

Metadata Nirvana

Data discovery and metadata creation
untouched by human hands

Jo Cook, Astun Technology

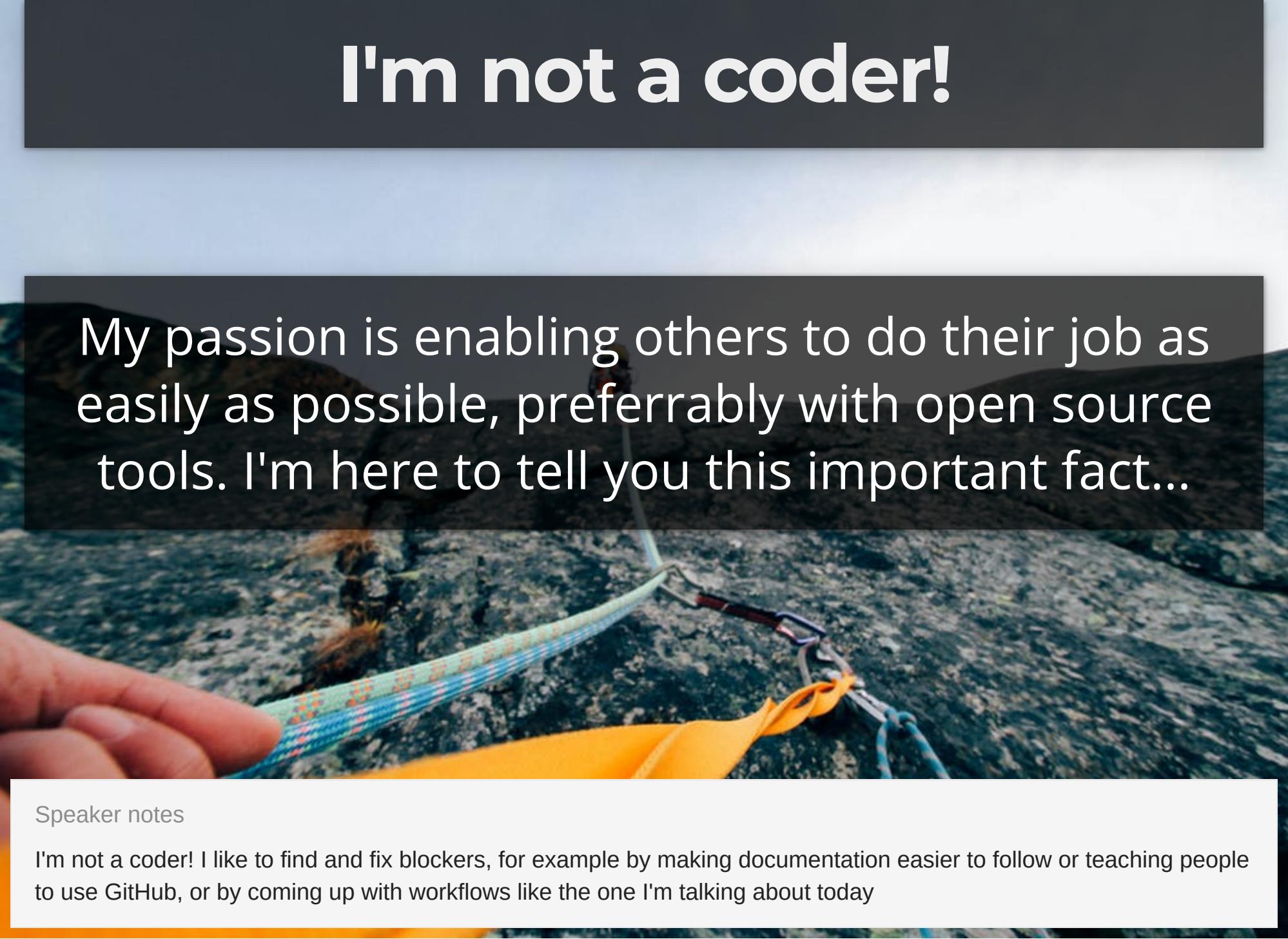
Hello! I'm Jo 🙌 I'm the Technical Evangelist for Data Discovery at Astun Technology.



Speaker notes

Astun Technology was founded in 2006. We're based in Epsom, but our 25-ish staff are spread across Europe. We do spatial and data "stuff", based on an open source technology stack. My job title is just a fancy way of saying that I help people find and share data.

I'm not a coder!

