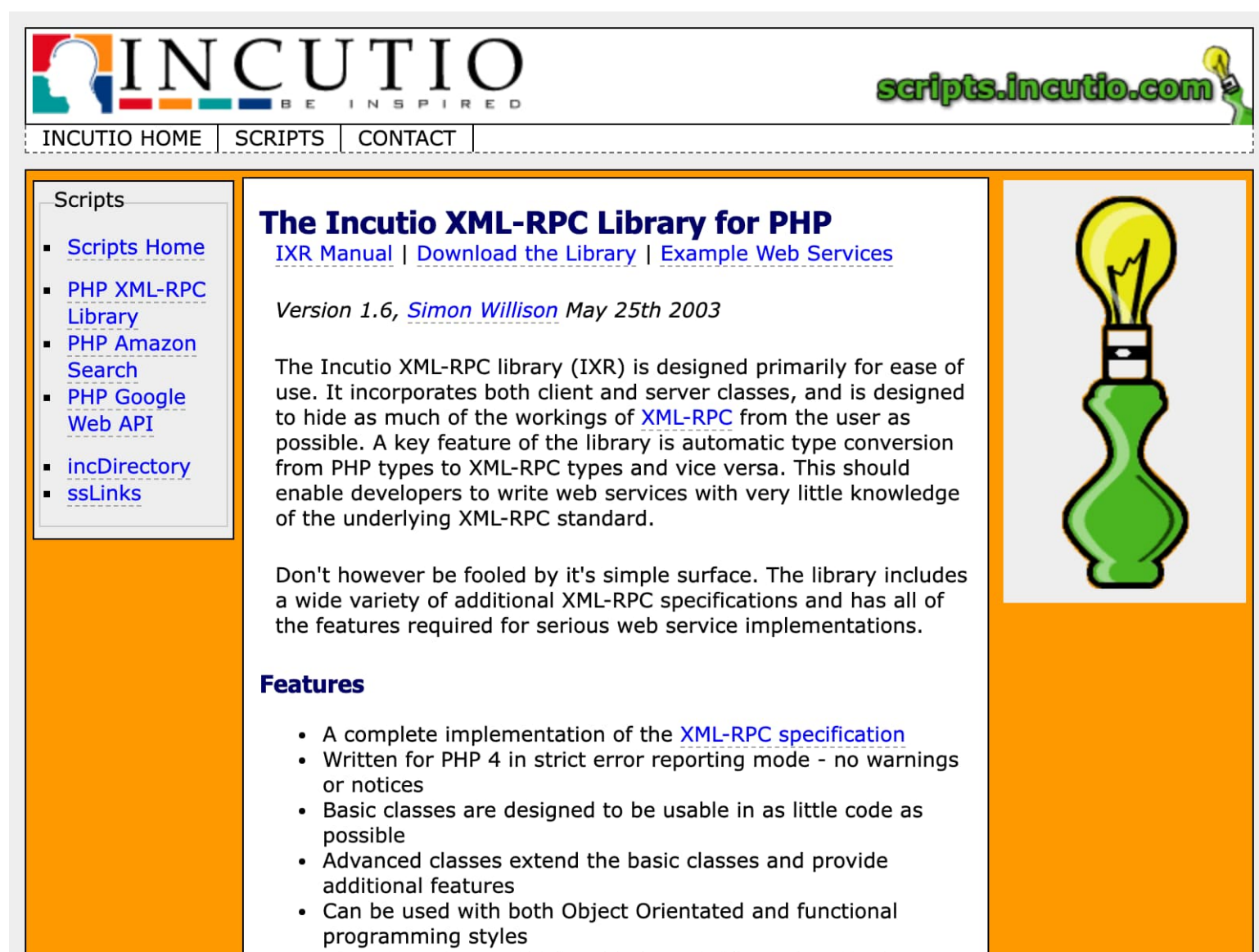# Twenty years of my blog

I started this blog on [June 12th 2002](#)—twenty years ago today! To celebrate two decades of blogging, I decided to pull together some highlights and dive down a self-indulgent nostalgia hole.

