CHESS Course: An Introduction to FAIR Data Management for Geoscientists, Syllabus

Day 2 - Session 1

Documentation of data

Torill Hamre, Nansen Environmental and Remote Sensing Center

NorDataNet

Norwegian Scientific Data Network

Session Topics

- Tools for documenting data
  - Rosetta (web application). NCO/CDO (command line), Python (netcdf4)
- Examples in Python
  - netcdf4, xarray
- Validation of metadata
  - What is actually validated?
- Validation tools for NetCDF/CF
  - NorDataNet validator, PUMA validator
- Rosetta in more detail
  - Profiles, time series, trajectory
  - Template concept and benefits
  - Converting a CTD profile from Seabird sensor

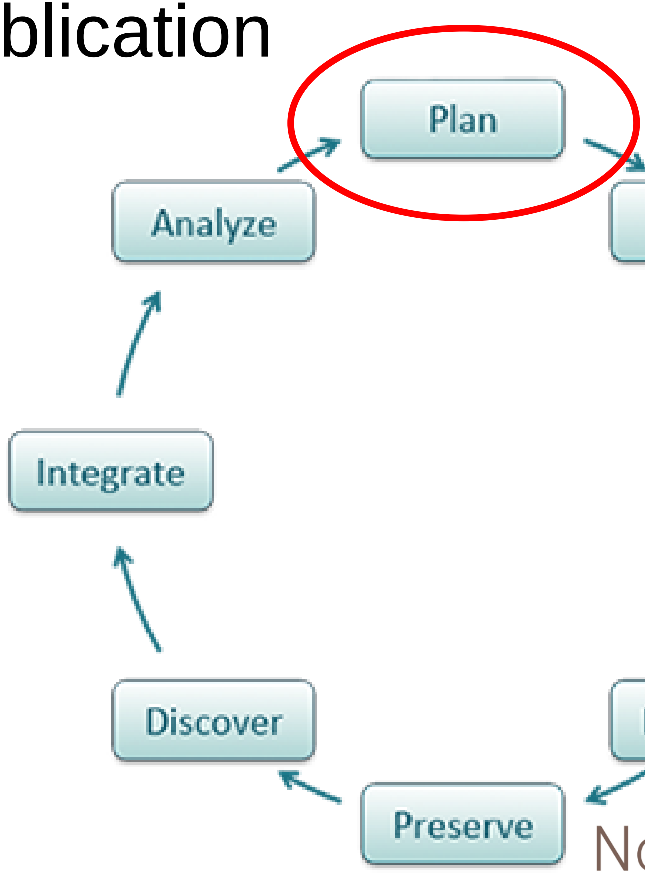
Learning Objectives

- Recognize the value of documenting your data
- Identify some tools for documenting data
- Understanding the difference between different categories of data
- Identify some tools for validating metadata
- Recognize the value of using standards

Preparing your data for publication

- Why is metadata important?
    - Documents the data
        - Content
        - Processing and quality control
        - Reference material
        - Contact for data provider
        - Data usage rights (license)
        - …

- How ensure enough metadata?
    - Include in planning of data collection and processing



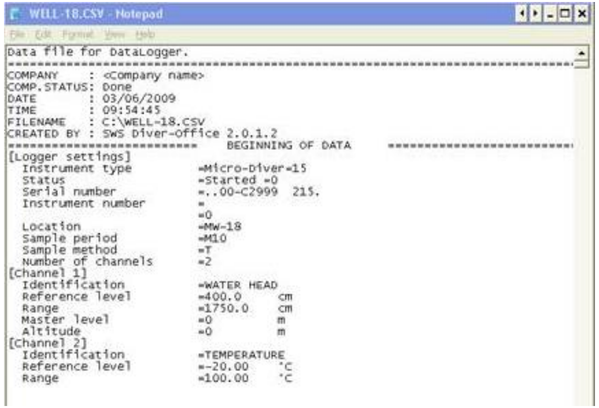
Preparing your data for publication

- Compile both metadata and other documentation
- Metadata
    - **Discovery metadata**, e.g. area covered, time period, parameters, ...
    - **Usage metadata**, e.g. processing history, units, data quality, ...
    - Avoid making new standards; utilize the existing ones
- Decide on suitable tools for formatting
    - Depends on formats used in processing and analysis tools
    - Select a **standard data format** and **metadata standard**
    - Data formats should be **self-describing**
    - Files should **include all needed metadata**

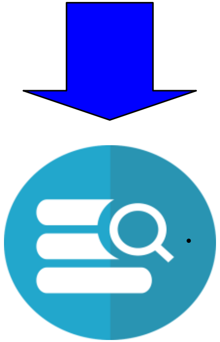
Preparing your data for publication

Why tools?

