European Public Sector Information Platform

Topic Report No. 2015 / 10

Web Scraping: Applications and Tools

Author: Osmar Castrillo-Fernández

Published: December 2015

Table of Contents

Keywords

web scraping, data extracting, web content extracting, data mining, data harvester, crawler, open government data

Abstract/ Executive Summary

Internet is the vastest information and data source ever built by mankind. However, it is a huge collection of heterogeneous and poorly structured data, difficult to collect in a manual way and complicated to use in automated processes.

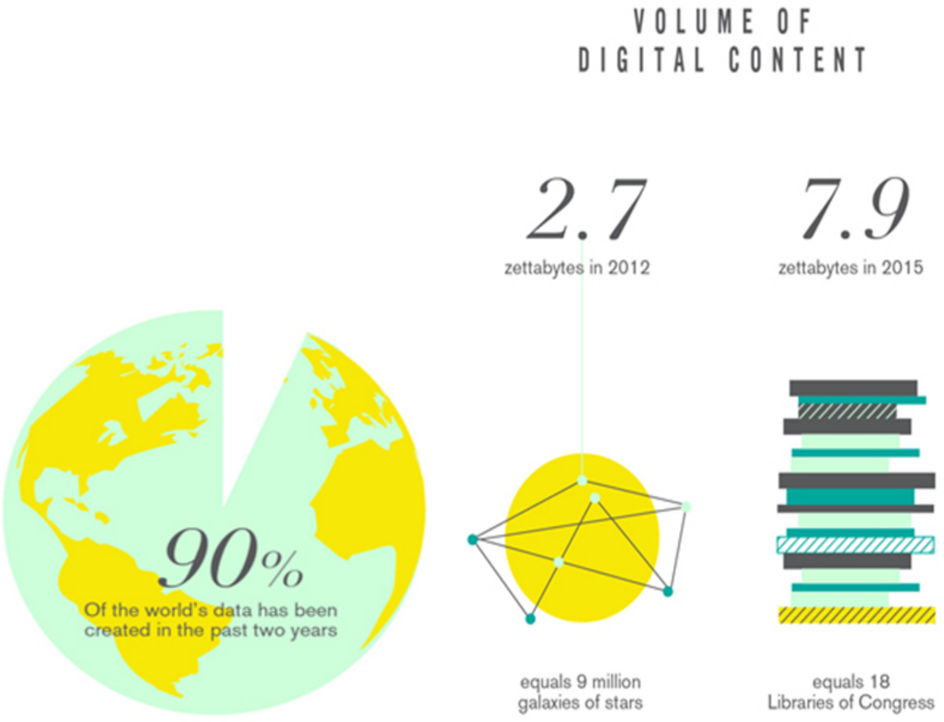
Over the last years, techniques and tools have surged, allowing data collection and conversion to structured data to be managed by B2C and B2B systems. This article offers an introduction to web scraping techniques and some of the most popular and novel techniques for data extraction and reuse in complex processes.

The possibilities to take benefit of such data are many, including areas like Open Government Data, Big Data, Business Intelligence, aggregators and comparators, development of new applications and mashups, among others.

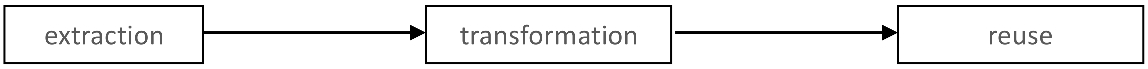
1 Introduction

Every single day, several petabytes of information are published via the Internet in various formats, such as HTML, PDF, CSV or XML. Curiously, HTML is not necessarily the most common format to publish content. For instance, 70% of the content indexed by Google is extracted from PDF documents. This is an additional obstacle for the different roles involved in data extraction from multiple sources. Journalists, researchers, data analysts, sales agents or software developers are some examples of professionals typically using the copy-and-paste technique to get information in a specific format and export it to a spreadsheet, an executive report or to some data exchange format such as XML, JSON or any of the several vocabularies available based on them.

Focusing on data acquisition from HTML (i.e, a web document), this article explains the mechanisms and tools that may help us to minimize the tedious duties of data extraction from the Internet.


VOLUME OF DIGITAL CONTENT
2.7
zettabytes in 2012
7.9
zettabytes in 2015
90%
Of the world's data has been created in the past two years
equals 9 million galaxies of stars
equals 18 Libraries of Congress

With regards to the **commitment of transparency and data openness that public administrations have assumed over the last decade, scraping and crawling are techniques that may be useful**. Whereas current information systems in use by public administrations consider these techniques, a relevant amount of web sites, content management systems and ECMs (Enterprise Content Management) do not. Exchanging these systems for new ones imply considerable economic efforts. Scraping tools provide an alternative to exchanging which minimizes such efforts. Open Government Data and Transparency Policies should take advantage of this opportunity.



2 What is web scraping?

One of the many definitions of this concept, and the favourite one for the author of this document, is:

> *A web scraping tool is a technology solution to extract data from web sites, in a quick, efficient and automated manner, offering data in a more structured and easier to use format, either for B2B or for B2C processes.*

Scraping processes may be written in different programming languages. The most popular are Java, Python, Ruby or Node. As it is obvious, expert programmers are required to develop and evolve them, and even to use them. Nonetheless, some software companies have designed different tools that enable other people to use scraping techniques by means of attractive and powerful user interfaces.

Web scraping tools are also referred as **Web Data Extractors**, **Data Harvesters**, **Crawling Tools** or **Web Content Mining Tools**.

The necessity to scrape web sites and PDF documents

As already stated, approximately **70% of the information generated in the Internet is available in PDF documents**, an unstructured and hard to handle format. However, a web page has a structured format (HTML code), although in a non-reusable way.

PDF scraping is not the object of the analysis of this article, although it is true that some tools exist to extract information, mainly related to data tables. This enormous amount of information published but captive of this kind of format is usually called "**the tyranny of PDF**". Some tools that are presented in later sections of this document can read PDF documents and return information in a structured format, although in a basic and rudimentary way.

Following with the main scope of this document (HTML documents), its structured nature multiplies the possibilities open by scraping techniques. **Web scraping techniques and scraping tools rely in the structure and properties of the HTML language**. This facilitates the task of scraping for tools and robots, stopping humans from the boring repetitive and error prone duty of manual data retrieval. Finally, these tools offer data in friendly formats for later processing and integration: JSON, XML, CSV, XLS o RSS.

The API mana

Data collection in a handcrafted way is truly inefficient: search, copy and paste data in a spreadsheet to later process them. This is a tedious, annoying and tiresome process. Therefore, it makes much more sense to automate this process. Scraping allows this kind of automation, as the majority of the available tools provide an API to facilitate access to the content generated. An API (Application Programming Interface) is a mechanism to connect two applications, allowing them to share data. Scraping tools facilitate a URL, an Internet address as those you may notice in the address bar of a web browser, giving access to the data scraped. In some cases, APIs are not only limited to a URL. They can also be programmed in any way to modify the final result of the scraping process. This feature is absolutely useful for B2B integration processes, enabling subsequent applications and services based on such data.

Therefore, it is recommended that a good scraping tool provides a simple and programmable API. The concept of API is very relevant in this topic. Occasionally, this term is also known as End Point.

3 Web scraping tools

Once that the basic concepts related to scraping have been commented, this documents focuses on tools currently available in the market. Basic information about their functionality and how they work is provided, avoiding deep technical details as much as possible.

Firstly, we suggest an initial tool segmentation:

1. Partial tools. These are typically **plugins** to third-party software. They usually **focus on a specific scraping technique** (for instance, HTML tables). They do not provide an API for B2B integration.
2. Complete tools. This segment includes the latest tools in the market and some older tools, created as a **general scraping service**. They offer features such as powerful graphical user interface, visual scraping utility, SaaS and/or desktop licensing models, query caching and recording, APIs or reporting and audit dashboards.

Partial tools for PDF extraction

This segment includes these tools oriented to read PDF documents and extract all or part of the information contained.

Within this type may include those oriented exclusively to open PDFs and extract all or part of its information. The figure below provides a brief description of some of these tools.