Some highlights

Some of my more influential posts, in chronological order.

- **[A new XML-RPC library for PHP](#)**—2nd September 2002

  I was really excited about XML-RPC, one of the earliest technologies for building Web APIs. IXR, the Incutio library for XML-RPC, was one of my earliest ever open source library releases. Here's [a capture of the old site](#).

  I've not touched anything relating to this project in over 15 years now, but it has lived on in both [WordPress](#) and [Drupal](#) (now only in Drupal 6 LTS).

  It's also been responsible for [at least one CVE vulnerability](#) in those platforms!

- **[getElementsBySelector()](#)**—25th March 2003

Andrew Hayward had posted a delightful snippet of JavaScript called [document.getElementsByClassName()](#)—like `document.getElementsByTagName()` but for classes instead.

Inspired by this, I built `document.getElementsBySelector()`—a function that could take a CSS selector and return all of the matching elements.

This ended up being very influential indeed! Paul Irish offers [a timeline of JavaScript CSS selector engines](#) which tracks some of what happens next. Most notably, `getElementsBySelector()` was part of John Resig's [inspiration](#) in creating the first version of jQuery. To this day, the jQuery source includes [this testing fixture](#) which is derived from [my original demo page](#).

I guess you could call `document.getElementsBySelector()` the original [polyfill](#) for [document.querySelectorAll()](#).

- **[I'm in Kansas](#)**—27th August 2003

In May 2003 Adrian Holovaty [posted about a job opportunity](#) for a web developer at at the Lawrence Journal-World newspaper in Lawrence, Kansas.

This coincided with my UK university offering a "year in industry" placement, which meant I could work for a year anywhere in the world with a student visa program. I'd been reading Adrian's blog for a while and really liked the way he thought about building for the web—we were big fans of Web Standards and CSS and cleanly-designed URLs, all of which were very hot new things at the time!

So I talked to Adrian about if this could work as a year-long opportunity, and we figured out how to make it work.

At the Lawrence Journal-Word Adrian and I decided to start using Python instead of PHP, in order to build a CMS for that local newspaper...

- **[Introducing Django](#)**—17th July 2005

... and this was the eventual outcome! Adrian and I didn't even know we were building a web framework at first—we called it "the CMS". But we kept having to solve new foundational problems: how should database routing work? What about templating? What's the best way to represent the incoming HTTP request?

I had left the Lawrence Journal-World in 2004, but by 2005 the team there had grown what's now known as Django far beyond where it was when I had left, and they got the go-ahead from the company to release it as open source (partly thanks to the example set by Ruby on Rails, which first released in August 2004).

In 2010 I wrote up [a more detailed history](#) of Django in a Quora answer, now mirrored to my blog.

- **[Finally powered by Django](#)**—15th December 2006

In which I replaced my duct-tape-and-mud PHP blogging engine with a new Django app. I sadly don't have the version history for this anymore (this was pre-git, I think I probably had it in Subversion or Mercurial somewhere) but today's implementation is still based on the same code, upgraded to Django 1.8 [in 2015](#).

That 2006 version did include a very pleasing Flickr integration to import my photos ([example on the Internet Archive](#)):

Simon Willison's Weblog

Thursday, 6th January 2005

France, Christmas 2004, a photoset
1:31 am / 31 photos

Photos from my Christmas in France. Including my dad's beautiful new horses. ⓪ 1:57 am

Big news... Six Apart and LiveJournal! (via) Brad's post on the buy-out, with a detailed FAQ. ⓪ 12:38 pm

Gate's Computer Crashes at CES. Sweet, sweet schadenfreude. ⓪ 2:13 pm

Chicago 2004, a photoset
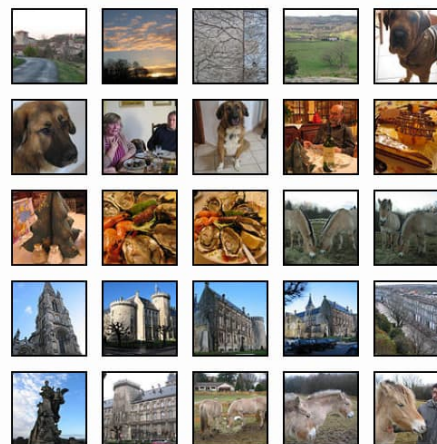Images from my trip to Chicago in September 2004.
3:40 pm / 52 photos

