

Scrapyd 源码目录

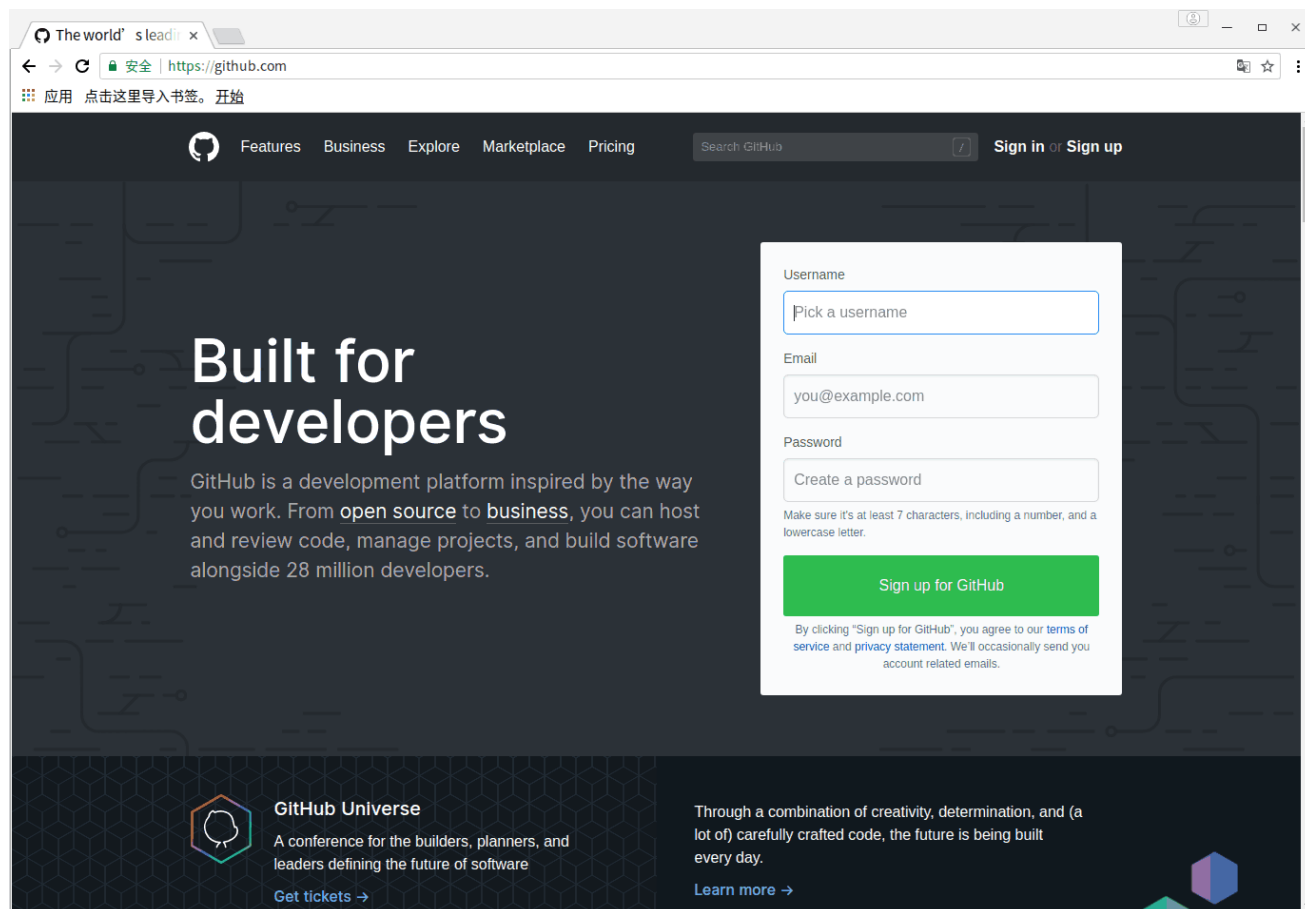
前面学习了 Scrapyd 的简介、文档及安装使用，基础部分已经学习完毕，或许你还有许多的疑问：

- API 的功能是如何实现的？
- Jobs 页面的爬虫运行状态是怎么统计出来的？
- 爬虫日志是如何生成的？
- Scrapyd 配置文件在哪里？
- Scrapyd 源码目录结构是什么样的？
-




















带着这些疑问，随我到源码中探个究竟。

Scrapyd 的源码

[Scrapyd 的源码](#)可以在 GitHub 上找到：



它的文件目录如下图所示：

 jdanford and Digenis Use SVG version of Travis CI status icon in `README.rst`		Latest commit 1713747 on 5 Jan
 bin	include twisted's license along with bin/trial	2 years ago
 debian	Add back default file sourcing	4 years ago
 docs	Merge branch 'list_all_jobs'	11 months ago
 extras	updating integration test script	3 years ago
 reqs	Require at least Scrapy 1.1	2 years ago
 scrapyd	Merge branch 'list_all_jobs'	11 months ago
 .bumpversion.cfg	Bump 1.2.0a1	2 years ago
 .coveragerc	automate differences building packages from master and feature branches	6 years ago
 .gitignore	gitignore cleanup	5 years ago
 .travis.yml	fix travis error in six-1.11 installation in py34	11 months ago
 LICENSE	add docs and debian packaging	6 years ago
 MANIFEST.in	Fixed manifest to include scripts	4 years ago
 Makefile.buildbot	fix bug in Makefile.buildbot and more simplifications to how packages...	5 years ago
 NEWS	added release notes to docs	6 years ago
 README.rst	Use SVG version of Travis CI status icon in `README.rst`	9 months ago
 setup.cfg	Build universal wheels	2 years ago
 setup.py	virt. package enum-compat fixes cond. req for py33	11 months ago
 tox.ini	Copy Twisted 16.3's bin/trial and run with it	2 years ago

