MODster/DODSter: Peer-to-Peer Sharing for Standard Remote Sensing Products

JAMES FREW
University of California, Santa Barbara

JAMES GALLAGHER
University of Rhode Island
Global Names: Simple

- Intrinsic
  - Obtained directly from object
    - checksum, hash, digital signature, ...
  - Often expensive to calculate
  - Sensitive to spurious changes
    - reformatting

- Arbitrary
  - Unique identifier obtained/calculated from single authority
    - UUID

- Difficult to dereference: who finds the object?
Global Names: Hierarchical

- General form: server/authority/name
  - **Server**: interprets names
    - Knows how to contact authority
    - If web server then name is a URL
  - **Authority**: assigns names
    - Must be well-known
  - **Name**: unique within authority
    - Can include any semantics the Authority wants

(based on Kunze’s ARK proposal)
Global Name Example: MODster

- MODIS granules have standard names
  - E.g. MOD03.A2001001.1550.002.2001017185332
- Create a NASA global name authority
  - E.g. MODIS
- Set up a global name server
  - E.g. http://essw.bren.ucsb.edu/modster
- Tell name server where MODIS granule lives
- Name server redirects
  
  http://essw.bren.ucsb.edu/modster/MODIS/MOD03.A2...

to your URL
GET MODster-URL
→

←
302 Found
Location: actual-URL

MODster server

↑
manual
↓
automatic

client

GET actual-URL
→

←
200 OK
(data stream)

data server
DODSter: Services for MODster

- **Subsetting**
  - DODS servers can subset
  - Constraint language meshes well with standard names

- **New formats**
  - DODS servers can transform formats

- **Information**
  - DODS servers can extract metadata
DODS URLs

- Parts of a DODS URL
  - <ext> selects type of information returned
    - Values: das, dds, info, html, dods, asc
  - <constraint> modifies data returns (dods, asc)
    - Select variables or parts of variables
    - E.g.: x,y,z[10:20]
- Metadata: das and dds (for machines); info (as html)
- WWW UI: html extension provides rudimentary UI
Using DODSter

- Combine a MODster URL with a DODS `<ext>` or `<ext>` and `<constraint>`
  - MODster URL
    http://essw.../modster/MODIS/MOD03.A2....hdf
  - DODS `<ext>` for data
    .dods?SST[1000:1500][525:625]
  - Complete URL
    http://essw.../modster/MODIS/MOD03.A2....hdf.dods?SST[1000:1500][525:625]

- This assumes every site participating in MODster offers DODS services.
When a MODster site lacks DODS

☐ Move service capabilities or data granules?
☐ Move the data!

- Places the burden of storage with users
  - Storage is expensive; this distributes costs to users
  - Maintenance of storage is complex; this distributes maintenance
- Available storage increases/scales with users/needs
- Moving data increases duplicates → increases robustness of the distributed data pool
  - Standard names and central registry manages duplicates
  - Duplicates can also be used to reduce leaf-node server network use (e.g., gnutella, et al. do this already)
Changes to MODster

- It must know where DODS services can be performed
  - DODS-capable MODster sites must register themselves
  - MODIS/\{regular expression\}, DODS base URL
  - MODIS identifies the namespace
  - \{regular ex\} part of the namespace this server can handle
  - DODS Base URL is the location of the relevant DODS server
Changes to DODS servers

- DODS servers must be able to move granules
  - Dictated by MODster redirection scheme
  - Extend DODS’ URL syntax
    - `http://…/nph-dods/<MODster URL>.<dods ext>[?<constraint>]`
    - E.g.: `http://…/nph-dods/http://…/modster/MODIS/MODA3….hdf.das`

- Moved granules
  - May be cached
    - MODster would always use the extended DODS syntax
  - May be stored if the site is also a MODster site
    - MODster database would be updated by the DODS server
Major Missing Pieces

- Local
  - Storage management
    - Everything online and accessible
  - Metadata management
    - Everything documented and findable
    - Everything connected to where it came from
    - Automatic MODster DB updates

- Global
  - Federation
    - Distributed data centers look like a single system
  - Service management
    - Discover, query, describe, and retrieve information
  - Namespace management
    - Same names for same things

→ Personal data centers ...

→ Digital libraries ...
Summary

- MODster
  - Use standard names & central directory to manage distributed data
  - Store data with users to increase capacity and robustness

- DODSter
  - Manage service availability using MODster DB
  - Service moves data (Data Diffusion)
Future work

- Leverage duplication
  - Load balancing
- Aggregate distributed granules
  - Must solve problem of ‘holes’ in the aggregate
- Active management of granules
  - Push granules to other sites