Making data access easier with OPeNDAP

James Gallapher (OPeNDAP™)
Duan Beckett (BoM)
Kate Snow (NCI)
Robert Davy (CSIRO)
Adrian Burton (ARDC)
Outline

- Introduction and trajectory (James Gallapher)
- OPeNDAP at BoM (Duan Beckett)
- OPeNDAP at NCI (Kate Snow)
- OPeNDAP at CSIRO (Robert Davy)
- Discussion and conclusion (Adrian Burton)
Making data access easier with OPeNDAP

James Gallagher
Thanks to Dave Fulker and Peter Fox
Outline

- Definitions
- About the protocol (aka Web API)
- Servers that implement the protocol
Distributed Oceanographic Data System (DODS)

➢ Conceived in 1993 at a workshop held at URI.
➢ Objectives were:
  – to facilitate access to PI held data as well as data held in national archives and
  – to allow the data user to analyze data using the application package with which he or she is the most familiar.
➢ Basic system designed and implemented in 1993-1995 by Cornillon, Flierl, Gallagher, and Milkowski with NASA funding.
➢ From 1995 to 2003 it was extended with NASA, NOPP, NSF and NOAA funding.
Some Definitions

**DAP = Data Access Protocol**
- Model used to describe the data;
- Request syntax and semantics; and
- Response syntax and semantics.

**OPeNDAP**
- The software;
- Numerous reference implementations;
- Core/libraries and services.

**OPeNDAP Inc.**
- OPeNDAP is a 501 c(3) not-for-profit corporation;
- Formed to maintain, evolve and promote the discipline neutral DAP that was the DODS core infrastructure.
Some Definitions

Syntax
➢ The computer representation of a data object - the data types and structures at the computer level; e.g.,
➢ T is a floating point array of 20 by 40 elements.

Semantics
➢ The information about the contents of an object; e.g.,
➢ T is sea surface temperature in degrees Celsius for a certain region of the Earth.
Considerations with regard to the development of OPeNDAP

- Many data providers
- Many data formats
- Many different client types
- Many different semantic representations of the data
- Many different security requirements
OPeNDAP

Fundamental Concept

- URL ≈ dataset* | URL with constraint ≈ subset
- Retrieve dataset descriptions (metadata)
dataset content (typed/structured)
- Retrieval protocol employed in many packages
  - flexible data typing
  - many, diverse clients
- arrays (~coverages)
tables (~features)

*dataset ≈ (file/granule | collection)
OPeNDAP
Datatype Philosophy

- Every dataset is a collection of variables
  - Variable: name, type, value(s) and attributes
  - Attribute: name, type and value(s)
- Internal data model has few data types
  - For simplicity…
- Types are domain-neutral but flexible
  - Structures & attributes ➔ rich syntax & semantics
- These types support many domain-specific needs
**URL ≈ dataset**
per OpenDap’s Data Access protocol (DAP)

---

<table>
<thead>
<tr>
<th>Domain name often is an organization’s web server.</th>
<th>Servers often have hierarchical collections.</th>
<th>Each URL references a distinct DAP “dataset.”</th>
<th>Suffixes specify return types.</th>
</tr>
</thead>
</table>

Depending on suffix, DAP returns metadata or content, with options for human- or machine-readable forms (XML, NetCDF4…). Suffix “dmr” ➔ metadata only.

*dataset ≈ (file/granule | collection)*
URL + Query \(\rightarrow\) Subset
& (future) results from other server functions

http://.../granule.nc4?dap4.ce=constraints&dap4.func=functions

<table>
<thead>
<tr>
<th>Dataset identifier as above, except return-type is NetCDF4 (= HDF)</th>
<th>DAP “constraint expressions” yield sub-arrays &amp; other proper subsets</th>
<th>DAP4 “function expressions” enable extensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constraints specify subsets by variable names, by array indices &amp; (for tables) by content. Likely extensions include statistics, UGRID subsetting, feature extraction…</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The query form &dap4.func=... enables DAP extensions \(\Rightarrow\) new server functions
DAP-based
Subset Selection (from arrays | tables)

- Select variables by name
  - For tabular data, this means selecting columns
- Select rows of a table via column-specific value constraints
  - Allows both domain-based & range-based subsetting
- Select sub-arrays by constraining their indices
(array-style)

