

1. TAP: The GAVO experience

Markus Demleitner (msdemlei@ari.uni-heidelberg.de)

Agenda

- Server side: DaCHS
- Client side: tapsh
- Registry substitute: GloTS

2. Server Side: DaCHS

DaCHS is GAVO's Data Center Helper Suite, an integrated package including ingestion, metadata management, and protocol interfaces (SCS, SIAP, SSAP, OAI, TAP...).

Implemented

- Hopefully all mandatory features
- Geometry support including regions
- Upload support

Not implemented yet

- Authentication
- Sensible quotes
- Queuing

Missing: A solid validator, extensive testing

3. Implementation considerations

- X.509/SSO probably is too heavyweight for a simple „I don't want other people to see my queries“
- UPLOAD parameter spec needs some clarifications – in particular: are quoted identifiers allowed as table names (in DaCHS, they are not)? What should happen if a URL or table name contains a comma or semicolon (in DaCHS, they are forbidden in table names and in URLs)? When people re-post an UPLOAD parameter, should uploads be added or replaced (in DaCHS, they are added)
- xtype=adql:REGION in uploads is at least very hard with pgSphere (DDL?)
- It would be good if the TAP spec could require a filename header in MIME parts of inline uploads. This helps in telling them apart from, say, PQL parameters.
- ADQL: Need language that comments are allowed wherever whitespace is allowed
- ADQL: Decay of INTERSECTS to CONTAINS for point arguments is a PITA. Can we deprecate it?

4. tapsh

Query TAP servers in the database shell-like – note that tapsh is exited in between:

```
msdemlei@victor:/home/msdemlei > tapsh -N
tapsh> server ivo://org.gavo.dc/...system_/tap/run
tapsh> select table_name from tap_schema.tables
tapsh> start
tapsh>
msdemlei@victor:/home/msdemlei > tapsh -N
tapsh> job
Job nicked cixute (remote id Pg2UTG)
phase COMPLETED, destruction due 2011-05-05T13:10:28, time limit 3600
-----
select table_name from tap_schema.tables
tapsh> dump
          table_name
tap_schema.schemas
tap_schema.tables
tap_schema.columns
[...]
tapsh> delete
```

5. Executable TAP

Here's an „executable“ tapsh script including a cross-server join:

```
#!/usr/local/bin/tapsh --echo-input
server ivo://vfau.roe.ac.uk/sdssdr7-dsa
select * from DR5QuasarCatalog as t where t.z between 0.1 and 0.12
run
nick quas
server ivo://org.gavo.dc/...system_/tap/run
select * from TAP_UPLOAD.uploaded as u join ppm1.main as p on \
  (1=contains(point('ICRS', p.raj2000, p.dej2000), \
    circle('ICRS', ra, dec, 0.5))) where kmag-jmag>0.5
upload result quas as uploaded
run
save "result.xml"
delete
delete quas
```

There should be a shortcut for the very common crossmatch expression. In tapsh, we let people enter a template by hitting ESC-M, but that's clearly suboptimal.

6. What's out there?

To find out what is already on TAP, you could ask the registry (still needs polish) or query each server's TAP_SCHEMA yourself (that's tedious).

For the time being we provide the Global TAP Schema GloTS¹.

tapsh interfaces to it through its `metasearch` and `tables` commands.

In the future, this task should be performed by the registries (that have comparable data).

¹ <http://dc.g-vo.org/glots/q/plain/form>

7. What next?

Tapsh and DaCHS are written in python and available under the GNU GPL. Get them at <http://soft.g-vo.org>². There's also a tapsh version for the JVM in a single jar file. The library tapsh is built on is available as seperately in GAVO's VOTable package.

Deploy (there are now at least three software packages implementing TAP and ADQL), and (please!) register.

We need much more data in TAP-queriable services. If you want, the GAVO data center will publish your data for you – just talk to me.

Cross-server joins are a major selling point for standard services, but they require uploads. Thus, upload support is highly desirable.

I'm not sure where VOSpace sits in the picture. Inline uploads at least make the issue less urgent.

² <http://soft.g-vo.org>