Setting Up Your Survey Data

What you need to consider when you want Data Central to host your astronomical data

data central

Outline

- High-level ideas/concepts to keep in mind
- How best to manage catalogues/tabular data
- How best to manage data files (e.g. images/spectra)
- What additional work is required to make your data release fully FAIR

High-level ideas/concepts to keep in mind

- It's easier to do things at the beginning correctly, rather than trying to backfill at the end
  - Information/Knowledge may be lost if not recorded correctly
- Data Central (and the rest of the Australian Software and Data teams) have seen many good and bad attempts, so ask us for help early (even in survey planning stages)
  - Asking us early not only reduces everyone's workload, but also means we can plan what new features we need and have time to develop them before the data is released
- See 2022ApJS..260....5C for some general advice around the data and paper publication process
  - Includes a checklist when publishing a paper

Have Unique IDs

- Everything must have a unique ID
- Unique IDs are required for:
    - Linking tables/catalogues together
    - Linking tables/catalogues to data files
    - Defining how data files are related (e.g. spectra of the same object)
    - Crossmatching between surveys, and building upon crossmatched results
    - Allowing other surveys to reference your data more precisely

Many simple things are easier to work with than a few complex things

- Don't put everything in a single table, have multiple tables which each contain one concept
- Don't put all the data for one object in one file, different data types (e.g. spectra, cubes) should be in different files
- Keeping things simple makes queries both easier and faster
- Allows tools to scale to large data volumes

Don't include data wholesale from other surveys

- Especially don't include data from other surveys and then modify it in undocumented ways
- Either:
  - Use references via unique IDs (e.g. cross-matching tables)
  - Normalise all data include to one format, with detailed provenance information
- If you think data that you depend on will disappear (e.g. it’s not published well), let us know and we can save it, don’t copy parts of it into your survey
- With the rise of well-maintained and managed services, including postage-stamps with the data is no longer needed—interactive Image Cutouts and HiPS can provide a more powerful replacement.

Automate validation of data

- Set this up early, and you’ll be in a much better position when it comes time to release it
- We'll give some links to downloadable tools which can help find potential problems
- Using a system like PAWS can make this easier

Managing Catalogues/Tables

- Always remember that catalogues/tables are stored in a database
  - PostgreSQL, MySQL and MariaDB are commonly used
- The rules about good database design apply to catalogues/tables
- Database systems, especially those designed to store tabular data and query with SQL, work best with normalised data
  - See wikipedia page on database normalisation, including the examples: https://en.wikipedia.org/wiki/Database_normalization#Normal_forms

Managing Catalogues/Tables

- The simplest way to think about this is each table should only deal with one concept
    - This makes it much easier to query: equality/comparative operators are much easier to use and faster than trying to use regex
- Try working with a subset of your data in **SQLite** with everything in one table vs. the normalised form—you can derive more interesting and powerful ideas with the normalised form (**pandas** can read and write to SQL databases)
- This is very similar to the concept of “tidy data” that R users may be familiar with

Example: Normalised Catalogues for Redshift Survey

For a redshift survey where the redshifts have been matched to templates, the following design could be used:

- A table for each of the objects observed, containing the ID of the object and its position
- A table for each spectra observed, containing the ID of the spectra, the ID of the object, and when it was observed
- A table for each redshift calculated, containing the ID of the redshift result, the ID of the spectra, the (unique) ID of template used, and a quality of fit value
- A table for each redshift template, containing the ID of the redshift template, and what it is of (e.g. spiral vs. elliptical)

Example: Normalised Catalogues for Redshift Survey

- “SELECT object_id, count(*) FROM spectra GROUP BY object_id”
  - Calculate the number of times objects were observed
- “SELECT templates.name, AVG(redshifts.quality) FROM redshifts JOIN templates ON redshifts.template_id = templates.id”
  - Calculate the average quality of fit for each template

Managing Data Files

- Must have a unique ID
- Either in a standard format (e.g. FITS) or come with documentation on how to handle it
  - If in a standard format, validators should be run on the files
  - astropy comes with **fitscheck** and **wcslint**, these should throw up no errors and ideally produce no warnings
  - **dfits** and **fitsort** allow for quick examination of FITS headers

Managing Data Files

- Check metadata within files (e.g. FITS headers)
  - Are the values correct and meaningful e.g. are there fields which have been copied over from raw which are no longer correct/meaningful, such as dates/times for stacked spectra?
  - Does the metadata include the unique ID, where the file is from, and when it was produced?
  - Are the units of the data stored correctly?
  - Where the format has common metadata standards (like FITS), are they included (e.g. position with equinox and system fully specified)?
  - Does every file include the metadata, rather than some including some metadata, and others missing it?