	PDFtoExcelOnLine	Zamzar	CometDocs	PDFTables	Tabula
	www.pdftoexcelonline.com	www.zamzar.com	www.cometdocs.com	pdftables.com	tabula.technology
	Works online. Allows several input and output formats, including Word, Excel, PowerPoint, PDF). The resulting file is sent via email. Free to use.	Works online. Large amount of input and output formats. The resulting file is sent via email. Free to use.	Works online. Converts PDF to Word, Excel and PowerPoint. Cloud-based document storage. Application versions for Desktop, Smartphone and Tablet. Free account with limited features. API for document translation.	Works online. Only converts existing tables in a PDF document to Excel. API available. Free account (very limited) plus paid accounts with various licensing terms.	Desktop tool. Focused on extraction of tables in PDF documents. Source code available. Oriented to software developers.
On-line	X	X	X	X	
Off-line					X
SaaS Free Service	X	X	X	X	
SaaS Paid Service			X	X	
Document formats	word, excel, power point, pdf	many	Converts PDF to Word, Excel and power point	Converts existing tables into PDF documents to Excel format	Only extracts data from tables into PDF documents
API			X	X	
Oriented to developers					X
Source code available					X

Partial tools to extract from HTML

Regarding techniques and tools for web content (HTML documents) partial scraping, some examples of tools are commented below.

Google Spreadsheets and the IMPORTHTML formula

This is a simple solution, but sufficiently effective to extract data from an HTML table to a Google Spreadsheets document. The actual format of the formula is:

=IMPORTHTML(“URL”;”table”;N)

Testing the IMPORTHTML formula is fairly simple, as long as value N is known. This value represents the order of the table in the list of tables available in the HTML code available in address URL. An example of use based in the format is:

```
=IMPORTHTML("http://www.euskalmet.euskadi.net/s07-
5853x/es/meteorologia/app/predmun_o.apl?muni=";"table";1)
```

Table Capture, a Google Chrome extension

Table Capture is an extension to the popular Google Chrome web browser. It enables users to copy the content of tables included in a web page in an easy manner. The extension is available from the aforementioned browser, typing the following URL in its address bar:

https://chrome.google.com/webstore/detail/table-capture/iebpjdmgckacbodjpijphcplhebcmeop

Once installed, if Chrome detects tables in the web page currently rendered, a red icon is shown to the right of the address bar. Clicking on it shows a listing of the tabled detected and two controls. The first one copies content to the clipboard, and the second one open a Google Docs document to later paste the content of the table. An example is shown in the following figure.

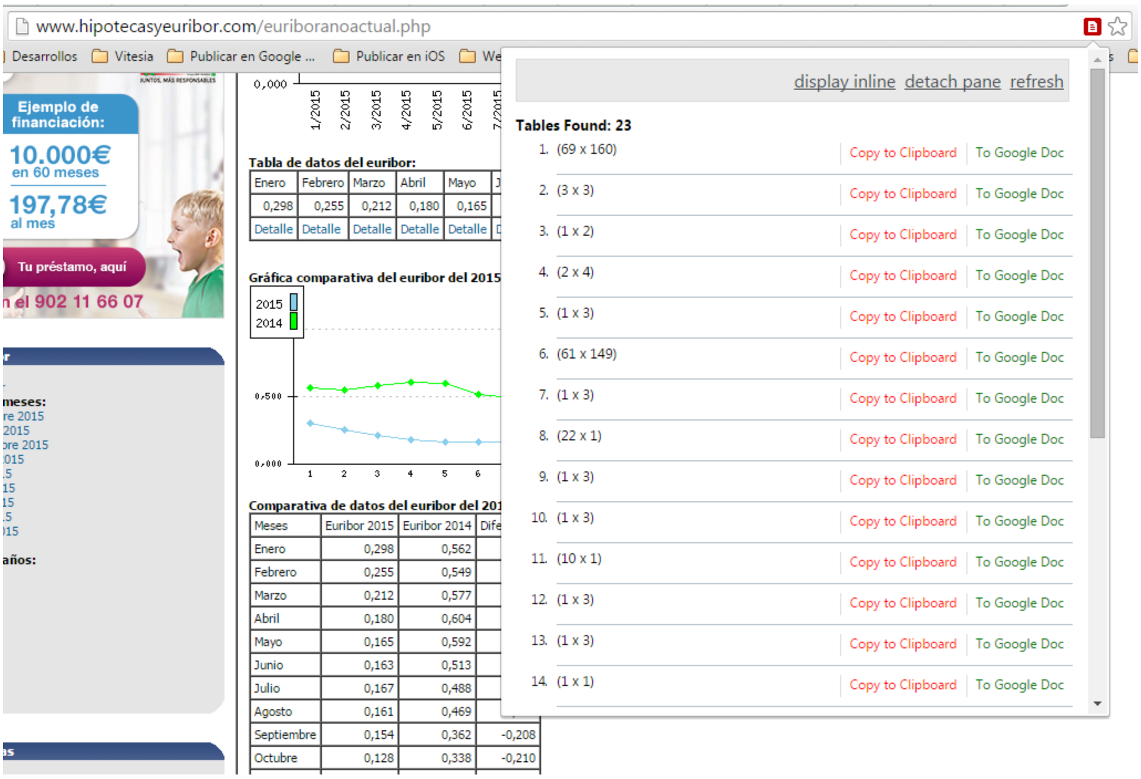


Tabla de datos del euribor:

Enero	Febrero	Marzo	Abril	Mayo	J
0,298	0,255	0,212	0,180	0,165	
Detalle	Detalle	Detalle	Detalle	Detalle	D

Comparativa de datos del euribor del 201

Meses	Euribor 2015	Euribor 2014	Dife
Enero	0,298	0,562	
Febrero	0,255	0,549	
Marzo	0,212	0,577	
Abril	0,180	0,604	
Mayo	0,165	0,592	
Junio	0,163	0,513	
Julio	0,167	0,488	
Agosto	0,161	0,469	
Septiembre	0,154	0,362	-0,208
Octubre	0,128	0,338	-0,210

Tables Found: 23

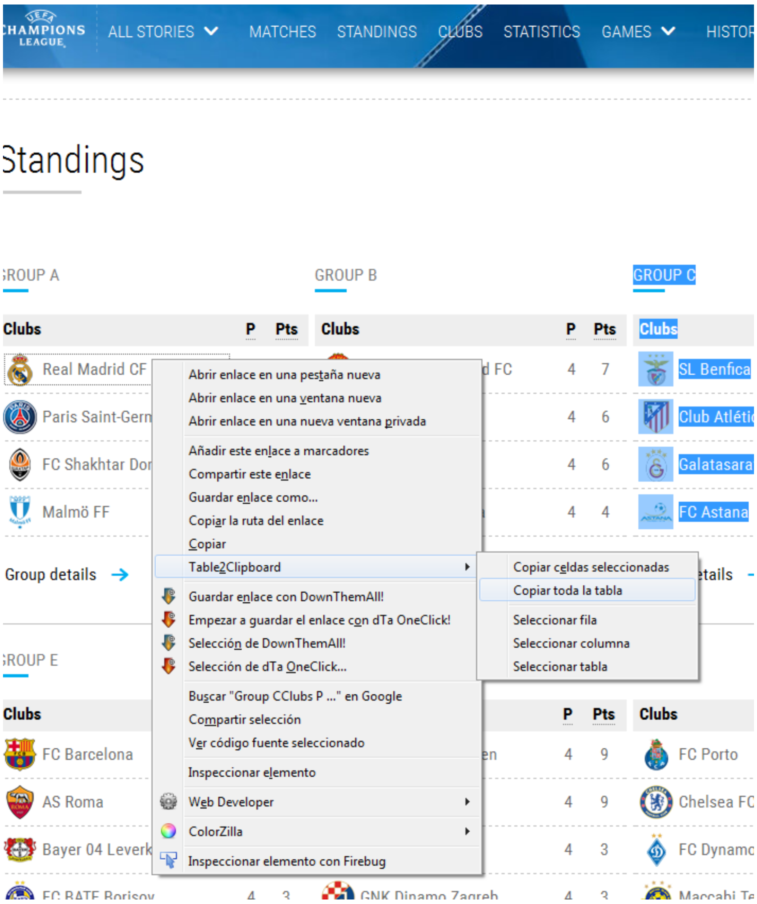
1. (69 x 160)	Copy to Clipboard	To Google Doc
2. (3 x 3)	Copy to Clipboard	To Google Doc
3. (1 x 2)	Copy to Clipboard	To Google Doc
4. (2 x 4)	Copy to Clipboard	To Google Doc
5. (1 x 3)	Copy to Clipboard	To Google Doc
6. (61 x 149)	Copy to Clipboard	To Google Doc
7. (1 x 3)	Copy to Clipboard	To Google Doc
8. (22 x 1)	Copy to Clipboard	To Google Doc
9. (1 x 3)	Copy to Clipboard	To Google Doc
10. (1 x 3)	Copy to Clipboard	To Google Doc
11. (10 x 1)	Copy to Clipboard	To Google Doc
12. (1 x 3)	Copy to Clipboard	To Google Doc
13. (1 x 3)	Copy to Clipboard	To Google Doc
14. (1 x 1)	Copy to Clipboard	To Google Doc

