e-inf-astro

GEFÖRDERT VOM
Bundesministerium
für Bildung
und Forschung

PUBLIC DOMAIN

VODataService 1.3 In PR

Markus Demleitner

IVOA 2026 Northern Spring Interop, Strasbourg

What's VODataService

VOResource (base) defines the basic *vr:Resource* type; that's abstract enough to also cover *vr:Organisation*, *vr:Authority*, or *vstd:Standard*, etc.

On top of that, VODataService defines Resource types suitable for data collections, things with tables (most importantly, *vs:CatalogService*, and things that speak HTTP (*vs:ParamHTTP*).

On top of that SimpleDALRegExt, TAPRegExt, VOEvent, ... define how to register services of concrete types.

Whatever we do in VODataService impacts almost all user-visible records.

News on 1.3

- *stats*: Giving distributions of table columns and characterising input parameters
- *productTypeServed*: Say whether you serve images, spectra, time series, dynamic spectra, or whetever.
- *dataSource*: Say whether you serve simulated or measured data.

Column Statistics

```
<column>
  <name>Vmag1</name>
  <description>? Magnitude of the main star (1)</description>
  <unit>mag</unit>
  <ucd>phot.mag</ucd>
  <stats>
    <min>2.2</min>
    <percentile03>2.8</percentile03>
    <median>9.25</median>
    <percentile97>11.5</percentile97>
    <max>11.7</max>
    <fillFactor>0.405</fillFactor>
  </stats>
  <dataType xsi:type="vs:VOTableType">float</dataType>
</column>
```

Discrete Stats

Stats also does discrete statistics:

```
<column>
   <name>hubble_type</name>
   <description>Morphological Hubble type for galaxies.</description>
   <ucd>src.morph.type</ucd>
   <stats>
     <option freq="0.110673636">E</option>
     <option freq="0.044598754">E-S0</option>
     <option freq="0.0020044383">E?</option>
     <option freq="0.007445057">I</option>
     <option freq="0.0007874579">IAB</option>
     <option freq="0.0040088766">IB</option>
     ...
   </stats>
</column>
```

It's On InputParam-s, Too

```
<param std="true">
  <name>TABLE</name>
  <description>The name of the table to run this against...
  <ucd>meta.id;meta.table</ucd>
  <stats>
    <option>gaia.dr3lite</option>
    <option>gaia.dr2light</option>
  </stats>
</param>
```

All that looks suspiciously like VOTable VALUES. So... should we rename *stats* to *values* for symmetry with VOTable?

Product Types

The old method of "if you're looking for images, look for SIAP services" has been broken ever since Obscore 1.2. Now there's at least a vocabulary that lets you express what product types you serve:

http://www.ivoa.net/rdf/product-type

Drop that into your resource records:

```
</coverage>
<productTypeServed>cube</productTypeServed>
<productTypeServed>image</productTypeServed>
<productTypeServed>spectrum</productTypeServed>
<productTypeServed>timeseries</productTypeServed>
<tableset>
```

productTypeServed in RegTAP

You can see the prototypes:

```
SELECT ivoid
FROM rr.res_detail
WHERE detail_xpath='/productTypeServed'
  AND 1=gavo_vocmatch(
    'product-type',
    'spectrally-resolved-dataset',
    detail_value)
```

Transition Plan: RegTAP 1.3 services will automatically image or spectrum for otherwise unannotated SIAP or SSAP service. Ok?

Theory Or Observation?

SSAP has had a feature to say “this is theoretical data” for a long time.

This shouldn’t depend on the product type, so VODataService 1.3 introduces the vocabulary

http://www.ivoa.net/rdf/data-source

There is also the new element *dataSource* that you can use like this:

```
SELECT ivo_string_agg(ivoid, ', '), detail_value
FROM rr.res_detail
WHERE detail_xpath='/dataSource'
GROUP BY detail_value
```

Do we expect structure in data-source? If so, this should do `ivo_vocmatch`, too.

Stuff To Think About

- Is there other low-cost and useful statistics that could go into *stats*?
- Should *stats* be *values* (and perhaps feed back into VOTable)?
- Can you add these this metadata to your records?
- Feed dataProductServed defaults from protocols in RegTAP?
- Have dataSource default to observation in RegTAP?