

Crawling the Web with Python and Scrapy

EXTRACTING DATA FROM THE WEB – CORE CONCEPTS



Eduardo Freitas

BUSINESS AUTOMATION & DATA CAPTURE SPECIALIST

<https://edfreitas.me>



Overview



Prerequisites

Fundamental concepts

- Crawling
- Scraping

Most common questions

- Considerations
- Counterarguments
- General advice

Why Scrapy?

Extracting data



Prerequisites



Course Prerequisites



Python programming know-how

Knowledge of HTML, CSS and web pages



Concepts: Crawling & Scraping



Crawling

Is the act of programmatically retrieving or downloading a web page's data, extracting its hyperlinks and following them.



Scraping

Is the act of programmatically retrieving or downloading a web page's data and extracting very specific information from it.



Crawling vs Scraping

Crawling

Retrieve, download and index web sites

Done by search engines, associated with mostly legitimate use

Good reputation

General: crawl pages from a specific site

Large-scale results, such as site indexes

Scraping

Retrieve data directly from web sites

Done for data and market analysis, to gain competitive advantage

Less credible reputation

Specific: Mine data from a specific site

Small-scale datasets



Crawling & Scraping: Legal or Illegal?



Crawling and scraping are not illegal by themselves.

After all, you can scrape or crawl your own website.



Problems May Arise When

Don't Obtain Permission

When crawling and scraping someone else's website without written permission

Disregard Terms of Service

When crawling and scraping someone else's website disregarding their ToS



Actions

Ignore you

Ban or block you

Cease/desist letter



But in reality, there's nothing that prevents them from suing you.



Legal Consequences



Common Suing Reasons



Violation of the Computer Fraud and Abuse Act (CFAA)



Violation of state, federal or a jurisdiction's penal code



Violation of the Digital Millennium Copyright Act (DMCA)



ToS breach of contract



Trespass and misappropriation



Typical Counterarguments



I can do whatever I want with publicly accessible data - **False**



I'm doing it for fair and a good cause – **False**



The browser does the same thing - **False**



I will simply get banned or blocked - **False**



ToS are not enforceable anyway - **False**



Do not assume that just because you're technically capable, that you can crawl and scrape someone else's site.



It is better to obtain written permission
before scraping
a site that is not yours and
adhere to their ToS.



General Advice



Be extra cautious with
web scraping.



General Advice



When an API is provided, use it and don't scrape



ToS – read, understand and respect them



Respect the rules of robots.txt



Use a reasonable crawl rate and don't overload the site with requests



When ToS or robots.txt prevent you from crawling or scraping, ask for written permission, before doing anything else



General Advice



Identify your web scraper or crawler with a valid User-Agent string



Do not publish any scraped data without verifying the license of the data, and obtaining a written permission from the copyright owner



Do not build your entire business around data scraping



Be extra cautious of any advice found on the web



If you are not sure of what you are doing, check with an attorney



When scraping someone else's website,
you might be putting yourself in a
vulnerable position.



Is This Something That Might Upset Someone?



Undercover

Are you willing to take the financial risk, if something were to go wrong?



Legitimate

There are a lot of gray areas around this topic, so it is better to respect the rules





User-Agent String

User-Agent String @ MDN web docs

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/User-Agent>



Why Scrappy?



Web Scraping



Fetch

The diagram consists of two adjacent rectangular boxes. The left box is purple and contains the word 'Fetch'. The right box is teal and contains the word 'Parse'. Both boxes are centered horizontally and vertically on the slide.