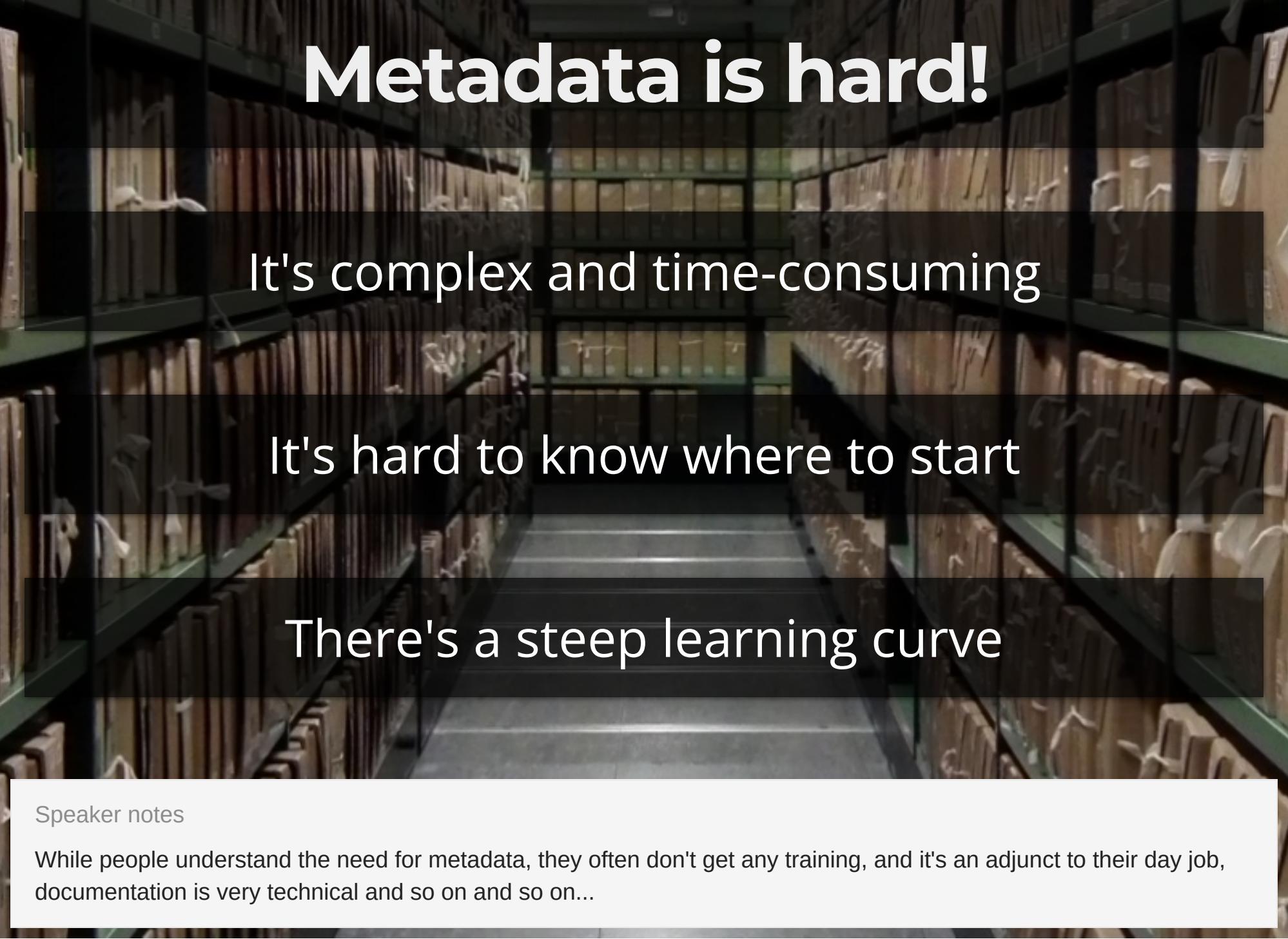


My passion is enabling others to do their job as easily as possible, preferably with open source tools. I'm here to tell you this important fact...

Speaker notes

I'm not a coder! I like to find and fix blockers, for example by making documentation easier to follow or teaching people to use GitHub, or by coming up with workflows like the one I'm talking about today

Metadata is hard!



It's complex and time-consuming

It's hard to know where to start

There's a steep learning curve

Speaker notes

While people understand the need for metadata, they often don't get any training, and it's an adjunct to their day job, documentation is very technical and so on and so on...



and even worse...