从 GitHub 的代码提交记录来看，最近的代码更新是在 11 个月以前，并且几乎每年都会有更新。代表这个项目是持续更新维护的，不用担心使用过后官方团队停止维护导致的项目管理隐患。

实际上我们安装 Scrapy 之后，用到的代码是在上图 `scrapyd` 目录中，安装时也相当于将 Scrapy 目录复制到 `site-packages` 内，至于其他的目录和文件，是在 PyPI 打包提交以及编写文档时所生成的文件。

启动文件、视图及 API

进入 Scrapy 目录，下图为 Scrapy 目录下的所有文件：

Digenis Merge branch 'list_all_jobs'		Latest commit 99a88ea on 20 Nov 2017
..		
scripts	Change script to allow run scrapyd as stand-alone Python application.	3 years ago
tests	drop tests for plaintext & pickle spider queue	a year ago
VERSION	Bump 1.2.0a1 → 1.2.0	2 years ago
init.py	Fix version reading	2 years ago
_deprecate.py	a deprecation cycle is at least a release's length	2 years ago
app.py	bind address default to 127.0.0.1	2 years ago
config.py	PY3 BytesIO/StringIO continued	2 years ago
default_scrapy.conf	bind address default to 127.0.0.1	2 years ago
eggstorage.py	PY3 Migrate zope.interface implements to implementer	2 years ago
eggutils.py	PY3 Fix decoding of spider list	2 years ago
environ.py	Merge pull request #179 from scrapy/python3-wip	2 years ago
interfaces.py	scrapy: added new cancel.json api to cancel pending/running jobs	7 years ago
launcher.py	Process API requests args as native strings	2 years ago
poller.py	fix poller race condition	2 years ago
runner.py	adding version support using a -d _version=145446.. argument	3 years ago
scheduler.py	PY3 Migrate zope.interface implements to implementer	2 years ago
script.py	scrapy command line runs with or without project depending on cwd	6 years ago
spiderqueue.py	PY3 Migrate zope.interface implements to implementer	2 years ago
sqlite.py	drop deprecated sqllitedict and sqlitedictpriorityqueue	a year ago
txapp.py	add scrapyd script, a twisted light wrapper for controlling scrapyd. so...	5 years ago
utils.py	Don't mask empty output from `scrapy list`	2 years ago
webservice.py	project arg in listjobs ws optional: list em all	11 months ago
website.py	rewrite jobs website	11 months ago

这个小节我们主要了解一下各个文件或目录的大致作用，知晓目录的结构以及文件分布。

至于启动文件、视图、API 文件的源码调试与解析会在后面的小节中重点讲解，以下列出每个文件或目录的作用释义：

- 1 * scripts - 里面只有 1个可用的文件：scrapyd_run.py，
- 2 它是整个项目的启动文件。
- 3 * init.py - Scrapyd 启动前的配置。
- 4 * tests - 用于测试 Scrapyd 功能的代码集。
- 5 * VERSION - Scrapyd 版本号文件。
- 6 * app.py - Scrapyd 应用配置文件读取并进行设置。
- 7 * config.py - Scrapyd 配置及相关设置。
- 8 * default_scrapy.conf - Scrapyd 配置文件
- 9 * eggstorage.py - 项目打包设置及版本生成。
- 10 * eggutils.py - 项目打包。
- 11 * environ.py - 项目打包目录读取
- 12 * interfaces.py - 处理和存储项目包的检索以及爬虫队列。
- 13 * launcher.py - 执行爬虫运行、记录状态以及进程池相关。
- 14 * poller.py - 队列相关。
- 15 * runner.py - 任务具体执行。
- 16 * scheduler.py - 任务及项目的状态与记录更新。
- 17 * script.py - 用于执行 Scrapy 传递的命令。

```
18 * spiderqueue.py - 爬虫队列相关。
19 * sqllite.py - sqllite 数据库相关。
20 * txapp.py - 通过 Twisted 启动 Scrapy。
21 * utils.py - 爬虫队列、项目列表 i 以及 JSON 视图。
22 * webservice.py - JSON 视图以及 Scrapy 中提供的所有 API。
23 * website.py - Web 视图、Home、Jobs 等页面及功能。
24
```

启动文件

启动文件是整个项目的入口。用于启动项目的文件位于 `/scrapy/script` 目录下，文件名为 `scrapy_run.py`，它通过载入指定 `txapp.py` 文件来启动 `scrapy` 项目，`scrapy_run.py` 代码如下：

```
1 from twisted.scripts.twistd import run
2 from os.path import join, dirname
3 from sys import argv
4 import scrapy
5
6 def main():
7     argv[1:1] = ['-n', '-y', join(dirname(scrapy.__file__), 'txapp.py')]
8     run()
9
```

`txapp.py` 则通过调用 `get_application` 来获取 Scrapy 配置并通过 `RUN` 命令启动 Scrapy，负责初始化 Scrapy 的 `txapp.py` 代码如下：

```
1 from scrapy import get_application
2 application = get_application()
3
```

关于文件源码的深层次剖析与调试，会在后面的章节《[动手调试 Scrapy 代码](#)》，通过代码调试的方式理解 Scrapy 的代码运行流程与逻辑。