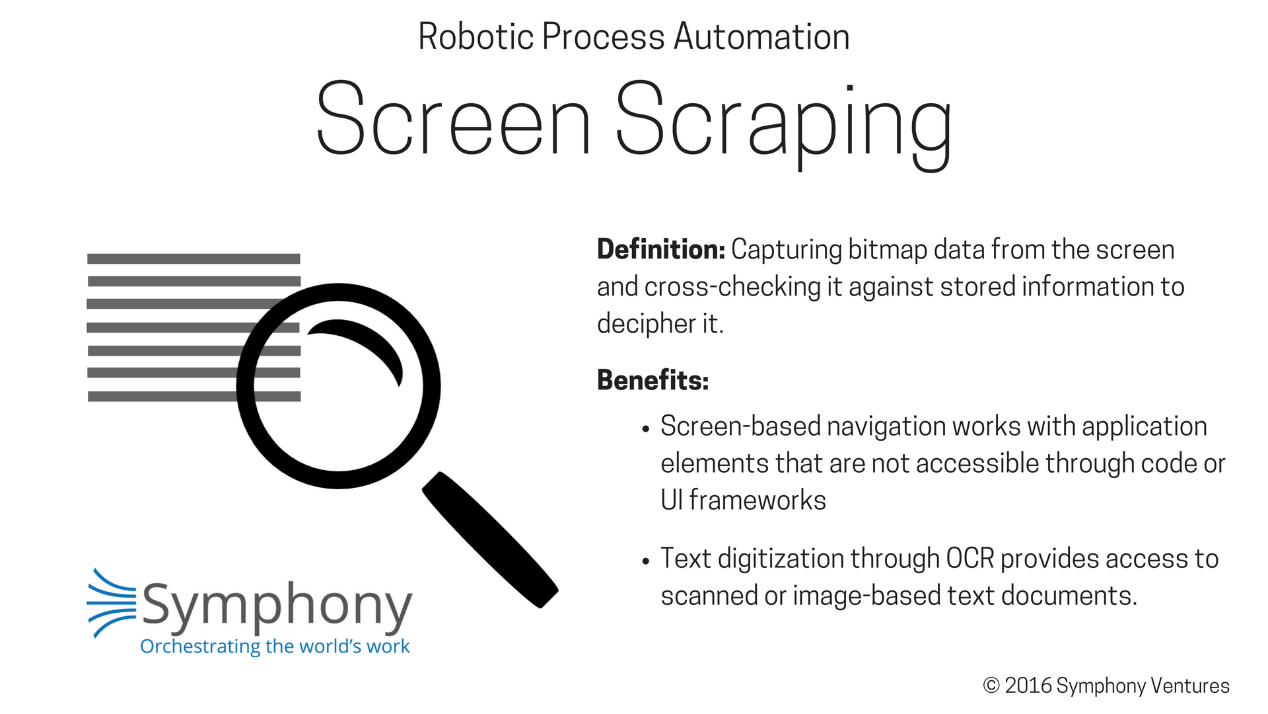
SCRAPING METHODS :

* Copy-pasting. The manual human examination and copy-pasting method may sometimes prove irreplaceable. ...
* DOM Parsing. ...
* HTTP Programming. ...
* Recognizing Semantic Annotation. ...
* Text Grepping. ...
* Web scraping Software.

**SCREEN SCRAPING DEFINITION :** 

SOURCE : <https://www.symphonyhq.com/rpa-technical-insights-part-10>

RPA DEFINITION :

Robotic Process Automation (RPA) is a business automation technology that helps in automating the humdrum repetitive tasks (data entry, banking) without human intervention.

SOURCE: https://www.edureka.co/blog/selenium-vs-rpa/

## Benefits of using RPA for web scraping

Leveraging an automated data scraping tool for websites provides the following benefits:

* The data collected by an RPA bot is more accurate in comparison to manual data entry.
* Automation makes the web scraping process faster. Tasks that take weeks to complete by hands get done within hours.
* Automation solutions are easy to implement and integrate within a company’s IT ecosystem.
* Low maintenance is needed as automated web scrapers require almost no upkeep in the long run.

RPA bots integrated with other technologies and tools such as machine learning makes the bot an even more powerful tool. For instance, machine learning combined with bots can locate companies’ websites from their logos.

SOURCE : <https://electroneek.com/blog/rpa/rpa-101-what-is-web-scraper/>

## Challenges of using RPA for web scraping

An RPA site scraper depends heavily on GUI elements to locate relevant data, making it challenging to automate when web pages don’t display content consistently. Some of the main web scraping limitations include:

* **User Interface (UI) features:** Some UI features such as infinite scrolling, popup messages and changing dynamic web content make it difficult for scrapers to handle data extraction. The solution would be additional bot programming for specific complex features.
* **”Load more” button:** Many pages load data in small parts using the ”Lload more” button. In such cases, the bot will stop at the end of a visible part end and miss further information without exploring more data. Creating an if-loop within the program of a bot to click the “lLoad more” button until no more buttons remain on a site.
* **Pop-up advertisements:** These advertisements often hide GUI elements from the RPA Bots view and impair data extraction. An effective solution would be to add the ADBlocker extension for your browser.
* **Protected systems:** Some web pages utilize advanced security technology to prevent their site from automated scraping, which becomes another roadblock. There are two possible solutions:; first,ly to either working with a company that offers web data as a service, or combining RPA with proxy servers to build bots that cannot be distinguished from humans.