Example of use of Table Capture

Table to Clipboard, Firefox add-on

As in the previous case, the Firefox web browser also supports add-ons (analogous to Chrome's extensions) to extract data from HTML tables. An example is Table2Clipboard, which can be downloaded and installed from https://addons.mozilla.org/es/firefox/addon/dafizilla-table2clipboard/?src=userprofile

In this case, a context menu showing upon a right-click on a table allows copying it up or just the clicked row or column. This is a mechanism quite useful and less intrusive that offers an interesting functionality in many cases.



Example of use of Table to Clipboard

Complete tools

Over the last years, a set of companies, may of them start-ups, have realized that the market was demanding tools to extract, store, process and render data via APIs. The software industry moves fast and in many directions and **a good web scraper can help in application development, Public Administration transparency, Big Data processes, data analysis, online marketing, digital reputation, information reuse or content comparers and aggregators**, among other typical scenarios.

But, what should a tool of this kind include at least to be considered as a serious alternative? In the opinion of the author:

- A powerful and **friendly graphical user interface**.
- An easy-to-use **API to link and integrate data**.
- A **visual access** to web sites to perform **data picking**.
- Data **caching and storage**.
- A **logical organization and management of the queries** used to extract data.

Import.io

Company located in London, specialized in data mining and Internet data transformation to users to exploit information.

Web site → import.io and enterprise.import.io

Motto → Instantly turn web pages into data

Indubitably, this is one of the reference tools in the market. It may be used in four different ways. The first one (named **Magic**, that can be classified as basic) is the access to import.io in order to type the address of the web site on which we want to perform scraping. The result is shown in an attractive visual tabular format. The main con is that the process is not configurable at all.

The second way of use is named **Extractor**. It is the most common usage of import.io technology: download, install and execute in your own computer. This tool is a customized web browser available for Windows, OS X and Linux. This way requires some previous skills using software tools and some time to learn how to use the tool. However, “picking” is offered in a quite reasonable manner –although open to improvement, at the same time. “Picking” is performed by clicking on the parts of the scraped web site that we want to extract, in a simple and visual way. This is a feature that any web scraping tools must include these days.

Once that queries have been created, output formats are only two: JSON and CSV. Queries may also be combined in order to page results (“Bulk Extract”) or aggregate them (“URLs from another API”). It is also relevant to note that users will have a RESTful API EndPoint to access the data source –which is a mandatory feature in any relevant complete scraping tool nowadays.

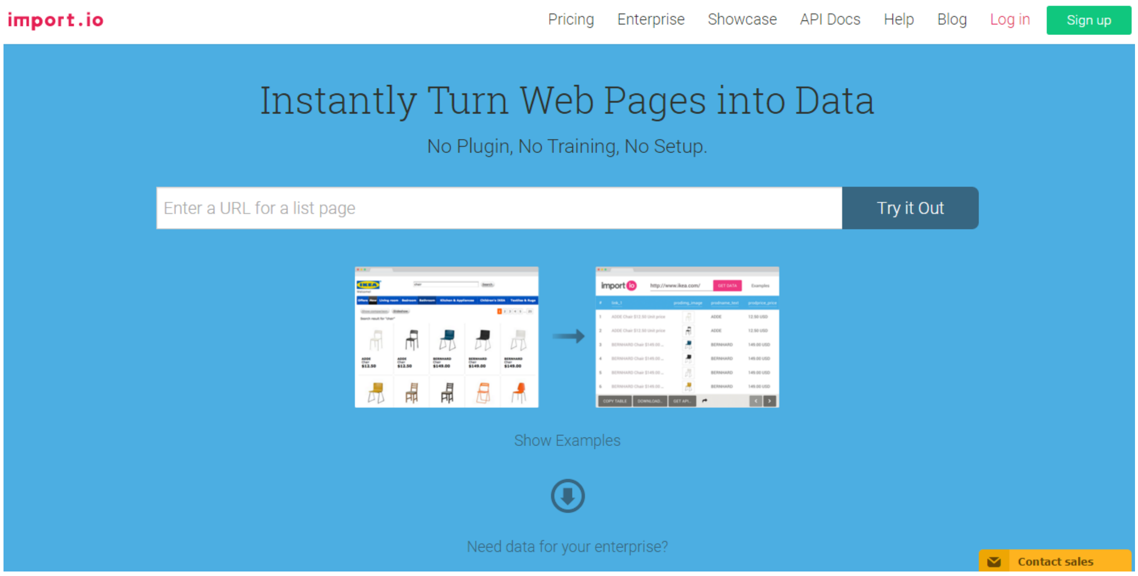
The import.io application requires a simple user registration process. With a username and password, the application can be used for free, with a set of basic features that may be sufficient for small developer teams without complex scraping requirements. For companies or professionals demanding more flexibility and backend resources, contact with import.io sales team is required.

The third and fourth ways of use provide more value to the tool. They are the **Crawler** and the **Connector**, respectively. The Crawler tries to follow all the links in the document indicated via its URL and it allows information extraction based on the picking process carried out in the initial document. In the tests carried out to write this article, we have not managed to finish all this process, as it seems to keep working all the time without producing any results. The Connector permits to record a browsing script to reach the web document from which to extract information. This approach is very interesting, for instance, if the data to be scraped are the result of a search.

In summary, import.io is a tool with interesting functionality **free to use**, with a **high level of maturity**, **an attractive and modern graphical user interface**, supporting cloud storage of the queries, which demands local installation to take full advantage of its features.

Strengths	Weaknesses
Visual Interface	Desktop installation
Blog and Documentation	Limited amount of output document formats
Allows pagination, aggregation, crawling and script recording	Learning curve

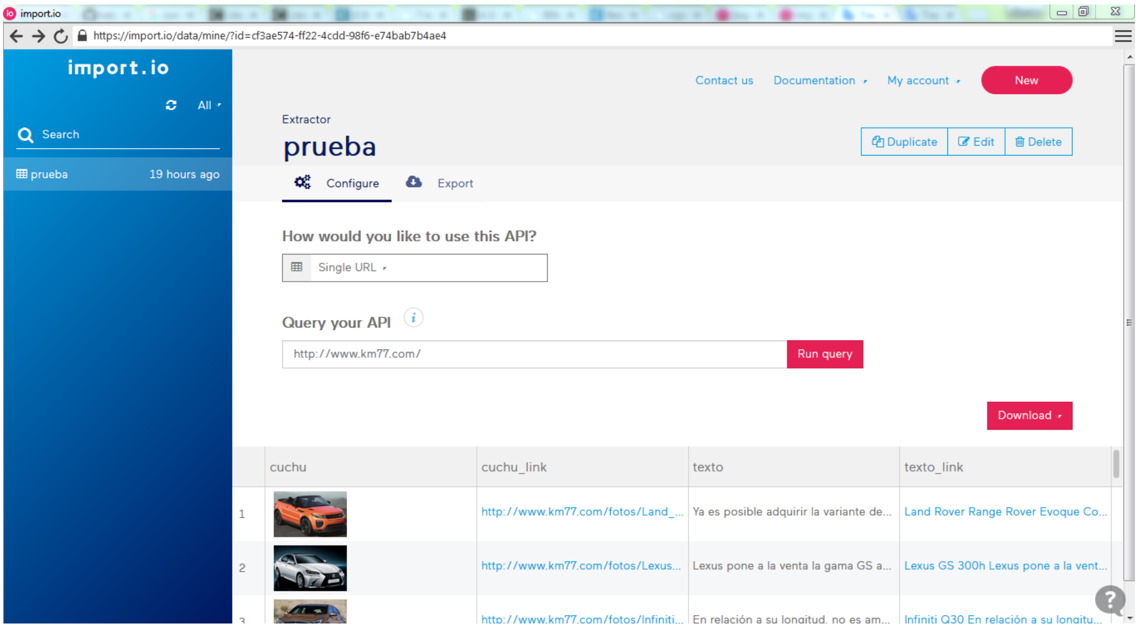
Strenghts vs Weaknesses for import.io



Instantly Turn Web Pages into Data

No Plugin, No Training, No Setup.

