netCDF-LD

Active working group members:

James Biard (j.biard@computer.org)

Jonathan Yu (jonathan.yu@csiro.au)

Mark Hedley (mark.hedley@metoffice.gov.uk)

Adam Leadbetter (Adam.Leadbetter@Marine.ie)

Acknowledgements

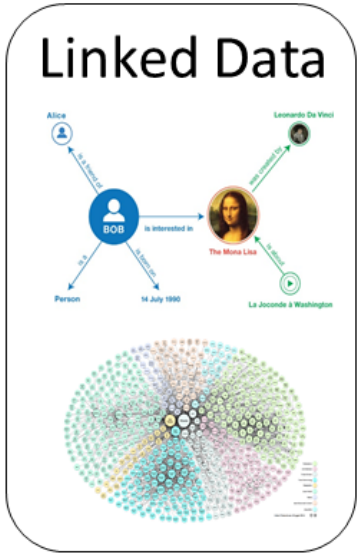
Nick Car & Alex Ip (Geoscience Australia)

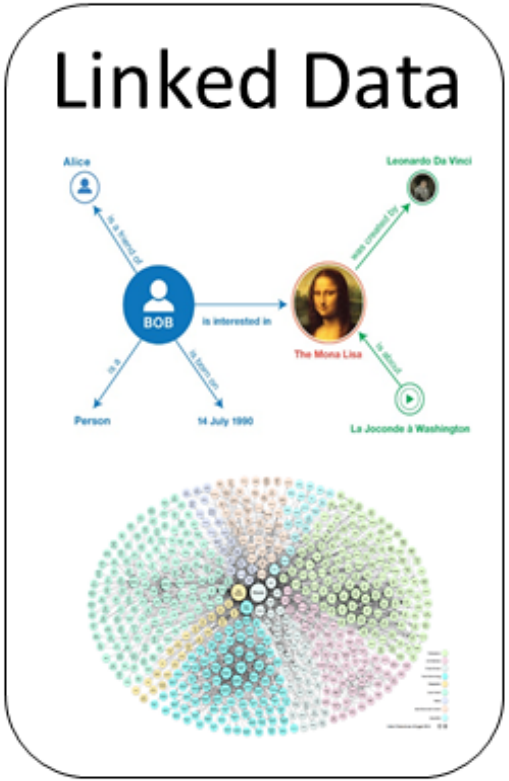
Kelsey Druken (NCI Australia)

Sean Arms (UCAR)

Contributors to bald repository

netCDF + Linked Data = netCDF–LD





Recommended best practice for exposing, sharing, and connecting pieces of data, information, and knowledge ***on the web.***

