SCRAPING THE WEB

IRE Tipsheet, Orlando Florida, 2011

Ryan McNeill
The Dallas Morning News
rmcneill@dallasnews.com

Cynthia O'Murchu
Financial Times
cynthia.omurchu@ft.com

Why scraping? Governments and organisation across the world are making data more easily available, and a proliferation of data are often available for download as Excel spreadsheets or database. But despite much progress journalists find that the government's rhetoric on data transparency does not always match reality. Information is often presented in "closed" formats such as PDF, or presented in badly structured form, or else dispersed across many different websites, which makes it difficult to pick up on trends. Scraping – using tools or coding from scratch – will help you overcome this.

Before you scrape

1. Try and get the data from the organisation that holds the data. Even if they are not making the data available in the desired format, they may be prepared to give you the data in database/Excel/CSV format. If the site presents data dynamically, rather than a static page, that's a pretty good indicator there's a database backend capable of exporting the data in a machine-readable format.
2. Check whether anyone else has already done the work. You'll find that sometimes academics, NGOs or commercial organisations have already created a database and you may be able to negotiate access. For example, TRAC at Syracuse compiles a heck of a lot of data from the federal court system.
3. Evaluate how important it is to get the data and how much it will cost in terms of work-hours. If you are going to spend a lot of time on it, be as sure as you can be that the story will pan out. But don't be afraid to try --- even if it might not end up in a story, you can treat scraping a site like a challenge to improve your skills.

What is scraping? "Scraping" is using code or software tools to extract data automatically from web pages (or PDFs). Usually, this technique is used when copying and pasting data from one format into another one would be too time-consuming. Scraping often involves having the computer perform repetitive tasks, such as filling in forms, clicking buttons to get to sub-pages and performing the equivalent of "copying and pasting" information by means of code.

The aim is to transfer the information into a format through which it can be analysed, re-used and collated. The advantage of using code to scrape data is that the code can be reused, or even put on a timer to download data repeatedly. This is particularly useful if a data set is being added to for example on a daily basis.

Web scraping: tools for non-programmers While some of the more complex websites can only be scraped using custom-built code, there are a number of out-of-the-box tools that journalists can use to scrape sites without having to learn how to code. All the tools have online tutorials on how to use them. Here is a selection:

- **ScraperWiki** *http://scraperwiki.com/*
  Scraperwiki is an online tool and community linking programmers with journalists, who can request datasets to be scraped. Has an extensive library of scrapers and data, from the UK and internationally

- **Downthemall** *http://www.downthemall.net/*
  Firefox plugin that allows you to download large amounts of pages from a website

- **iMacros** *https://addons.mozilla.org/en-US/firefox/addon/imacros-for-firefox*
  Firefox plugin; rovides a number of templates for scraping and lets you 'record' HTML-based macros to extract data, fill in forms, and perform a host of other repetitive tasks. The iopus.com site (makers of imacros) have a host of tutorials on how to use the software.

- **OutWitHub** http://www.outwit.com/products/hub/
  Another Firefox extension. Outwithub's interface includes a number set data extraction presets such as data, images, emails, but also lets the user make custom adjustments The extracted data can be exported to CSV, HTML, Excel or SQL databases.

- **Needlebase** http://needlebase.com/
  Needlebase - recently acquired by Google - is useful for grabbing data stored on multiple pages and sub pages. The user can "teach" the tool by pointing and clicking through one or two examples and showing the it how the site is structured and which data fields you want to extract. Needle then follows this pattern to scrape data. Needle exports into Excel, CSV and XML. The tool can also be used to aggregate data from different sites and create mash-ups. Needlebase is quite pricey (it has a monthly pricing structure). There is a free version however, but it requires the scraped datasets to be public.

Programming options Scripting languages like Ruby or Python offer you a certain amount of customization that ready-made options on the web might not offer. Don't be afraid. Learning Ruby or Python is easier than you think. There are tons of resources out there, from online forums to books, and they will help you along the way. For example, O'Reilly produces *Learning Ruby* (the one with the giraffes on the front). Don't forget the NICAR-L listserv, populated by tons of people just like you who've probably faced the same obstacles in learning a scripting language.

Remember: Good programmers write good code. Great programmers steal great code. Lucky for you, there are quite a few smart people out there designing ready-made code that's easily implemented into your scripts. In Ruby, they are called gems, Python has packages. Think of them like plugins. These plugins allow you to accomplish incredibly complex tasks with just a few lines of code.

For example, Ryan uses a Ruby gem called *Firewatir*. It allows you to control Firefox from a script using a few lines like so:

```
Require 'firewatir'
browser = Watir:Browser.new
browser.goto("https://pcl.uscourts.gov/search")
browser.text_field(:name => "loginid").set "LOGIN"
browser.text_field(:name => "passwd").set "PASSWORD"
browser.button(:name => "faction").click
```

These six lines will open up Firefox, go to the PACER site, enter a login and password, then enter the site. Six lines.

So don't be afraid to dip your toe into programming, even if it's a bit at a time.

PDFscraping: tools for non-programmers It is possible to write code to extract data from PDFs using for example Ruby or Perl - two coding languages - but there are also a number of free or inexpensive, but powerful software tools that accomplish this task. Essentially, they do the

reverse of what a PDF writer tool does. Cracking open PDFs this way can be difficult time-consuming, but will not - and should not - deter a determined journalist.

- **AbbyFineReader** http://finereader.abbyy.com/
  Works well with different languages and scripts such as Cyrillic/Greek, with built-in OCR

  Docudesk UnPDF http://www.docudesk.com/deskunpdf_product_home.shtml
  Similar to Abbey, has OCR

Tips to bear in mind when using PDF scrapers

1. Make sure you eyeball the data and cross check it before and after scraping. As good as many of the PDF extraction tools are, they don't always capture everything.
2. Do integrity checks. You can sum columns, count the number of records, look for missing fields --- these are just a few ways you can test to make sure the data are accurate.

Ideas - Projects

1) http://www.newscientist.com/article/dn18806-revealed-pfizers-payments-to-censured-doctors.html (this one involved scraping this site: http://www.pfizer.com/responsibility/working_with_hcp/payments_report.jsp)
2) http://kansas.watchdog.org/5648/10-counties-voted-not-to-retain-kansas-supreme-court-justices/ (the scraping is explained here: http://www.franklincenterhq.org/2170/screen-scraping-mapping-in-r/)
3) http://data.baltimoresun.com/orioles-home-runs/
4) http://www.azcentral.com/news/articles/2010/05/02/20100502tax-rebate-errors.html (Said scraper Matt Wynn: "County said it couldn't possibly find out which homes were improperly receiving a tax credit. Further, they couldn't provide the databases so we could do our own analysis. Luckily, those apparently impossible databases were available online. So I ran a scraper of mammoth proportions that did the matches for us, and bam. Story.")

Further reading

Pro Publica Scraping for Journalism: A Guide for Collecting Data
http://www.propublica.org/nerds/item/doc-dollars-guides-collecting-the-data

Want to learn how to get started with Ruby?
http://www.ruby-lang.org

Want to learn how to get started with Python?
http://www.python.org/

Also, for other guides on web scraping, check out tipsheets at www.ire.org.