Fig. 1

1. Blind Discovery in the VO

Markus Demleitner
msdemlei@ari.uni-heidelberg.de

Blind discovery is looking for resources by resource properties:

- Coverage (space, time, spectrum)
- Observables (e.g., temperatures, redshifts)
- Characterisation (C14n: e.g., spectral resolution, limiting magnitude)

This is against the currently dominating discovery mode: “Where is the Gaia data?”

(cf. Fig. 1)

2. Resource vs. Dataset Discovery

This is about doing discovery of *Resources* in the VO Registry, i.e., roughly data collections.

For blind dataset (i.e., image, spectrum,. . .) discovery, the VO’s traditional recipe is to globally query all services of a certain kind with your constraints.

There are limits to that, though. For instance, SIAP version 1 didn’t let you constrain by time, and it’s still hard to constrain by sensitivity in all our current protocols (to be fair: that’s hard to define).

Also, as the number of services grows, you will want to have a resource (i.e., service) discovery phase in front of the all-VO query in order to weed out services guaranteed to have nothing. That way, typcial queries will be able to skip 50% to 99% of the registered services, resulting in comparable speed-ups for global dataset discovery.

3. What Physics?

VODataService tablsets have had UCDs forever. So, to find resources that have both temperatures and proper motions, you can use a RegTAP query like

```
select distinct ivoid
from rr.table_column
where ucd like ’phys.temperature;%’
intersect
select distinct ivoid
from rr.table_column
where ucd like ’pos.pm;%’
```

Implementation status: Most major providers now have tablesets with their resource records, but UCD quality. . . varies.

4. Coverage: Declaration

VODataService 1.1 (in late RFC right now) adds a few elements to the pre-existing coverage:

```
<coverage>
  <spatial>6/2623,2666-2667,2709,2752</spatial>
  <temporal>55819.5 55819.5</temporal>
  <spectral>2.95119e-19 4.08734e-19</spectral>
</coverage>
```

That’s a MOC (in ICRS) for coverage on the sky, zero or more MJD intervals for time, and zero or more intervals of observed messenger particle energy in Joule.

For solar-system data, non-ICRS MOCs can be defined, but no non-ICRS frames for MOCs have been identified yet, so there are no records using that.

Implementation status: 16800 (of 24000) resources have spatial coverage.

Only 87 have spectral, only 94 temporal coverage. If you have VO resources, please fix their resource records!

5. Coverage: Querying

RegTAP 1.2 (early draft) will define `rr.stc_spatial`, `rr.stc_spectral`, and `rr.stc_temporal` tables closely reflecting the VODataService 1.1 representation. For simple spectral querying, a UDF `ivo_specconv` is planned.

For 2 μm observations in M31, you can run a query like

```
select ivoid
from rr.stc_spatial
natural join rr.stc_spectral
where
  1=intersects(moc(6, circle(10.7, 41.3, 1)), coverage)
and
  gavo_specconv(2, ’um’, ’J’)
    between spectral_start and spectral_end
```

Implementation status: Available on http://reg.g-vo.org/tap. The explicit MOC conversion in the query above is currently still necessary, but it will soon be superfluous. And the future `ivo_specconv` is still `gavo_specconv` as per the IVOA rules for defining cross-service (i.e., ivo-prefixed) UDFs.

6. C14n: Declaration

Full treatment of sensitivities and such is *really* hard. First (and perhaps sufficient) stand-in: column statistics. See:
http://ivoa.net/documents/Notes/colstatnote

For continuous variables:

```
<column
    g-colstat:fillFactor="1.0"
    g-colstat:max-value="14.08"
    g-colstat:median="8.470000267028809"
    g-colstat:min-value="-0.62"
    g-colstat:percentile03="5.96999979019165"
    g-colstat:percentile97="10.739999771118164">
  <name>mv</name>
  <unit>mag</unit>
  <ucd>phot.mag;em.opt.V</ucd>...
```

So, tableset columns are annotated with a fill factor (ratio of non-null values to number of rows), min/max (mainly for VOTable compatibility), "2σ" percentiles, and the median. I would hope that is about enough to cover most interesting use cases for column statistics.

These currently sit on namespaced attributes of column because they don't break VODataService that way. Once this hits VODataService, I plan to add a `stats` child to the column element that would carry these items.

Implementation Status: DaCHS 2.4 and newer can produce this and compute the underlying statistics.

7. C14n: Querying (continuous)

The note is proposing a `num_stat` table with the stat attributes as columns in RegTAP. http://dc.g-vo.org/tap has a prototype of that (`g_num_stat`).

Implementation status: You can do "Give me resources reaching 15^{m} in K" like this:

```
SELECT ivoid, table_name, percentile97
FROM rr.table_column
NATURAL JOIN rr.res_table
NATURAL JOIN rr.g_num_stat
WHERE ucd=âphot.mag;em.ir.kâ
AND percentile97>15
```

– but not much more yet. That's because basically nobody produces this metadata yet, and thus there are only 6265 rows in `g_num_stat` at the moment (full `rr.table_column` has almost 1.3 million). But well, that's what an early draft is about: We're at 0.5% of the way at this point.

8. C14n: Spots of Pain

- Non-reals: We represent our stats as floats in the database; there's no way of having dates, say, in parallel (but they could be put into the tableset).
- TAP_SCHEMA: The stats ought to be there for symmetry with the tableset, too.
- Discrete domains: That's a can of worms. See the note. Ideas welcome.

9. Conclusion

Blind discovery by UCD and spatial constraint is close to being useful. Try it, report bugs, and make it worthwhile for the data centres to provide the metadata.

Discovery by time and spectra is largely stable but suffers from a bad lack of metadata on the side of the data centres. Change this if you can (by fixing your records or complaining to your favourite data provider).

Dealing with column statistics has just started. Get on the boat now! Your ideas are most welcome.