

# **InterOpJune2025 College Park, USA: Registry Session #1: Tuesday June 3, 2025 @11:00-12:30 ET (Session #3) Room 1309**

**Renaud Savalle (PADC) (remote) Intro and Documents  
Status**

**Tess Jaffe (NAVO) (in person): The new Registry GitHub  
Project board and Spring Cleaning hackathon objectives**

**Hendrik Heintz (INAF) (remote): Onboarding to the VO -  
- Services, Data(models), People**

**Renaud Savalle (PADC) (remote): StandardRegExt and  
OpenAPI**

**Gilles Landais (CDS) (remote): Use of VODataService  
for Vizier records**

# Markus Demleitner (Univ. Heidelberg/GAVO) (in person) : Metadata for continually updated resources

## All: Discussion

Intro: Renaud reviews standards recently updated, in progress, or planned to be updated. Try out searching for resources with UAT terms in a new PyVO as implemented by Markus using the Registry metadata.

TJ roughly outlines what the hackathon is about

Hendrik Heinl: Talks about one thing that leads to the problems to be discussed in the hackathon, namely new service providers who don't find the VO very easy to understand. How do we onboard these providers? Workshops online. "House calls"? Funding is of course the issue. Work starting on SimDM. New generation of users of the VO looking for existing solutions. Standards/services are not always aging well. Need to harness the former to help with the latter. Discussion of how useful VESPA workshops are and how nice it would be if they were better funded

Renaud on StandardsRegExt work: see slides. Gilles comments on was: DCAT uses OpenAPI... to investigate

Gilles: Vizier records in the Registry. 20k records, each with its own DOI and Registry metadata. EOSC (interdisciplinary) harvest the IVOA registry (TJ note: NASA's new Science Discover Engine similarly). Compare Vizier, GAVO, and EOSC, different numbers of resources related to "agn". Some resources include more than one catalog but a set of them, and we don't have a good way to handle this. See slides with rather non-trivial diagram of a possible solution and its pros and cons.

Markus points out that we need to think through the data discover use cases. Going through relationships gives him the creeps. The real problem is the Cone Search. What if we change Cone to work like TAP with one service and one capability. Gilles: doesn't think this fixes anything. Suppose you search for datasets with "parallax". Then how do you query it? This could be solved by updating cone to take the table name. But Gilles points out that there may be multiple tables in that resource and one has parallax and another in that same resource does not.

We conclude we need another running meeting and Markus wants us to focus on how it would look in the Registry.

Markus continuously updated data and the registry. Two use cases: 1. temporal coverage ; 2. declaring specific updates so a user who already checked the resources knows to come back and look again. Use `stc_temporal` with the `time_start` and `time_end`. One idea, if you update your data, then update your registry. But that's hard for lots of providers. Register with the estimated lifetime of the project? Update Registry regularly at a cadence you can handle, 5 years, whatever. Then there's a need for something in the `res_date` table with `value_role="updated"` and a time range. So you can ask the Registry whether the resource has been updated since the last time it was updated.

Gus says, what if the provider just puts a very large range that isn't helpful to the user. Curators need to know about this.

JoshuaF asks what else is in there about curation related dates. Markus: 1. date that the resource itself was updated vs when the record in the Registry was updated. The IVOID though refers to both the registry record and the data resource itself. Library science uses two different IDs.

PierreLS: What about a data stream?

Giles asks about the proliferation of dates. Vizier has catalogs that don't have a planned end. Just flag it as something that is regularly updated so check every time. Markus says providers won't come back to say when it's over. Markus' proposal has a built-in date so users don't query them unnecessarily, or be overwhelmed by a bunch of resources who never said they were no longer being updated.

Anais: data updated every second like SIMBAD. Then you always have to query the service. which TJ points out that this is a choice the publisher can make by setting the coverage far in the future.

Anne: Coverage date; processing date; curation date. Not cleanly separated in PDS systems. Markus says we have these clear in the registry. Gus says users need to accept that datasets can change. A DOI should be changed when a specific dataset is now different. But the user has to know what they want.

PierreLS: we've been discussing this a long time, how much info in the registry, where to put the complexity. Not sure coverage is well enough defined or maintained. People can't and don't trust it. Markus: but we still need the standard. Maybe talk about how well these coverages work on Thursday?

# **InterOpJune2025 College Park, USA - Registry Session #2: Spring Cleaning Hackathon: Thursday June 5, 2025 @14:00-15:30 ET (Session #12) Room 1309**

Registry hackathon:

We split into three groups

- o HEASARC worked on our subject keywords. TJ's notebook turned up a bunch that are invalid. The group was headed by Antara with Markus as consultant, with other HEASARCers Michael Preciado, Meredith Gibb, and Ed Sabol on line. This exercise pointed out a gap in HEASARC's metadata pipeline, that we will now fix.
- o MASTers Joshua and TheresaD saw something odd in TJ's notebook regarding some MAST registry queries. They dug through their infrastructure to find a nasty bug deep down that would affect multiple things, and know how to fix it when they get the right team clued in.
- o TJ went through the notebook with the remainder of the attendees who weren't familiar with the notebook or didn't understand what the queries were doing. We talked about the queries and how

useful it is to have the whole Registry Database structure printed out in a sheet on your wall. Several useful suggestions were made or explanations, that TJ will use to update the notebook.

- \* Sébastien said for instance that obs.field is ok as part of a UCD but not by itself. It has to be something like "meta.id;obs.field".

- \* TJ asked about whether it's really going to work if we have large string MOCs in the Registry for a lot of resources. How efficient will that be?

- \* Somebody (sorry!) suggested checking the units as well. There's a TAPLINT validator to check the units.

Joshua has some tooling that might be useful in <https://github.com/jwfraustro/vounitlex>.

At the end, we were all happy enough that we'd like to do a Southern Spring Cleaning in Goerlitz.

Update: the MAST issue was fixed by the backend DBA less than 2 hours later. Good job!