- To add documentation to dataset files
  - **Discovery metadata**
  - **Usage metadata**
- To prepare datasets in standard formats
- Makes data easier to reuse (also by others)



```
Data file for DataLogger.
==========================================================================
COMPANY     : <Company name>
COMP.STATUS: Done
DATE        : 03/06/2009
TIME        : 09:54:45
FILENAME    : C:\WELL-18.CSV
CREATED BY : SWS Diver-Office 2.0.1.2
=====================     BEGINNING OF DATA     =====================
[Logger settings]
  Instrument type         =Micro-Diver=15
  Status                  =Started =0
  Serial number           =..00-C2999  215.
  Instrument number       =
                          =0
  Location                =MW-18
  Sample period           =M10
  Sample method           =T
  Number of channels      =2
[Channel 1]
  Identification          =WATER HEAD
  Reference level         =400.0      cm
  Range                   =1750.0     cm
  Master level            =0          m
  Altitude                =0          m
[Channel 2]
  Identification          =TEMPERATURE
  Reference level         =-20.00     °C
  Range                   =100.00     °C
```



(data + metadata)

NorDataNet
Norwegian Scientific Data Network

Tools for documenting metadata

What tools?

- GIS (commercial, free)
- Data processing and analysis tools (commercial, free)
- Rosetta (free)

QGIS

netCDF

netCDF4

xarray

Rosetta

Tools for documenting metadata

Rosetta

UCAR COMMUNITY PROGRAMS | unidata
Data Services and Tools for Geoscience

Main features

- Read text files (CSV)
- Adds standard metadata
- Generates NetCDF
- Templates for conversion
- Web applicaiton