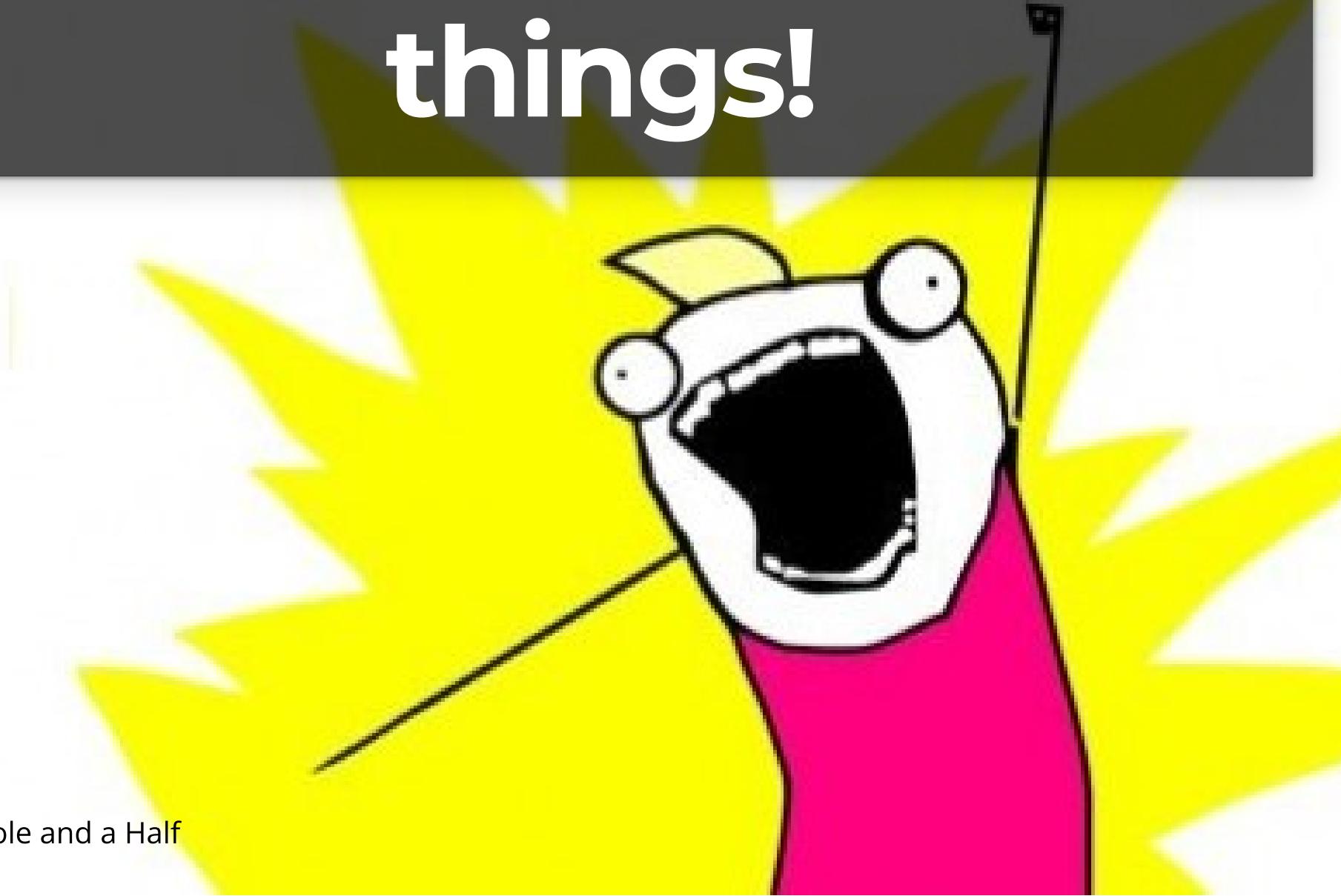
Manual metadata entry doesn't scale

Solutions that work for tens of datasets don't
work so well for thousands

Speaker notes

This leads to problems with completion, accuracy and currency, and metadata creators don't necessarily know about all the data, or who is responsible for it

Automate all the things!



Automation is not new!

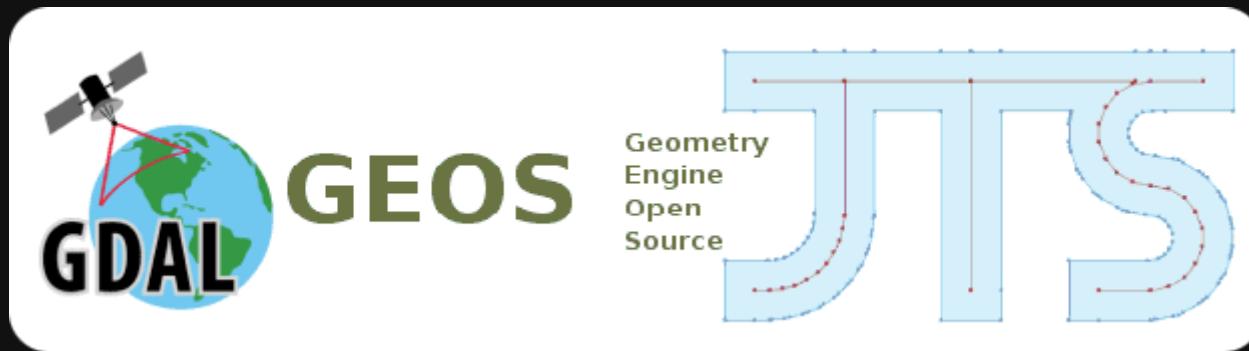


Speaker notes

There are many tools out there that will derive at least some metadata elements for you, including the USGS Metadata Wizard, which has been around for many years

Using FOSS is not even very original!

There are many libraries that can be used to derive some metadata



Speaker notes

GDAL/OGR, JTS, GEOS, Boost..., not only in geospatial but for things like CityML



Layer Properties - high_water_polyline | Information

Information from provider

Name	high_water_polyline
Path	/home/jo/Documents/geodata/boundaryline/Data/GB/high_water_polyline.shp
Storage	ESRI Shapefile
Comment	
Encoding	ISO-8859-1
Geometry	Line (MultiLineString)
CRS	EPSG:27700 - OSGB 1936 / British National Grid - Projected
Extent	5512.998499999996944,5336.68600000000222 655653.849999999767169,1220301.502000000949947
Unit	meters
Feature count	47,357

Identification

Identifier	
Parent Identifier	
Title	
Type	dataset
Language	
Abstract	
Categories	
Keywords	

Extent

CRS	Spatial Extent
-----	----------------

Help Style Apply Cancel OK

Some metadata elements can be easily derived from the dataset

Speaker notes

You can get at this sort of metadata in your file browser- think titles, locations, change dates, or tools like QGIS, which will show you spatial extents.

