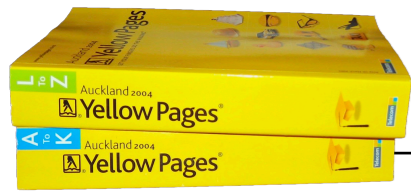


# Linking capabilities with tablesets: an update

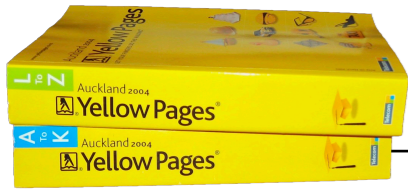
Renaud Savalle (PADC/Obs Paris) and  
Gilles Landais (CDS)

2024-05-21 - Registry WG



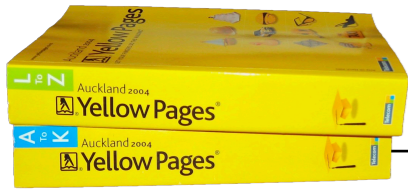
## Outline

- `<capability>` and `<tableset>` in VOResources
- CDS problem and proposal
- Report of Running Meeting
- A new proposal
- Discussion ?



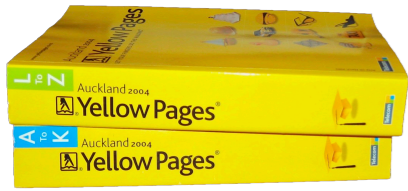
## <capability> and <tableset> in VOResources

- ▼ ● **ri:Resource** "<http://www.ivoa.net/xml/RegistryInterface/v1.0>" Gaia DR3 Part 1. Main source
  - **title** Gaia DR3 Part 1. Main source
  - **shortName** I/355
  - **identifier** ivo://CDS.VizieR/I/355
  - **altIdentifier** doi:10.26093/cds/vizier.1355
  - > ● **curation** CDS
  - > ● **content** Surveys
  - **rights** [https://cds.unistra.fr/vizier-org/licences\\_vizier.html](https://cds.unistra.fr/vizier-org/licences_vizier.html)
  - > ● **capability**
  - > ● **capability**
  - > ● **capability** "<ivo://ivoa.net/std/TAP#aux>"
  - > ● **capability** "**cs:ConeSearch**" Cone search capability for table I/355/gaiadr3 (Gaia
  - > ● **capability** "<ivo://ivoa.net/std/hips#hips-1.0>" Hips catalogue for table I/355/gaiadr3
  - > ● **capability** "**cs:ConeSearch**" Cone search capability for table I/355/paramp (1D astrophysical
  - > ● **capability** "**cs:ConeSearch**" Cone search capability for table I/355/paramsup (Additional
  - > ● **capability** "**cs:ConeSearch**" Cone search capability for table I/355/epphot (Light
  - > ● **capability** "**cs:ConeSearch**" Cone search capability for table I/355/rvsmean (RVS
  - > ● **capability** "**cs:ConeSearch**" Cone search capability for table I/355/xpsummary (Auxiliary
  - > ● **capability** "**cs:ConeSearch**" Cone search capability for table I/355/xpsample (BP/RP
  - > ● **capability** "**cs:ConeSearch**" Cone search capability for table I/355/xpcont (Mean
  - > ● **capability** "**cs:ConeSearch**" Cone search capability for table I/355/spectra (Gaia
  - **facility** Gaia
  - > ● **coverage** 0/0-11 6/
  - ▼ ● **tableset**
    - ▼ ● **schema** default
      - **name** default
      - > ● **table** I/355/gaiadr3
      - > ● **table** I/355/paramp
      - > ● **table** I/355/paramsup



## Initial discussion and Running Meeting

- ▶ <http://mail.ivoa.net/pipermail/registry/2024-March/005572.html>  
CDS: “We didn't find any clean method to link capabilities with tablesets described in a VORegistry record.” ...
- ▶ [IVOA\\_Reg\\_RM20240506.pdf](#)
  - CDS described a new proposal
  - A lively discussion followed



## CDS proposal to link tablesets with services in the registry



**Issue:** one resource in registry possibly contain a collection of datasets and a collection of services but it is **not possible to attach a service to the datasets**.

