Fig. 1

Fig. 2

Fig. 3

Fig. 4

1. Register using purx

(cf. Fig. 1)

Markus Demleitner
msdemlei@ari.uni-heidelberg.de

(cf. Fig. 2)

purx = PUblishing Registry Proxy

- Why Register?
- Registration Options
- The purx solution
- A sample registration

(cf. Fig. 3)

2. Why Register?

(when there's Google)

There's longer answers, but the short answer is:

structured metadata enables powerful discovery

As in: "Where can I find far infrared spectra obtained for objects in the vicinity of Sco-Oph?"
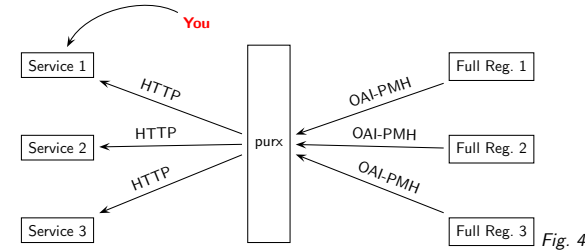
3. Several Ways Into The Registry

Full story:

http://wiki.ivoa.net/twiki/bin/view/IVOA/GettingIntoTheRegistry

In short:

- Use a web page and forms.
- Run a "publishing registry"
- And now, purx.

4. Enter purx

(cf. Fig. 4)

Essentially, purx regularly does http requests (with if-modified-since headers) to registered URLs to establish if any changes have been made to the resource records prepared by the services. If there are any, they will be included in incremental harvests from the full registries.

5. So?

- To register, just grab an example file from the purx enrollment site,
- change what you need to change,
- put that file to a stable web location,
- and tell purx

(cf. Fig. 5)

Here are links to some sample records:

- A simple browser service: http://dc.zah.uni-heidelberg.de/getRR/dexter/ui/ui
- A browser service with metadata for the underlying table: http://dc.zah.uni-heidelberg.de/getRR/ucds/ui/ui
- A standard cone search (SCS) service with a custom browser service: http://dc.zah.uni-heidelberg.de/getRR/amanda/q/cone
- An image (SIAP) service with a custom browser service: http://dc.zah.uni-heidelberg.de/getRR/lswscans/res/positions/sia
- A spectral (SSAP) service: http://dc.zah.uni-heidelberg.de/getRR/flashheros/q/ssa
- A spectral line (SLAP) service: http://dc.zah.uni-heidelberg.de/getRR/toss/q/q
- A TAP service (but again, don't write capability and tableset by hand, your toolkit can already build these): http://gavo.aip.de/getRR/__system__/tap/run

Fig. 5

6. In Practice I

DaCHS, for instance, spits out appropriate files itself:

```
$ curl http://voparis-rr.obspm.fr/getRR/__system__/tap/run | \
  xmlstarlet fo | head -25 | tail -15
<ri:Resource
  created="2009-12-01T10:00:00"
  status="active" updated="2017-03-08T14:59:34"
  xsi:type="vs:CatalogService">
  <title>PADC TAP Server on voparis-rr.obspm.fr TAP service</title>
  <shortName>PADCC TAP</shortName>
  <identifier>ivo://vopdc.obspm/tap</identifier>
  <curation>
    <publisher>Paris Astronomical Data Centre</publisher>
    <creator>
      <name>PADC</name>
      <logo>http://voparis-srv-paris.obspm.fr/logos/PADC_small.png</logo>
    </creator>
    <date role="updated">2017-03-08T14:59:34</date>
    <contact>
      <name>PADC support team</name>
      <address>77 av. Denfert Rochereau, 75014 Paris, FRANCE</address>
```
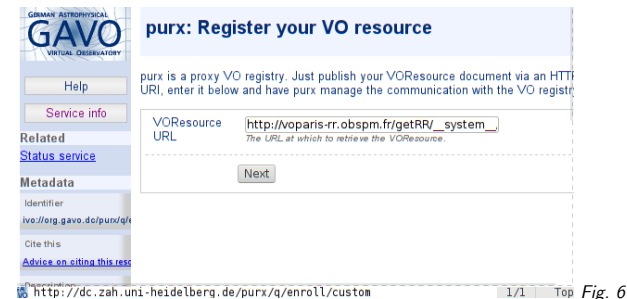

purx: Register your VO resource

Fig. 6

Fig. 7

7. In Practice II

So, submit the URL where the VOResource XML is located:

(cf. Fig. 6)

(this will also validate your resource record)

8. Receive a Mail

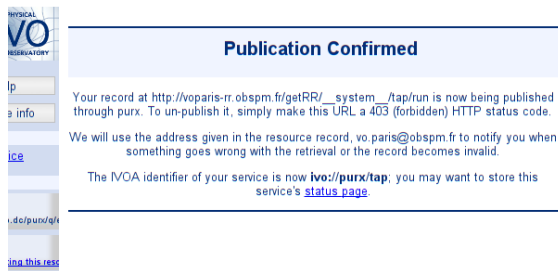(cf. Fig. 7)

This went to contact/email.

Fig. 8

Publication Confirmed

Your record at http://voparis-rr.obspm.fr/getRR/__system__/tap/run is now being published through purx. To un-publish it, simply make this URL a 403 (forbidden) HTTP status code.

We will use the address given in the resource record, vo.paris@obspm.fr to notify you when something goes wrong with the retrieval or the record becomes invalid.

The IVOA identifier of your service is now **ivo://purx/tap**; you may want to store this service's status page.

9. Confirm Publication

When you retrieve the URL in the mail, you get:

(cf. Fig. 8)

That's it.

purx will warn contact/email after a while when the document becomes unavailble or invalid.

You can immediately get rid of the publication by returning a 403 from the XML URL.

And you can inspect the status of your publication at any time.

10. Do you want to know more?

purx info (including sample records) at

http://dc.g-vo.org/purx/q/enroll/info

Thanks.

5