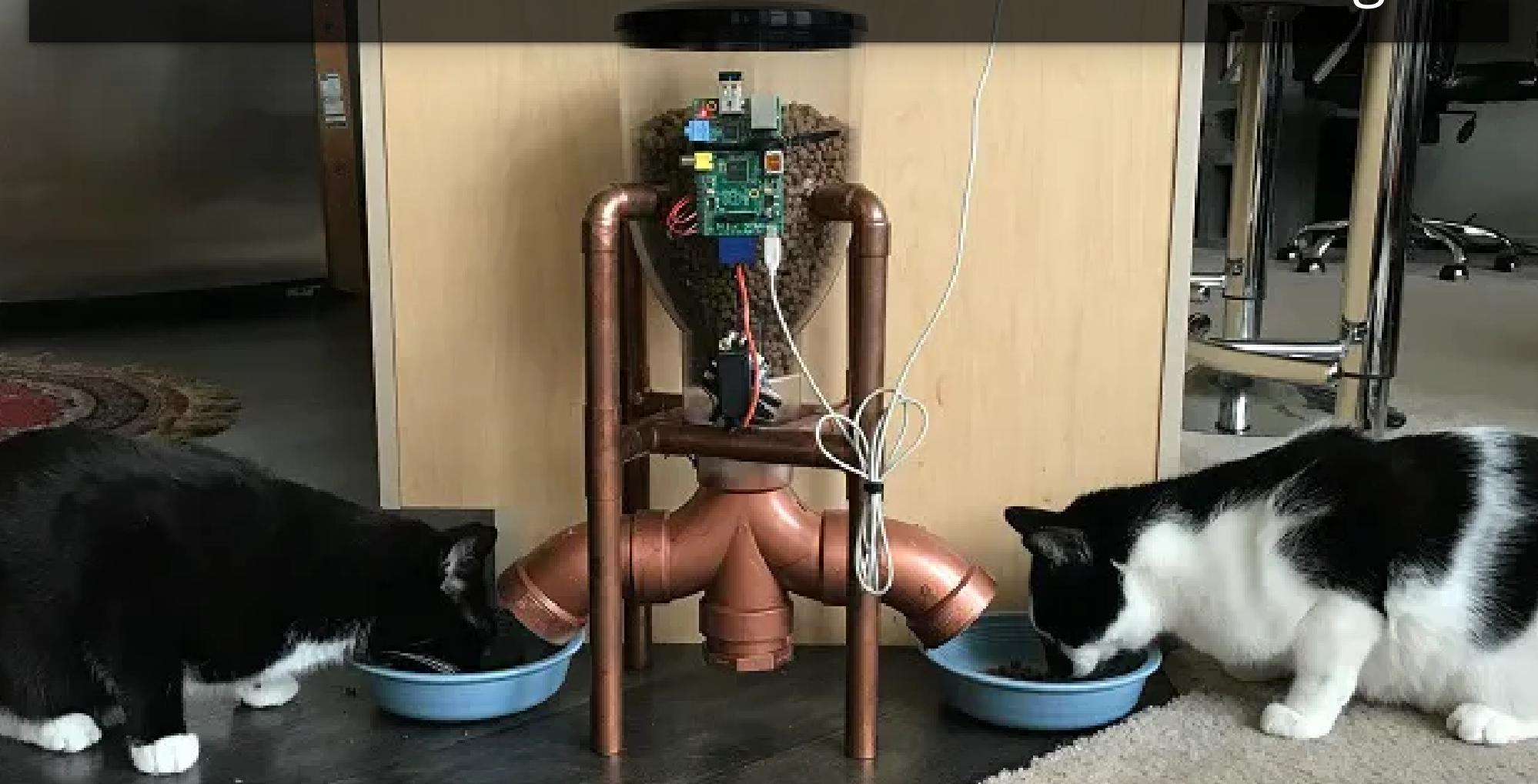
A fluffy, light-colored kitten with dark spots is sitting on a red feathered surface. The kitten is looking up and to the right with wide, curious eyes. The background is dark and out of focus.

But we need a way to derive some elements such as human-readable titles, descriptions and keywords

Speaker notes

Human-readable titles, descriptions, keywords and so on are much harder to do programatically, and also hard for users to do at scale. What works manually for a few records doesn't scale to thousands.

We've bolted together a number of open source tools and libraries to overcome this challenge



Speaker notes

Some of it is very proof of concept at the moment so I don't have a slick demo

Part One: Metadata Crawler

Crawler is a script for discovering data (spatial or non-spatial) in file systems, databases and on websites. For each data source it finds, it derives as much of the metadata as it can.



Speaker notes

Crawler builds on some ETL libraries built by Titellus, for Talend ETL's spatial plugin. It's open source. It can be run as a web service (using tomcat) or as a cross-platform shell script. We've adapted it to work with non-spatial datasets, and also to output metadata in the UK Gemini metadata profile

STEP 1: FROM DATA TO PLACEHOLDER METADATA



```
<?xml version="1.0"  
encoding="UTF-8"?>  
<gmd:MD_Metadata>  
...  
</gmd:MD_Metadata>
```

Speaker notes

Crawler creates an xml metadata record for each data source it discovers, with place-holders for the elements it can't derive. It can insert records directly into any Metadata Catalog with a CSW-T (Transactional CSW) endpoint, not just GeoNetwork. It can create new or update existing records.

Part Two: Ye Olde Approache



A	B	C	D	E
	abstract Descriptive text about the table. This should be at least 100 characters long to satisfy the Gemini 2.3 guidance. The purpose of an abstract is to provide a clear narrative summary that enables the reader or user to understand the content of the data. See the README sheet for information on the other fields to be completed	keywords	updatefrequency	inspirekeyword
chema.table		location	daily weekly fortnightly monthly annual	keywords, abstracts, and contact information

Speaker notes

Metadata Crawler derives titles using a set of rules (for databases it uses the form Database.Schema.Tablename) so we can use the same when auto-generating the rows in the excel spreadsheet to ensure a match. To ensure we get quality results from this process, we can use controlled text fields for elements where that is appropriate. This may seem clumsy but excel is well-known and used. Data owners can copy and paste in bulk to populate "their" records quickly and easily.