View of the basic use of import.io (available at https://import.io)



https://import.io/data/mine/?id=cf3ae574-ff22-4cdd-98f6-e74bab7b4ae4

Extractor

prueba

How would you like to use this API?

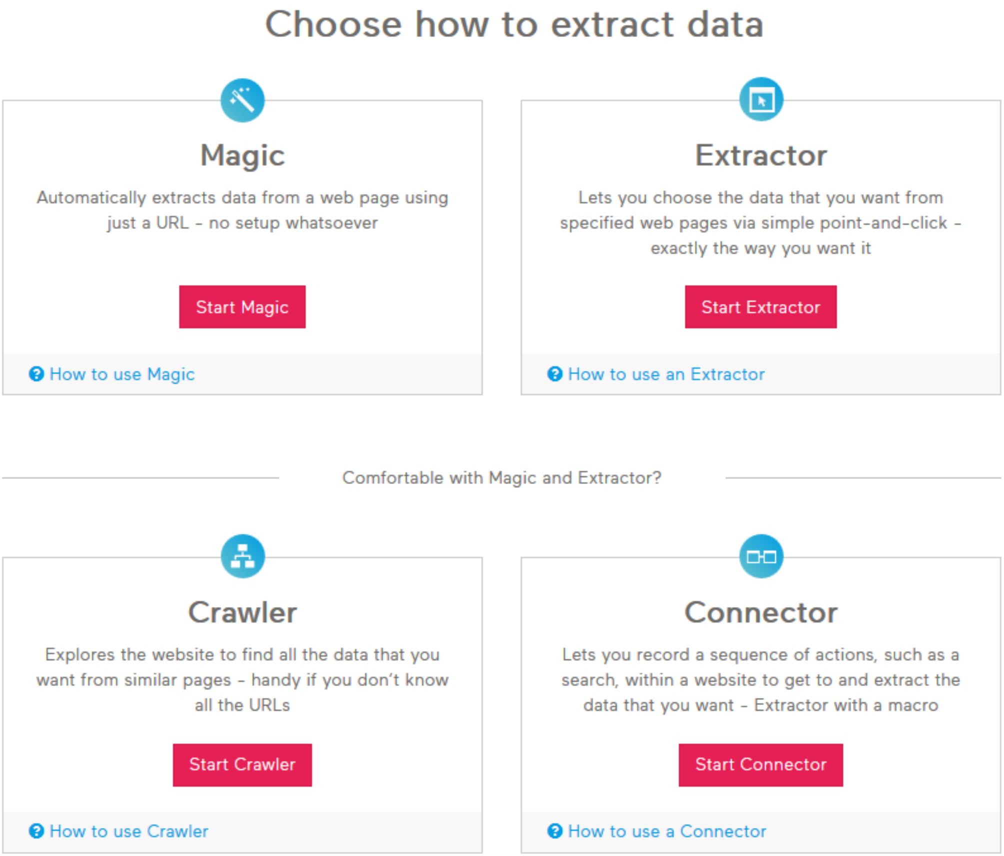
Single URL

Query your API

http://www.km77.com/

	cuchu	cuchu_link	texto	texto_link
1		http://www.km77.com/fotos/Land_...	Ya es posible adquirir la variante de...	Land Rover Range Rover Evoque Co...
2		http://www.km77.com/fotos/Lexus...	Lexus pone a la venta la gama GS a...	Lexus GS 300h Lexus pone a la vent...
3		http://www.km77.com/fotos/Infiniti...	En relación a su longitud, no es am...	Infiniti Q30 En relación a su longitu...

import.io desktop application



Choose how to extract data

Magic

Automatically extracts data from a web page using just a URL – no setup whatsoever

Extractor

Lets you choose the data that you want from specified web pages via simple point-and-click – exactly the way you want it

Comfortable with Magic and Extractor?

Crawler

Explores the website to find all the data that you want from similar pages – handy if you don't know all the URLs

Connector

Lets you record a sequence of actions, such as a search, within a website to get to and extract the data that you want – Extractor with a macro

Screen to select data extraction mode in import.io

Kimonolabs

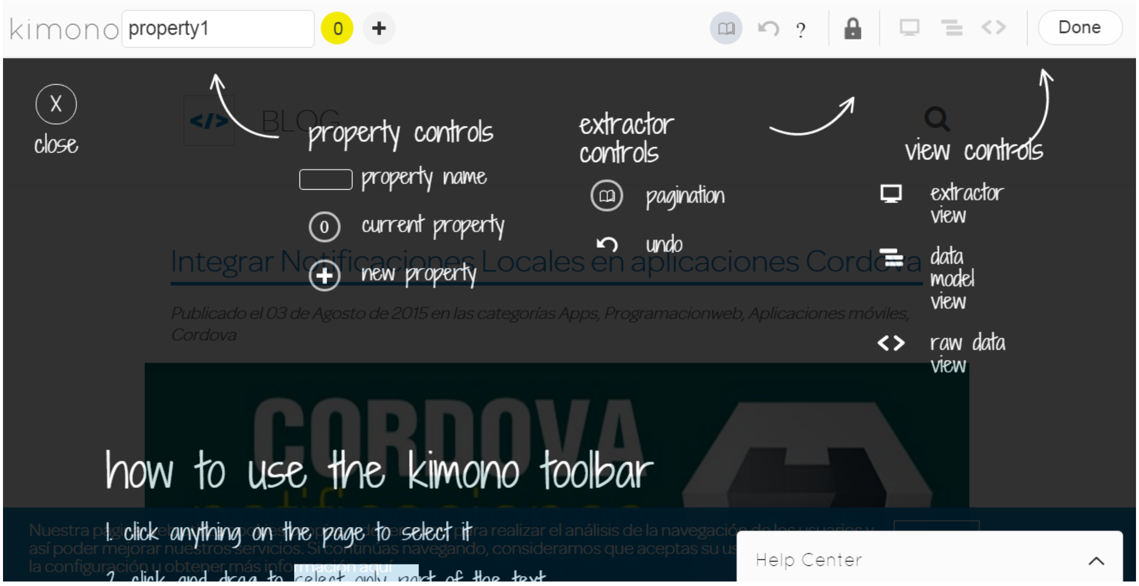
Kimono Labs is a company located in Mountain View, California. They offer a SaaS tool after the use of a Chrome extension. It allows extracting data from webdocuments in an intuitive and easy way. They provide results in JSON, CSV and RSS formats.

Web site → www.kimonolobas.com

Motto → Turn websites into structured APIs from your browser in seconds

Kimono Labs is another key player in the field of web scraping. It uses a strategy similar to import.io, using a web browser. In their case, they offer a Chrome extension, rather than embedding a web browser in a native desktop application. Therefore, the first step to use this tool is to install Google Chrome and then this extension. Registration is optional. It is a quick and simple process that is recommended.