Managing Images

- Must use standard FITS WCS keywords
  - Required for both the cutout tool and SIA to process images
- Should include additional observational metadata (allows for more specific SIA queries):
  - Resolution of image
  - Wavelength/frequency/energy band
  - Time coverage of image
  - Polarisation of image (if relevant)
- Ensure both data and WCS have two axes:
  - **wcslint** should pick this up

Managing spectra

- Data Central uses specutils (Astropy Coordinated Package) to handle reading the variety of formats spectra are stored in. If needed, we write additional loaders for surveys which allow both Data Central, and more widely any specutils user, to load spectra hosted by Data Central.
    - Data Central's SSA service serves out the original spectra, as well as a simple format designed to be plugged into tools like splat without extra effort.

Managing spectra

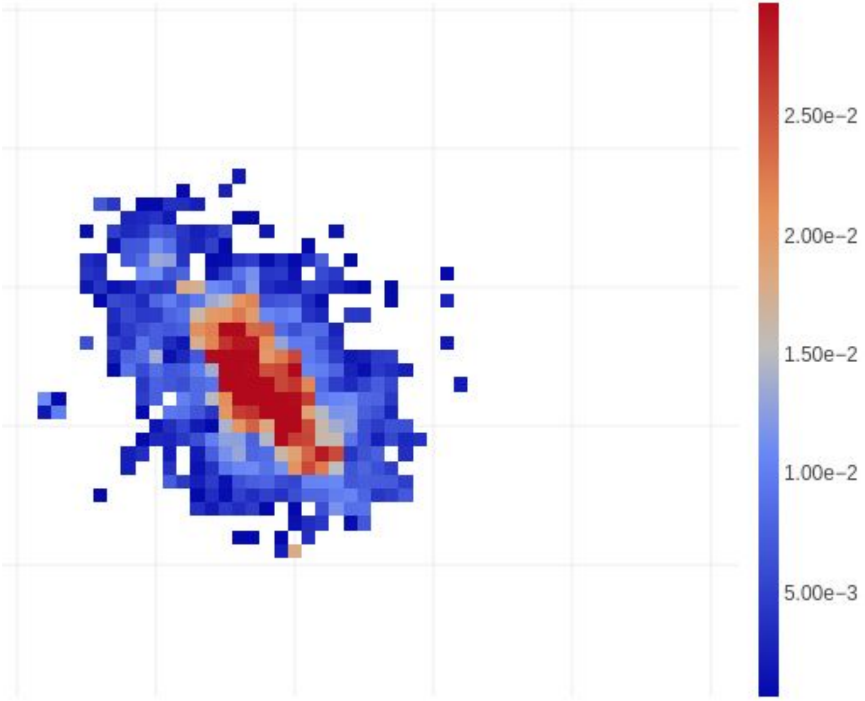
- Whilst we can adapt to any format with enough time, there are two formats which are relatively well behaved across different tools:
  - A spectrum per file, where either the wavelength or frequency is linear, and further extensions include uncertainty, sky, masks etc. (preferred)
  - A spectrum per file, where the flux, wavelength etc. are in a table (which allows for non-linear wavelengths/frequencies)

Managing spectra

- The FITS WCS standard does allow for logarithmic wavelengths/frequencies, but we've found that support for this is either patchy or does not behave in a consistent way across software.
    - Do not stick logarithmic data into FITS and set a linear WCS

Managing IFS

- Visualisation currently designed around LZIFU
- 2D Maps are the only format currently visualised



data central

Managing other data types/other data formats

- Best to come and discuss with us, to answer following the two key questions
    - What metadata is needed to find and query the data?
    - How do we read and visualise the data?
- We're planning on introducing a "blob" or "opaque" type for files which are supplementary and don't fall into one of the current grouping we have.

What is needed beyond the VO for FAIR data

- Data usage license
  - Astronomy has taken a laissez faire approach
  - Funders (e.g. ARDC) want more clarity—consistency with other fields
  - Astronomy Data Centres picking either CC-BY or CC-0—there have been recent ADASS discussions on what is best, see 2020arXiv201212994G as a starting point
- Provenance metadata
  - FITS header COMMENT and HISTORY usage
  - Paper references (use DOI)
  - Documentation on Document Central

data central

Coming soon

- Refactored ingestion system:
  - Faster ingestions—faster turnaround time for larger datasets
  - More detailed feedback—notify team as to what the current status of ingestion
  - Will allow teams to run test ingestions without having to wait on the Data Central team
- New data types:
  - Blob/opaque data type for field-specific formats: e.g. PFD files from pulsar astronomy
  - Data Cubes
  - New ingestion system will allow adding new data types faster

Questions?