

# Identify associated data sets

## Story

"e.g. Find and download direct and dispersed images that overlap on the sky via an archive query"

Minimum: As a scientist I want to get a set of WFC3 grism exposures with their associated direct images for a given pointing or a position on the sky in order to carry out an analysis. Only identify direct images in the same visit as the grism.

Better: As a scientist I want to get all the ACS and WFC3 direct and grism exposures for a given pointing or a position on the sky in order to carry out an analysis that provides me with the deepest spectrum with the widest wavelength coverage. The matched photometry across all HST bands will allow me to do further analysis.

## Inputs

- RA + Dec position on the sky

or

- Proposal ID

or

- Dataset association ID or some other method of specifying a particular HST visit

Optional, if only a subset of the data is needed (say only WFC3/IR data or only one grism or only certain filters of imaging), the user should be able to limit the search to those that are needed.

If only a proposal or association is specified, the user should be able to find all observations that overlap with the footprint of the visit(s) with some buffer around them. This is specifically handy to identify direct images in the area surrounding a grism exposure. Constructing a catalog that has a wider sky coverage allows the user to better account for contamination from objects off the edge of the direct image.

## Outputs

Astropy table object/FITS table with all the matching data, including association, ra and dec of pointing, instrument, filter/grism, date of observation, exposure time, program ID, target name, pointing footprint.

A straight-forward way to get from this list to a list of files that need to be downloaded in order to carry out analysis

The user should be able to edit the output of the query to decide if they do not want to continue processing some dataset or they can go back and re-execute a query that is better tuned to their needs.

An plot showing all the overlapping data:



## Computations

Finding all data at a given position is pretty easy with astroquery. Though some improvements there are possible:

<https://github.com/gbrammer/mastquery/blob/master/mastquery/query.py>

The part where you need to find all the overlaps with a footprint is the part that is hard. Astroquery does take a region file as input but if you need to modify the default one (say expand or co-add several region files) then you are in for a lot of messy work. Here is code that shows what's needed:

<https://github.com/gbrammer/mastquery/blob/392823efad196d8427f16d380fa81122ee12e4f6/mastquery/overlaps.py#L111>

## Drawbacks

This is not grism specific and a lot of the functionality is in astroquery.mast already so it is not clear how much (if any) of the heavy lifting should be done by a grism specific package rather than pushing functionality upstream to astroquery.mast.