Index-Constrained Subsetting

Input Source Array ➔ Output Sub-Array
Index-Based Subsetting

- Excellent if desired subset is a bounding box parallel to source array (indices ↔ coordinates)
- Less useful when
  - Subset selection not based on domain coordinates
  - Source is not organized as coordinate-mapped arrays
  - Desired subset is polygonal or is skewed (relative to source-array orientation)
There are several different DAP servers

- Hyrax, developed by OPeNDAP, inc.
- TDS, developed by Unidata
- PyDAP, developed by Roberto De Almeida
- ERDDAP, developed by NOAA
- Others...
Architecture Overview of Hyrax

A widely-used DAP server

- NetCDF files
- HDF files
- SQL database
- Text files
- Extensible...

Apache Server Framework

Core DAP Services (Hyrax Front-End)

DAP-Extending Services

Other Web Services

Data User, Browser-Only
XML or JSON

Data-User App (other standard)
OGC-Compliant

Data-User App (Python/Java/C++)
Native-DAP Libraries

Data-User App (netCDF-based)
NetCDF Libraries

Hyrax Back-End Server with Encoding-Specific Handlers
OPeNDAP services
Can Function as Middleware

- Plugin-like handlers ⇔ multiple ingest encodings
  - Hence a growing set of source-data types

- Data output ⇔ multiple response encodings
  - Native DAP—useful in Python, Java, C/C++, Fortran…
  - netCDF3/4, GeoTIFF Jpeg2000, ASCII/CSV
  - XML (⇒ HTTP via style sheets)
  - Recently added: WMS, W10n, WCS, CovJSON
New(er) features in Hyrax

- Authentication (NASA Earthdata login)
- User-specified aggregation
- Cloud-based data stores
For more information

- www.opendap.org
- support@opendap.org
- jgallagher@opendap.org
OpenDAP and THREDDS
A web developers perspective

Duan Beckett
03 96168397
duan.beckett@bom.gov.au
The Bureau maintains several THREDDS servers

I develop web application on top of THREDDS – OpenDAP and NcWMS

Would like to share some views from this experience
Stats for September 2018

11,173,664 HTTP Requests

- 93%
- 3%
- 2%
- 2%
- Other

OpenDAP
Catalog access
File download
Other
Stats for September 2018

16.7TB Bytes transferred (TB)

- OpenDAP: 52%
- NetCDFSubset: 18%
- File download: 30%
- Other: 1%
Monthly requests from 2012 to present:
Monthly TB transferred from 2012 to present:
Two very beneficial features of THREDDS for web development are:

- Ease of deployment
- Ability to create aggregated datasets

What is a THREDDS aggregation?

- A virtual representation of a collection of files as a single file
- Creates a single entry point to data
Web development

Request for i.e. time dimension, can now be simplified to a single request:

http://thredds_url/dodsC/ocean_fc_agg.ascii?time
Web development

Services leveraging THREDDS:

• Operational ocean forecasting
  • Bluelink verification site
• Atmospheric transport modeling
  • TAPPAS/Spread
• Data compression application
Bluelink verification site part 1

- Displays multiple ocean datasets in map viewer (WMS)
- A growing number of server side processes
- Benefitted from using aggregated datasets in development
Bluelink verification site part 2

- Monitoring of Argo floats in the oceans
- A large number of server side statistical features
- Data not aggregated
- Aggregation was simulated via regular scans and database
- Increased development time
TAPAS/Spread site

- Models wind dispersal of pathogens using atmospheric model data for a run
- Pointing to several different datasets via OpenDAP
- Benefitted from accessing remote datasets
Data compression site

- Ocean forecast compression application
- Uses instance of THREDDS for visualization (WMS) and interrogation of data (OpenDAP)
- Benefitted from ease of deployment as deployed as a Docker image
Final thoughts

• THREDDS and OpenDAP are very powerful tools to build interesting applications off

• Data discovery vs web development

• From data providers I would like to see more use of requesting aggregating datasets for ease of development
THREDDS wish list

- JSON response from an OpenDAP call
- Ways to update/modify aggregations and documentation of how aggregations work behind the scenes
- Update individual catalogs
- Ways to mitigate the impact of outside users overloading the server
Thank you…

Duan Beckett
03 96168397
duan.beckett@bom.gov.au
NCI makes available national reference datasets – especially those produced by the government agencies and the universities. A range of communities and data collections make use of OPeNDAP at NCI.