Standard format …

Reachable …

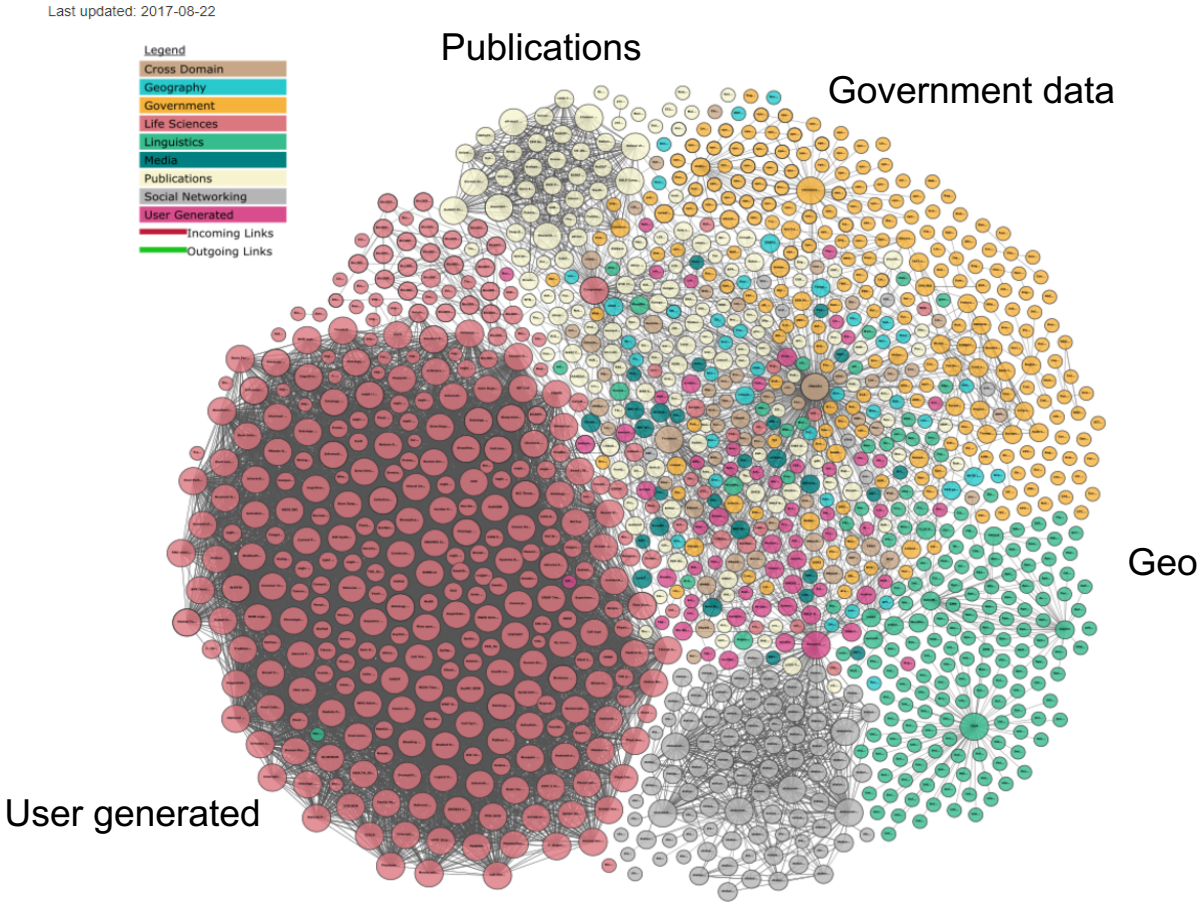
Relationships between data …

Collection of interrelated data → Linked Data

Key concept: Give each *thing* in the data an individual identity or URI

https://www.w3.org/standards/semanticweb/data

Linked Open Data Cloud http://lod-cloud.net/



32 billion triples in 2014

…

192 billion+ triples in 2017

See http://stats.lod2.eu/stats

Can we plug netCDF/HDF data in?

Other motivations

Encode and interpret nc files that use multiple metadata standards/conventions effectively (e.g. check naming and codelist conflicts) - CF often combined with other conventions (e.g. ACDD + CF)

Exploit Web and Linked Data tech to ***enhance discovery*** across large collections of files (e.g. represent separate files as graphs)

Represent nc/hdf files as close to the spirit of a binary array data model (vs. transform into other data models like RDF Data Cube (yet))

People are ***already*** linking to external references but not consistently …

Design principles

Work with current netCDF files

Design a simple mechanism to that works with existing netCDF files as-is to encode in a Linked Data friendly format.

Allow consistent and precise naming of each *thing* in netCDF/HDF metadata

Implies introducing new syntax (compatible with netCDF / HDF) to build URIs for each attribute name and property value

Enable consistent way to link to references, e.g. model, instrument, etc.

Overview

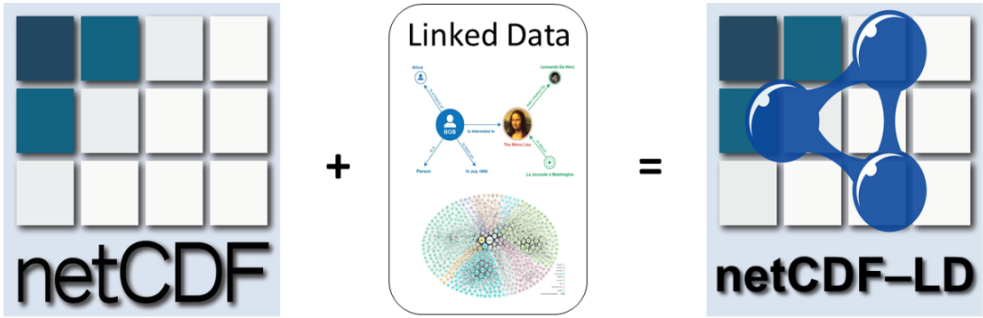
What have we been up to?

