Old loves, new views...

Author : Tim Sherratt

Tagged as : archives, recordsearch, userscripts, zotero

Date : August 29, 2012

I'm deeply in love with the collections of the National Archives of Australia. They move me, they inspire me, they make me want to *do something*. How do I express my love? I've written stories about things like atomic bombs, progress, astronomy and weather forecasting — pursuing lives and events documented in the Archives' rich holdings. I work on projects like Invisible Australians, hoping to bring the compelling remnants of the White Australia Policy to broader public attention. And I build things. I make tools that help other people explore, understand and use the Archives. I do this because these riches *need* to be used. They need to be shared. They need to be part of the fabric of our lives.

A few years ago I created a little script for Firefox that put a fresh face on the display of digitised records in the National Archives' RecordSearch database. It's publicly available and has been installed more than 500 times. Demonstrating this script at the 'Doing our bit' Build-a-thon a few weeks ago made me realise again both how useful it was and how much work it still needed.

One of the most exciting features when I first created the script was the ability to display the records on a '3D wall', courtesy of a Firefox plugin called CoolIris. But CoolIris uses Flash and is no longer being supported. Time for a new approach.

Say hello to the **Archives Viewer** (naming things isn't really one of my strengths). Instead of rewriting my existing script I decided to create a completely new web application. Why? Mainly because it gave me a lot more flexibility. I could also make use of a variety of existing tools and frameworks like Django, Bootstrap, Isotope and FancyBox. Standing upon the code of giants, I had the whole thing up and running in a single weekend. The code is available on GitHub.

What does it do? Simply put, just feed the **Archives Viewer** the barcode of a digitised file in RecordSearch and it grabs the metadata and images and displays them in a variety of useful ways. It's really pretty simple, both in execution and design.

Yep, there's a wall. It's not quite as spacey and zoom-y as the CoolIris version, but perhaps that's a good thing. It's just a flat wall of page image thumbnails with a bit of lightbox-style magic thrown in. But when I say *just*, well... look for yourself. There's something a bit magical about seeing all the pages of a file at once, taking in their shapes and colours as well as their content. This digital wall provides a strangely powerful reminder of the physical object.

working for the triumph of content over form, ideas over control, people over systems
http://discontents.com.au

Of course you can also view the file page by page if you want. Printing is a snap — just type in any combination of pages or page ranges and hit the button. The images and metadata are assembled ready to print. No more wondering 'which file did this print out come from?'.

But perhaps the most important feature is that each page has it's own unique, persistent url. Basic stuff, but oh, so important. With a good url you can share and cite. Find something exciting? Tell the world about it! I've included your typical social media share buttons to help you along.

> 1891 draft of the Australian Constitution — with doodles. [bit.ly/Qus4D5](bit.ly/Qus4D5)
>
> — Tim Sherratt (@wragge) August 23, 2012

One disadvantage over the original userscript is that the viewer isn't directly linked to RecordSearch. You probably don't want to have to cut and paste the barcode every time you view a file. So I've also created a couple of connectors that ummm… connect things up.

The first connector is just a bookmarklet. A bookmarklet is just a little piece of javascript code disguised as a browser bookmark. Just drag this link — Archives Viewer — to your browser's bookmark toolbar. Then when you're on the item page of a digitised file in RecordSearch, just click the bookmarklet and you'll be instantly transported to the wall.

The second connector is a bit smarter. It's an enhanced version of another userscript I wrote to display the number of pages in a digitised file. It still does that, but now it also rewrites the links to the digitised files so that they automatically open in the Archives Viewer. It's a bit harder to install. You need Chrome or Firefox and the add-ons Greasemonkey (for Firefox) or Tampermonkey (for Chrome). Then just go to the userscript page and hit the big 'Install' button.

You might be wondering about Zotero (at least I hope you are). My Zotero-RecordSearch translator lets you capture page images and metadata direct to your own research database, so what happens when you're transported across to the Archives Viewer? Never fear, I've written a new translator that lets you save pages as you could in RecordSearch. Even better, you get a persistent, context-enriched url, and the ability to capture multiple pages at once. Yippee!

But that's not quite all. Buried within the pages is some lovely Linked Open Data. To be truthful, it's not really very 'linked' yet, but it does expose the basic metadata in a machine-readable form, borrowing from the vocabularies of projects like Locah and the Archival Ontology. It's an experiment, as is the Archives Viewer itself. We can learn by doing.

> Makin' triples from dhistory Archives Viewer RDFa: file – bit.ly/PKmjmJ & page bit.ly/PKmm1O
>
> — Tim Sherratt (@wragge) August 26, 2012

I've given quite a few talks over recent times encouraging people to take up their tools and start hacking away at the digital collections of our cultural institutions. Yes, I admit it, I'm an impatient

[historian](#) (and a grumpy one at that). But it's also because I think it's important that we recognise that access is [never just something you're given](#). It's something that we make through our stories, our projects, and our tools. It's something that's grounded in respect and powered by love.

Share this:

- [Click to email this to a friend (Opens in new window)](#)
- [Click to print (Opens in new window)](#)
- [Click to share on Twitter (Opens in new window)](#)
- [Share on Facebook (Opens in new window)](#)
- [Click to share on Google+ (Opens in new window)](#)
-