DAL & utypes

some thoughts on how VO-DML and the utype mapping proposal effect DAL, now and in the future

DAL & utypes

- VO-DML is for data modellers, but does constrain utypes syntax

  - <prefix>:<identifier>

  - identifiers are opaque strings

- would prefer a limited charset for identifiers

  - no URL-encoding required

  - could be used in URIs (where? TBD)

  - no use case, just future-proof

DAL & utypes

- utype mapping proposal is all about GROUPs
    - replaces/standardises:
        – STC-in-VOTable note(s)
        – PhotDM-in-VOTable note
- GROUP metadata: optional
    - can be added to VOTables now
    - does not break anything
- GROUP metadata: required
    - makes services incompatible
    - requires major version update

DAL & utypes

- ## mapping proposal
  - does not specify use of utype attribute of FIELDs or plain old PARAMs
- ## options for FIELD utypes (query response)
  - DAL specs could use those utype fields
    - to preserve app compatibility
  - DAL specs could leave them unspecified
    - implementors would be free to use them for their own purposes (e.g. to simplify your own apps using your own services)s
  - a future version of utypes mapping might specify it: DAL would care

DAL & utypes

- VO-DML, utypes, and mapping: DAL has nothing to worry about

- implementors **should** add VO-DML GROUPs to make their output metadata-rich

- DAL: use FIELD utypes to maintain app compatibility

  - until next major version: it gets broken anyway

- DAL: require VO-DML GROUPs and could abandon FIELD utypes at next major versions