After installation and registration, we can use Chrome to reach a web document with interesting information. In this moment, we click the icon installed by the Kimono extension and the picking process starts. This process provides help to the user at first execution with a visual and attractive format. Its graphical user interface is really polished and the tool results very friendly.



how to use the kimono toolbar

1. click anything on the page to select it

2. click and drag to select only part of the text

Integrar Notificaciones Locales en aplicaciones Cordova

Publicado el 03 de Agosto de 2015 en las categorías Apps, Programacionweb, Aplicaciones móviles, Cordova

Initial help view offered by Kimono

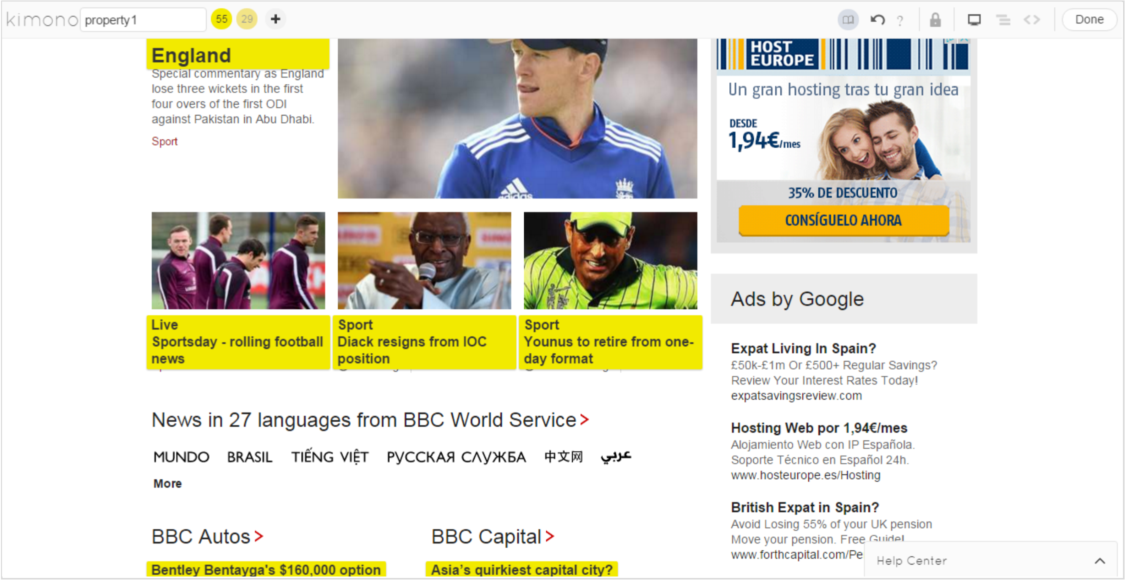
To start working with Kimono, the user must create a "ball" in the upper end of the screen. By default, a ball is already created. The various balls created are shown with a different colour. Afterwards, sections of the document may be selected to extract data an, subsequently, they are highlighted with the colour of the associated ball. At the same time, the ball shows the number of elements that match the selection. Balls may be used to select different zones of the same document, although their purpose is to refer zones with certain semantic consistency.

For instance, in the web site of a newspaper, we might create a ball named "Title" and then select a headline in a web page. Then, the tool highlights one or two additional headlines as interesting. After selecting a new headline, the tool starts highlighting another 20 interesting elements in the web page. We may notice that there are some headlines which are not highlighted by the tool. We select one of them and now over 60 headlines are highlighted. This process may be repeated until the tool highlights all the headlines after selecting a small amount of them. This process is known as "selector overload" and is available in several scraping tools.

Once that all the headlines have been selected, we can try doing the same with the opening paragraph of each news item: create a ball, click on the text area of an opening paragraph, then another one, and go on with the process until having all the desired information ready for extraction.

Although the idea is really good, our tests have found that the process is somehow bothersome. Sometimes, box highlighting in web pages does not work well with, for instance, problems in texts which are links at the same time.

Once that the picking process is finished by clicking on the "Done" button, we can name our new API and parameterize its temporal programming. With this, the system may execute the API every 15 minutes, hour, day, etc. and store the results in the cloud. Whenever the user calls the API by accessing the associated URL, Kimono does not access the target site but returns the most recent data stored in its cache. This caching mechanism is highly useful but not exclusive of Kimono.



England

Special commentary as England lose three wickets in the first four overs of the first ODI against Pakistan in Abu Dhabi.

Sport

Live
Sportsday - rolling football news

Sport
Diack resigns from IOC position

Sport
Younus to retire from one-day format

News in 27 languages from BBC World Service >

MUNDO BRASIL TIẾNG VIỆT РУССКАЯ СЛУЖБА 中文网 عربي

More

BBC Autos >

Bentley Bentayga's $160,000 option

BBC Capital >

Asia's quirkiest capital city?

HOST EUROPE

Un gran hosting tras tu gran idea

DESDE 1,94€/mes

35% DE DESCUENTO

CONSÍGUELO AHORA

Ads by Google

Expat Living In Spain?
£50k-£1m Or £500+ Regular Savings? Review Your Interest Rates Today!
expatsavingsreview.com

Hosting Web por 1,94€/mes
Alojamiento Web con IP Española. Soporte Técnico en Español 24h.
www.hosteurope.es/Hosting

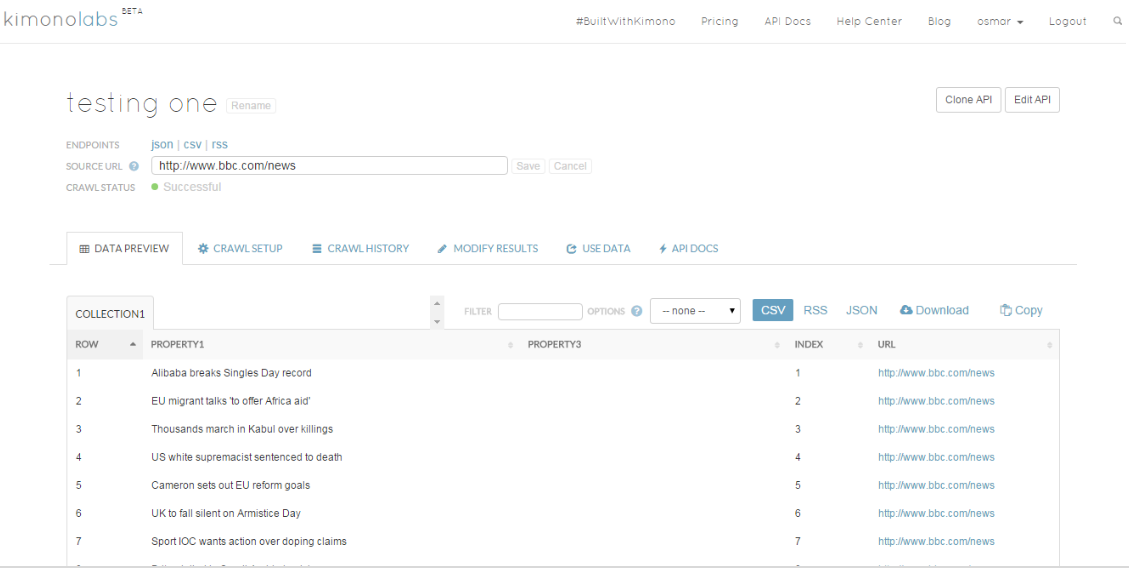
British Expat in Spain?
Avoid Losing 55% of your UK pension Move your pension. Free Guide!
www.forthcapital.com/Pe

Example of highlighting in the picking process of kimono

The query management console, available at www.kimonolabs.com, provides access to the newly created API and various controls and panels to read data and configure how they are obtained. This includes an interesting feature, which are email alerts to be received when data change in the target site.

There is an additional interesting option named "Mobile App" that integrates the content of the created API in a view resembling a mobile application, allowing some styling configuration. However, the view generated by this option is a web document accessible by the URL announced, aimed to be rendered in a mobile browser. Unluckily, the name of the option misleads users and does not generate a mobile application to be published in any mobile application store. Still, it **may be a useful option for rapid prototyping**.

