Exploring GLAM data

Tim Sherratt, November 2021
Notes from a key story presentation at ResBaz Queensland 2021.

I'm coming to you today from beautiful nipaluna, on the unceded lands of the muwinina people. I'd like to pay my respects to their elders – past, present, and emerging.

If you'd you'd like to play along, the slides are here… https://slides.com/wragge/resbaz-2021

Today I'm going to explore some of the possibilities of GLAM data. But given that this is a key story, I thought I'd take you on a bit of a biographical journey before getting stuck into some of the more technical stuff.

But first let's unpack our acronyms, GLAM is:

- galleries
- libraries,
- archives
- and museums

Cultural heritage institutions if you like. I've been working between academia and the cultural heritage sector for more than thirty years.

In 1994, I was working for a small organisation called the Australian Science Archives Project when I first became aware of this thing called the 'World Wide Web'. As a small organisation with limited resources, we realised we could use the web to communicate with researchers – to help them find and use archival materials relating to the history of Australian science. We created the first website about archives in Australia, and one of the first Australian history sites.

In the years that followed I developed a lot of websites, and taught myself the basics of coding in PHP and Javascript. But it wasn't until about 2007, that I realised how computational methods could be used to manipulate cultural heritage data – to create new access points, new ways of seeing.

Seeing differently

I was working for the National Archives of Australia where a new exhibition was being planned on World War I. The Archives had recently digitised over 370,000 WWI service records, and we were thinking about how these might be used in a digital resource to accompany the physical exhibition.

These records are held in Series B2455, and here's how they look in the National Archives' online database, RecordSearch. If we focus on the file titles, we see that contain specific pieces of information – names, service numbers, and places of birth and enlistment, separated by colons. It's really structured data, of the kind you might find in a CSV file or spreadsheet. This was a deliberate, and farsighted, choice by the NAA staff – creating descriptions that would be open to future uses, whatever they might be.

So I suggested that grab all of the file titles, extract all the data, geolocate the place names, and make a map interface – so that users could find service records, not by searching for a name, but by simply clicking on a map.

Of course it wasn't quite that simple – the data wasn't always consistent, and place names couldn't always be matched to current locations. I think I counted 12 or thirteen different spellings of the word 'Lieutenant'. But GLAM data represents human activity, not the output of machines, so there'll always be these sorts of challenges.

In the end we created a site that was quite innovative. Both in its use of maps to explore archival collections, and in some of the user engagement features we added. I even got to demonstrate it to the PM.

But to get the interface running in the web browsers of the time, we had to make many compromises. We could only show a limited number of markers or else Internet Explorer would explode. So we had to split the results by state, and cluster the markers into groups. Why was this a problem?

At some point I decided to look at the complete dataset by loading it into Google Earth. This is what I saw.

Markers… everywhere. As I zoomed in, the names resolved, and they were just… everywhere. All of those little towns sent their young people to fight and die. It was a different way of seeing the impact of the war.

I left the Archives not long after, but I continued to work with their data. I taught myself enough Python to be dangerous and built a screen scraper to extract structured data from RecordSearch – this meant I could download metadata and images from thousands of files. The scraper still works, in fact I gave it a major overhaul earlier this year.

My partner, Kate Bagnall, and I were particularly interested in records relating to the administration of the White Australia Policy. Kate is a historian of Chinese Australia and has worked with these records extensively. They include special certificates that non-white residents had to carry if they travelled overseas. Without this documentation, they could be subjected to the infamous Dictation Test on their return, and refused entry. There are many thousands of these certificates in the Archives.

In 2011, I fired up my scraper and downloaded all the available images from one series of these documents – I ended up with 12,502 pages. I then used a facial detection script to locate the portrait photos, and crop them from the page images. After weeding out the false positives, we had a collection of more than 7,000 faces. We displayed them using the simplest means possible – on one long, scrolling wall.