InfoWorld: Year of the enterprise Wiki (via) Wikis for business collaboration make a whole ton of sense. ⓪ 4:42 pm

SxSW 2004: Confirmed Speakers. Whoa! It's like an A-Z of cool

2005 » **January**

M	T	W	T	F	S	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

... 86 photos

- **How to turn your blog in to an OpenID**—19th December 2006

In late 2006 I got very, very excited about OpenID. I was convinced that Microsoft Passport was going to take over SSO on the internet, and that the only way to stop that was to promote an open, decentralized solution. I wrote posts about it, made screencasts (that one got 840 diggs! Annoyingly I was serving it from the Internet Archive who appear to have deleted it) and gave a whole bunch of conference talks about it too.

I spent the next few years advocating for OpenID—in particular the URL-based OpenID mechanism where any website can be turned into an identifier. It didn't end up taking off, and with hindsight I think that's likely for the best: expecting people to take control of their own security by chosing their preferred authentication provider sounded great to me in 2006, but I can understand why companies chose to instead integrate with a smaller, tightly controlled set of SSO partners over time.

- **A few notes on the Guardian Open Platform**—10th March 2009

In 2009 I was working at the Guardian newspaper in London in my first proper data journalism role—my work at the Lawrence Journal-World had hinted towards that a little, but I spent the vast majority of my time there building out a CMS.

In March we launched two major initiatives: the Datablog (also known as the Data Store) and the Guardian's Open Platform (an API that is still offered to this day).

The goal of the Datablog was to share the data behind the stories. Simon Rogers, the Guardian's data editor, had been collecting meticulous datasets about the world to help power infographics in the paper for years.

The new plan was to share that raw data with the world.

We started out using Google Sheets for this. I desperately wanted to come up with something less proprietary than that—I spent quite some time experimenting with CouchDB—but Google Sheets was more than enough to get the project started.

Many years later my continued mulling of this problem formed part of the inspiration for my creation of Datasette, a story I told in my 2018 PyBay talk How to Instantly Publish Data to the Internet with Datasette.

- **Why I like Redis**— 22nd October 2009

I got interested in NoSQL for a few years starting around 2009. I still think Redis was the most interesting new piece of technology to come out of that whole movement—an in-memory data structure server exposed over the network turns out to be a fantastic complement for other data stores, and even though I now default to PostgreSQL or SQLite for almost everything else I can still find problems for which Redis is a great solution.

In April 2010 I gave a three hour Redis tutorial at NoSQL Europe which I wrote up in Comprehensive notes from my three hour Redis tutorial.

- **Node.js is genuinely exciting**— 23rd November 2009

In December 2009 I found out about Node.js. As a Python web developer I had been following the evolution of Twisted with great interest, but I'd also run into the classic challenge that once you start using event-driven programming almost every library you might want to use likely doesn't work for you any more.

Node.js had server-side event-driven programming baked into its very core. You couldn't accidentally make a blocking call and break your event loop because it didn't ever give you the option to do so!

I liked it so much I switched out my talk for Full Frontal 2009 at the last minute for one about Node.js instead.

I think this was an influential decision. I won't say who they are (for fear of mis-representing or mis-quoting them), but I've talked to entrepreneurs who built significant products on top of server-side JavaScript who told me that they heard about Node.js from me first.

- **Crowdsourced document analysis and MP expenses**—20th December 2009

This was my biggest data journalism project at the Guardian.

The UK government had finally got around to releasing our Member of Parliament expense reports, and there was a giant scandal brewing about the expenses that had been claimed. We recruited our audience to help dig through 10,000s of pages of PDFs to help us find more stories.

The first round of the MP's expenses crowdsourcing project launched in June, but I was too busy working on it to properly write about it! Charles Arthur wrote about it for the Guardian in The breakneck race to build an application to crowdsource MPs' expenses.