The console menu also offers the "Combine APIs" option. Initially, it may look like an aggregator, assembling the data obtained from several heterogeneous APIs in a single API. Nevertheless, help information in this option indicates that the aggregated APIs must have the same exact name of data collections. The conclusion is that this option is useful to paginate information, but not to aggregate.



testing one

ENDPOINTS json | csv | rss
SOURCE URL http://www.bbc.com/news
CRAWL STATUS Successful

FILTER OPTIONS -- none --

ROW	PROPERTY1	PROPERTY3	INDEX	URL
1	Alibaba breaks Singles Day record		1	http://www.bbc.com/news
2	EU migrant talks 'to offer Africa aid'		2	http://www.bbc.com/news
3	Thousands march in Kabul over killings		3	http://www.bbc.com/news
4	US white supremacist sentenced to death		4	http://www.bbc.com/news
5	Cameron sets out EU reform goals		5	http://www.bbc.com/news
6	UK to fall silent on Armistice Day		6	http://www.bbc.com/news
7	Sport IOC wants action over doping claims		7	http://www.bbc.com/news

Management console of kimono

In summary, kimono is a free tool, with a high level of maturity, a very good graphical user interface, providing cloud storage for queries, requiring Chrome browser and their extension – both of them installed locally.

Strengths	Weaknesses
Visual interface	Chrome browser dependency
Documentation	Does now allow aggregation
Picker	Weak mobile app option

Strenghts vs Weaknesses for kimono

myTrama

myTrama is a web crawling tool developed by Vitesia Mobile Solutions, a company located in Gijón, Spain. myTrama allows any user to extract data located in different Internet sites and obtain them in an ordered and structured way in various formats (JSON, XML, CSV and PDF).

Web sites → www.mytrama.com and www.vitesia.com

Motto → Data is the new oil

myTrama is a new web crawling tool positioned as a clear competitor to those previously commented. It is a purely SaaS service, thus avoiding the need for users to install any software nor to depend on a specific web browser. myTrama works on Chrome, Firefox, Internet Explorer and Safari. It is available at https://www.mytrama.com.

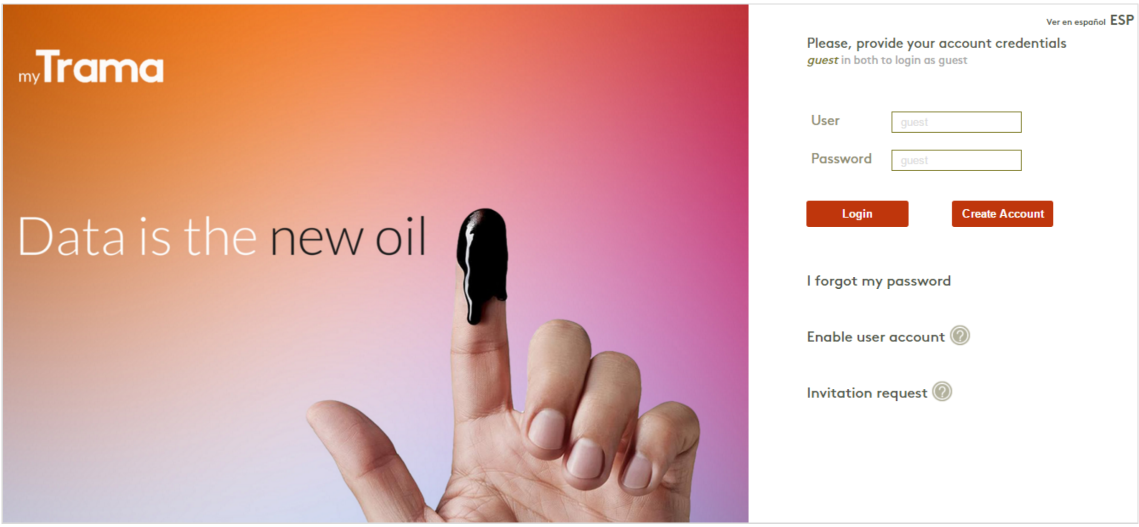
A general analysis of this tool suggests that myTrama takes the best ideas of import.io and kimono. It presents information in a graphical user interface, perhaps not so good but more compact and with the look and feel of a project management tool. Some of the features which seem more interesting in this tool are commented below:

- Main view is organized in a way similar to an email client, with 3 vertical zones: 1) folders, 2) queries, and 3) query detail. It is efficient and friendly.
- Besides JSON, XML and CSV, the classical structured formats for B2B integrations, it adds PDF for quick reporting and sends results in an easily viewable and printable format.
- It includes a query language named Trama-WQL (quite similar to SQL), which is simple to use while powerful. It is useful when visual picking is not sufficient, providing a programmatic manner to define the picking process. Documentation of this language is available in the tool as a menu option.
- The "Audit" menu option gives access to a compact control panel with information about the requests currently being made to each of the APIs (EndPoints).
- The picker is completely integrated. It is not necessary any type of additional software. It is similar to the approach used in kimono, although it uses "boxes" instead of "balls". A subtle differentiation is that a magic wand replaces the default mouse pointer when picking is available. In addition, the picking process may be stopped by right-clicking on the area being picked.
- myTrama permits grouping boxes within boxes, although only one level of grouping

and only a group with query are allowed. This is a very useful feature in order to have results properly grouped. Hopefully, the development team will improve this feature soon to provide users with more flexibility.

- Query configuration allows update frequency with a granularity of minutes, from 0 to 9999999. Zero means real time (this is, accessing the target site upon each request to the EndPoint). For any other value, information is obtained from the cache –as in kimono.
- APIs may be programmed using parameters sent via GET and POST requests. Unfortunately, the dev team has not published sufficient documentation related to this feature. For example, it is possible to use the URL of an API and overwrite the FROM parameter (the URL referencing the target document) in real time. It is also possible to pass parameters via GET and POST in the same API. Additionally, there is a service that allows the execution of a Trama-WQL sentence without any query created in the tool. As these are not very well documented features, the best choice is to contact the people at Vitesia.
- Paging queries and aggregation of heterogeneous queries are supported in a fairly simple and comfortable way.
- For those preferring the browser extension way of scraping, a Chrome extension is also available. This mechanism allows users to browse sites and start the scraping process by clicking on the button installed by the extension. This plugin is not yet published but can be requested to Vitesia.
- PDF is not only a format available as output, but also as input. Therefore, a URL may reference only HTML documents but also PDF. For instance, users will be able to extract information from PDFs and generate JSON documents that feed a database for later information analysis. The business hypothesis to support this is based on the evidence initially commented at the introduction of this article that stated that 70% of the content published in the Internet is contained in PDF documents. Vitesia consider that this may be a differentiating feature between myTrama and their competitors.
- APIs preserve session. This allows chaining calls to queries in myTrama and fulfil business processes, such as searches, step-based forms (wizards) or access information available behind a login mechanism.
- It is available in two languages: English and Spanish.
- Access to this platform is based on invitation. Users remain active for 30 days. Later contact with the dev team is required in order to move to a stable user.

Among all the tools analysed, myTrama seems to be the most complete and compact, although its user interface is one step behind kimono and import.io. For users with software development skills, myTrama seems to be the best choice –although requiring direct contact with Vitesia.