This was the Real Face of White Australia. I upgraded the underlying technologies recently, but left the design much the same. Even after all these years, scrolling through the seemingly endless wall of faces is powerful, and disturbing.

GLAM data can help us to see differently – to feel differently.

Asking new questions

The first beta version of the National Library's digitised newspaper project was released in 2008. A year later it merged into their new discovery service, Trove. Trove's digitised newspapers have fundamentally changed research in a number of humanities disciplines, particularly history. Trove delivers metadata, page images, and the OCRd text extracted from those images. Importantly though, this data is segmented into individual articles, not pages, or issues. When you couple article-level OCR with a good search engine, you have a way of looking beyond established historical narratives and observing the individual fragments of lived experience – the small stories.

But what about the big picture? I started poking around inside Trove about 2010. Once again, I created a screen scraper to extract structured data from the web interface. Using this data, I started to create visualisations of search queries, showing the number of matching articles per year. Instead of a list of search results, I could *see* everything.

This was an early version of a tool I called QueryPic. It's been through many iterations, and I created a completely new version earlier this year. There are many caveats and qualifications to apply to these sorts of visualisations – I tend to say that they're not arguments, but they can help you frame new questions. They highlight possible patterns and trends – shifts in language, changes in technology, the impact of specific events. And as you can see from these examples, you can also compare search terms.

QueryPic enables you to zoom out of the search interface, but once you have access to the underlying data, you can also zoom in – analysing in detail the contents and contexts of individual articles. I created a harvester to help researchers build custom datasets of metadata and text – again it's been through many iterations, but is still in service. You can use it to harvest thousands of articles. Indeed, in one early experiment examining changes in the content of newspaper front pages, I harvested details of 4 million articles.

- front pages
- immigrants

In 2012, Trove released a public API, or Application Programming Interface. APIs deliver structured data directly upon request – no screen scraping necessary. This meant I could update my tools, but it also opened up a new range of possible integrations and analyses. Instead of just a website, it became a platform – something to build upon.

The nature of access

What happens when you turn something like QueryPic back onto Trove itself? If we don't use a search query to limit our results, what we see are the total number of digitised articles in Trove per year. This is a representation of the complete corpus – what we're searching when we type stuff in the search box. I've been creating these sorts of charts at irregular intervals over the years and they show some interesting changes. Between 2011 and 2014 we can see a dramatic increase in the number of articles from the period around WWI. Why? In the lead up to the centenary of the war it was decided to focus digitisation dollars on newspapers from that era.

The impact of that decision can still be seen today.

Of course priorities have to be set, decisions about funding have to be made. The point is that these decisions shape online collections, they help construct what we mean by 'access'. However, the effects of these priorities are rarely exposed through the standard interfaces we use to search collections. We only see them when we look at the data underneath.

One thing that's interested me over a number of years are the files in the National Archives of Australia that we're **not** allowed to see. Under the Archives Act, government files more than 20 years old are opened to the public. However, before they're released, they go through a process known as 'access examination' to make sure there's nothing in them that might, for example, damage national security, or infringe on individual privacy. Most files are made fully open, some have pages removed or redactions applied. A much smaller number are withheld completely – in RecordSearch, they are assigned the access status of 'Closed'. The files are not available, but metadata about them is.

In 2016, I harvested the details of all the 'closed' files from RecordSearch and created an interface where we could see what we couldn't see. I've been repeating this harvest on or around the first of January each year to see what changes. It's a way of examining access not as a set of rules laid down in the Archives Act, but as a changing historical process.

As I mentioned, some files are opened, but with words and phrases redacted. I wondered if it might be possible to quantify the scale of redaction. So, of course, I downloaded many thousands of page images from publicly available ASIO surveillance files, and developed a redaction finder.

However, I got a bit distracted from the idea of quantification by the aesthetic qualities of the redactions themselves. (And yes you can buy your own # redactionart scarf).

This was particularly so when I discovered these guys lurking amidst the ASIO files. Yet another example of how the construction of access is a very human process.