In December we launched round two, and I took the time to write about it properly.

Here's a Google Scholar search for guardian mps expenses—I think it was pretty influential. It's definitely one of the projects I'm most proud of in my career so far.

- **WildlifeNearYou: It began on a fort...**—12th January 2010

In October 2008 I participated in the first [/dev/fort](#)—a bunch of nerds rent a fortress (or similar historic building) for a week and hack on a project together.

Following that week of work it took 14 months to add the "final touches" before putting the site we had built live (partly because I insisted on implementing OpenID for it) but in January 2010 we finally went live with WildlifeNearYou.com (sadly no longer available). It was a fabulous website, which crowdsourced places that people had seen animals in order to answer the crucial question "where is my nearest Llama?".

Here's [what it looked like](#):

Although it shipped after the Guardian MP's expenses project most of the work on WildlifeNearYou had come before that—building WildlifeNearYou (in Django) was the reason I was confident that the MP's expenses project was feasible.

- [**Getting married and going travelling**](#)—21st June 2010

One June 5th 2010 I married Natalie Downe, and we both quit our jobs to set off travelling around the world and see how far we could get.

We got as far as Casablanca, Morocco before we accidentally launched a startup together: Lanyrd, launched in August 2010. "Sign in with Twitter to see conferences that your friends are speaking at, attending or tracking, then add your own events."

We ended up spending the next three years on this: we went through Y Combinator, raised a sizable seed round, moved to London, hired a team and shipped a LOT of features. We even managed to ship some features that made the company money!

This also coincided with me putting the blog on the back-burner for a few years.

Here's an early snapshot:

In 2013 we sold Lanyrd to Eventbrite, and moved our entire team (and their families) from London to San Francisco. It had been a very wild ride.

Sadly the site itself is no longer available: as Eventbrite grew it became impossible to justify the work needed to keep Lanyrd maintained, safe and secure. Especially as it started to attract overwhelming volumes of spam.

Natalie told the full story of Lanyrd on her blog in September 2013: Lanyrd: from idea to exit—the story of our startup.

- **Scraping hurricane Irma**—10th September 2017

  In 2017 hurricane Irma devastated large areas of the Caribbean and the southern USA.

  I got involved with the Irma Response project, helping crowdsource and publish critical information for people affected by the storm.

  I came up with a trick to help with scraping: I ran scrapers against important information sources and recorded the results to a git repository, in order to cheaply track changes to those sources over time.

  I later coined the term "Git scraping" for this technique, see my series of posts about Git scraping over time.

- **Getting the blog back together**—1st October 2017

  Running a startup, and then working at Eventbrite afterwards, had resulted in an almost 7 year gap in blogging for me. In October 2017 I decided to finally get my blog going again. I also back-filled content for the intervening years by scraping my content from Quora and from Ask Metafilter.

  If you've been meaning to start a new blog or revive an old one this is a trick that I can thoroughly recommend: just because you initially wrote something elsewhere doesn't mean you shouldn't repost it on a site you own.

- **Recovering missing content from the Internet Archive**—8th October 2017

  The other step in recovering my old blog's content was picking up some content that was missing from my old database backup. Here's how I pulled in that content by scraping the Internet Archive.

- **Implementing faceted search with Django and PostgreSQL**— 5th October 2017

  I absolutely love building faceted search engines. I realized a while ago that most of my career has been spent applying the exact same trick—faceted search—to different problem spaces. WildlifeNearYou offered faceted search over animal sightings. MP's expenses had faceted search across crowdsourced expense analysis. Lanyrd was faceted search for conferences.

  I implemented faceted search for this blog on top of PostgreSQL, and wrote about how I did it.

- **Datasette: instantly create and publish an API for your SQLite databases**—13th November 2017

  I shipped the first release of simonw/datasette in Nevember 2017. Nearly five years later it's now my number-one focus, and I don't see myself losing interest in it for many decades to come.