Please, provide your account credentials
guest in both to login as guest

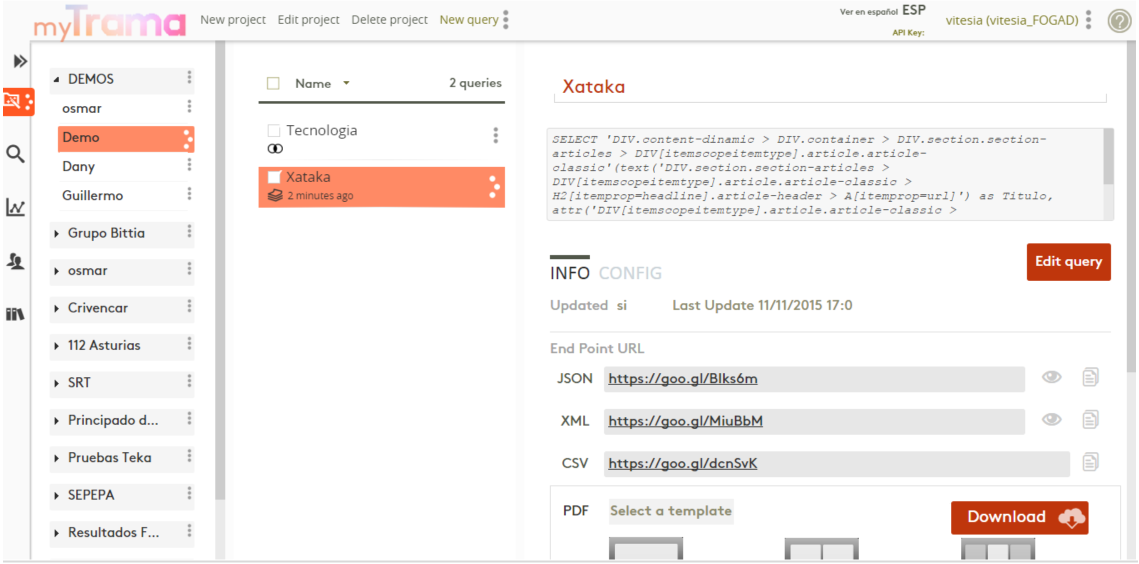
User guest
Password guest

I forgot my password

Enable user account

Invitation request

Initial screen of myTrama



Xataka

```
SELECT 'DIV.content-dinamic > DIV.container > DIV.section.section-
articles > DIV[itemscopeitemtype].article.article-
classic'(text('DIV.section.section-articles >
DIV[itemscopeitemtype].article.article-classic >
H2[itemprop=headline].article-header > A[itemprop=url]') as Titulo,
attr('DIV[itemscopeitemtype].article.article-classic >
```

Updated si   Last Update 11/11/2015 17:0

End Point URL

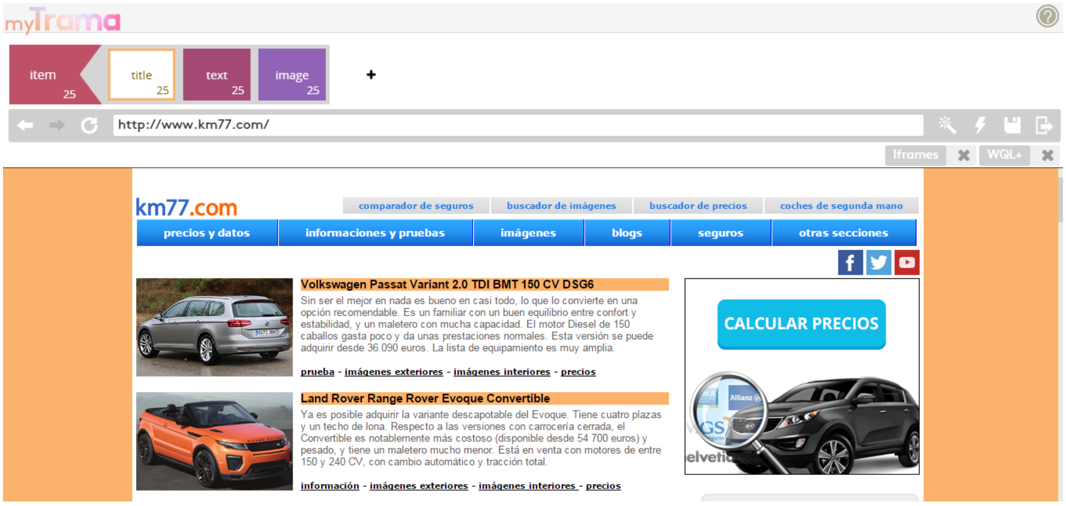
JSON https://goo.gl/Blks6m

XML https://goo.gl/MiuBbM

CSV https://goo.gl/dcnSvK

PDF Select a template

Query management console in myTrama



Volkswagen Passat Variant 2.0 TDI BMT 150 CV DSG6

Sin ser el mejor en nada es bueno en casi todo, lo que lo convierte en una opción recomendable. Es un familiar con un buen equilibrio entre confort y estabilidad, y un maletero con mucha capacidad. El motor Diesel de 150 caballos gasta poco y da unas prestaciones normales. Esta versión se puede adquirir desde 36 090 euros. La lista de equipamiento es muy amplia.

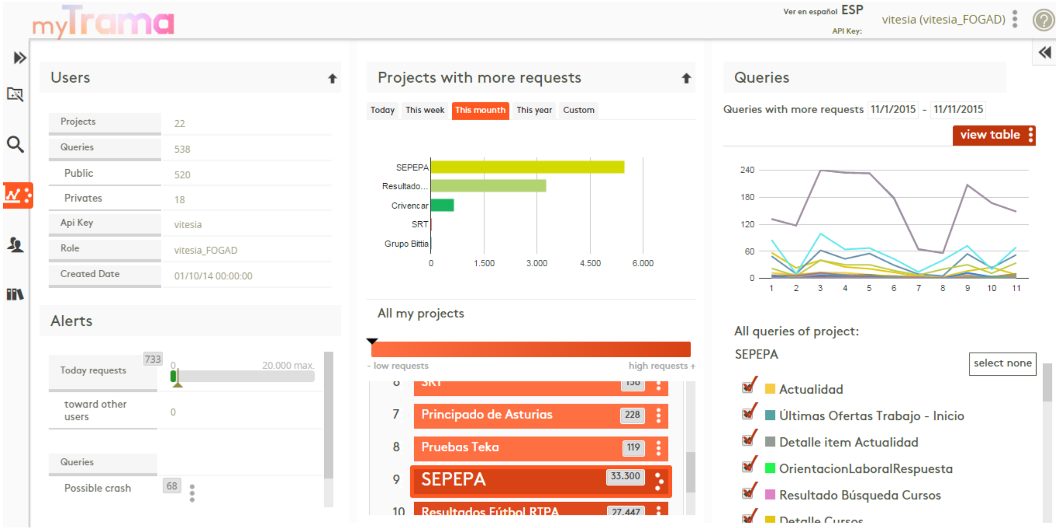
prueba - imágenes exteriores - imágenes interiores - precios

Land Rover Range Rover Evoque Convertible

Ya es posible adquirir la variante descapotable del Evoque. Tiene cuatro plazas y un techo de lona. Respecto a las versiones con carrocería cerrada, el Convertible es notablemente más costoso (disponible desde 54 700 euros) y pesado, y tiene un maletero mucho menor. Está en venta con motores de entre 150 y 240 CV, con cambio automático y tracción total.

información - imágenes exteriores - imágenes interiores - precios

Picking process in myTrama



Projects	22
Queries	538
Public	520
Privates	18
Api Key	vitesia
Role	vitesia_FOGAD
Created Date	01/10/14 00:00:00

Today requests	733	0
toward other users		0
Queries		
Possible crash	68	

6	SRT	156
7	Principado de Asturias	228
8	Pruebas Teka	119
9	SEPEPA	33.300
10	Resultados Fútbol RTPA	27.447

- Actualidad
- Últimas Ofertas Trabajo - Inicio
- Detalle item Actualidad
- OrientacionLaboralRespuesta
- Resultado Búsqueda Cursos
- Detalle Cursos

Dashboard screen in myTrama

In summary, myTrama is a tool solely offered as a SaaS service, very complete to carry our scraping processes, with cloud storage and that may be operated with any web browser. Its major weakness is the lack of documentation of many differentiating issues relevant to developers interested in taking advantage of scraping processes.