- climate and weather models
- satellite images (Himawari, MODIS, LandSat)
- MT Data
- bathymetry and elevation
- hydrology
- geophysics
- Also: optical astro, genomic and social sciences
• Within these disciplines, data span a wide range of:
  - Gridded
  - Non-gridded (i.e., trajectories/profiles, point data)
  - Coordinate reference projections
  - Resolutions
• Collections are being accessed and utilised from a broad range of options
  - Direct access on filesystem
  - Web and data services
  - Data portals
  - Virtual labs
  - Virtual desktop interface (VDI)
THREDDS (Thematic Realtime Environmental Distributed Data Services) data server (TDS) developed by Unidata (UCAR) allows for browsing, downloading and programmatically accessing data.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPeNDAP (DAP2)</td>
<td>Protocol enabling data access and subsetting through the web</td>
</tr>
<tr>
<td>NetCDF Subset Service (NCSS)</td>
<td>Web service for subsetting files that can be read by the netCDF java library</td>
</tr>
<tr>
<td>Web Map Service (WMS)</td>
<td>OGC web service for requesting static images of data</td>
</tr>
<tr>
<td>Web Coverage Service (WCS)</td>
<td>OGC web service for requesting data in some output format</td>
</tr>
<tr>
<td>HTTP File Download</td>
<td>Direct downloading</td>
</tr>
</tbody>
</table>

THREDDS Usage Stats

- HTTP: 0.6
- OPeNDAP: 0.4
- WMS: 0.2
- NCSS: 0.1
- WCS: 0.1
How we use OPeNDAP and why?

- At NCI we serve OPeNDAP as part of THREDDS
  - Provides other web service endpoints important to our user community.
  - NetCDF/HDF5: most common data formats at NCI.
  - File aggregations.

- Other OPeNDAP options:
  - Hyrax
    - NCI have had a server in the past and its mostly similar to THREDDS.
    - It doesn’t cover all user web-services.
  - ERDDAP
    - Shows potential advantages such as a database search interface.
    - Need to consider resources to provide support for such an additional service.
  - PyDAP
    - Popular stand-alone python server
Popular tools use OPeNDAP:

- Python libraries
- R
- MATLAB
- Panoply
- QGIS
- Ferret
- CDO
- NCO
- NCL
- ncview
- ncdump
- ...

- IPython
- Interactive Computing
Data Quality Strategy (DQS): What does it involve?

1. Underlying HPD file format
2. Close collaboration with data custodians and managers
   - Planning, designing, or reassessing the data collections
3. Quality control through compliance with recognised community standards
4. Data assurance through demonstrated functionality across common platforms, tools, and services
Supporting Enhanced Features?

**netCDF4 enhancements with HDF5**

- **Grouping:**
  - allows users to group variables together with parent variables.
  - works with OPeNDAP but causes issues with other services.

- **Ragged Arrays (variable length arrays):**
  - can be stored with HDF5 but ragged arrays don’t work with other services.
  - did not work in Hyrax.

*Are others experiencing similar issues on THREDDS?*

*Is there more interest in using these advanced features?*
Discussion Topics

• VERSION: Current THREDDS production version at NCI is 4.6.10:
  • Includes OPeNDAP DAP 2.0.
  • What versions are currently being used? (e.g., Hyrax has DAP 4.0)

• AUTHORISATION: NCI currently does not have general authorization for THREDDS services but its on our 2019 roadmap.
  • Known issues with both identity and authorization systems for programmatic access.
  • Some tools don’t support modern web2.0 enabled libraries.
  • What authorization should be in place that works internationally and with all software?

• AGGREGATIONS: Success using time aggregation but not for lat/lon aggregates - a shortcoming in many domains.
  • Have other institutions investigated and had success with using aggregations?

• PERFORMANCE: We have a diverse range of users (10K+ per quarter) and we record performance metrics in our DQS
  • Many cases are now recorded through our Data Quality Strategy records and most perform very well.
  • Improving the service is dependent on users providing specific feedback on any issues experienced.
  • We experience performance issues in some cases.
  • Some cases are known deficiencies in the software/server and where possible we have developed alternatives.
  • Other performance issues are data issues.
  • What performance issues are prominent in your experience of OPeNDAP servers?
OPeNDAP at CSIRO (Robert Davy)