Tools

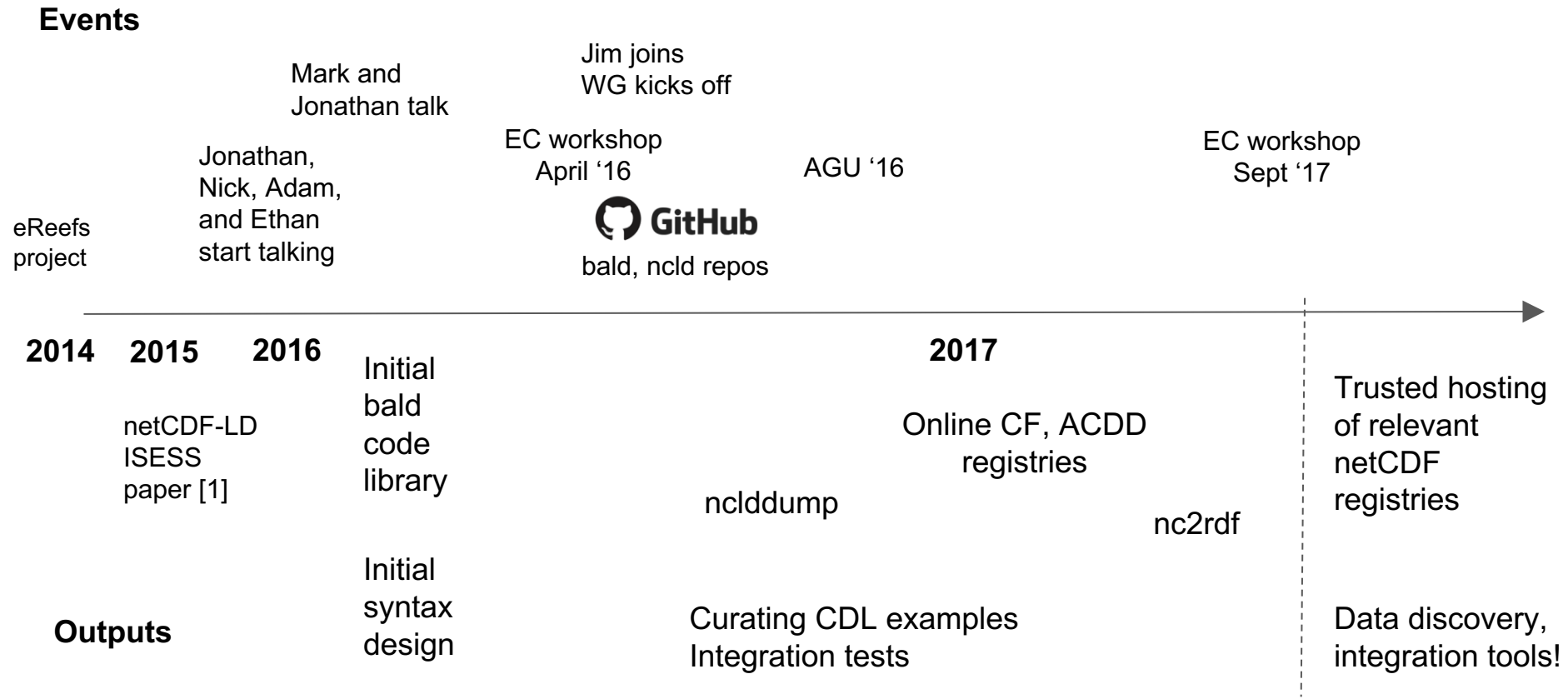
Syntax (aliasing, prefixes)

Supporting registries

Next steps



What have we been up to?



[1] Yu et al. 2015, Towards Linked Data Conventions for Delivery of Environmental Data Using netCDF, https://doi.org/10.1007/978-3-319-15994-2_9

Tools

Python libraries (Github bald repo) - (bald = binary array linked data)
https://github.com/binary-array-ld/bald

Command line tools (in development):