Creating pathways

But online access is not just something delivered to a grateful audience, it has to be taken. It's not just determined by the interfaces we build, or the datasets we publish, but also by the skills and confidence of those who make might use of them.

What's the point of an API if the people who might benefit from it most don't know how to use it, or just don't see the point of it?

More and more GLAM data is becoming available:

- metadata
- text
- images
- born digital
- transcribed etc

Lots of rich and interesting data to explore – but how do we help people make use of it?

Events like ResBaz, of course, make an important contribution. As do training programs like Software Carpentry, or sites like Programming Historian. But the work of finding potential users, and exposing them to the possibilities of GLAM data is ongoing and evolving.

That's why I spend a lot of my time now developing the GLAM Workbench.

The GW includes the latest versions of tools that I've created over the last 10 years or so – things like QueryPic and the harvesters for Trove newspapers and RecordSearch. But it's growing all the time. Not just a collection of tools, but also tutorials, examples, hacks, and pre-harvested datasets.

Has resources relating to a variety of GLAM collections…

The GW makes use of Jupyter notebooks, and understanding what Jupyter makes possible was really a turning point for me. It offers *me* easier ways of maintaining, sharing, and documenting the sorts of tools I create. But it also helps users overcome some of those initial barriers that make digital research seem all too hard. Jupyter notebooks can build confidence by simply allowing users to try things in their browser – to try running a piece of code, querying an API, or building a visualisation.

This has helped me start to think of the GLAM Workbench not just as a collection of tools, but as a series of possible pathways.

For example…

The how and the why

So a GLAM organisation creates a public API that lets users explore collection data… FANTASTIC! But why would the researchers bother? What can they do with the API that they couldn't do before? What's in it for them?

If we want researchers to spend some of their valuable time exploring GLAM data, then we have to illustrate the 'why' as well as the 'how'.

Jupyter notebooks combine text and code into what's been called a 'computational narrative' that runs in your web browser. So if, for example, you're documenting how an API works, you can not only talk about things like endpoints and parameters, you can provide live code snippets that make real queries, return real data, and answer real questions.

The GW includes an introductory 'getting started' notebook that demonstrates how notebooks themselves work. But it does this by retrieving and visualising real data from the National Museum of Australia's collection API.

In the NMA section of the GW, there are more notebooks that explore in detail the sorts of data that's available from the API, and how you can use it to visualise the NMA's collections over time and space. The API is introduced, not as a series of technical specifications, but through the sorts questions it makes possible.

The same applies to the other collections APIs that are explored in the GW:

- Trove
- DigitalNZ
- Te Papa
- Museums Victoria
- and a variety of web archives

And of course it's not just happening in the GW. A number of GLAM organisations around the world are recognising the value of Jupyter notebooks in supplementing conventional forms of documentation — making users aware of some of the possibilities embedded within their data.

Jupyter notebooks can be both tool and tutorial. Users can learn about a new GLAM data source not by just reading about it, but by undertaking real research tasks.

Places to start & places to go

If someone arrives at the GLAM Workbench and finds it all a bit too confusing and scary, I suggest they start by playing with a tool called the GLAM CSV Explorer. It's a web app that pulls together hundreds of datasets shared by GLAM organisations though open data portals. You just select a dataset from the dropdown list, and the tool will analyse the contents and build a series of visualisations. It gives you a peek inside the dataset, and helps you to think about the research possibilities.

The GLAM CSV Explorer runs as a web app, but it's actually a Jupyter notebook. Jupyter notebooks can be viewed and used in different ways – for example, as slideshows, dashboards, or web apps. This means I can create multiple versions of tools aimed at users with different levels of skill or confidence.

QueryPic is, for example, available as a simple web app – no code to be seen – just fill in a couple of boxes and click on the button. But what does this simplicity hide? Researchers should be encouraged to ask critical questions of the tools and interfaces we create. To help unpack some of the assumptions underlying QP I've created another notebook that demonstrates how it uses search facet data from the Trove API. As a researcher's skills and questions develop, they can analyse and modify the code. They can move beyond the interface. They can follow their questions.

