DAL+Registry Session Notes

IVOA Interop. - Nov. 2025

2025-11-14

Simple Cone Search 2 and Transitioning To It (Markus Demleitner)

- Output data model: each row = ID + position (ra, dec)
- Alternatives:
  - using SIAP2-style POS attribute
  - DM markup with utypes
  - drop VERB
- Why new CS? Make it compatible with DALI
  - DALI compliance:
    - mandatory: vosi capabilities and tables, scs2 endpoint name, RESPONSEFORMAT, MAXREC
    - optional: SIAP2 POS parameter (make sense?), UPLOAD parameter (a table with RA, DEC, SRC columns for multicone ; maybe to complicated to require it)
- Currently: one service for each data collection
  - Solution: a TABLE parameter ; the table to target
- Transition plan:
  - a lot of failed attempt in the past ; it is hard
  - Goal: no more ConeSearch-1.x service at the end of the transition
  - Current transition plan:
    - After SCS-2.0 (2026), deprecated SCS-1.x (mid 2027).
    - Then need of a transitioning team, when 90% of publishers have taken up SCS2 (2028), clients should raise a warning.
    - A migration team in 2030.
    - After 1-2 years, declare SCS-1 invalid. This is an optimistic plan.
  - No code written yet. Call for volunteers.

- Questions
  - ??: timeline is pretty long. Taking the example of Google. Should be better to have it shorter. It would be more effective with a shorter timeline.
  - MM: For me it is actually a short timeline. IVOA provides standards to live on a long time period. There is no obligation to upgrade services ; that's why it takes time.
  - MD: We are not Google. We work in consensus. However it means we have to walk slower.

Finally: Towards TAPRegExt 1.1 (Markus Demleitner)

- TAP has a lot of SHOULD and MAY
  - Examples:
    - may support several lang
    - languages with optional features
    - uploads optional,
    - ...
- TAP has 2 exec modes: sync and async (queue and batch queries)
  - Some services have different limits for each mode (ex of DaCHS: 10s in sync 7200s in async)
- 1st attempt:
  - idea: specify the default execution mode (see TAPRegExt PR#8 on GitHub)
  - 1st design: limit groups depending on mode especially anonymous queries
  - abandoned idea ; not unanymous solution
- 2nd attempt
  - a better alternative proposed in PR#8 discussion:
  - an additional XML attribute for each limit: `forMode`=sync|async
  - no need for language feature yet
- DALIInterface:
  - TAP services registered with their base URL => bad practice
  - See http://ivoa.net/documents/caproles document
  - Upgrade idea: have a root base URL in <accessURL> and as many <endpoint> elements as endpoints (sync, async, ...)
  - Like to attach RDF triples to endpoints using RDFa attributes ; example: a meta element with a property hasArchived in a given endpoint element
- Deprecate dataModel:
  - this should be a property of the tables and schemas not of TAP capabilities
- What's implemented so far:
  - in DaCHS: per mode limits (but PyVO ugly warnings), DALIInterface (but not anywhere in production yet)
- Opinions:
  - Do you find merit in the limits element?
  - Do you find fault with forMode?
  - Are there modes other than sync and async?
  - How scared are you of DALIInterface?
  - Should it go to DALI in the first place?
  - How scary is RDFa in VOResource to you?
  - Would you like to wait with deprecating dataModel?

Questions:

GL: interested in the RDF. RDF does not appear in the examples s10.
MD: It is an IVOA XML piece. Besides, it is RDFa and not RDFx. It is possible but not required.

The Euro-VO Registry is adding validation support for the Object Observability Simple Access Protocol (Erik Mellegård)

- ObjObsSAP: Object Observability Meeting
  - 1st attempt: ObjVisSAP in 2018
  - Goal: retrieve constraint free observability time intervals for given object coordinates

- Demo
  - EuroVO Registry Webpage
  - Several service types supported, and now ObjObsSAP too
  - Give the root URL of the service, run and you get the validation result (here the Joshua's service as the SWIFT one does not work for few days). Then you get the compliance level (here A++), and a list of expected requirements or warnings defined the ObjObsSAP document.
- An HTTP API is also available. It returns the same validation result in an XML document.

- Questions:
  - ??: Is it intended to be integrated to registered services or just for standalone services?
    - EM: Both. It is run auatomatically at a given frequency for registered services.
  - P Skoda: Can it be extended to given a time interval in order to target transient objects.
    - EM: Probably. Have to check the specification. The standard can also be extended with custom parameters.

Registry Ideas (Paul Harrison)

- Have created a new Publishing Registry Implementation
  - Enables x-query
  - Interfaces:
    - OAI-PMH interface
    - HTTP basic auth
    - Swagger-UI
    - Dockerized solution
- Possible new standard interfaces
  - search interface => just stick to RegTAP
  - retrieval interface:
    - idea: returns all or parts if record in JSON given the IVOID
  - admin interface:
    - trigger immediate harvest
    - std upload
- Other approaches
  - SPASE
  - NAS PDS
- DM solution (refactor)
  - clear potential in DM, Regsitry and Semantics
  - ProposalDM needs concepts like Observatory, Telescop and Instrument
- DataCollection to model services so that they serve the same dataset use case
- DataResource is a service and so is expected to have Capabilities (but which is optional)
- Feedback are welcome on this new Publishing Registry Implementation

Questions:

- MD: registry output in JSON. Most (TAP) services provides a JSON output, but not standardized.
- MD: those interfaces should be in the publishing registry. Who would query them?
- PH: It would be searchable registries.

Simulations in the registry (Hendrik Heinl)

- SimDM goal is to describe data coming from simulations
- Complexity of SimDM has lead to no existing service using it now.
  - It is not a bad standard, but just a bit overwhelming.
  - questions at different variable granularity (e.g. for specific physics down to single parameters)
- What's missing?
  - A use case. What question a user should ask to the registry to find services using SimDM
    - Analogously: how to find specific data collections. What question should be asked?
  - What's needed to make it meaningful?
- SimDM diagram:
  - a lot of things but people are generally interesting to the kind of physics being simulated ; a small but important part of SimDM.
  - There is a label element in the Physics element of the SimDM standard that depends from a vocabulary (PhysicalPhysics) that does not exist ; broken link.
- How to solve the lack of vocabulary?
  - use an IVOA vocab. which will have to be maintained.
  - An erratum would be needed in SimDM to fix the link to the Vocabularies.
- Is high granularity...
  - ...a curse? describing experiment with parameters is difficult
  - ...a blessing?
- What's next?
  - To correct the voca bularies with an erratum
  - to find reliable way to maintaing vocabularies; volunteers are welcome
  - To find examples
- the TheoryIG is needed but no more active ; volunteers?

Questions

- James T: part of the challenge is to describe everything. Suggest to split this depending in several use cases. Provide with a TAP service.
- HH: agree and disagree. Not easy to find in a TAP service. Prupose of this service is different as the other, so not easy question
- ??: most people from simulations may even not agree on most of the parameters. THe problem is not SimDM, but because people cannot agree on what data and their meaning to put their.
- HH: having experience doing simulation, who are not very enthousiast to publish data this way.
- Kaï P: Support publication of simulations. We need simulations.
- MD: SimDB in therory sane. It will take years to make it used. Let's try. The document can be improved too. Need a flag in the registry to say these data are simulated or observed ; let's have a look at the GitHub repo for the related Issue.

Discussion (All)

- MD: People publishing ObsCore servicess should take a look at https://ivoa.net/documents/Notes/TableReg Note which contains good practices.