AWS