→ A general issue in VizieR, Esa, Ukidss, etc.

example: ivo://CDS.vizieR/i/355

**Consequences:** clients (eg: PyVO, registry module) are **not capable today to link properly the datasets with the services**

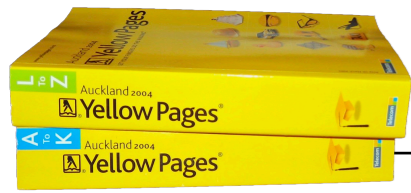
Examples :

- VizieR catalogue:  
<https://cdsarc.cds.unistra.fr/viz-bin/cat/J/AJ/165/45>
- VizieR Notebook:  
<https://cdsarc.cds.unistra.fr/vizieR/notebook.gml?source=J/AJ/165/45>

### Proposal: Update VOResource

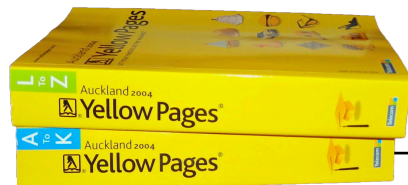
```
<capability xsi:type="cs:ConeSearch"
  standardID="ivo://ivoa.net/std/ConeSearch">
  <services>
    <reftable name='table1' />
    <reftable name='table2' />
  </services>
</capability>
...
<tableset>
  <schema>
    <table>
      <!-- todo ADD CONTENT-TYPE HERE -->
      <name>table1</name>
    </table>
  </schema>
  ...
</tableset>
```

**Don't break  
the current  
framework**



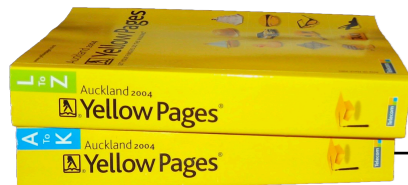
## The Resource Granularity Problem

- ▶ “collections” in VOResources seem to be the problem
  - Legacy: Vizier Catalog  $\Leftrightarrow$  VOResource: 1-1 mapping
  - History: VOResource first written **manually**: factorization of metadata
- ▶ And also...
  - High granularity VORs have caused a similar problem with TAP before
  - Solved with #aux cap - cf Endorsed Note Discovering Data Collections within Services (2019)
- ▶ BUT
  - Such large VOR tend to “hide” their details, hindering discovery efforts:  
Ex: which table(s) ? is concerned by a IsServedBy relationship?
  - Redundancy not an issue since VORs are programmatically **generated**



## The “mini-VOResources” pattern

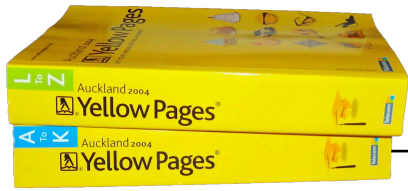
- ▶ Split such “large” VOResources into smaller ones
  - Associate *closely related* <capabilities> and <table>s
  - Limit case (SCS): 1 <capability> and/or 1 <tableset> with 1 <table>
- ▶ Benefits
  - No change in Standards, just a “new” usage pattern
  - Solves CDS issue: a service is naturally attached to a (set of) table(s)
  - Several VOR elements become more accurate (as they apply to one 1 capability/table): <source>, <altIdentifier>, <coverage> etc.
  - Useful when we’ll add <productTypeServed>
- ▶ *Closely Related* ?
  - Can discoverability drive this criteria ?
  - Ex: Tables describing *similar kind of* data (same productTypeServed) ?
  - NB: ~~“Tables which can usefully be joined together: not a good criteria~~



## Impacts of using “mini-VORs”

- ▶ For existing Publishers:
  - Discoverability will increase only when you “upgrade”
  - Upgrade could be costly for some (design new publishing layer)
  - Some ivoids may change (OK, they are not PIDs)
  - + Need to use new <relationship>s to GROUP mini-VORs together
  
- ▶ For Users:
  - They will see more resources in results
  - + They can discover better “labeled” data more easily
  
- ▶ For Harvesters:
  - may see same DOIs attached to several resource. Is it an issue ?
  
- ▶ For Clients Apps and their ADQL queries:
  - They might have to display related VORs using <relationship>s





## Conclusions

- ▶ Consensus:
  - Discoverability is our challenge, and should drive changes
  - Small resources ease data discovery
  
- ▶ CDS will explore the impact of their proposal on discovery queries
  
- ▶ Maybe it's not necessary to modify the schema...  
...but the way we use it ?