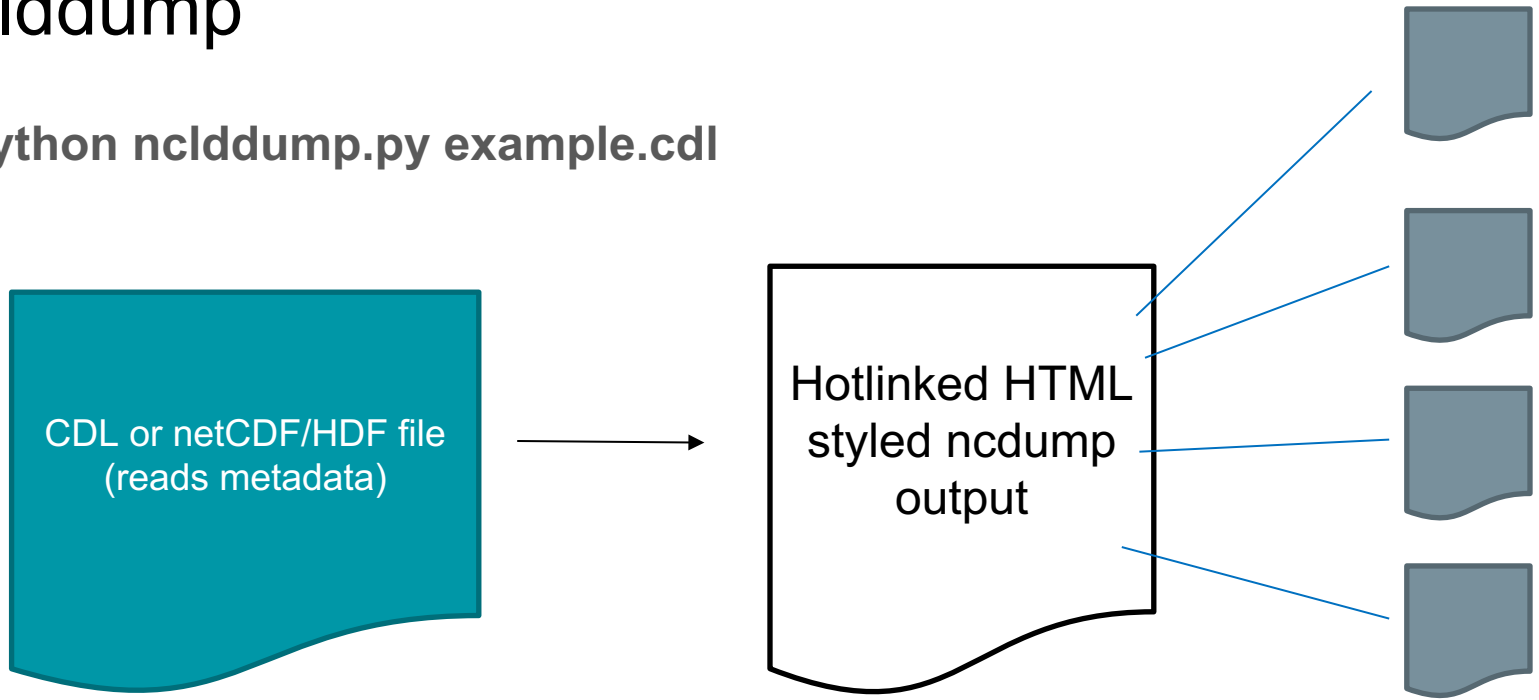
ncldddump

nc2rdf

Demos

nclddump

$ python nclddump.py example.cdl



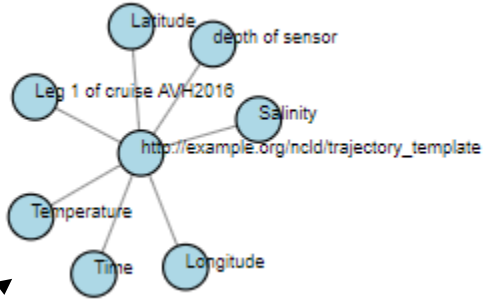
Example on https://binary-array-ld.github.io/netcdf-ld/

nc2rdf

$ python nc2rdf.py example.cdl

CDL or netCDF/HDF file
(reads metadata)

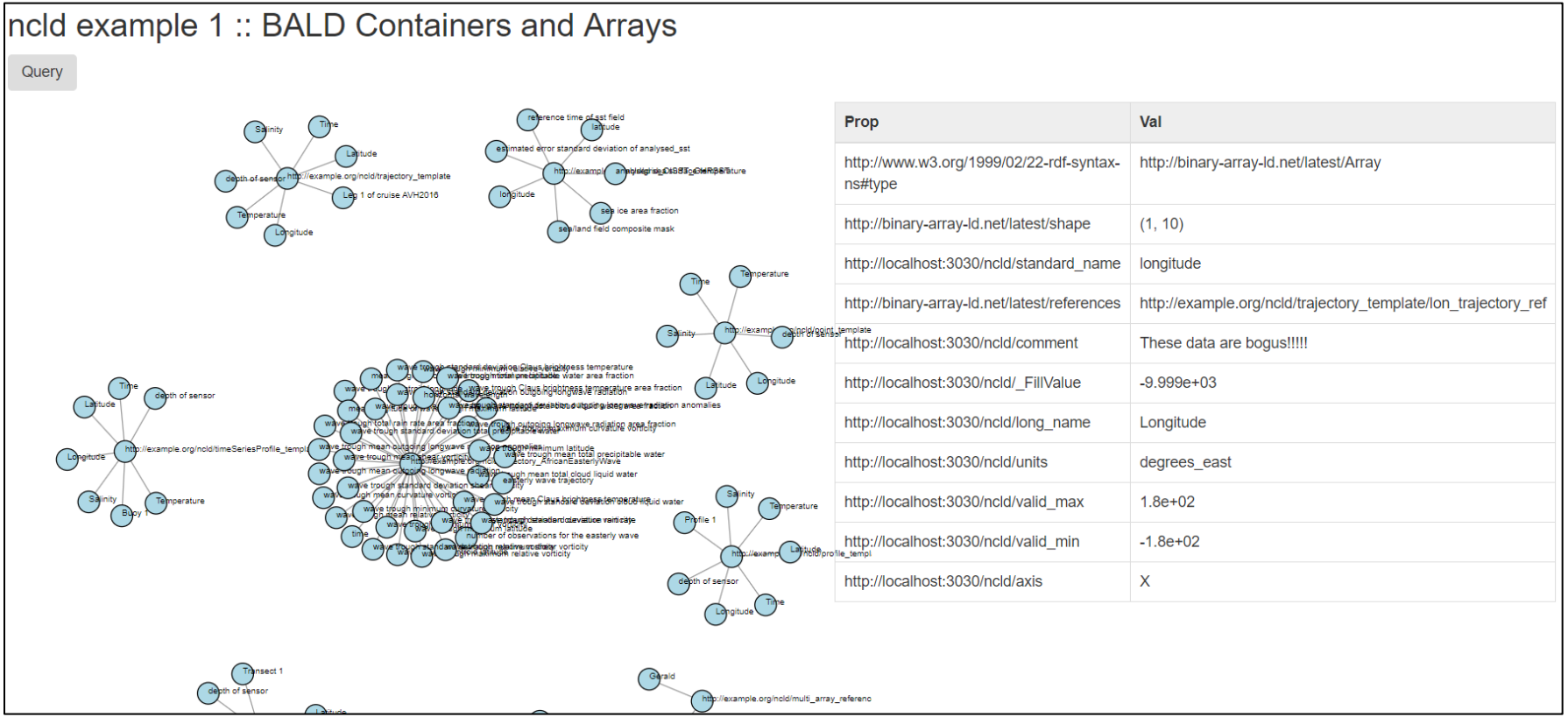
(visualisation representation)