STEP 2: METADATA FROM CSV

```
title,abstract,keywords,updatefrequency  
db.geo.buses,"this is a dataset about buses",buses,daily
```

```
host:8080/geonetwork/srv/eng/q?title  
db.geo.buses"
```



```
[  
 {  
   "uuid": "xxx"  
   "this is a dataset about buses" :  
   "<snip>/gmd:abstract/gco:CharacterString"
```

Speaker notes

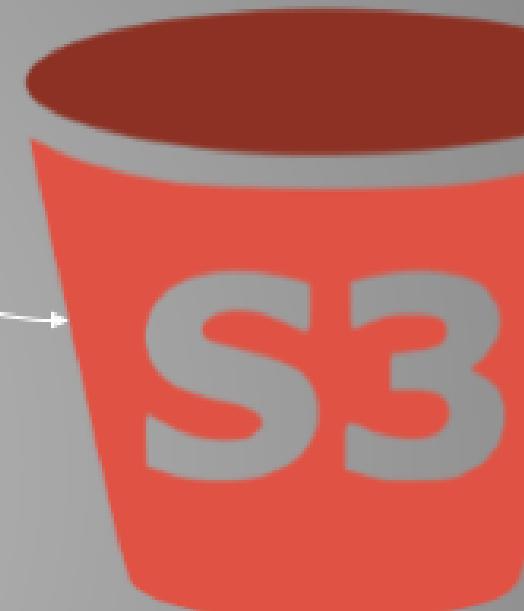
Then we use the GeoNetwork API wrapped in a python script to update the records in the catalog from the spreadsheet. Even though GeoNetwork is hosted on AWS in the cloud, we can let users run these scripts using environments such as Cloud9

STEP THREE: EMAILING CSV

metadata@astuntechnology.com

cordtitle,abstract,keywords
datefrequency
Table mydb.geo.buses,this is
dataset about
ses,buses,daily

Recordtitle,abstract,keywords
.updatefrequency
Table mydb.geo.buses,this is
a dataset about
buses,buses,daily

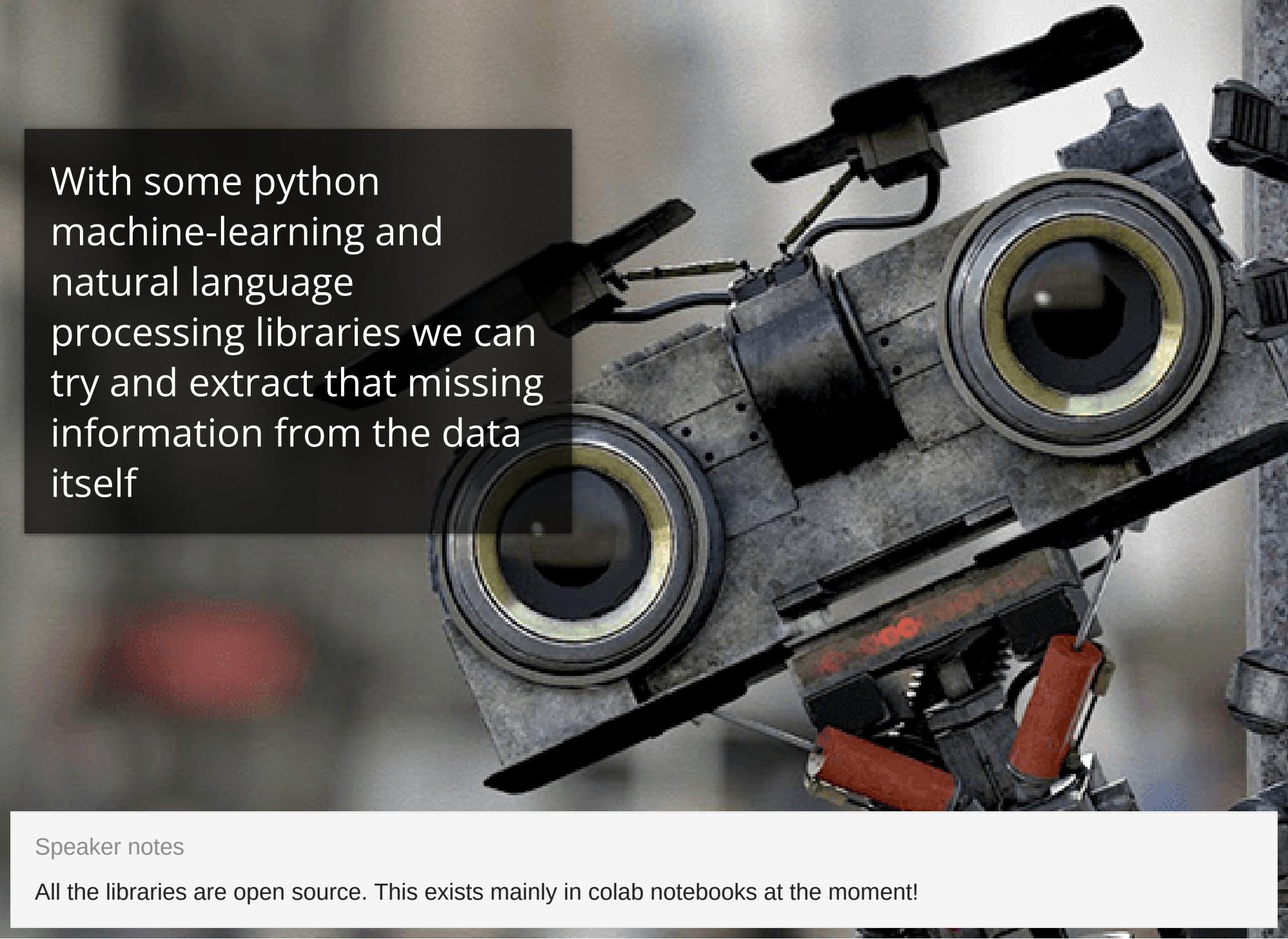


Speaker notes

Alternatively, the completed csv can be emailed to an address associated with an S3 bucket accessible to the GeoNetwork processing script

Part Two: The Cool New Approach

**BEYOND THIS POINT
YOU SHOULD ENGAGE
A GUIDE**



With some python
machine-learning and
natural language
processing libraries we can
try and extract that missing
information from the data
itself

Speaker notes

All the libraries are open source. This exists mainly in colab notebooks at the moment!