Strengths	Weaknesses
The Trama-WQL language	Limited documentation
Dashboards	Free license only for 30 days
Picker	More oriented to developers
Session preservation between API requests	

Strengths vs weaknesses for myTrama

Confronting the three tools

		import.io	kimonolabs BETA	myTrama
distribution	SaaS model		X (requires installation of chrome extension)	X
	Desktop installer	X (Windows, OS X and Linux)		
	Chrome extension		X (required)	X (optional)
	Free license	X	X	X
	Cross-browser compatibility	Own browser	Only Chrome	X
operation	Valuation of user interface	Good	Very good	Good
	APIs creation simplicity	X	X	X
	Visual picking	X	X	X
	Caching and storage	X	X	X
	Query organization	X	X	X
	Own query language			Trama WQL
	Statistics and audit dashboards			X
	PDF extraction			X
	Output formats	JSON, CSV	JSON, CSV, RSS	JSON, CSV, XML, PDF
	APIs creation simplicity	X	X	X
	Session preservation between API invocations			X
features	Automatic crawling	?		
	Maturity level	High	High	Medium/High
	Complex of use	Medium/High	Medium	Medium
	Cloud storage	X	X	X
	Query pagination	X	X	X
	Query aggregation	X		X
	Real time data	?	?	X
others	Orientation to software development	Medium/Low	Medium/Low	Medium
	Google Docs integration	X	X	
	Level of documentation	High	High	Low

Other tools

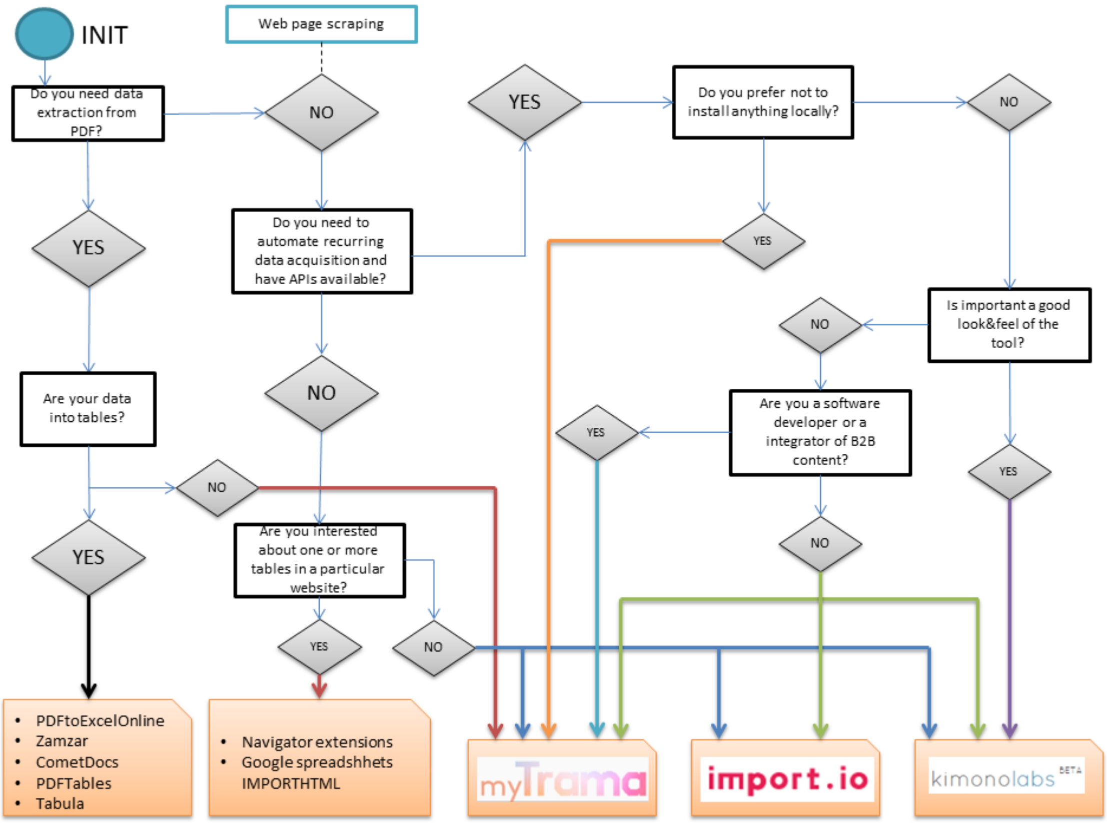
This article analyses three tools in the category of complete tools but many other exist. For the sake of brevity, and for those readers interested in this kind of tools, some other interesting tools, utilities or frameworks permitting web scraping are listed below.

Mozenda	QuBole	ScraperWiki	Scrapy	Apache Nutch
Scrapinghub	ParseHub	Ubot Studio 5	Scraper (Chrome Plugin)	Outwit Hub
Fminer.com	80legs	Content Grabber	CloudScrape	Webhose.io
UIPath	Winautomation	Visual Web Ripper	AddTolt	Agent Community
All in One Stats	Automation Anywhere	Clarabridge Enterprise	Darcy Ripper	Data Integration
Data Crops	Dataddo	Diffbot	Easy Web Extract	Espion
Feedity	Ficstar Web Grabber	ForNova Big Data Platform	Helium Scraper	Kapow Katalyst
PDF Collector	PDF Plain Text Extractor	RedCritter	Scrape.it	Solid Converter
Spinn3r	SyncFirst Standard	TextfromPDF	Trapeze	Unit Miner
Web Content Extractor	Web Data Extraction	Web Data Miner	Web Robots Scraping	WebHarvy

If you are not convinced to use any of the three recommended tools, Mozenda or ParseHub may be interesting alternatives.

4 Decision map

The following diagram may be helpful in the process of deciding which tool meets which scraping requirements. Obviously, the diagram could be more complex, as more questions may be asked in a decision process. However, the rise of complexity in the figure suggests keeping it simple, but sufficiently illustrative.


INIT
Web page scraping
Do you need data extraction from PDF?
NO
YES
Do you prefer not to install anything locally?
NO
YES
Do you need to automate recurring data acquisition and have APIs available?
YES
Is important a good look&feel of the tool?
NO
Are your data into tables?
NO
Are you a software developer or a integrator of B2B content?
YES
YES
NO
YES
Are you interested about one or more tables in a particular website?
NO
YES
NO
PDFtoExcelOnline
Zamzar
CometDocs
PDFTables
Tabula
Navigator extensions
Google spreadshhets IMPORTHTML
myTrama
import.io
kimonolabs BETA

5 Scraping and legislation

The question to be made is simple: is **web scraping legal**?

The answer is not as simple. Firstly, we need to know the specific legislation of each country concerned. The United States of America are more permissive than Europe. In Spain, where the author lives and works, laws are fairly more restrictive. However, there are various statements (including one from the Supreme Court of Spain), where web scraping is considered legal under specific conditions.

For example, users must be careful with legal conditions in certain web sites. If their content is proprietary and oriented to commercial purposes**, scraping might be an illegal activity**. Author rights are another legal characteristic of special interest before scraping the information of a web site. For instance, news from digital newspapers cannot be extracted to be published in a blog or application without stating the source.

As expected, all this generates doubts, so it is a good practice to consult lawyers specialized in digital content, intellectual property, unfair competition and licensing. In any case, if data are clearly open, you will not need to worry.

6 Conclusions and recommendations

Web scraping is a familiar term that has gained importance because of the need to “free” data stored in PDF documents or web pages. Many professionals and researchers need the data in order to process it, analyse it and extract meaningful results. On the other hand, people dealing with B2B use cases need to access data from multiple sources to integrate it in new applications that provide added value and innovation.

The market demands holistic web scraping solutions, that encompass cloud storage and ease to build interoperable APIs. In this article we have analysed and compared 3 web scraping solutions: import.io, Kimono and myTrama. These solutions differ in their implementation details but share more common ground than believed: a visual picker, JSON results, data caching, background robots to gather data, programmatic APIs, etc.

An experienced team, using these solutions, can develop services such as content aggregators, ranking tools, mobile apps, data monitoring systems, reputation management applications, business intelligence solutions, Big Data, etc.

About the Author

Osmar Castrillo Fernández has more than 15 years of experience in the IT industry. He holds a Bachelor's degree in Computer Science from the University of Oviedo. His main expertise fields are web development, service-oriented architectures and public administration.

In April 2004 he started working for CTIC Foundation in the R&D team what was in charge of developing a new J2EE framework for the Principality of Asturias (FWPA). The FWPA was a key element in the success of the model of E-Government in the Principality of Asturias and helped to simplify and homogenise the development of new government-related applications and services.

He was a teacher in the first and second editions of the J2EE-FWPA course for IT professionals in Asturias. He also taught several other courses related to UML 2.0 and n-tier architectures.

In October 2012 he founded and become CTO of Vitesia Mobile Solutions, a Company devoted to extracting and analysing published data on the Internet. At the start of 2013 he started leading the TRAMA project, a technology to facilitate web scraping, that later became the tool myTrama.

Copyright information