Zooming in & zooming out

I've already described how using QueryPic and the Trove Newspaper Harvester you can zoom out to view a complete set of search results, then zoom back in to analyse features of interest. The Harvester like QueryPic, is available as a web app, so with just a couple of clicks you can generate your own research dataset, with metadata, OCRd text, and even images. Once you have your dataset, you could, for example, load all the OCRd text into a text analysis program, like Voyant Tools, to examine the language in detail.

Harvesting bulk data, creating custom datasets for in-depth analysis – these are common tasks that you might want to apply to a range of collections and data types. There are a wide variety of examples in the GLAM Workbench.

You can, for example:

- Harvest the OCRd text from Trove's digitised books and journals. As well as newspapers, there are journals like a hundred years worth of *The Bulletin* that also have text available.
- Harvest the front covers of the Australian Women's Weekly. Or pages from one of the digitised journals.
- Harvest newspaper articles from Papers Past, the NLNZ's digitised newspaper service.
- Harvest thousands of press releases and interviews from federal politicians, via Trove and the Parliamentary Library.
- Harvest the millions of tags that Trove users have applied to resources.
- Harvest details of Radio National programs from the past twenty years.
- Harvest the text contents of a collection of archived web pages.

To get you started, and again, to address the 'why' as well as the 'how', the GLAM Workbench includes a series of pre-harvested datasets – data packaged and ready to go. This includes, for example a set of 3,471 full-page editorial cartoons harvested from *the Bulletin* – one for every issue between 1886 to 1952. As well as a collection of 12,619 press releases and interviews from federal politicians talking about refugees.

Web archives, in particular, illustrate the possibilities and challenges of working across different scales. The GLAM Workbench tries to make the mountains of data preserved in web archives available in manageable chunks. So you can, for example, examine how an individual web page changes over time. But you can also look across a whole domain.

There's one notebook that lets you download and explore all the Powerpoint files under defence.gov.au.

Platforms & environments

There is no core application that runs the GLAM Workbench, no digital platform that needs to be maintained. It's a collection of repositories containing Jupyter notebooks. And each of the repositories contains configuration files that enable the notebooks to be run on a variety of platforms. Different users will have different needs at different times.

Throughout the GLAM Workbench you'll find links that say 'Run live on Binder'. Just click those links and a customised computing environment will be created in the cloud using a service called Binder. Once it's ready, the notebook will load, ready to do real work. One click and you're away.

Using Binder helps the GW overcome some of the initial barriers that might confront a novice user – there's no logging in, no creation of accounts, no installation of software. Everything is configured for you. This convenience encourages experimentation – just have a go! – but it does have significant limits. Binder sessions will timeout if no activity is detected, and won't save any files that you've created or changed. You have to manually download anything you want to keep. Every Binder session starts with a clean slate.

If you're harvesting large datasets from somewhere like Trove's newspapers or a web archive, these limits will quickly become annoying. But if your project grows to this point, you can set up your own dedicated GLAM Workbench repository using Reclaim Cloud or Nectar. These are both cloud services, and spinning up your own GLAM Workbench is quick and easy. But unlike Binder, the environment you create will persist between sessions, so your data will be preserved.

And of course if you're comfortable installing software, and managing your own system, you can run the GLAM Workbench on your own computer – either by using one of the pre-built Docker images, or setting things up from scratch using Python.

As your research needs change, as your skills develop, as your project grows, you have choices, you have ways to move forward.

A work in progress

As least that's the plan. The GLAM Workbench is, and will always remain, a work in progress. For me, this whole enterprise is a continuing exploration of the possibilities of GLAM data.

I hope you've seen something of interest, something that will encourage you to do some exploring of your own. If you do, I'll do my best to help…

Openness important.

So take what you want, share it, change it. That's what it's for.

#presentation