Rosetta

Tools for documenting metadata

NCO/CDO

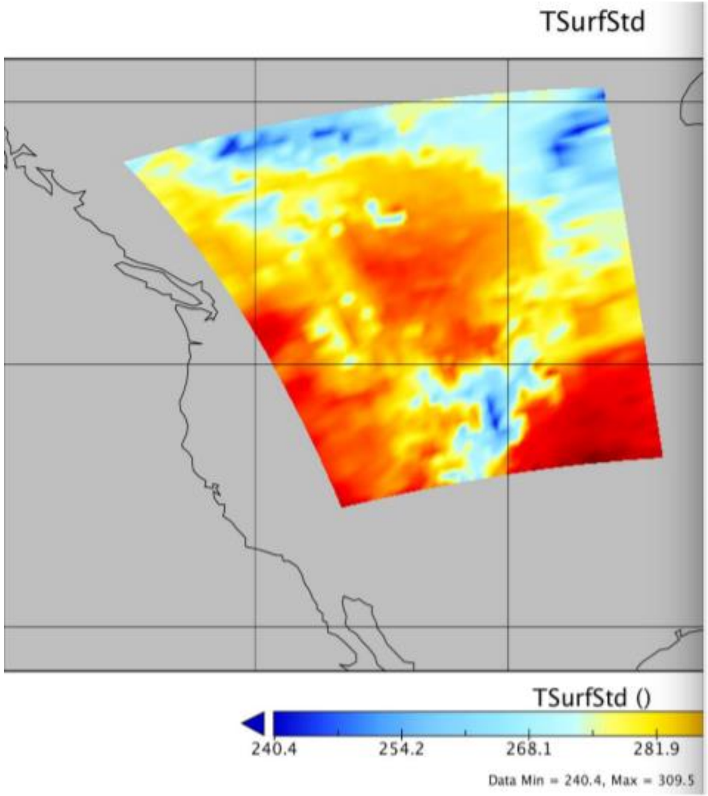
Main features

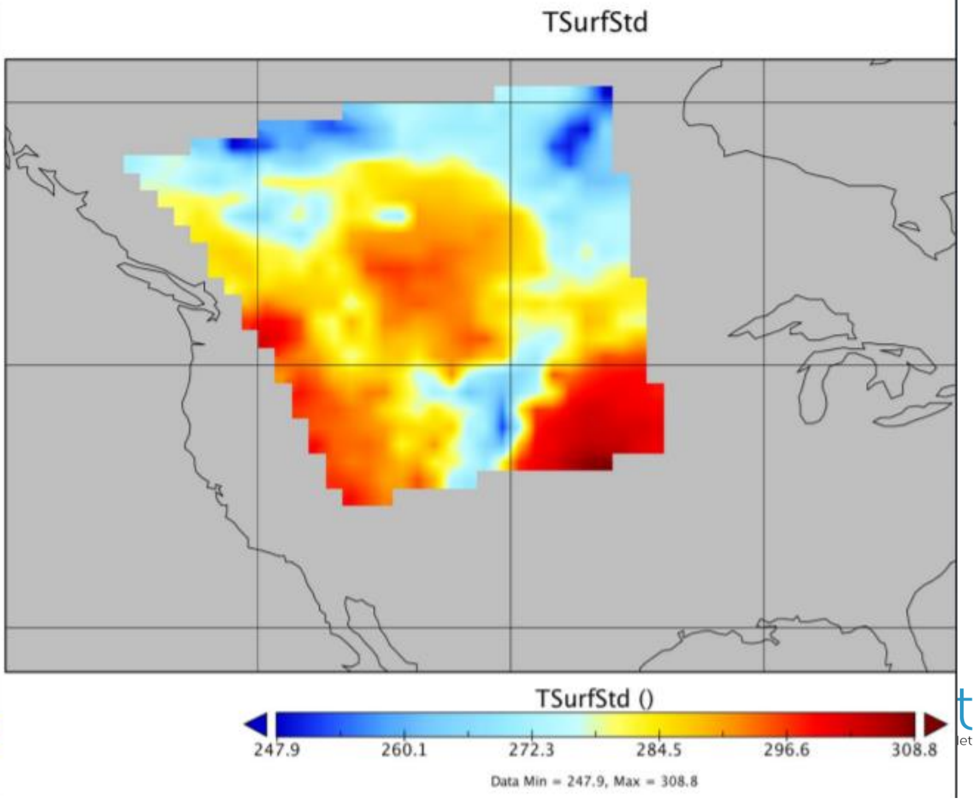
- Reads, manipulates and stores NetCDF
- Add/modify metadata
- Remapping, subsetting
- Analysis (e.g. statistics)
- Supports NetCDF, DAP, HDF4, HDF5
- Command line

netCDF Operators (NCO) Software Stack

Regrid NASA AIRS Level 2 Swath Data in raw HDF4 format from regional curvilinear 45x30 source grid to equiangular 1x1 degree:

```
% ncremap -s AIRS.2014.10.01.202.L2.RetStd.v6.0.11.0.G14275134307.hdf -d dst_1x1.nc
```