  Datasette was inspired by the Guardian Datablog, combined with my realization that Zeit Now (today called Vercel) meant you could bundle data up in a SQLite database and deploy it as part of an exploratory application almost for free.

My blog has [284 items tagged datasette](#) at this point.

- **Datasette Facets**—20th May 2018

  Given how much I love faceted search, it's surprising it took me until May 2018 to realize that I could bake them into Datasette itself—turning it into a tool for building faceted search engines against any data. It turns out to be my ideal solution to my favourite problem!

- **Documentation unit tests**—28th July 2018

  I figured out a pattern for using unit tests to ensure that features of my projects were covered by the documentation. Four years later I can confirm that this technique works *really well*—though I wish I'd called it Test-driven documentation instead!

- **Letterboxing on Lundy**—18th September 2018

  A brief foray into travel writing: Natalie and I spent a few days staying in a small castle on the delightful island of Lundy off the coast of North Devon, and I used it as an opportunity to enthuse about letterboxing and the Landmark Trust.

- **sqlite-utils: a Python library and CLI tool for building SQLite databases**—25th February 2019

Datasette helps you explore and publish data stored in SQLite, but how do you get data into SQLite in the first place?

[sqlite-utils](#) is my answer to that question—a combined CLI tool and Python library with all sorts of utilites for working with and creating SQLite databases.

It recently had its [100th release](#)!

- **[I commissioned an oil painting of Barbra Streisand's cloned dogs](#)**—7th March 2019

  Not much I can add that's not covered by the title. It's a really good painting!

- **[My JSK Fellowship: Building an open source ecosystem of tools for data journalism](#)**—10th September 2019

  In late 2019 I left Eventbrite to join the JSK fellowship program at Stanford. It was an opportunity to devote myself full-time to working on my growing collection of open source tools for data journalism, centered around Datasette.

  I jumped on that opportunity with both hands, and I've been mostly working full-time on Datasette and [associated projects](#) (without being paid for it since the fellowship ended) ever since.

- **Weeknotes: ONA19, twitter-to-sqlite, datasette-rure**—13th September 2019

  At the start of my fellowship I decided to publish weeknotes, to keep myself accountable for what I was working on now that I didn't have the structure of a full-time job.

  I've managed to post them roughly once a week ever since—[128 posts and counting](#).

  I absolutely love weeknotes as a format. Even if no-one else ever reads them, I find them really useful as a way to keep track of my progress and ensure that I have motivation to get projects to a point where I can write about them at the end of the week!

- **Using a self-rewriting README powered by GitHub Actions to track TILs**—20th April 2020

  In April 2020 I started publishing TILs—Today I Learneds—at til.simonwillison.net.

  The idea behind TILs is to dramatically reduce the friction involved in writing a blog post. If I learned something that was useful to me, I'll write it up as a TIL. These often take less than ten minutes to throw together and I find myself referring back to them all the time.

  My main blog is a Django application, but my TILs run entirely using Datasette. You can see how that all works in the simonw/til GitHub repository.

- **Using SQL to find my best photo of a pelican according to Apple Photos**—21st May 2020

  Dogsheep is my ongoing side project in which I explore ways to analyze my own personal data using SQLite and Datasette.

  dogsheep-photos is my tool for extracting metadata about my photos from the undocumented Apple Photos SQLite database (building on osxphotos by Rhet Turnbull). I had been wanting to solve the photo problem for years and was delighted when osxphotos provided the capability I had been missing. And I really like pelicans, so I celebrated by using my photos of them for the demo.

- **Git scraping: track changes over time by scraping to a Git repository**—9th October 2020

  If you really want people to engage with a technique, it's helpful to give it a name. I defined Git scraping in this post, and I've been promoting it heavily ever since.

  There are now 275 public repositories on GitHub with the `git-scraping` topic, and if you sort them by recently updated you can see the scrapers on there that most recently captured some new data.

- **Personal Data Warehouses: Reclaiming Your Data**—14th November 2020

  I gave this talk for GitHub's OCTO (previously Office of the CTO, since rebranded to GitHub Next) speaker series.

