## Integrating Selenium with Scrapy

Asked 9 years, 8 months ago Modified 9 years, 8 months ago Viewed 7k times



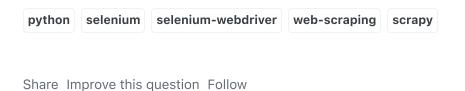
Is there any way to effectively integrate Selenium into Scrapy for it's page rendering capabilities (in order to generate screenshots)?





A lot of solutions I've seen just throw a Scrapy request/response URL at WebDriver after Scrapy's already processed the request, and then just works off that. This creates twice as many requests, fails in many ways (sites requiring logins, sites with dynamic or pseudo-random content, etc.), and invalidates many extensions/middleware.

Is there any "good" way of getting the two to work together? Is there a better way for generating screenshots of the content I'm scraping?





edited Jul 13, 2015 at 18:35

asked Jul 13, 2015 at 18:16

alecxe 474k 1:

**474k** 127 1.1k 1.2l

Rejected 4,501 2 27 4

## 1 Answer



Sorted by: Highest score (default) \$



Use Scrapy's <u>Downloader Middleware</u>. See my answer on another question for a simple example: <u>https://stackoverflow.com/a/31186730/639806</u>





Share Improve this answer Follow



answered Jul 14, 2015 at 13:58

JoeLinux 4.307 1 31 31



I've looked at this, and while it does fix one of the issues (doubling up on requests), it bypasses many features Scrapy provides. It discard user-agent configuration, proxy configurations, headers, and offers zero persistence between calls (no sessions/cookies). Furthermore, it's impossible to submit POST requests in Selenium, so things like FormRequests will break or have very unexpected results. – Rejected Jul 14, 2015 at 15:44

It does bypass those things. It's a very simple example, but a lot of those things can be duplicated in Selenium (such as cookies, headers and user-agent string). In fact, most of that info you can pull using the request information that's available as an arg to the process\_request method. Also, you won't need to POST through Selenium. No reason you can't do that through Scrapy in parse after pulling the Selenium response. – JoeLinux Jul 14, 2015 at 15:49

Wouldn't the FormRequest be 'hijacked' by the Selenium Downloader Middleware as it passed through, and then processed as a driver.get(url)" by Selenium? How could this be prevented? – Rejected Jul 14, 2015 at 16:02

Use a conditional (e.g., if should\_process\_js(request): ), and just return return request to continue processing normally if whatever conditions are false (such as the request being a POST, or whatever you decide). − JoeLinux Jul 14, 2015 at 16:03 ✓

I've worked on this and found other issues, that I was curious if you had any thoughts on. Returning an HtmlResponse doesn't fire off the response\_downloaded signal, and anything relying on it breaks (such as throttling). CustomHeaders, most importantly "Referer" cannot be manually set on WebDriver. – Rejected Jul 15, 2015 at 17:11

## Start asking to get answers

Find the answer to your question by asking.

Ask question

## **Explore related questions**

python selenium selenium-webdriver web-scraping scrapy

See similar questions with these tags.