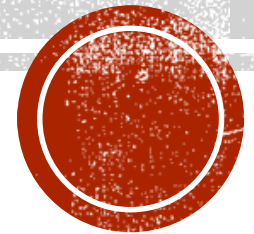


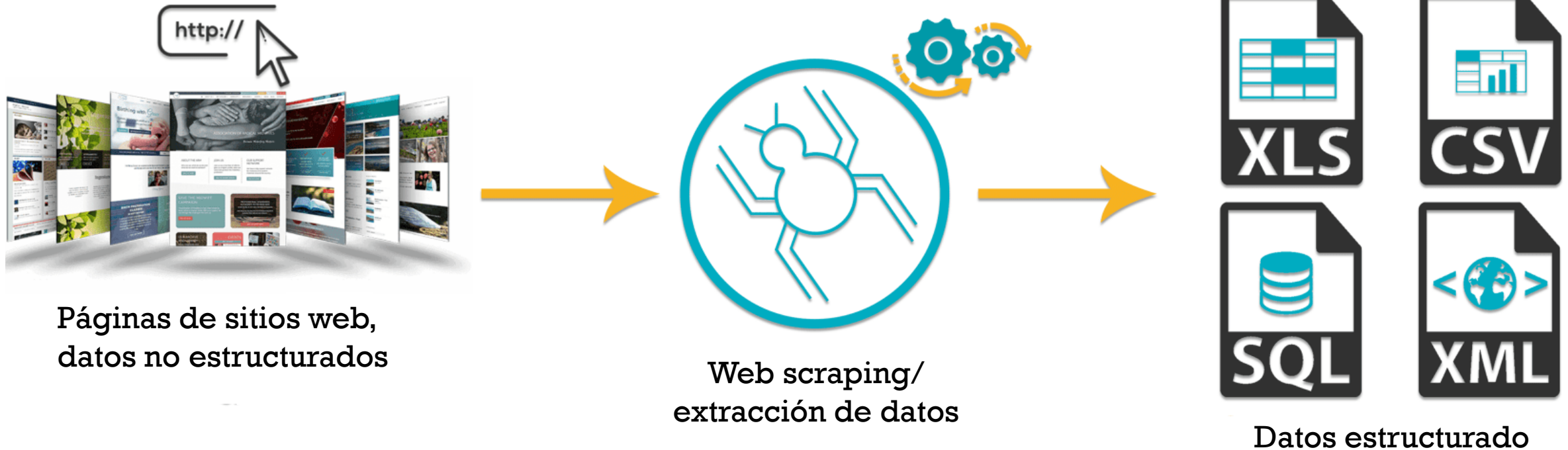
# WEB SCRAPING EN R



Por Ángel Sandoval

# ¿QUÉ ES WEB SCRAPING?

Es un tipo de "minería de internet" que implica extraer información relevante de un sitio web en particular para su posterior análisis.