Parse



Fetching Content – HTTP Requests

Urllib

Urllib2

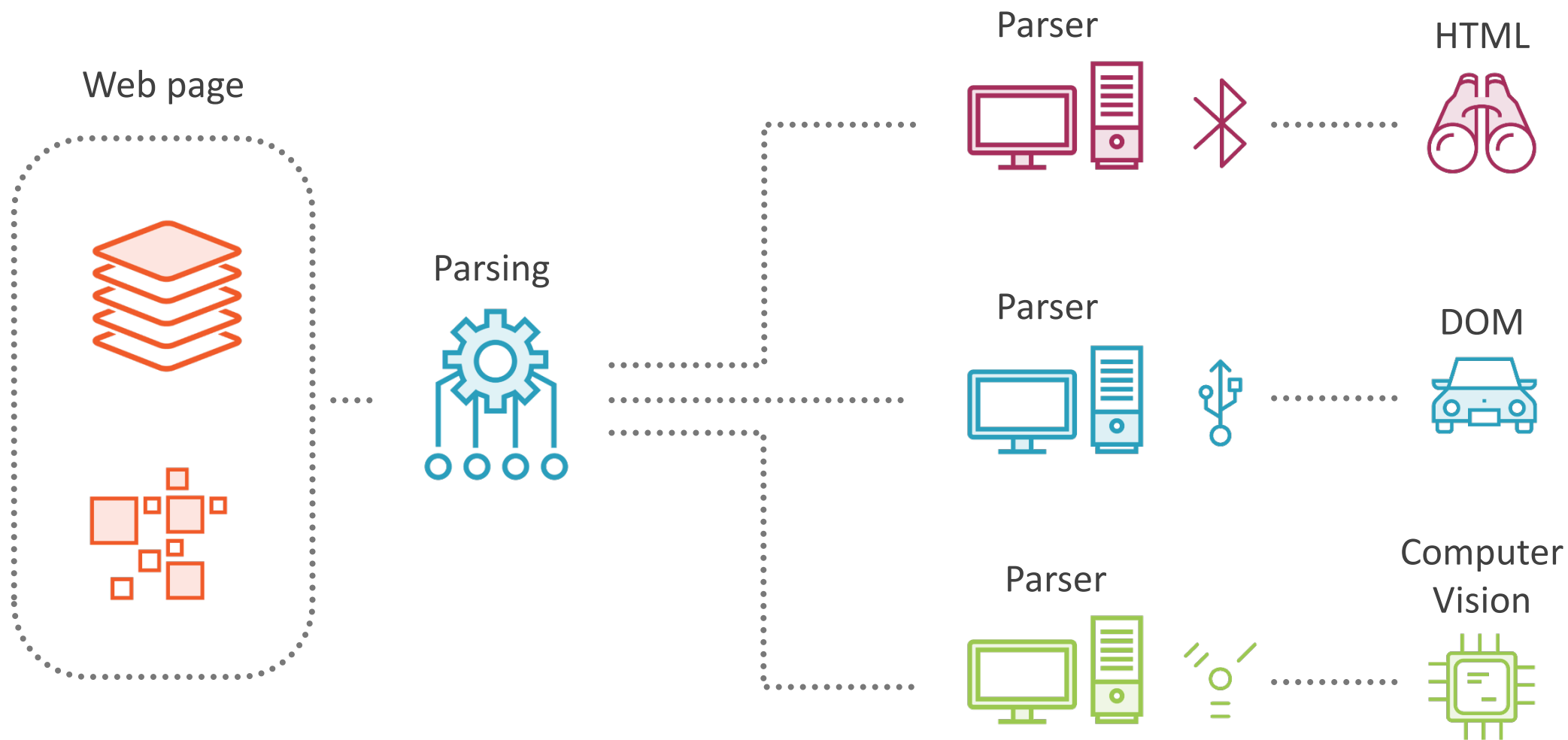
Requests

Httplib

Httplib2



Web Scraping – Parsing Content



Scrapy

Scrapy is an application framework for crawling web sites and extracting structured data which can be used for a wide range of useful applications, like data mining, information processing or historical archival.

<http://docs.scrapy.org/en/latest/intro/overview.html>



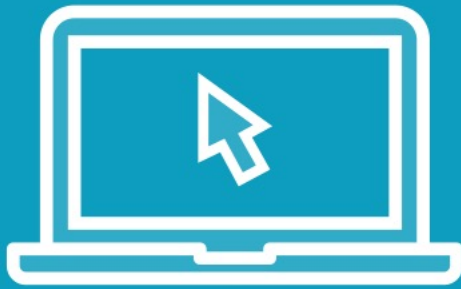
Scrapy Can Also Be Used For

Work with APIs

Generic crawler



Demo



Extracting Data without Scrapy



Summary



Crawling vs scraping

Problems that may arise

Typical counterarguments

Advice in general

HTTP libraries

Types of content to parse

Extracting data without Scrapy