Other
data

Load into
triple store DB
for semantic queries

Demo visualisations of graphs from CDL examples in bald repo



Prop	Val
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://binary-array-ld.net/latest/Array
http://binary-array-ld.net/latest/shape	(1, 10)
http://localhost:3030/ncld/standard_name	longitude
http://binary-array-ld.net/latest/references	http://example.org/ncld/trajectory_template/lon_trajectory_ref
http://localhost:3030/ncld/comment	These data are bogus!!!!!
http://localhost:3030/ncld/_FillValue	-9.999e+03
http://localhost:3030/ncld/long_name	Longitude
http://localhost:3030/ncld/units	degrees_east
http://localhost:3030/ncld/valid_max	1.8e+02
http://localhost:3030/ncld/valid_min	-1.8e+02
http://localhost:3030/ncld/axis	X

http://waterinformatics-ext1-cdc.it.csiro.au/ncld-demo/

ereefs convention example

variables:

float eta(time, j, i) ;

eta:units = "metre" ;

eta:long_name = "Surface elevation" ;

eta:standard_name = "sea_surface_height_above_sea_level" ;

eta:medium_id = “ocean”

eta:scaledQuantityKind_id = “sea_surface_elevation”

eta:substanceOrTaxon_id = “ocean_near_surface”

ereefs convention example - what we wanted

variables:

float eta(time, j, i) ;

eta:units = "metre" ;

eta:long_name = "Surface elevation" ;

eta:standard_name = "sea_surface_height_above_sea_level" ;

eta:medium_id = “ocean”

eta:scaledQuantityKind_id = “sea_surface_elevation”

eta:substanceOrTaxon_id = “ocean_near_surface”

Who defines these terms?

How do I check validity?

ereefs convention example - what we ended up with

variables:
    float eta(time, j, i) ;
        eta:units = "metre" ;
        eta:long_name = "Surface elevation" ;
        eta:standard_name = "sea_surface_height_above_sea_level" ;
        eta:medium_id = "http://environment.data.gov.au/def/feature/ocean" ;
        eta:scaledQuantityKind_id =
            "http://environment.data.gov.au/def/property/sea_surface_elevation" ;
        eta:substanceOrTaxon_id =
            "http://environment.data.gov.au/def/feature/ocean_near_surface" ;

I can check validity over the web (HTTP)

Not very scalable or extensible :(
Also not that readable…

Binary Array LD Syntax (for netCDF and HDF)

Methods to encode or process nc/hdf for translating to RDF / Linked Data ready

Aliasing

Lookup table for ‘well-known’ or declared mappings

Can be explicit or implicit

Pros: Easy to convert current nc files
Cons: Resolving clashes

e.g. `title` → `acdd:title`

(netcdf) (RDF)

