Web Scraping 101 with F.O.S. Goutte

By Joshua Copeland

About Me

Josh Copeland

@OGProgrammer

- CTO of Engaged Nation
- PHP Developer for 6+ years
- Java, .NET, and C/++ exp.
- Serial Entrepreneur
- Prior Real Estate Agent
- ♥'s Family, Tech, & Skating
- Self Proclaimed Computer

What is Web Scraping?

Web scraping is the process of automatically collecting information from the web.

Requires breakthroughs in text processing, semantic understanding, artificial intelligence and human-computer interactions.

Current web scraping solutions range from the ad-hoc, requiring human effort, to fully automated systems that are able to convert entire sites into structured information, with limitations.

Traditional Methods

In 2009 there was no “all-in-one” library with both an HTTP Client & a HTML Parser. These where your choices in PHP back then:

- ➔ **Tidy Extension**
  Wasn't designed for extraction, only HTML error fixing.
- ➔ **DOM**
- ➔ **SimpleXML**
- ➔ **XMLReader**
- ➔ **CSS Selectors**
  Works fine for HTML parsing but isn't a crawler.

Introducing Goutte!

A simple PHP Web Scraper.

[Examples from this presentation available at

https://github.com/php-vegas/web-scraper-examples](https://github.com/php-vegas/web-scraper-examples)

Did you know?

Goutte was built by Fabien Potencier who also built the Symfony Framework.

FriendsOfSymfony is the group that maintains this package and others in the Symfony world.

What does Goutte use?

- Symfony Components
  a. BrowserKit
  b. CssSelector
  c. DomCrawler
- Guzzle HTTP Component.

Did you know?

Fabien Potencier also built these Symfony components.

You should check out his github profile where his username is “fabpot”.

He's kind of a big deal.

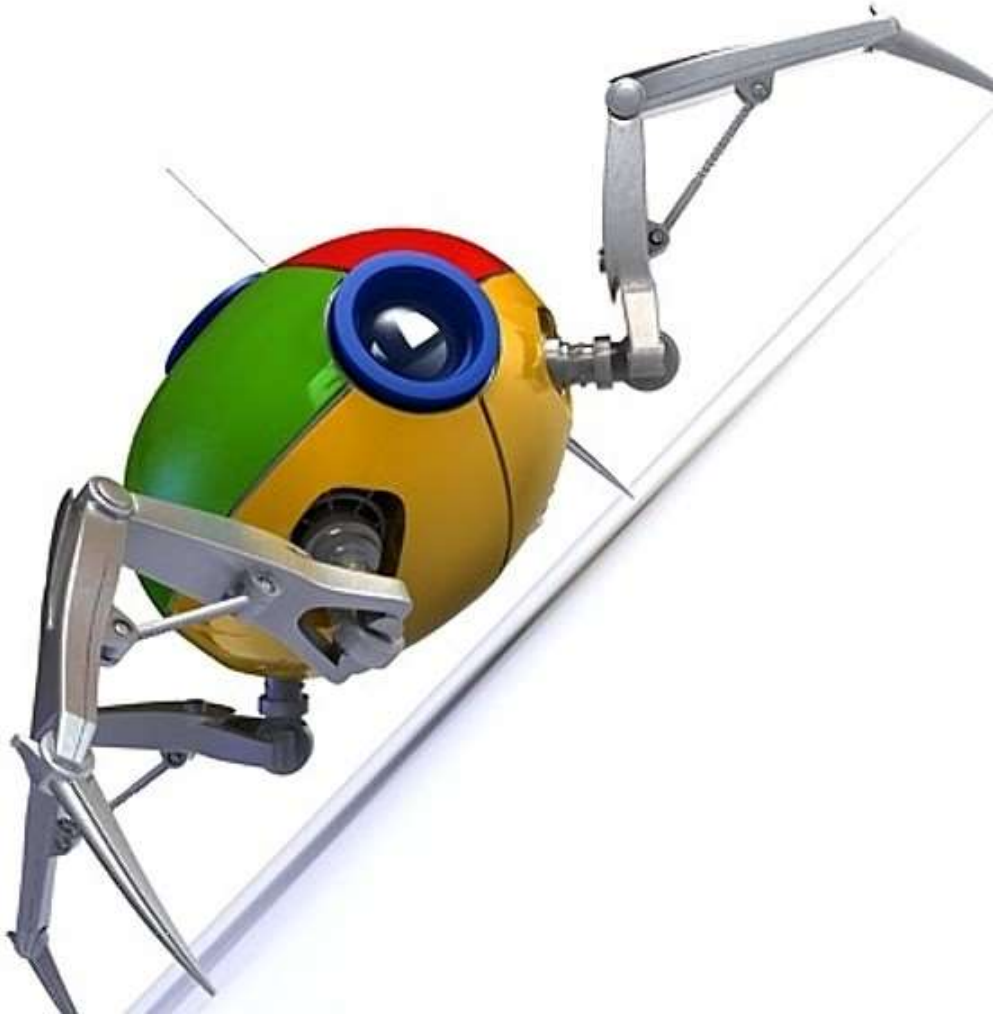
What does Goutte do

- Uses Guzzle (cURL, streams, sockets, or event loops)
  - GET/POST Requests
- Fine tune cURL settings
- Follow links - Crawl the site
- Extract data
  - XPath, CssSelector
- Submit forms
  - Login!