4.12 `ncremap` netCDF Remapper

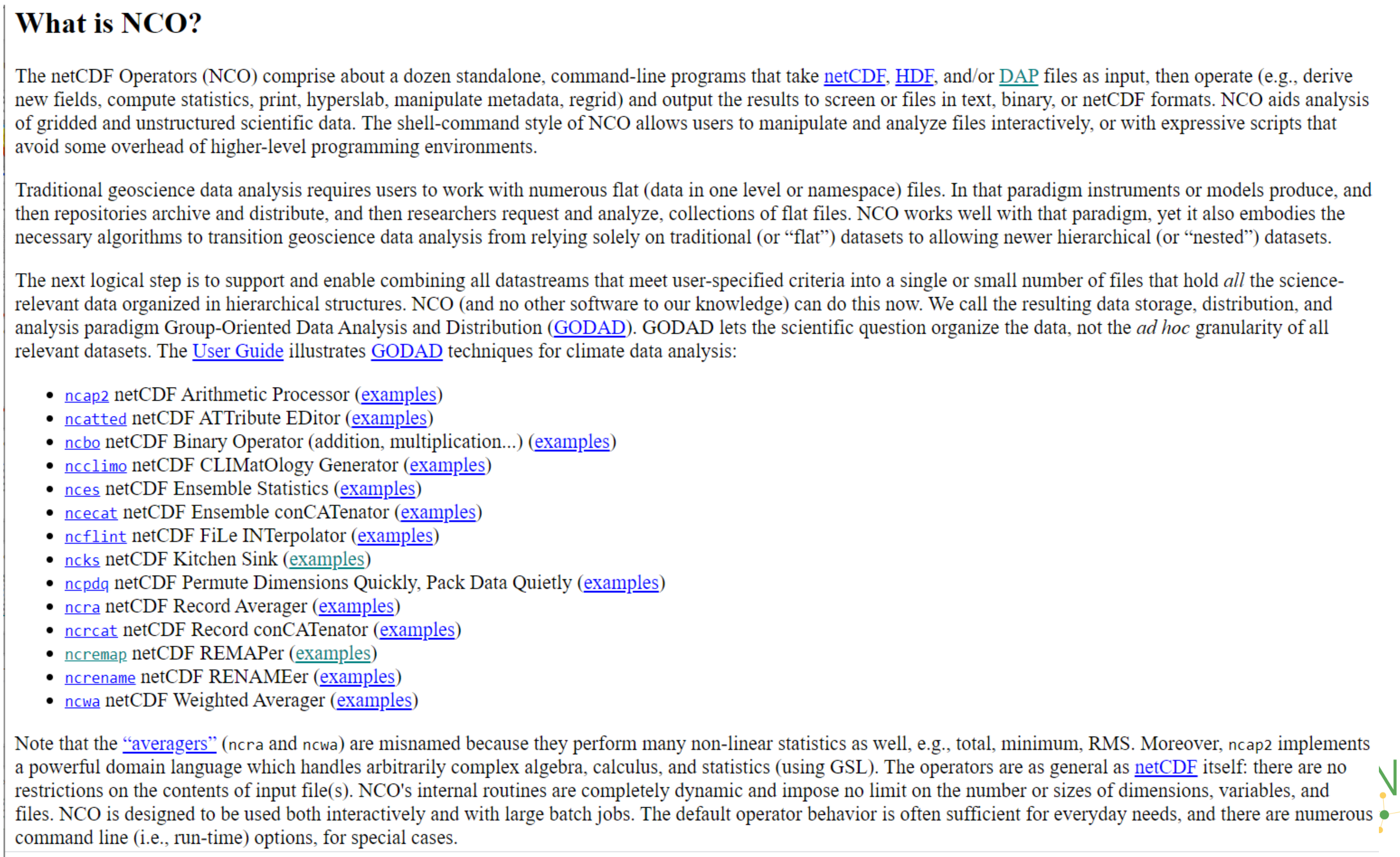
SYNTAX

```
ncremap [-3] [-4] [-5] [-6] [-7] [-a alg_typ] [--a2o]
[-D dbg_lvl] [-d dst_fl] [--d2f] [--dpt] [--dpt_fl=dpt_fl]
[--dt_sng=dt_sng] [--esmf_typ=esmf_typ]
[--fl_fmt=fl_fmt] [-G grd_sng] [-g grd_dst]
[-I drc_in] [-i input-file] [-j job_nbr] [-L dfl_lvl]
[-M] [-m map_fl] [--msh_fl=msh_fl]
[--msk_dst=msk_dst] [--msk_out=msk_out] [--msk_src=msk_src] [--mss_val=mss_val]
[-n nco_opt] [--nm_dst=nm_dst] [--nm_src=nm_src]
[--no_cll_msr] [--no_frm_trm] [--no_stg_grd]
[-O drc_out] [-o output-file] [-P prc_typ] [-p par_typ]
[--pdq=pdq_opt] [--ppc=ppc_opt] [--preserve=prs_stt]
[-R rgr_opt] [--rgn_dst] [--rgn_src] [--rnr_thr=rnr_thr]
[--rrg_bb_wesn=bb_wesn] [--rrg_dat_glb=dat_glb] [--rrg_grd_glb=grd_glb]
[--rrg_grd_rgn=grd_rgn] [--rrg_rnm_sng=rnm_sng]
[-s grd_src] [--sgs_frc=sgs_frc] [--sgs_msk=sgs_msk] [--sgs_nrm=sgs_nrm]
[--skl=skl-file] [--stdin] [-T drc_tmp] [-t thr_nbr]
[-U] [-u unq_sfx] [--ugrid=ugrid-file] [--uio]
[-V rgr_var] [-v var_lst[,…]] [--version] [--vrb=vrb_lvl]
[--vrt_fl=vrt_fl] [--vrt_ntp=vrt_ntp] [--vrt_xtr=vrt_xtr]
[-W wgt_opt] [-w wgt_cmd] [-x xtn_lst[,…]] [--xcl_var]
[--xtr_nsp=xtr_nsp] [--xtr_xpn=xtr_xpn]
[input-files] [output-file]
```