Prefixing

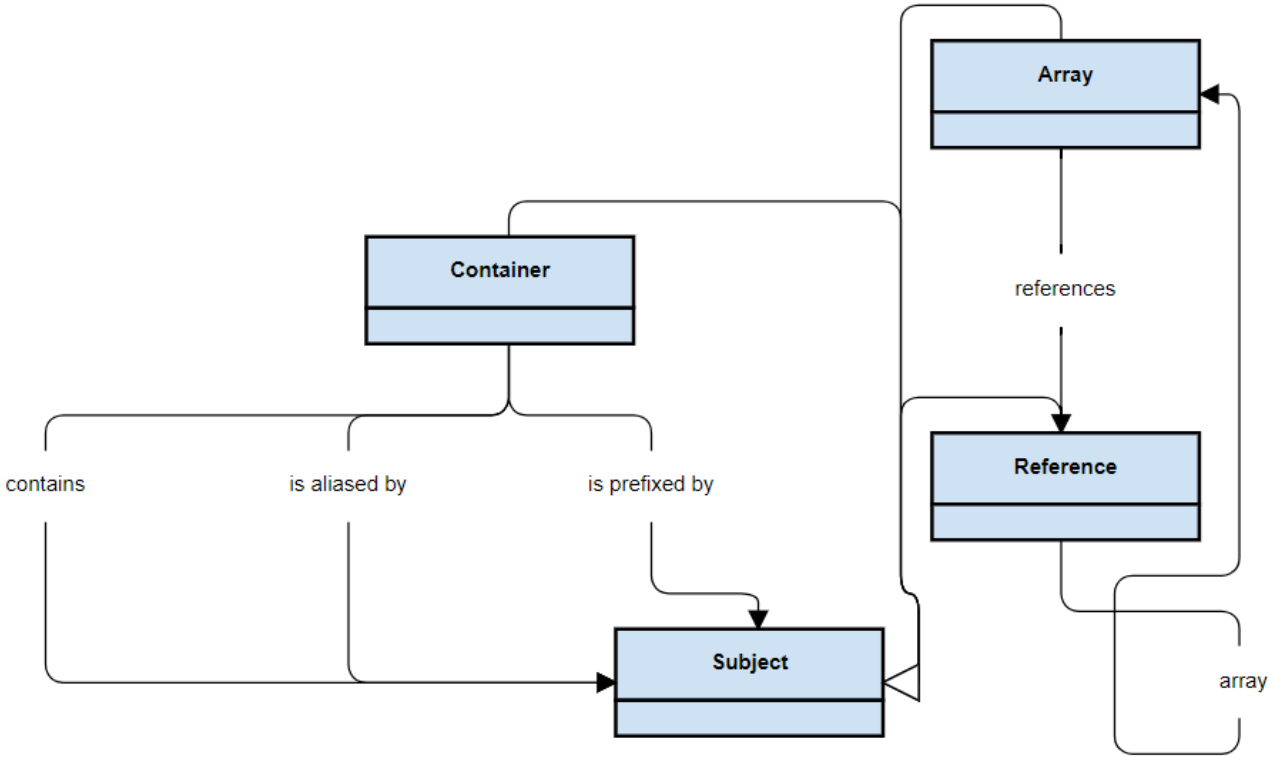
Kinda like namespacing

Pros: Easy to convert conformant files
Cons: Current files need tweaking

e.g. `acdd___title` → `acdd:title`

(netcdf) (RDF)

Binary Array Linked Data (BALD) model



http://binary-array-ld.net/_latest?classView=true

Aliasing example

variables:

int variable(pdim0, pdim1) ;
variable:**SDN_ParameterDiscoveryCode** = "**BactTaxaAbundSed**" ;

int cfvariable(pdim0, pdim1) ;
cfvariable:**standard_name** = "**air_temperature**" ;

// global attributes:
:isAliasedBy = "alias_list" ;

Variable metadata

Aliasing example – adding context (explicit aliases)

variables:

int alias_list ;
alias_list:**SDN_ParameterDiscoveryCode** = "http://vocab.nerc.ac.uk/isoCodelists/sdnCodelists/cdicsrCodeList.xml#SDN_ParameterDiscoveryCode" ;
alias_list:**BactTaxaAbundSed** = "http://vocab.nerc.ac.uk/collection/P02/current/BAUC/" ;
alias_list:**standard_name** = "https://def.scitools.org.uk/CFTerms/standard_name" ;
alias_list:**air_temperature** = "http://vocab.nerc.ac.uk/collection/P07/current/CFSN0023/" ;

Mappings

int variable(pdim0, pdim1) ;
variable:**SDN_ParameterDiscoveryCode** = "**BactTaxaAbundSed**" ;

int cfvariable(pdim0, pdim1) ;
cfvariable:**standard_name** = "**air_temperature**" ;

Variable metadata

// global attributes:
:isAliasedBy = "alias_list" ;

Aliasing example – RDF representation

<example> a bald:Container ;
bald:contains <variable>, <cfvariable> .
…

<variable> a bald:Array ;
ns1:SDN_ParameterDiscoveryCode
<http://vocab.nerc.ac.uk/collection/P02/current/BAUC/> ;

<cfvariable> a bald:Array ;
ns2:**standard_name**
<http://vocab.nerc.ac.uk/collection/P07/current/CFSN0023/> .

Prefix example – ereefs running example

variables:

float eta(time, j, i) ;
eta:units = "metre" ;
eta:long_name = "Surface elevation" ;
eta:standard_name = "sea_surface_height_above_sea_level" ;
eta:medium_id = “ocean”
eta:scaledQuantityKind_id = “sea_surface_elevation”
eta:substanceOrTaxon_id = “ocean_near_surface”

Variable metadata

Prefix example – ereefs with prefixes added

variables:

float eta(time, j, i) ;
eta:units = "metre" ;
eta:**cf**__long_name = "Surface elevation" ;
eta:**cf**__standard_name = "**cfsn**__sea_surface_height_above_sea_level" ;
eta:**ereefs**_medium_id = “**feature**__ocean”
eta:**ereefs**_scaledQuantityKind_id = “**property**__sea_surface_elevation”
eta:**ereefs**_substanceOrTaxon_id = “**feature**__ocean_near_surface”

Variable metadata

Prefix example – added prefix mappings

variables:

int prefix_list;

prefix_list:cf__ = https://def.scitools.org.uk/CFTerms/

prefix_list:cfsn__ = http://mmisw.org/ont/cf/parameter/

prefix_list:feature__ = "http://environment.data.gov.au/def/feature/" ;

prefix_list:property__ = "http://environment.data.gov.au/def/property/" ;

prefix_list:ereefs__ = "http://registry.it.csiro.au/sandbox/ncld/ereefs-attributes/ " ;

