



Fig. 1



Fig. 2

```

<?xml-stylesheet href="/static/xsl/datalink-to-html.xsl" type="text/xsl"?>
<DESCRIPTION>Size of the resource
<VALUES null="1"/>
</FIELD>
<DATA>
<TABLEDATA>
<TR>
<TD>ivo://org.gavo.dc/~?kapt
<TD>http://localhost:8080/kapt
</TD>
<TD>Low-res photo with plate b
<TD>#preview-image</TD>
<TD>image/jpeg</TD>
<TD>229620</TD>
</TR>
<TR>
<TD>ivo://org.gavo.dc/~?kapt
<TD>http://localhost:8080/kapt
</TD>
<TD>Greyscale wedge scanned wi
<TD>#calibration</TD>
<TD>image/fits</TD>
<TD>73454240</TD>
</TR>

```

Fig. 3

1. Datalink and XSLT

(cf. Fig. 1)

Markus Demleitner
msdemlei@ari.uni-heidelberg.de

(cf. Fig. 2)

Datalink is great. But client uptake lags.

(Temporary?) remedy: XSLT.

- What's the plan?
- Obstacles
- Solutions
- Help!

2. The Plan

Use XSLT to turn this:

(cf. Fig. 3)

into this in common browsers:

(cf. Fig. 4)

TABLE LINKS



Where?	Description	What?
Link (2273227 Bytes)	Low-res photo with plate borders	#preview-image ivo://org.gavo.dc/~?kapt
Link (21003840 Bytes)	Greyscale wedge scanned with the data	#calibration ivo://org.gavo.dc/~?kapt
Link (268971840 Bytes)	The full dataset.	#this ivo://org.gavo.dc/~?kapt
	A preview for	#preview

Fig. 4

3. Obstacle 1

Browsers will ignore

```

<?xml-stylesheet
  href="/static/xsl/datalink-to-html.xsl"
  type="text/xsl"?>

```

on application/x-votable+xml documents.

However, datalink mandates delivering Datalink documents with

application/x-votable+xml;content=datalink

4. Solution 1

Cheat just a bit.

Browser requests look like this:

```

GET /flashheros/q/sdl/dlmeta?ID=... HTTP/1.1
Host: localhost:8080
Accept: text/html,application/xhtml+xml,application/xml;...
DNT: 1
...

```

Actual datalink clients have no business Accepting HTML.

So: When a request comes in accepting both HTML and XML, we decide it's a browser and return a text/xml media type.

5. Obstacle 2

VOTable is not exactly XSLT-friendly. I ended up dumping abominations like

```
<xsl:variable
  name="access_url_index"
  select="count(vot:VOTABLE/vot:RESOURCE[@type='results']
    /vot:TABLE/vot:FIELD[@name='access_url']
    /preceding::vot:FIELD)+1"/>
```

Yikes.

6. Datalinks Tossed Around

I now start to include datalink columns in my tables – as in

- <http://dc.g-vo.org/flashheros/q/web/form>¹ or
- <http://dc.g-vo.org/kapteyn/q/web/form>².

As long as key clients understand our DL service blocks – perhaps that's reasonable?

7. Desidrata

The XSLT only formats plain datalinks.

→ adhoc:service RESOURCES
Basic support should be straightforward

The XSLT doesn't properly handle multiple datasets.

→ Need hack to group by ID

The XSLT produces visually unattractive documents.

→ Hm. Style it?

8. Action?

- Improve the XSLT (icons! flashy graphics! support for services.)
- Legalise the behaviour in an erratum?
- Identify columns linking datalink documents?

¹ <http://dc.g-vo.org/flashheros/q/web/form>

² <http://dc.g-vo.org/kapteyn/q/web/form>