

Data Access Layer Interface

DALI

Purpose of a DALI Spec

- identify standard practices in DAL services
 - specify in a single standard for reference by future DAL standards
 - save 10+ pages per DAL specification
 - enable re-usable software (client & service)
- promote consistency in future specifications
 - identify inconsistent usage and **stop!**

DALI

- standard request execution using REST-ish
 - resources: list of jobs
 - synchronous execution: DALI-sync
 - asynchronous execution: DALI-async
- execution pattern is orthogonal to nature of the job

DALI-async

- defines how DAL services use the UWS pattern
 - job specified as parameters (key=value)
 - some parameters specified in DALI
 - some parameters are service-specific
 - specify how results are returned
 - Constrain how results are named?
 - specify how errors are reported (VOTable)
 - specify how overflow is handled (VOTable)
 - Try to define this in CSV? TSV?

DALI-sync

- defines pattern for synchronous job execution
 - jobs specified the same way as w/ DALI-async
 - equivalent use of GET and POST
 - explain allowed/optional use of HTTP redirects
 - service may return response directly
 - service may redirect: PRG pattern

actual simple AP URLs with query strings not harmed by this proposal :-)

DALI-async & DALI-sync

- two web resource patterns
- service specs define the jobs (parameters)
- service specs say which they use, or both
 - could be same job in both (e.g. TAP)
 - could be multiple types of jobs on one type
 - could be some job types on one, some on the other

Standard Parameters

- REQUEST

- specifies the service operation
- allowed values are service specific, must be at least one

- VERSION

- service version negotiation
- client specifies what version of the protocol they are using, service uses it to interpret all other parameters

Standard Parameters

- FORMAT
 - value can be a mimetype or a shorthand value
 - supported values are specific service
 - meaning and usage should be consistent, but it is not...

Standard Parameters

- **FORMAT**
 - in TAP, it controls the output format
 - e.g. `FORMAT=application/x-votable+xml`
 - e.g. `FORMAT=text/csv`
 - in simple AP, it is used as a query parameter (e.g. a PQL param) that constrains the type of the data
 - e.g. `FORMAT=application/fits`
 - e.g. `FORMAT=image/jpg`

Standard Parameters

- **FORMAT**
 - in TAP, it controls the output format
 - e.g. FORMAT=application/x-votable+xml
 - e.g. FORMAT=text/csv
 - in simple AP, it is used as a query parameter (e.g. a PQL param) that constrains the type of the data
 - e.g. FORMAT=application/fits
 - e.g. FORMAT=image/jpg

- ObsCore has a field called access_format

LANG=PQL&FORMAT=application/fits

Standard Parameters

- FORMAT
 - option 1: DALI specifies FORMAT as output format of the request being performed
 - other specs cannot override this meaning
 - PQL would need to define AFORMAT (e.g.) as to constrain ObsCore.access_format
 - option 2: continue to use FORMAT in custom ways (inconsistently)
 - other specs define meaning
 - issue for format negotiation in simple APs, e.g. getting text/csv results – which users want!

Standard Parameters

- MAXREC
 - user-specified limit on the number of records
 - data discovery only?
 - service may have default and maximum value
- RUNID
 - services should accept this string value and log it in activity logs
 - should propagate it if making other web service calls as part of job

rules: case sensitivity

- case
 - parameter names are not case sensitive
 - shown in upper case in documents and examples for clarity only
 - parameter values are case sensitive unless specifically stated otherwise

rules: literal values

- literal values
 - numbers
 - dates are iso8601:
 - e.g. 2010-12-10 (yyyy-MM-dd)
 - datetime are iso8601 with T separator:
 - e.g. 2010-12-10T14:30:45 (fractional seconds optional)
 - date and datetime values are always UTC
 - boolean: 1, true, 0 , false
 - case sensitive?

rules: range of values

- range of values: slash-separated pair
 - A/B means A to B inclusive
 - /B means less than or equal to B
 - A/ means greater than or equal to A
 - ranges can be used with numbers, dates, and datetimes?

rules: list values

- list of values: comma-separated list
- list can be unstructured:
 - e.g. GENRE=folk,classical,jazz
- list can be structured:
 - e.g. POS=20,30
- structured vs unstructured is specific to the parameter

rules: combine list and range

- list and range may be combined to mean a list of ranges
 - e.g. FOO=1/2,4/5,8/

rules: qualifiers

- qualifier appended with semi-colon separator
- used to modify meaning of previous values
 - e.g. POS=20,30;GALACTIC

rules: cardinality

- current specs only allow one use of each parameter
- use list of values
- current practice is that a DAL service chooses one arbitrarily and ignores others
 - Should it be an error instead?
 - Should we allow multiple use of same parameter?