SOURCE : SOURCE : <https://electroneek.com/blog/rpa/rpa-101-what-is-web-scraper/>

**BEAUTIFUL SOUP DEFINITION :**

Beautiful Soup is **a Python library** that is used for web scraping purposes to pull the data out of HTML and XML files. It creates a parse tree from page source code that can be used to extract data in a hierarchical and more readable manner. ... 2, and it supports Python 3 and Python 2.4 as well.

SOURCE : <https://analyticsindiamag.com/beautiful-soup-webscraping-python/>

**SCRAPY DEFINITION :**

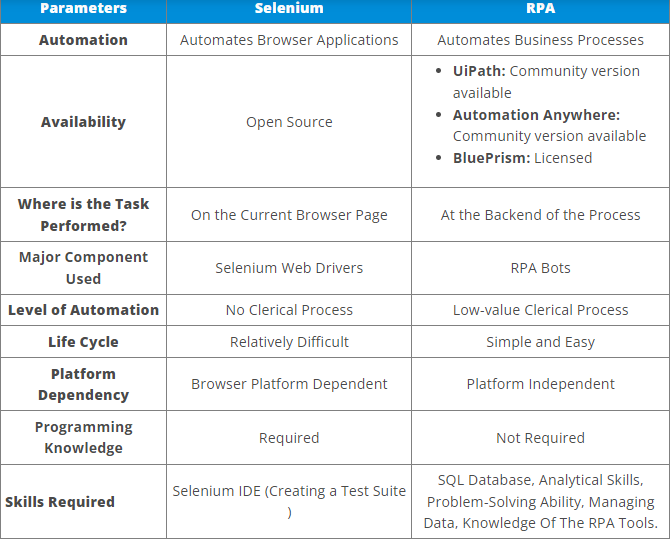
Scrapy is a Python framework **for large scale web scraping**. It gives you all the tools you need to efficiently extract data from websites, process them as you want, and store them in your preferred structure and format

SOURCE : <https://www.analyticsvidhya.com/blog/2017/07/web-scraping-in-python-using-scrapy/>

**SELINIUM DEFINITION :**

Selenium is an open-source tool that **automates web browsers**. It provides a single interface that lets you write test scripts in programming languages like Ruby, Java, NodeJS, PHP, Perl, Python, and C#, among others.

DIFFERENCE ENTRE SELENIUM ET RPA :



## Extensibility

**Scrapy:**The architecture of **Scrapy** is well designed to customize the middleware to add our own custom functionality. This feature helps us our project to be more Robust and flexible.

One of the biggest advantages of Scrapy is that we can able to migrate our existing project to another project very easily. So for the large/Complex projects, Scrapy is the best choice to work out.

If Your project needs proxies, data pipeline, then Scrapy would be the best choice.

**Beautiful Soup:**When it comes to a small project, Or low-level complex project Beautiful Soup can do the task pretty amazing. It helps us to maintain our code simple and flexible.

If you are a beginner and if you want to learn things quickly and want to perform web scraping operations then Beautiful Soup is the best choice.

**Selenium:** When you are dealing with Core Javascript featured website then Selenium would be the best choice. but the Data size should be limited.

## Performance

**Scrapy:**It can do things quickly because of its built-in feature i.e usage of asynchronous system calls. The Existing libraries out there not able to beat the performance of Scrapy.

**Beautiful Soup:**Beautiful Soup is pretty slow to perform a certain task but we can overcome this issue with the help of Multithreading concept but However the programmer need to know the concept of multithreading very effectively. This is the downside of Beautiful Soup.

**Selenium:**It can handle up to some range butn’t equivalent to Scrapy.

## EcoSystem

**Scrapy:**It has a good ecosystem, we can use proxies and VPN’s to automate the task. This is one of the reasons for choosing the library for complex projects. we can able to send multiple requests from the multiple proxy addresses.

**BeautifulSoup:** This library has a lot of dependencies in the ecosystem. This is one of the downsides of this library for a complex project

**Selenium:** It has a good ecosystem for the development but the problem is we can’t utilize the proxies very easily.

SOURCE: medium.com

DASH DEFINITION :

Dash is **a python framework created by plotly for creating interactive web applications**. Dash is written on the top of Flask, Plotly. js and React. js. With Dash, you don't have to learn HTML, CSS and Javascript in order to create interactive dashboards, you only need python.

SOURCE : <https://towardsdatascience.com/dash-for-beginners-create-interactive-python-dashboards-338bfcb6ffa4#:~:text=Dash%20is%20a%20python%20framework,dashboards%2C%20you%20only%20need%20python>.

UML :

https://www.lucidchart.com/pages/what-is-UML-unified-modeling-language