DESCRIPTION

`ncremap` remaps the data file(s) in *input-file*, in *drc_in*, or piped through standard input, to the horizontal grid specified by (in descending order of precedence) *map_fl*, *grd_dst*, or *dst_fl* and stores the result in *output-file*(s). If a vertical grid *vrt_fl* is provided, `ncremap` will (also) vertically interpolate the input file(s) to that grid. When no *input-file* is provided, `ncremap` operates in "map-only" mode where it exits after producing an annotated map-file. `ncremap` was introduced to NCO in version 4.5.4 (December, 2015).

`ncremap` is a "super-operator" that orchestrates the regridding features of several different programs including other NCO operators. Under the hood NCO applies pre-computed remapping weights or, when necessary, generates and infers grids, generates remapping weights itself or calls external programs to generate the weights, and then applies the weights (i.e., regrids).



What is NCO?

The netCDF Operators (NCO) comprise about a dozen standalone, command-line programs that take netCDF, HDF, and/or DAP files as input, then operate (e.g., derive new fields, compute statistics, print, hyperslab, manipulate metadata, regrid) and output the results to screen or files in text, binary, or netCDF formats. NCO aids analysis of gridded and unstructured scientific data. The shell-command style of NCO allows users to manipulate and analyze files interactively, or with expressive scripts that avoid some overhead of higher-level programming environments.

Traditional geoscience data analysis requires users to work with numerous flat (data in one level or namespace) files. In that paradigm instruments or models produce, and then repositories archive and distribute, and then researchers request and analyze, collections of flat files. NCO works well with that paradigm, yet it also embodies the necessary algorithms to transition geoscience data analysis from relying solely on traditional (or “flat”) datasets to allowing newer hierarchical (or “nested”) datasets.