What Goutte doesn't do

- Does not interpret the response in any way.
  - Will not execute JavaScript
    - Which means no AJAX
      - Could simulate the AJAX request
    - Try Google cached versions of the site
    - Use PhantomJS, Spiderling, CasperJS, Selenium
- Can't render or screenshot the page
  - Could save the HTML & assets

Let’s get started!



What you'll need

- **Recommend using Composer**
  Easiest way to install PHP libraries
- **Alternatively could use PHAR**
  Available releases on their GitHub
- **Version 3**
  - PHP 5.5+
  - Guzzle 6+
- **Version 2**
  - PHP 5.4
  - Guzzle 4-5
- **Version 1**
  - PHP 5.3
  - Guzzle 3

Require Goutte in your project

```
composer require fabpot/goutte
```

Basic Example

```php
use Goutte\Client;

$client = new Client();

// Go to the symfony.com website
$crawler = $client->request('GET', 'http://www.symfony.com/blog/');

// Click on the "Security Advisories" link
$link = $crawler->selectLink('Security Advisories')->link();
$crawler = $client->click($link);

// Get the latest post in this category and display the titles
$crawler->filter('h2 > a')->each(function ($node) {
    print $node->text()."\n";
});
```

Guzzle Settings Example

```php
use Goutte\Client;
use GuzzleHttp\Client as GuzzleClient;

// Create the guzzle client with your default options
$guzzle = new GuzzleClient(
    array(
        // base_uri isn't supported due to BrowserKit, anyone want to make a PR on github for this?
        // 'base_uri'        => 'https://www.symfony.com',
        'timeout'         => 0,
        'allow_redirects' => false,
        'cookies'         => true,
        // Proxy from proxylist.hidemyass.com
        'proxy'           => 'tcp://63.150.152.151:3128'
    )
);

$client = new Client();
$client->setClient($guzzle);
```

Check out all the Guzzle options at http://docs.guzzlephp.org/en/latest/request-options.html

Basic HTTP Auth Example

```php
$client = new Client();

// Params are username, password, and auth type (basic & digest)
$client->setAuth('test', 'test', 'basic');

$crawler = $client->request('GET', 'http://browserspy.dk/password-ok.php');

print $client->getResponse()->getStatus();
// 401 = no good, 200 = happy
```

Form Login Example

```
$crawler = $client->request('GET', 'http://github.com/');

$crawler = $client->click($crawler->selectLink('Sign in')->link());

$form = $crawler->selectButton('Sign in')->form();

$crawler = $client->submit($form, array('login' => 'fabpot', 'password' => 'xxxxxx'));

$crawler->filter('.flash-error')->each(function ($node) {
    print $node->text()."\n";
});

// outputs “You can't perform that action at this time.”
```

Getting info from the Response

```php
// Get the URI
print 'Request URI : ' . $crawler->getUri() . PHP_EOL;

// Get the Symfony\Component\BrowserKit\Response object
$response = $client->getResponse();

// Get important stuff out of the Response object
$status = $response->getStatus();
$content = $response->getContent();
$headers = $response->getHeaders();
```

Watch out for...

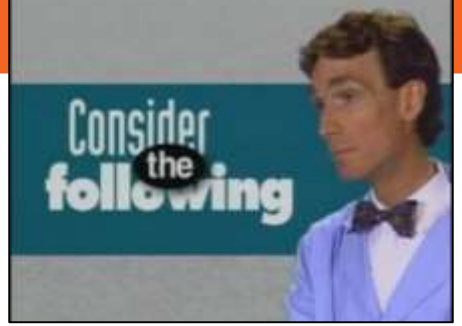
- DDOSing a site, put a sleep(x) between calls
  - Good way to get your IP banned, use a proxy
- Pulling bad/malformed data
  - Write tests to make sure this doesn't happen
- Fetching elements by unique IDs, hashes, etc
  - Get creative, find an RSS feed, API, or structured data
- Protections against scrapping like javascript or AJAX
  - Some buttons have JS events attached

More examples in GitHub

This presentation + dell product info scraper
https://github.com/php-vegas/web-scraper-examples

CSRF Scanner
https://github.com/marlon-be/marlon-csrfscanner

There are extensions for WP, Laravel, mink, & others.
Just search pacakgist.org for “goutte”.



In theory you could use PHPv8 (v8js engine) to execute javascript and create a handler aka middleware in Guzzle.

This would be awesome but there are existing projects out there that already do this. Just that PHP doesn't have a way right now.

What were those projects?

- PhantomJS
- Spiderling
- CasperJS
- Selenium

JavaScript

Questions?

@OGProgrammer

Rate this talk
http://spkr8.com/t/68291