Fig. 1

Fig. 2

Fig. 3

1. Reuse XSLT for Service DOIs

(cf. Fig. 1)

Markus Demleitner
msdemlei@ari.uni-heidelberg.de

(cf. Fig. 2)

- Solving what?
- VOResource to DataCite
- VOResource to landing page

(cf. Fig. 3)

2. Solving what?

General topic: Improving citeability. We now advise our users to cite services like this:

```
@MISC{vo:Flash_Heros_SSAP,
  year=2012,
  title={Flash/Heros {SSAP}},
  author={Wolf, B. and Kaufer, A. and Mandel, H. and Stahl, O.},
  url={http://dc.zah.uni-heidelberg.de/flashheros/q/ssa/info},
  howpublished={{VO} resource provided by the {GAVO} Data Center},
  doi = {10.21938/oUuqajnWmofP2w75.Acc8g}
}
```

The DOI adds credibility and lets cross-discipline citation engines count statistics. Neither the URL nor the IVOID would work for that.

Also, there's http://dx.doi.org/*doi*. Whether or not *we* think citation-counting is a worthwhile exercise: A working infrastructure for doing that will matter to our data providers.

3. VOResource-to-Datacite

DOIs are managed by registering metadata in a Schema called **Datacite Metadata Kernel** (DaCi) with a registrar.

DaCi metadata isn't too far away from VOResource.

A stylesheet for doing this translation is available from

```
https://volute.g-vo.org
    /svn/trunk/projects/registry/dois
```

The repo also contains some python wrapper code that can generate DOIs and talk to registrars.

4. vor-to-doi usage

To generate DaCi metadata from VOResource, just apply the XSLT to an OAI-PMH response:

```
$ xsltproc --stringparam doi_for_record 10.5072/testrec \
>> vor-to-doi.xslt vor-example.xml  xmlstarlet fo

<d:resource xmlns:d="http://datacite.org/schema/kernel-3">
  <d:identifier identifierType="DOI">10.5072/testrec</d:identifier>
  <d:creators>
    <d:creator>
      <d:creatorName>Schwan, H.</d:creatorName>
    </d:creator>
    ...
  </d:creators>
  <d:titles>
    <d:title>ARIGFH identified objects</d:title>
  </d:titles>
  <d:publisher>The GAVO DC team</d:publisher>
  <d:publicationYear>2011</d:publicationYear>
  <d:subjects>
    <d:subject>Stars: Positions</d:subject>
    ...
  </d:subjects>
```

Note how you need to pass in the DOI – since it is a matter of local policy and subject to uniqueness requirements, there is no way the stylesheet can figure it out. The oai2datacite.py script that comes with the stylesheet contains DOI generation logic, but still no uniqueness enforcing.

Most items map fairly straightforwardly from VOResource to DaCi. When translating in the other direction, multi-language support in DaCi (e.g., multiple titles) could become a challenge – and of course DaCi is missing our capability and table metadata.

5. vor-to-doi usage II

```
  <d:resourceType resourceTypeGeneral="Service"
    >AstroObjects</d:resourceType>
  <d:alternateIdentifiers>
    <d:alternateIdentifier alternateIdentifierType="ivoid">ivo://org.gavo.dc/arigfh/q/cone</d:alternateIdentifier>
  </d:alternateIdentifiers>
  <d:relatedIdentifiers/>
  <d:formats>
    <d:format>application/x-votable+xml</d:format>
  </d:formats>
  <d:descriptions>
    <d:description descriptionType="Abstract">ARI's "Geschichte des Fixsternhimmels" is an att-
empt to collect all
astrometrically useful observations from before ca. 1970 in a way
comparable to what has been done to construct the FK* series of
fundamental catalogs. About 7e6 published positions are included.

In GAVO's DC, we provide tables of identified and non-identified stars
together with the master catalog that objects were identified against.</d:description>
  </d:descriptions>
</d:resource>
```

The AstroObjects resourceType is a free invention for VO purposes (but we're allowed to do this). The XSLT also uses AstroImage, Spectrum, SpectralLines, and AstroData for SIAP, SSAP, SLAP, and TAP services. It'd be great if we could agree on a common set of terms within the VO.

The style sheet doesn't yet support non-service resource types. Let's tackle these as we need them.

Our IVOID here is an alternate identifier.

The XSLT handles a number of relationships already. We should probably revisit this as VOResource 1.1 comes along, as that unifies the relationship terms between VOResource and DaCi (no solution for service-for and served-by yet, though).

6. VOResource to Landing Page

DOI custom is to let the DOI resolve to a "Landing Page" listing metadata and access modes.

Where VOResource `capability` isn't represented in DaCi, this is where we need it.

There's an XSLT turning VOResource into a landing page on

`https://svn.ari.uni-heidelberg.de`
`/svn/gavo/hdinputs/voidoi/res`

Apart from branding, this should again work fine for third parties.
(cf. Fig. 4)

The landing page is a fairly straightforward translation of VOResource metadata; see an example[1].

Tricky question: How should we represent access URLs for VO services?

8. So...

There's certainly room for improvement...

...it'd be great if we could share the efforts.

[1] `http://dx.doi.org/10.21938/puTViqDkMGcQZu8LSDZ5Sg`

Fig. 4