

Registry Working Group - IVOA InterOp Oct 2022



Organization: Chair: Renaud Savalle, Vice-Chair: Tess Jaffe

Mailing list: registry@ivoa.net

Slack: [#registry](https://ivoa.slack.com)

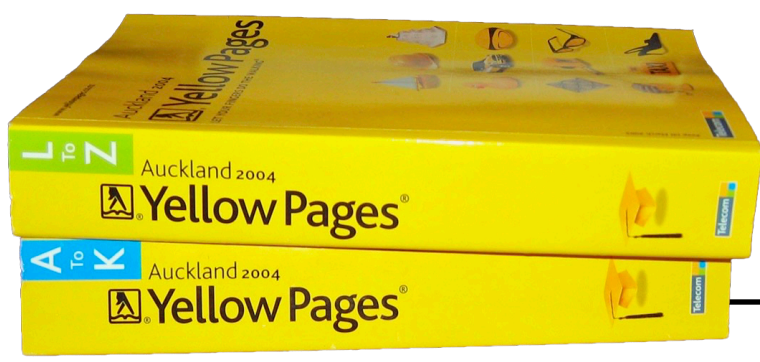
Session#10 Summary (1/2)

1. Christophe Arviset and Mattias Wångblad - Euro-VO Registry Status and Plans

- Full searchable and publishing registry with Search and Service Validation functionalities + Registry Ecosystem Health Reports
- **New staffing** will allow code refactoring and fixing validators with an agile dev approach
- Version 4.0 planned for summer 2023 will feature **REST API** for resource validation, and keep up to date with standards

2. Markus Demleitner - StandardsRegExt, Endorsed Notes, and why we should bother (a review)

- An IVOA REC Registry Extension designed to store metadata about IVOA documents offering Standard Keys used in IVOIDs
- New need to register the Endorsed Notes (ENs)
- Clarification of what to register there → RECs and ENs requiring Standard Keys and/or standard_ids
- Editors will write StandardsRegExt records with the help of the **ivoatex** package
- Management of Standard Keys needs to be re-considered
- → Need for new revision: **StandardsRegExt-1.1**: Editor job accepted, help welcome !



Registry Working Group - IVOA InterOp Oct 2022



Organization: Chair: Renaud Savalle, Vice-Chair: Tess Jaffe

Mailing list: registry@ivoa.net

Slack: [#registry](https://ivoa.slack.com)

Session#10 Summary (2/2)

3. Markus Demleitner - A New Registry API for pyVO

- A new Registry Search Interface to discover data collections, implemented in **PyVO** v1.4
- New design pattern with constraint classes, ex: registry.UCD() registry.Spatial()
- **ADQL** query automatically generated, get_RegTAP_query() will show it → learning tool!
- A list of resources is returned, from which **DAL** services can be called
- **Demo** with **Jupyter notebook**, (see session video @38:00),
- Ranking results (by usage? availability? validation?) still an open issue...

4. Tom Donaldson - Integrating PyVO with the NAVO Registry + Discussion

- PyVO 1.4 and MAST Registry: issues found while working on **Jupyter notebooks** for a NAVO workshop. Corrective actions:
- API should allow choice of which registry to query at runtime with regsearch()
- PyVO could then add automated compatibility testing across several RegTAP implementations/sites via CIs
- **TAP** hard output limit was incorrectly required → solved (no longer is)
- Some *not-yet-standard* **ADQL** features were used (COALESCE, UNION) by API but missing from servers. Workarounds possible.
- Food for thoughts:
 - Implementation must be tested before REC but can break interoperability
 - REC does not mean immediate wide adoption !
 - Users and **Apps** must consider addressing different versions/sites: use **VOSI** /capabilities and “experimental” PyVO features