

DAP4

James Gallagher & Ethan Davis

OPeNDAP and Unidata

Motivation

- Long-standing goal to correct flaws in DAP2
 - Sequence constraints ill defined
 - Support for curvilinear grids lacking
- Other related technology has advanced
 - Groups in HDF5 & NetCDF4
 - CDM Shared dimensions
- Web Services now pervasive
 - leverage this to increase DAP's usefulness

Goals

- Define DAP4
 - Data model
 - Web services
 - Extensible
- Publish specification
- Release two servers
- Release two clients

DAP4 Specification

Volume 1: Data Model and Constraints

Volume 2: Web Services

Extensions:

- CSV/Text (aka ASCII) data response*
- NetCDF data response*
- JSON responses*
- Asynchronous response

DAP4 Data Model

- Coverages: Replace Grids with a more general model: Arrays, shared dimensions and maps
 - 'Coverage' is slang for 'discrete coverage' (cf OGC)
 - Maps define the Domain of a discrete function
 - The Array defines the Range
 - Shared dimensions provide the connection between Domain and Range
- Tabular data: Sequences (tables that 'nest')
- Projection: subsetting based on data *types*
- Filters: subsetting based on *values* (Seq now; Arrays maybe?)

DAP4 Web Service

- Design to support simple scripts
 - Resource types identified by standard file extensions (“.dsr”, “.dmr”, “.dap”)
- Designed with REST in mind
 - Resource types identified by content negotiation
 - Hypertext with links

The Services Response

- DSR: Dataset Services Response
- DSR: Dataset Services Response is the 'service endpoint'
- This contains links to all of the other responses available from the server for a specific dataset
- A key point for/of REST
- In DAP2 the 'base URL' was undefined; in DAP4 the base URL for a dataset is the DSR

Content Negotiation

- DSR provides links to all of the other responses
- However, a server must implement Content Negotiation.
- i.e., a browser asking for the DMR winds up displaying a (HTML) web interface while a smart client parses XML

Dataset Metadata

- Called the DMR, this holds the information from DAP2's DDS & DAS
- Encoding: XML
- This response holds all of the variables and Attributes
- This defines the environment in which the Constraint will be evaluated

Data and Constraints

- As with DAP2, Data is returned in a two-part response
 - Binary uses 'reader makes right' (not XDR)
- Can request one or more variables and/or Structure fields
- Can 'slice' the dimensions of array variables
- Can filter Tabular (Sequence) data
 - A flaw in DAP2 where multiple Sequences could not be requested has been fixed
 - The DAP4 'filter' syntax could also support Array filtering

Status

- Project ends on 31 July 2014
- Budget cut (remember the recession ;-)
- But, it's actually in pretty good shape...

Goals - Revisited

- Define DAP4 - Yes
 - Data model
 - Web services
 - Extensible
- Publish specification - On the web
- Release two servers - Hyrax & TDS close
- Release two clients
 - netCDF Java; C not yet
 - C++ library

Online Information

Specification: Online at docs.opendap.org*

Test Servers: Hyrax-based**; TDS-based

Software: C++ (libdap), Java (netCDF Java)

DAP4 Adoption Use Cases

Use cases GSFC

Other use cases

Topics:

- New data model elements

- New web services

- Extensions - new return types and behaviors

UC1 Subsetter

Spatial/Variable Subsetter. Our Simple Subset Wizard (SSW) works by constructing URLs to download spatial / variable subsets from the server as netCDF files (<http://disc.gsfc.nasa.gov/SSW>). This has allowed us to stop writing custom subsetter programs for each dataset or group of datasets.

- Constructing URLs is different but this difference is primarily syntax and not semantics
- DDS & DAS replaced by the DMR means new parsing code must be written OR a new library used
- The CE syntax is slightly different (although slicing arrays uses the same syntax with some new features that the code might find useful)
- The DSR is completely new and using that could simplify working with new data sources

UC2 Giovanni

Giovanni (<http://giovanni.gsfc.nasa.gov/giovanni/>) provides data exploration capabilities for datasets from GES DISC as well as a few select other providers. Data are acquired from servers via OPeNDAP, saved as netCDF/CF1, which is the Giovanni internal standard. This allows the deployment of many netCDF-capable tools for analysis and data manipulation purposes.

- Wait for netCDF-C to support DAP4
- NCO already supports netCDF enhanced data model (not sequences)
 - Should work without change on any dataset not containing sequences

UC3 MapServer vrt files

MapServer provides WMS services for GES DISC data. Rather than site individual instances on each data server, we use the .vrt (Virtual files) capability in MapServer to set up the OPeNDAP connections needed to acquire data for the MapServer.

- This will depend on GDAL updating to support DAP4 - not currently planned, but also not overwhelmingly hard - there is an active user community for GDAL and it may update the 'DAP' driver. The code currently uses libdap (C++) which supports DAP4.

UC4 NcML

We are currently experimenting with the NcML aggregation capability to see if it is feasible to offer the ability to generate time series at a point using just Hyrax and the NcML handler.

- The NcML handler in Hyrax will need modifications to work with DAP4. This may take significant effort since the NcML software is moderately complex.
- Complicating the above, Unidata may drop support for NcML aggregation

UC5 Sequence filtering

We have code that filters sequences

- The syntax for filtering sequences has changed, but not much
- The old syntax: Given a Sequence 's' the old (DAP2) filter is 's.time, s.lat, s.lon, s.sst & s.lat>10 & s.lon<20'
- The new (DAP4) syntax is:
 - 'd4ce = /s | lat>10, lon<20' or
 - 'd4ce = /s.time; /s.lat; /s.lon; /s.sst | lat>10, lon<20' or
 - 'd4ce = /s.{time;llat;lon;sst} | lat>10, lon<20'
- Note that the new syntax does not use '&' so we can use the query string of the URL as per the IETF specification
- We can ask for two Sequences and provide separate filters for both:
 - 'd4ce = \s1 | lat >10, lat < 20; \s2{time,sst} | lon >-80, lon <-20

UC6 Using netCDF-C

My client application uses the netCDF-C library for OPeNDAP access.

- Wait for netCDF-C to support DAP4
- Needed changes depend on how netCDF-C is used and on the structure of the dataset being accessed
 - If dataset is recognized by netCDF-C as a classic model dataset:
 - Don't need to change anything!
 - If dataset is recognized by netCDF-C as an enhanced model dataset
 - No change if client understands enhance data model
 - Otherwise, need to update for enhanced data model.
 - DAP4 sequence filtering is not yet supported in netCDF-C