Prefix Mappings

float eta(time, j, i) ;

eta:units = "metre" ;

eta:cf__long_name = "Surface elevation" ;

eta:cf__standard_name = "cfsn__sea_surface_height_above_sea_level" ;

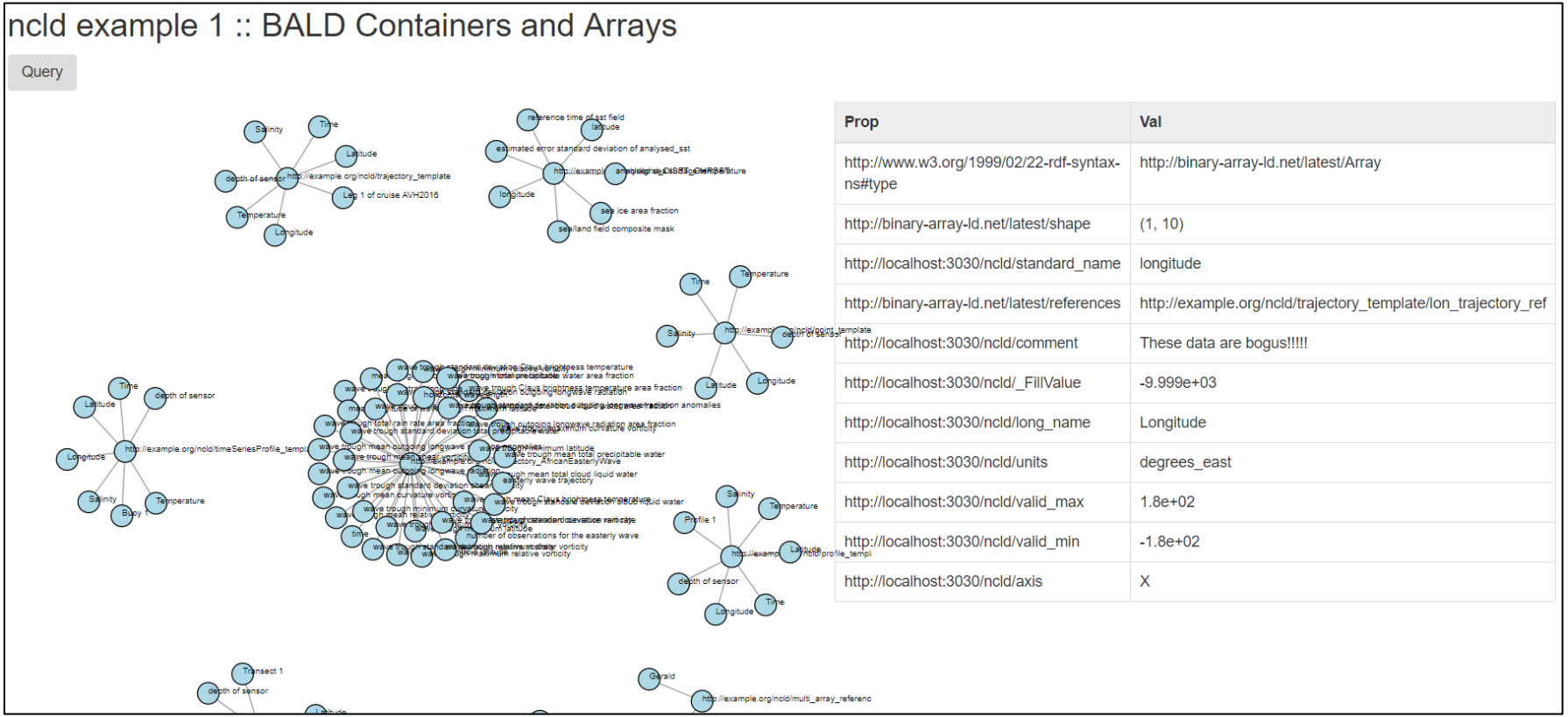
eta:ereefs__medium_id = “feature__ocean”

eta:ereefs__scaledQuantityKind_id = “property__sea_surface_elevation”

eta:ereefs__substanceOrTaxon_id = “feature__ocean_near_surface”