IRL Example!

Auto-Summarise:

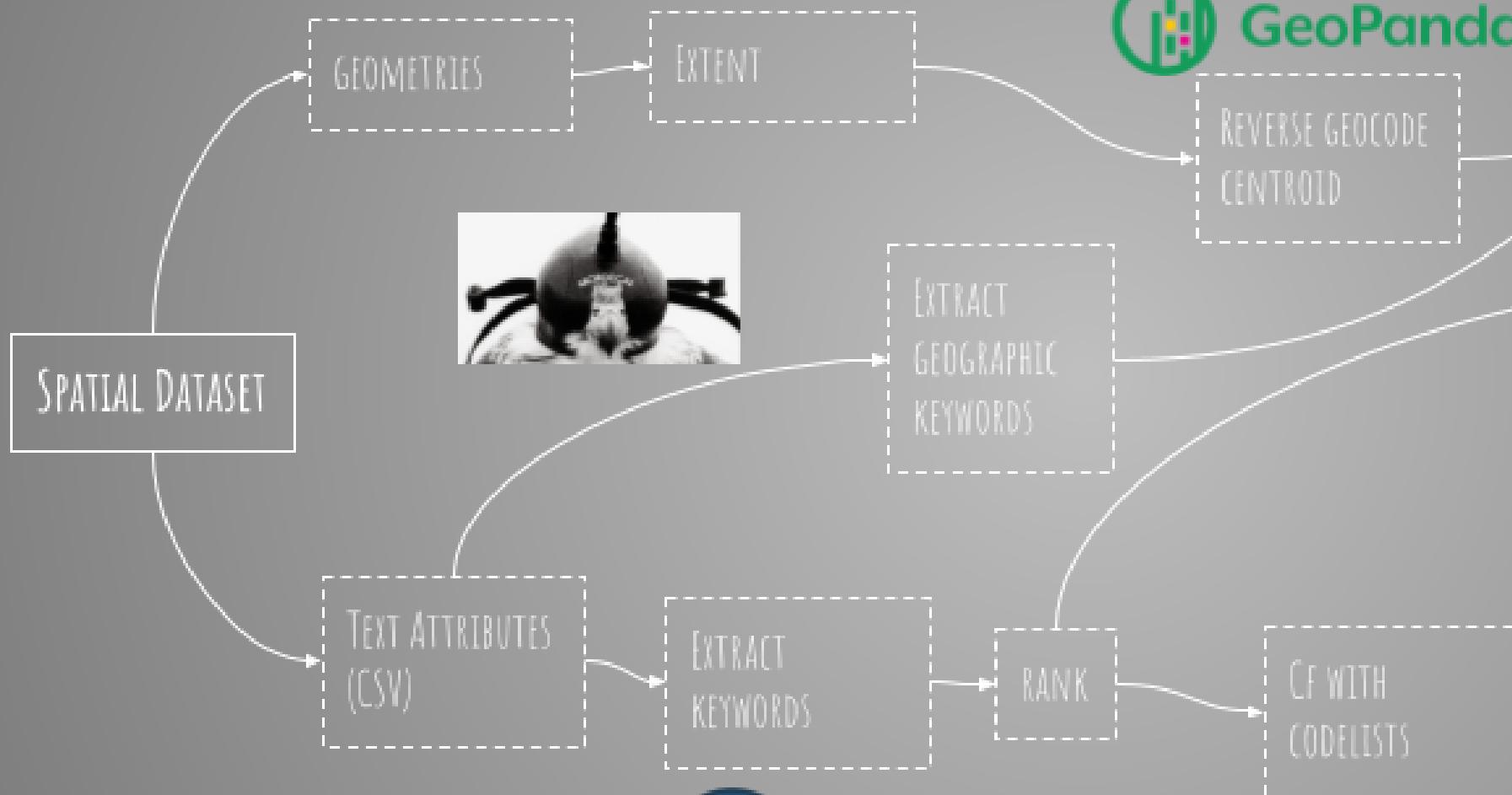
A deterministic river temperature model to prioritise management of riparian woodlands to reduce summer maximum river temperatures and a riparian woodland prioritisation score that looks to maximise the benefits of Riparian tree planting for protecting Scotland's rivers from the adverse effects of climate change.

Details of the modelling work that produced the river temperature and climate sensitivity predictions can be found in the peer reviewed manuscript: Jackson et al (2018) 'A spatio-temporal statistical model of maximum daily river temperatures to inform the management of Scotland's Atlantic salmon rivers under climate change.' (see link under 'Additional Information' Tab).

Speaker notes

This is a real example of putting a complex block of text through our prototype ML workflow. We can generate keywords and geographic keywords, and auto-summarise to create a short description. Metadata keywords should be used inside the abstract. This needs to be done intelligently, eg as part of sentences rather than lists. You should include keywords that work for all levels of expertise, and take into consideration variations in spelling, synonyms and so on. Geographic keywords help to provide context for the user.

