NRC·CNRC

Toward TAP-1.2

Patrick Dowler
Canadian Astronomy Data Centre

IVOA Interop 2026 / Perth

Overview

- What’s new in TAP-1.2?
- Which standards does TAP-1.2 depend on (details)

TAP-1.2: user managed tables

- TAP-1.1 provided access to existing tables
- TAP-1.2
  - main new feature: users can CREATE, LOAD, and DELETE tables
  - secondary: users can control permissions
  - secondary: users can create indexes

  - still to figure out: users and services can negotiate table lifetime
  - still to figure out: visibility of tables in tap_schema, registry records

Relevant Standards

- TAP-1.1 depends on primarily on DALI, VOSI, UWS, VOTable
- VOSI-1.1 defines /tables:
  - GET /tables – list schema(s) and table(s)
  - GET /tables/{table name} – table details
- naturally, WD-VOSI-1.2 extends this with optional
  - PUT, POST, DELETE /tables/{table name}

Relevant Standards

- CREATE INDEX : inherently an async operation
- currently, WD-VOSI-1.2 defines new optional endpoint
    - POST /table-ops – create UWS job
- so far so good, it works (see previous talk by AK)
- but it leaves some gaps…

Relevant Standards

- user can create an index, but they cannot see it (list)
- users can create an index, but they cannot drop it, e.g. if they make a mistake or it just doesn’t work as intended
- as noted earlier, tap_schema.columns.indexed flag is not sufficient for multi-column indices (not a new problem)

Questions – Ideas – Discussion

- Question: is DROP INDEX also async?
  - could add to VOSI-1.2 table-ops job
- Question: add tap_schema.indexes table in TAP-1.2? optional?
  - would satisfy “list indexes”, but then create|drop index is specified in VOSI and list is specified in TAP (ugh)... move table-ops to TAP?
- Question: update VOSI table metadata to describe multi-column indexes? (actually VODataService)
  - better registry metadata, another update dependency

questions?