  It's the Dogsheep talk, with a better title (thanks, Idan!) It includes a full video demo of my personal Dogsheep instance, including my dog's Foursquare checkins, my Twitter data, Apple Watch GPS trails and more.

  I also explain why I called it Dogsheep: it's a devastatingly terrible pun on Wolfram.

  I'm frustrated when information like this is only available in video format, so when I give particularly information-dense talks I like to turn them into full write-ups as well, providing extra notes and resources alongside screen captures from the talk.

  For this one I added a custom template mechanism to my blog, to allow me to break out of my usual entry page design.

- **Trying to end the pandemic a little earlier with VaccinateCA**—28th February 2021

In February 2021 I joined the VaccinateCA effort to try and help end the pandemic a little bit earlier by crowdsourcing information about the best places to get vaccinated. It was a classic match-up for my skills and interests: a huge crowdsourcing effort that needed to be spun up as a fresh Django application as quickly as possible.

[Django SQL Dashboard](#) was one project that spun directly out of that effort.

- **[The Baked Data architectural pattern](#)**—28th July 2021

  My second attempt at coining a new term, after Git scraping: Baked Data is the name I'm using for the architectural pattern embodied by Datasette where you bundle a read-only copy of your data alongside the code for your application, as part of the same deployment. I think it's a really good idea, and more people should be doing it.

- **[How I build a feature](#)**—12th January 2022

  Over the years I've evolved a processes for feature development that works really well for me, and scales down to small personal projects as well as scaling up to much larger pieces of work. I described that in detail in this post.

Picking out these highlights wasn't easy. I ended up setting myself a time limit (to ensure I could put this post live within a minute of midnight UTC time on my blog's 20th birthday) so there's plenty more that I would have liked to dig up.

My [tags index page](#) includes a 2010s-style word cloud that you can visit if you want to explore the rest of my content. Or use the faceted search!

A few more project release highlights:

- [GraphQL in Datasette with the new datasette-graphql plugin](#)—7th August 2020
- [git-history: a tool for analyzing scraped data collected using Git and SQLite](#)—7th December 2021
- [shot-scraper: automated screenshots for documentation, built on Playwright](#)—10th March 2022
- [Django SQL Dashboard](#)—10th May 2021
- [Datasette Desktop—a macOS desktop application for Datasette](#)—8th September 2021
- [Datasette Lite: a server-side Python web application running in a browser](#)—4th May 2022

Evolution over time

I started my blog in my first year of as a student studying computer science at the University of Bath.

You can tell that Twitter wasn't a thing yet, because I wrote [107 posts in that first month](#). Lots of links to other people's blog posts (we did a lot of that back then) with extra commentary. Lots of blogging about blogging.

That first version of the site was hosted at `http://www.bath.ac.uk/~cs1spw/blog/`—on my university's student hosting. Sadly the Internet Archive doesn't have a capture of it there, since I moved it to `http://simon.incutio.com/` (my part-time employer at the time) in September 2002. Here's [my note](#) from then about rewriting it to use MySQL instead of flat file storage.

This is the earliest capture I could find [on the Internet Archive](#), from June 2003:

Simon Willison's Weblog

PHP, Python, CSS, XML and general web development

3rd June

Using bookmarklets to experiment with CSS

I'm in the middle of a whole bunch of exams at the moment, but here's a quick tip that should make experimenting with and learning CSS a great deal easier. It involves bookmarklets. If you haven't seen them before, bookmarklets are bookmarks that embed javascript; when you click the bookmark, the javascript is executed in the context of the currently loaded page. What that means is that in a suitably advanced browser bookmarklets can be used to modify pages, analyse their structure and do a whole host of other useful things.

A fair number of bookmarklets work in both IE and Gecko engine browsers, but the really interesting ones (at least as far as learning CSS is concerned) tend to be Gecko only. If you're serious about learning CSS it's well worth getting a Gecko browser at any rate - Gecko's CSS support is more standards compliant than anything else currently available, so I recommend designing for the standards with Gecko and tweaking to support IE afterwards. Some people advocate designing for IE because it has buggier support, but once you've seen the tools available for Gecko I think you'll agree it really is a superb development platform. If you don't have access to a Gecko engine browser (that means Mozilla, Netscape 7, Camino, or Phoenix/Firebird) go and grab Firebird - it's only a 6 MB download and is available for most major platforms.