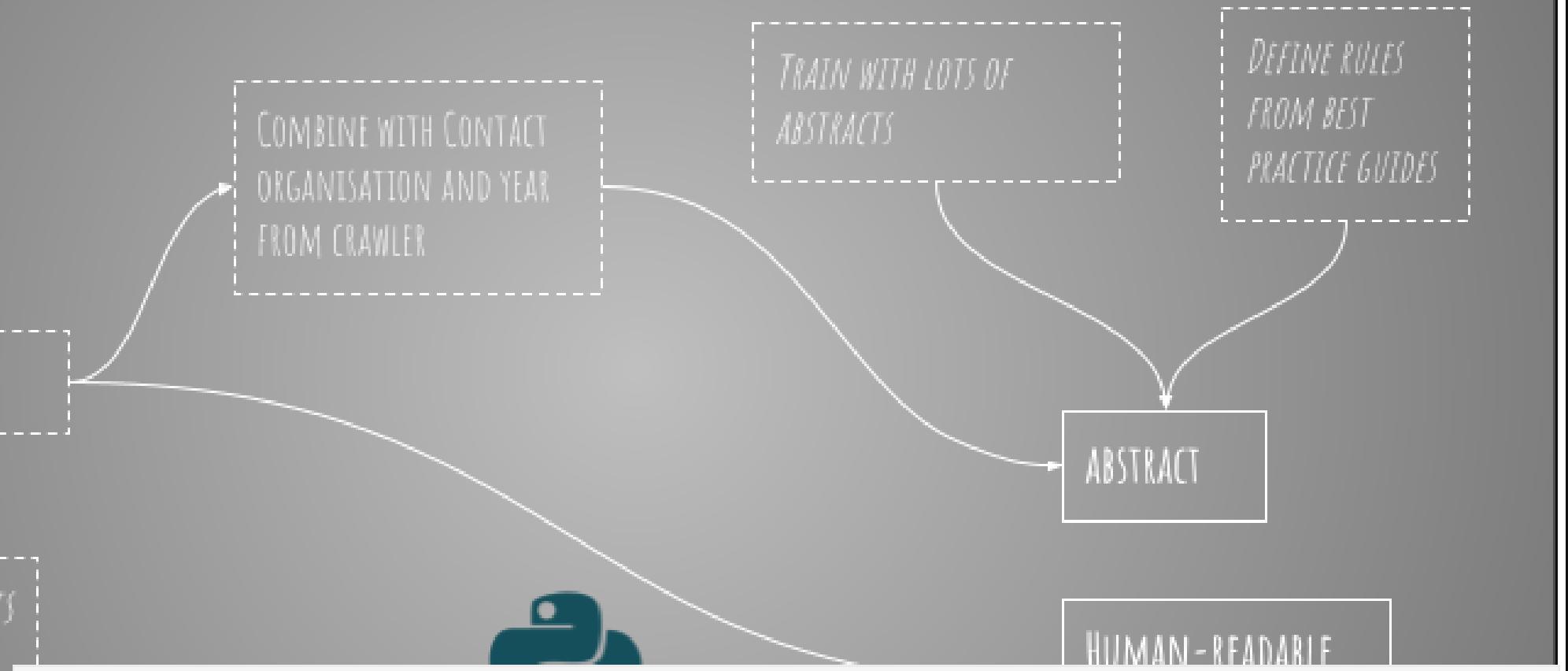
MACHINE-LEARNING STEP ONE



Speaker notes

We can then rank our results and refine our keywords by comparing with codelists

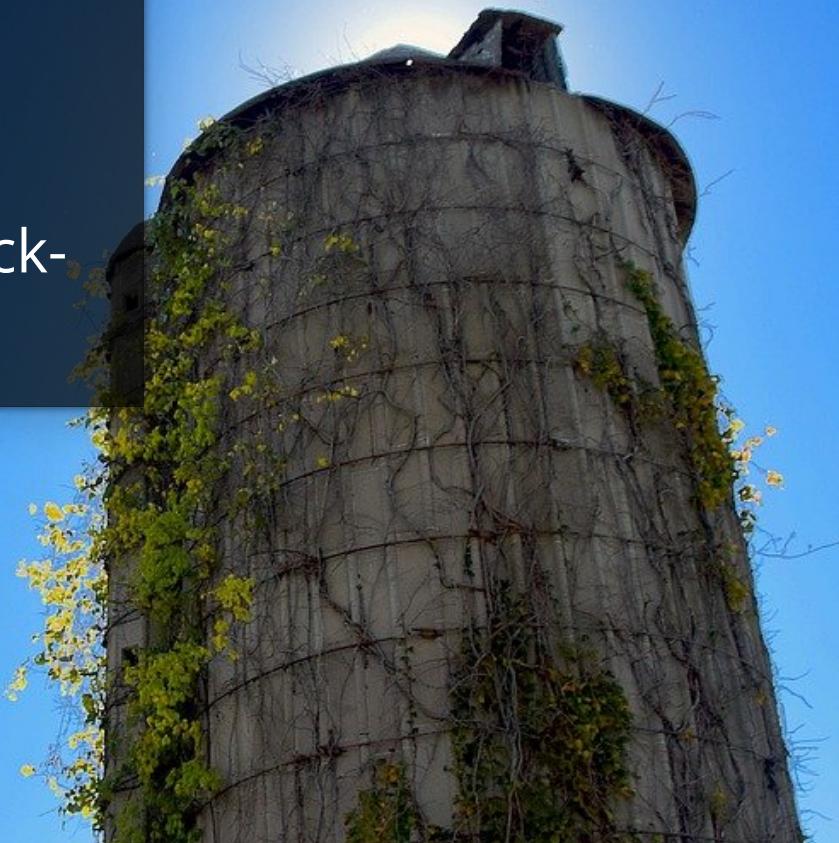
MACHINE-LEARNING STEP TWO



Speaker notes

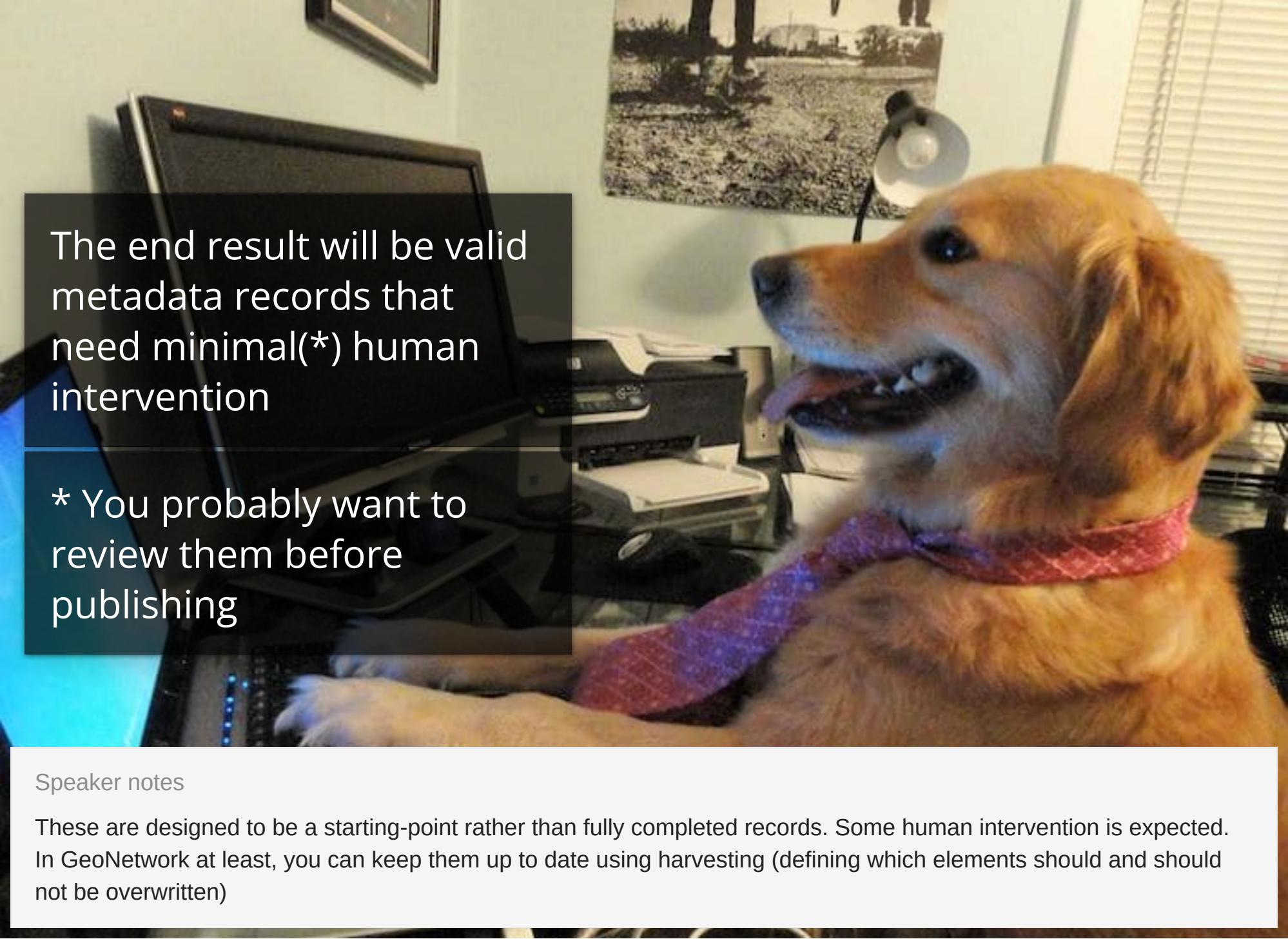
Once we have these additional elements we can combine them with the output from Crawler to complete our metadata. We have a large corpus of metadata, and some best practice guidelines for data sharing to help train our model and define some rules. The guidance is from an SEO perspective and indicates how the elements like titles and abstracts might be displayed in standard web search results. So (eg) the min and max length for an abstract

We've designed this workflow to be fully modular to avoid technological silos or lock-in.



Speaker notes

You can use any approach to get the derived metadata to feed into the models, and output the results into any metadata schema.



The end result will be valid metadata records that need minimal(*) human intervention

* You probably want to review them before publishing

Speaker notes

These are designed to be a starting-point rather than fully completed records. Some human intervention is expected. In GeoNetwork at least, you can keep them up to date using harvesting (defining which elements should and should not be overwritten)

The usual caveats apply!



Speaker notes

We have all the pieces but need to glue them together into a usable product. It's a bit heath robinson right now. There are lots of moving parts, and areas where we're going to get expert assistance such as on SEO (to test our ML results against what they would do) and to ensure we're using the right ML models and so on.

Useful References

talend-spatial.github.io

www.gov.uk/government/publications/search-engine-optimisation-for-publishers-best-practice-guide

Thank You!

Jo Cook, Astun Technology

jocook@astuntechnology.com | @archaeogeek
astuntechnology.com | @astuntech