PAD for Collaborative Notes for the

Registry Session at November 2023 Interop in Tucson, Arizona (USA) 2023-11-11

https://wiki.ivoa.net/twiki/bin/view/IVOA/InterOpNov2023Registry

Renaud Savalle, Hendrik Heinl - Report on recent data publishers onboarding activites

Markus Demleitner - The harvest trigger service

Discussion: Python as a registry client for the end users

Tess Jaffe leads the discussion on making it easy to discover services, in PyVO and/or registry.

Changes needed in PyVO or in the Registry ?

List of use-cases gathered from the Registry Working Group

1. How to find DOIs in the Registry

Markus: When you have the DOI, just dereference it to get to the data. However, there may be a point to search for "data from publication X". Adding a Constraint for that would totally make sense, but you'd have to explain to people how to specify the source (bibcode vs. DOI vs. free-form).

Oh, and: Where this really matters is D-links from ADS. Getting D-links into the ADS is the topic of a draft note: https://github.com/ivoa/BibVO.

Anne Raugh: FAIRness implies to resolve DOI. Less accessible if user has to ask the IVOA for it. AR: Also note that it is entirely proper for an IVOA Registry to resolve a DOI to something other than a landing page. That's appropriate behaviour for an IVOA service/data set DOI resolver. Thus the DOI can be used as a data/service access hook on the actual landing page, for example.

Somewhat tangentially, Gilles suggested an ADS-like "tag:content"-type interface to the registry. This does not exist as such now but could be implemented with reasonable effort; see http://dc.g-vo.org/WIRR for an interface on top of the registry with a menu UI.

=> NEED MORE DISCUSSION AND A CLEARER USE-CASE

2. How do I find catalogs/services with redshift information?

UCD finder from CDS, called by find_ucd(desc) funtion
Markus: this function could go into PyVO as an utility function => PR to do (and then fix the docstring on the UCD constraint).

HH: Great feature. Only comment: the actual use case is 'redshift', but the query is for 'infrared' hence the resulting ucd does actually not fit. ;-) RS: Sorry I wanted to show a term with 2 words so I have used "infrared magnitude" in that example.
HH: I think we can simply adapt the use case accordingly (it's actually a more complex one). Sure, please give me a term that makes sense with redshift ;-) Or just run the example with "redshift", it works.

3. Use-case by Gilles Landais was the same as 2.

4. User looking for data: not sure what kind is avail or how to query it ?

Does he have to interpret our servicetypes ('conesearch') ? Tess: Use-case: Look for a paper in VizieR then interfaces to access that table, get "conesearch" services -> There should be a documentation "you can call this with center and a radius".
Markus: improve interface.describe() showing PyVO documentation.
GG: get_service() component seems obscure, would prefer astroquery than having to go thr an service interface after the registry.search() query. Would like to have a good brain-dead default service chosen
Markus: You'll get many services of different types, a blind choice won't work here. But see also Markus talk in the Apps2 session "The Image Discovery API" about "give me images". Images and Spectrum services are better suited for that use-case

5. Looking for UV images without knowledge of SIA vs SIAv2

=> as well, see Markus talk in the Apps2 session "The Image Discovery API". and associed PyVO PR. (discover/? module) =>
https://github.com/astropy/pyvo/pull/470
Markus: 20% done in that PR, need volunteers to join the effort.

Brief teaser: The code is perhaps 20% there, collaborators solicitated. What's required to make this work well: Proper STC coverage declared by services.

RS: There is also a plan to bring dataproduct_type to the Registry - see my opening slides in State of the TCG yesterday

6. Hendrik's use case: will be solved by the dataproduct_type addition in the Registry

7. Use-case by Gilles Landais: Fuzzy text search using Lemmatization

Currently asteroid give different results than asteroidS
Markus: RegTAP has ivo_hasword() which does a full-text search in Postgres but it's not the only backend.
Gilles Landais:

Markus: oh, and doing that as a keyword search *is* inviting trouble, because that also matches without lemmatisation against res_subject, where people may have either asteroid or asteroids. Please, publishers, use the UAT as per
http://www.ivoa.net/rdf/uat!

8. Manon: "The HiPS are ServiceCatalog but it's hard to search for hips".

Need more details about the question; this should really work by saying

- natural join rr.capability where standard_id='(look up in hips spec)'

9. Baptiste

Complex user-case with querying lots of services and having problems with timeouts from some. Broader topic than Registry search

10. Markus use-case about "Use your constraints ("Covers M1 in XRay")"

*** Reg chairs should look at that ***

11. Markus: Duplication Use-Case

CONCLUSION

- We should do this regularly maybe also with other client interfaces...