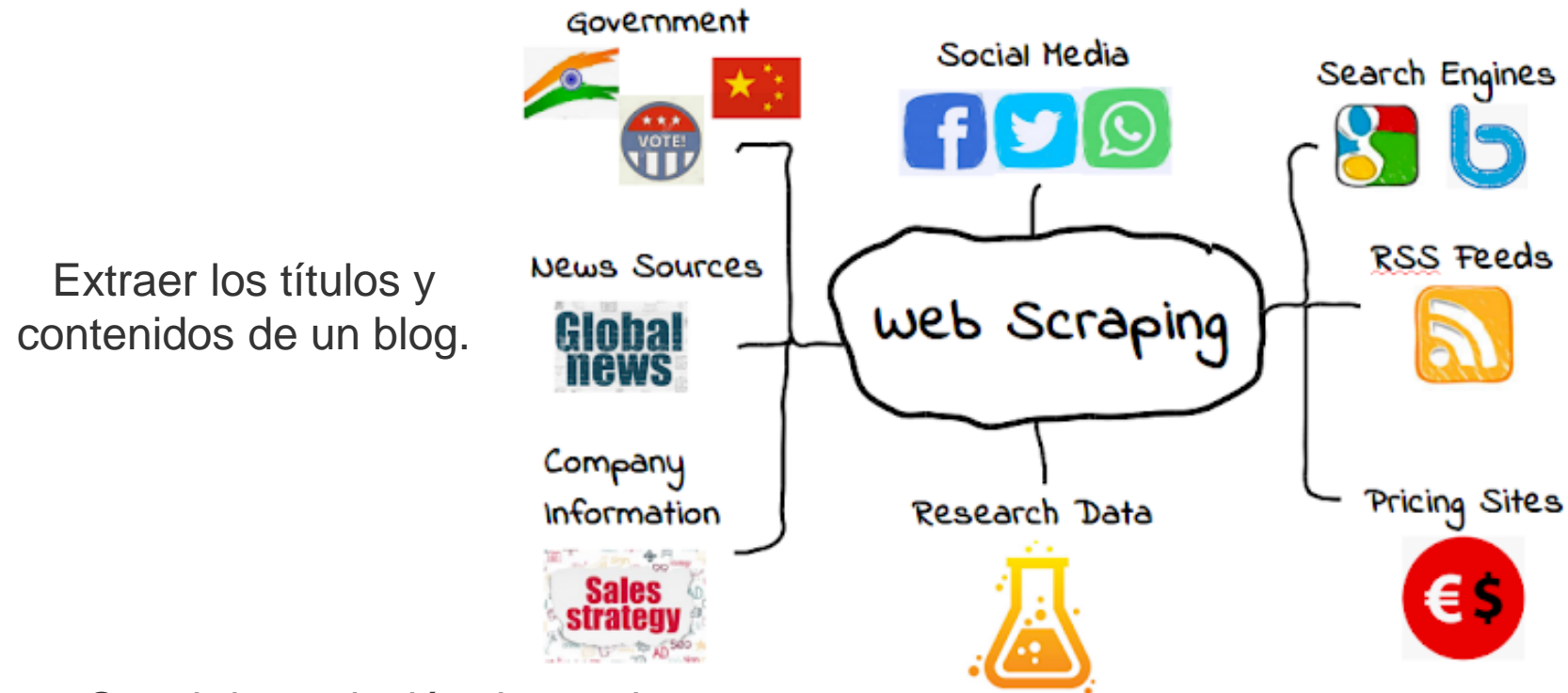


Se puede extraer información manualmente pero la idea es automatizar el trabajo utilizando bots



# USOS MÁS COMUNES

Extraer datos de contacto  
como por ejemplo email.



Extraer los títulos y  
contenidos de un blog.

Crear un canal RSS de los  
contenidos de una página web.

Seguir la evolución de precios  
de distintos productos.



# CONOCIMIENTOS PREVIOS

Estructura básica de una página web

Lenguaje html



Captura de pantalla de un navegador web (Chrome) mostrando la estructura HTML de una página. La URL es `reporteria.supercias.gob.ec/portal/cgi-bin/cognos.cgi?b_action=cognosViewer...`. El título de la página es "PORTAL DE INFORMACION - ESTADO FINANCIEROS CONSOLIDADOS POR E". El contenido principal muestra un formulario de búsqueda con los campos:

- TIPO ENTE: TIPO\_ENTE
- PERIODICIDAD: VALOR
- FECHA CORTE: FECHA\_CORTE

Hay un botón "Finalizar" debajo de los campos. A la derecha, se muestra el panel de desarrollo del navegador (Elements, Console, Sources) con la estructura HTML visible:

```
<!doctype html PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html lang="es">
  <head>...</head>
  <body role="application" topmargin="3" bottommargin="0"
    marginheight="3" rightmargin="5" leftmargin="5" marginwidth="
    3" onclick="if (typeof window.oCV_NS_ != 'undefined')
    {window.oCV_NS_.rvMainWnd.hideOpenMenus();}" style="overflow:
    auto" class="viewer">
    <div id="cvSkipToReport_NS_" style="display:none;" class=
    "skip">...</div>
    <div id="cvSkipToNavigation_NS_" style="display:none;"
    class="skip">...</div>
    <form action="/portal/cgi-bin/cognos.cgi" name=
    "..." method="post">
      <input type="text" value="run">
      <input type="text" value="javascript:...">
      <table border="1">
        <tr>
          <td>...

El menú de desarrollo del navegador está abierto, mostrando opciones como "Atrás", "Reenviar", "Volver a cargar", "Guardar como...", "Imprimir...", "Enviar...", "Enviar a tus dispositivos", "Traducir a español", "AdBlock: el mejor bloqueador de anuncios", "Ver código fuente de la página" (Ctrl + U) y "Inspeccionar" (Ctrl + Mayús + I).


```

# PASO PREVIOS PARA CREAR UN SCRAPIADOR EN R

- 1. Descargar driver para controlar navegador (ejemplo Chrome).
  - [https://chromedriver.storage.googleapis.com/2.31/chromedriver\\_win32.zip](https://chromedriver.storage.googleapis.com/2.31/chromedriver_win32.zip)
  - <https://selenium-release.storage.googleapis.com/3.14/selenium-server-standalone-3.14.0.jar>
- 2. Instalar librerías
  - `devtools::install_github("johndharrison/binman")`. Para gestionar la descarga de archivos binarios
  - `devtools::install_github("johndharrison/wdman")`. Proporciona funciones para descargar estos archivos binarios y para gestionar los procesos que los involucran
  - `devtools::install_github("ropensci/RSelenium")`



# COMO HACER QUE FUNCIONE LA HERRAMIENTA

- 1. En el CMD setear la dirección donde se guardo los driver
- 2. Ejecutar el driver especificando un puerto
  - `java -jar selenium-server-standalone-3.14.0.jar -port 8002.`

La sentencia anterior debe arrojar como resultado en el cmd:  
Selenium Server us up and running port 8002