Variable metadata

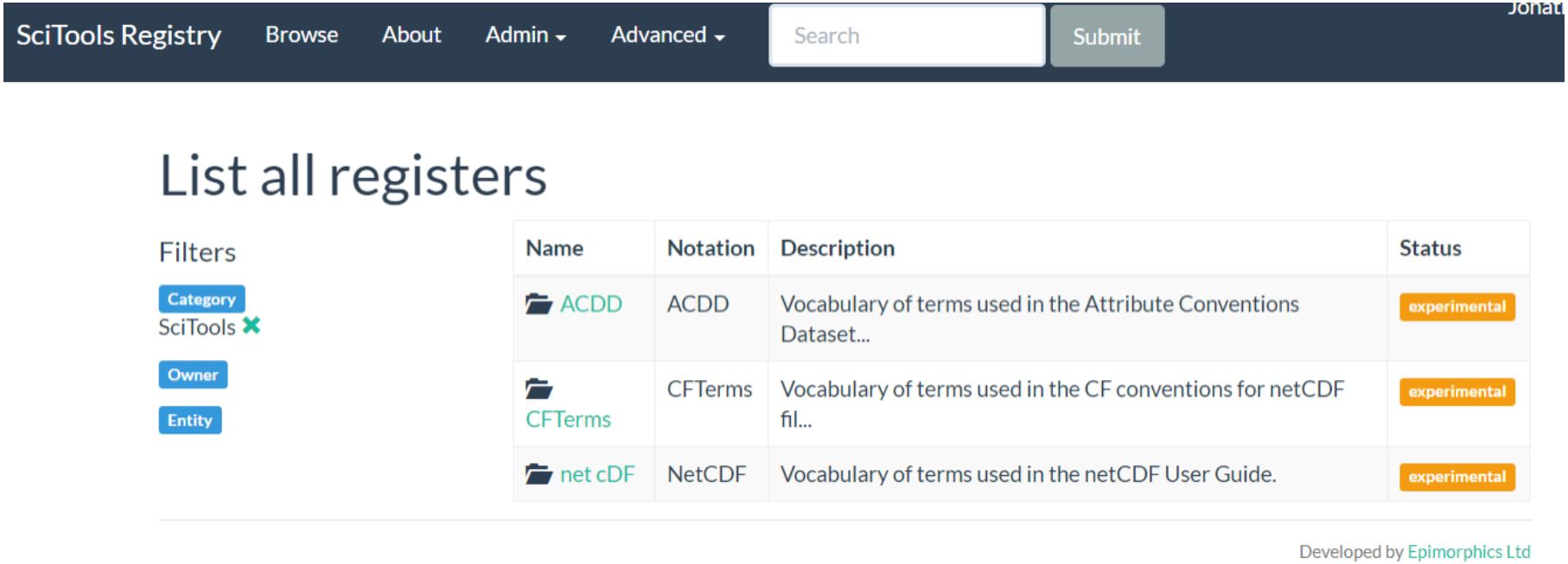
Demo visualisations of graphs from CDL examples in bald repo



Prop	Val
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://binary-array-ld.net/latest/Array
http://binary-array-ld.net/latest/shape	(1, 10)
http://localhost:3030/ncld/standard_name	longitude
http://binary-array-ld.net/latest/references	http://example.org/ncld/trajectory_template/lon_trajectory_ref
http://localhost:3030/ncld/comment	These data are bogus!!!!!
http://localhost:3030/ncld/_FillValue	-9.999e+03
http://localhost:3030/ncld/long_name	Longitude
http://localhost:3030/ncld/units	degrees_east
http://localhost:3030/ncld/valid_max	1.8e+02
http://localhost:3030/ncld/valid_min	-1.8e+02
http://localhost:3030/ncld/axis	X

http://waterinformatics-ext1-cdc.it.csiro.au/ncld-demo/

Supporting registries



https://def.scitools.org.uk/

Next steps

Establishing trusted registers online - CF terms, NUG, ACDD

Process THREDDS servers and explore integration and visualisations

Explore opportunities to link to other codelists

- Area type http://vocab.nerc.ac.uk/collection/P30/current
- Standardised regions (P29) http://vocab.nerc.ac.uk/collection/P29/current/

Build tools and demonstrators showing discovery across existing netCDF CF repositories (e.g. via THREDDS)

Want to contribute? Submit nc samples to the bald repo

Thanks

netCDF–LD

Active working group members:

James Biard (j.biard@computer.org)

Jonathan Yu (jonathan.yu@csiro.au)

Mark Hedley (mark.hedley@metoffice.gov.uk)

Adam Leadbetter (Adam.Leadbetter@Marine.ie)

GitHub Python libraries (bald = binary array linked data)
https://github.com/binary-array-ld/bald

http://tinyurl.com/netcdf-ld

Demo http://waterinformatics-ext1-cdc.it.csiro.au/ncld-demo/

(Aliasing example)

```
variables:
  int alias_list ;
        alias_list:standard_name = "https://def.scitools.org.uk/CFTerms/standard_name" ;
        alias_list:sea_surface_elevation =
                        "http://environment.data.gov.au/def/property/sea_surface_elevation" ;
        alias_list:ocean_near_surface =
                          "http://environment.data.gov.au/def/feature/ocean_near_surface" ;
        alias_list:ocean = "http://environment.data.gov.au/def/feature/ocean" ;

  float eta(time, j, i) ;
        eta:units = "metre" ;
        eta:long_name = "Surface elevation" ;
        eta:standard_name = "sea_surface_height_above_sea_level" ;
        eta:medium_id = “ocean”
        eta:scaledQuantityKind_id = “sea_surface_elevation”
        eta:substanceOrTaxon_id = “ocean_near_surface”
```