Now that you're tooled up with a good browser, drag the following links on to

Full entry on Using bookmarklets to experiment with CSS.

By November 2006 I had redesigned from orange to green, and started writing Blogmarks—the name I used for small, bookmark-style link posts. I've collected 6,304 of them over the years!

Simon Willison's Weblog

PHP, Python, CSS, XML and general web development

Tamarin   *[3 days, 18 hours ago]*

On Tuesday, the Mozilla Foundation and Adobe announced the Tamarin project, an open-source ECMAScript virtual machine based on the ActionScript engine used by Flash Player 9.

Frank Hecker's overview of what this means is useful, but the Tamarin source code itself provides this interesting piece of historical insight:

> AVM+ is the ActionScript Virtual Machine
>
> AVM+ offers an order of magnitude performance increase over the "Classic AVM" in Flash Player 7. Our performance target is 10X.
>
> AVM+ implements ActionScript 3.0, the new version of the ActionScript language that is compliant with the ECMAScript Edition 4 standard.
>
> AVM+ is also built for modularity. It will be part of the Flash Player, but is a self-contained module which can be incorporated into other programs with ease. It may also be submitted to the ECMA standards organization as a reference implementation of ECMAScript Edition 4.

By 2010 I'd reached more-or-less my current purple on white design, albeit with the ability to sign in with OpenID to post a comment. I dropped comments entirely when I relaunched in 2017—constantly fighting against spam comments makes blogging much less fun.

Simon Willison's Weblog

On [javascript](), [python](), [django](), [openstreetmap](), [node](), [...]()   [Search]

Recent entries

[Getting married and going travelling]() one month ago

It's been a busy month. On Saturday the 5th of June I married the wonderful [Natalie Downe]() in a beautiful ceremony at Roedean School in Brighton. The reception had [owls](), [cheese](), [a ferret](), [a golden eagle](), amazing Turkish food, [Jewish chair dancing]() and [lovely guests](). It was the happiest day of my life.

Elsewhere

22nd July 2010

[Jeremiah Grossman: I know who your name, where you work, and live](). Appalling unfixed vulnerability in Safari 4 and 5 —if you have the "AutoFill web forms using info from my Address Book card" feature enabled (it's on by default) malicious JavaScript on any site can steal your name, company, state and e-mail address —and would be able to get your phone number too if there wasn't a bug involving strings that start with a number. The temporary fix is to disable that preference. ②

21st July 2010

[What to do when PyPI goes down](). My deployment scripts tend to rely on PyPI these days (they install dependencies in to

The source code for the current iteration of my blog is available [on GitHub]().

Taking screenshots of the Internet Archive with shot-scraper

Here's how I generated the screenshots in this post, using [shot-scraper]() against the Internet Archive but with a line of JavaScript to hide the banner the display at the top of every archived page:

```
shot-scraper 'https://web.archive.org/web/20030610004652/http://simon.incutio.com/' \
  --javascript 'document.querySelector("#wm-ipp-base").style.display="none"' \
  --width 800 --height 600 --retina
```

mgdlbp on Hacker News [pointed out]() that you can instead add `if_` to the date part of the archive URLs to hide the banner, like this:

```
shot-scraper 'https://web.archive.org/web/20030610004652if_/http://simon.incutio.com/' \
  --width 800 --height 600 --retina
```

Posted [12th June 2022]() at 10:59 pm · Follow [@simonw]() on Twitter

blogging 31

Next: [Weeknotes: datasette-socrata, and the last 10%...]()

Previous: [A tiny web app to create images from OpenStreetMap maps]()

It's the 20th anniversary of the launch of my blog, https://t.co/gimNW2jc1b!

I decided to celebrate by trying to pull together some highlighted content from the past 20 years https://t.co/57vFA22bdH — Simon Willison (@simonw) June 12, 2022