RegTAP: A Relational Registry for the VO

Markus Demleitner, Florian Rothmaier

Astronomisches Rechen-Institut Heidelberg

January 16, 2013
Why RegTAP?

- the Registry Interface 1 ("RI1") relies on deprecated technologies
  - SOAP-based web services based on XQuery and on ADQL 1.0

- RI1 underspecifies what is being queried against
  - even simple queries behave in a rather different way on the various registries

- the RI1 infrastructure is in need of a lot of work and maintenance
  - it seemed to be less work to use existing components like TAP servers and ADQL 2.0 in the medium term
RegTAP: Overview

- internal working draft on volute¹

- the interface for searching registries: a complete rewrite based on TAP ("RegTAP")
  - RegTAP consists of **13 official tables** plus four tables for the STC extension (wish list)

- **three ADQL user-defined functions** for pattern matching, string comparison

- design guidelines: our goal was not to entirely cover the VORegistry schema but to provide a **robust and user-friendly interface**

The resource Table

- contains most atomic members of `vr:Resource` (type defined in the VOResource schema)

<table>
<thead>
<tr>
<th>Name</th>
<th>UType</th>
</tr>
</thead>
<tbody>
<tr>
<td>ivoid</td>
<td><code>vor:resource.identifier</code></td>
</tr>
<tr>
<td>res_type</td>
<td><code>vor:resource.type</code></td>
</tr>
<tr>
<td>created</td>
<td><code>vor:resource.created</code></td>
</tr>
<tr>
<td>res_title</td>
<td><code>vor:resource.title</code></td>
</tr>
<tr>
<td>updated</td>
<td><code>vor:resource.updated</code></td>
</tr>
<tr>
<td>reference_url</td>
<td><code>vor:resource.content.referenceURL</code></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

- given UTypes refer to the VOResource schema equivalent
- primary key of the table: `ivoid` (all tables have this column)
- further table columns: information about the creator of the resource, the coverage, the waveband, the instrument, ...
The interface Table

- this table subsumes the attributes of the various VOResource interfaces

<table>
<thead>
<tr>
<th>Name</th>
<th>UType</th>
</tr>
</thead>
<tbody>
<tr>
<td>ivoid</td>
<td>vor:resource.identifier</td>
</tr>
<tr>
<td>intf_type</td>
<td>vor:interface.type</td>
</tr>
<tr>
<td>res_title</td>
<td>vor:resource.title</td>
</tr>
<tr>
<td>query_type</td>
<td>vor:interface.queryType</td>
</tr>
<tr>
<td>result_type</td>
<td>vor:interface.resultType</td>
</tr>
<tr>
<td>access_url</td>
<td>vor:interface.accessURL</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

- unlike the VOResource schema, RegTAP only allows one URL per interface
  - queries simplified by this design decision
The res_details Table

- this table gives utype-value pairs for registry extensions

<table>
<thead>
<tr>
<th>Name</th>
<th>UType</th>
</tr>
</thead>
<tbody>
<tr>
<td>ivoid</td>
<td>vor:resource.identifier</td>
</tr>
<tr>
<td>cap_index</td>
<td></td>
</tr>
<tr>
<td>detail_utype</td>
<td></td>
</tr>
<tr>
<td>detail_value</td>
<td></td>
</tr>
</tbody>
</table>

- in particular this key-value mechanism is intended as an easy fallback for new VOResource extensions

- example: “Find all TAP endpoints offering the relational registry.”
The res_details Table

- this table gives utype-value pairs for registry extensions

<table>
<thead>
<tr>
<th>Name</th>
<th>UType</th>
</tr>
</thead>
<tbody>
<tr>
<td>ivoid</td>
<td>vor:resource.identifier</td>
</tr>
<tr>
<td>cap_index</td>
<td></td>
</tr>
<tr>
<td>detail_utype</td>
<td></td>
</tr>
<tr>
<td>detail_value</td>
<td></td>
</tr>
</tbody>
</table>

- in particular this key-value mechanism is intended as an easy fallback for new VOResource extensions

- example: “Find all TAP endpoints offering the relational registry.”

```sql
SELECT access_url
FROM rr.interface
  NATURAL JOIN rr.capability
  NATURAL JOIN rr.res_detail
WHERE standard_id='ivo://ivoa.net/std/tap'
  AND detail_utype='vor:capability.datamodel.ivoid'
  AND detail_value='ivo://ivoa.net/std/regtap/vor'
```
Other Tables

- **capabilities**: descriptions of what a resource does and how to use it
- **res_role**: information about creators, publishers and contacts
- **intf_parameters**: description of the input parameters that can be provided to the service as a name=value argument
- **res_schema, res_table, table_column**: basically VODataService (metadata on columns of tables / descriptions of tables pertaining to the resource)
Other Tables

- **capabilities**: descriptions of what a resource does and how to use it
- **res_role**: information about creators, publishers and contacts
- **intf_parameters**: description of the input parameters that can be provided to the service as a name=value argument
- **res_schema, res_table, table_column**: basically VODataService (metadata on columns of tables / descriptions of tables pertaining to the resource)
- **subject**: keywords
- **res_date**: date associated with an event in the life cycle of the resource
- **relationship**: descriptions of relationships between resources
- **validation**: validation levels for resources and capabilities
On Our Wish List: The STC Extension

- separate “Data Model” with IVORN
  ivo://ivoa.net/std/RegistryInterface#stc

- many use cases for this extension

- four STC tables in RegTAP: rr.stc_spatial, rr.stc_temporal, rr.stc_spectral, rr.stc_redshift
  - e.g. table rr.stc_temporal

<table>
<thead>
<tr>
<th>Name</th>
<th>UType</th>
</tr>
</thead>
<tbody>
<tr>
<td>ivoid</td>
<td>vor:resource.identifier</td>
</tr>
<tr>
<td>time_start</td>
<td>stc:AstroCoordArea.TimeInterval.StartTime</td>
</tr>
<tr>
<td>time_end</td>
<td>stc:AstroCoordArea.TimeInterval.EndTime</td>
</tr>
</tbody>
</table>

- service coverage is modelled as a union of intervals (or bounding boxes)
“Find all SIA services that provide infrared images.”

```sql
SELECT ivoid, access_url
FROM rr.capability
    NATURAL JOIN rr.resource
    NATURAL JOIN rr.interface
WHERE standard_id='ivo://ivoa.net/std/sia'
    AND waveband LIKE '%infrared%'
```
Use Cases

- “Find all SIA services that provide infrared images.”

```
SELECT ivoid, access_url
FROM rr.capability
    NATURAL JOIN rr.resource
    NATURAL JOIN rr.interface
WHERE standard_id='ivo://ivoa.net/std/sia'
    AND waveband LIKE '%infrared%'
```

- “Find all TAP services; return their accessURLs.”

```
SELECT ivoid, access_url
FROM rr.capability
    NATURAL JOIN rr.interface
WHERE standard_id='ivo://ivoa.net/std/tap'
```
our system is running
  ▶ in Heidelberg and in the very near future on a mirror in Potsdam

the ESA will adopt our system pretty soon

TAP client libraries are already available

registry web interface planned by GAVO Heidelberg

try it and consider updating your own clients
References

K. Benson et al.: IVOA Registry Interfaces Version 1.0, IVOA Recommendation (2009-11-04),
http://www.ivoa.net/Documents/RegistryInterface/20091104/REC-RegistryInterface-1.0.pdf.


M. Demleitner: A relational model for VOResource, talk given at the IVOA interop in Urbana (May 2012).

M. Demleitner: Towards Registry Interfaces 2, talk given at the IVOA interop in Sao Paulo (October 2012).