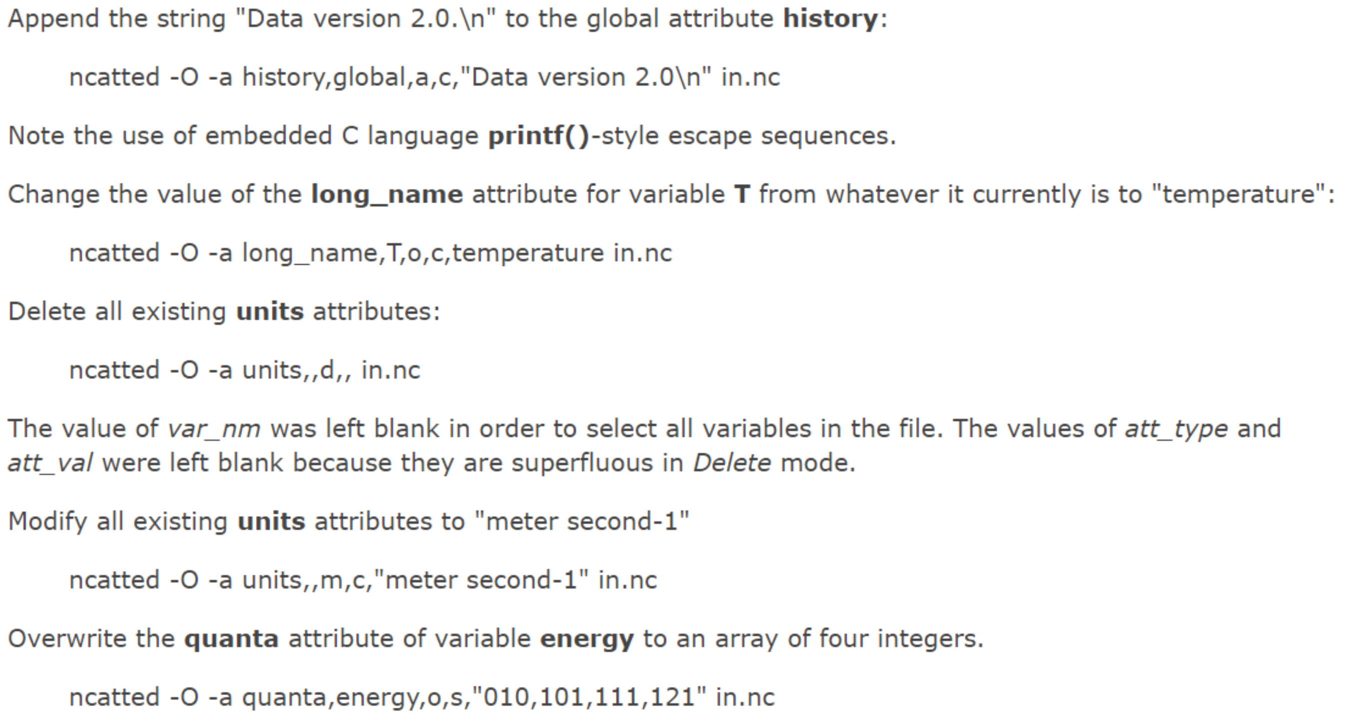
The next logical step is to support and enable combining all datastreams that meet user-specified criteria into a single or small number of files that hold *all* the science-relevant data organized in hierarchical structures. NCO (and no other software to our knowledge) can do this now. We call the resulting data storage, distribution, and analysis paradigm Group-Oriented Data Analysis and Distribution (GODAD). GODAD lets the scientific question organize the data, not the *ad hoc* granularity of all relevant datasets. The User Guide illustrates GODAD techniques for climate data analysis:

- ncap2 netCDF Arithmetic Processor (examples)
- ncatted netCDF ATTribute EDitor (examples)
- ncbo netCDF Binary Operator (addition, multiplication...) (examples)
- ncclimo netCDF CLIMatOlogy Generator (examples)
- nces netCDF Ensemble Statistics (examples)
- ncecat netCDF Ensemble conCATenator (examples)
- ncflint netCDF FiLe INTerpolator (examples)
- ncks netCDF Kitchen Sink (examples)
- ncpdq netCDF Permute Dimensions Quickly, Pack Data Quietly (examples)
- ncra netCDF Record Averager (examples)
- ncrcat netCDF Record conCATenator (examples)
- ncremap netCDF REMAPer (examples)
- ncrename netCDF RENAMEer (examples)
- ncwa netCDF Weighted Averager (examples)

Note that the “averagers” (ncra and ncwa) are misnamed because they perform many non-linear statistics as well, e.g., total, minimum, RMS. Moreover, ncap2 implements a powerful domain language which handles arbitrarily complex algebra, calculus, and statistics (using GSL). The operators are as general as netCDF itself: there are no restrictions on the contents of input file(s). NCO's internal routines are completely dynamic and impose no limit on the number or sizes of dimensions, variables, and files. NCO is designed to be used both interactively and with large batch jobs. The default operator behavior is often sufficient for everyday needs, and there are numerous command line (i.e., run-time) options, for special cases.

Tools for documenting metadata

Example – modify metadata with **ncatted** (https://linux.die.net/man/1/ncatted)



Append the string "Data version 2.0.\n" to the global attribute **history**:

```
ncatted -O -a history,global,a,c,"Data version 2.0\n" in.nc
```

Note the use of embedded C language **printf()**-style escape sequences.

Change the value of the **long_name** attribute for variable **T** from whatever it currently is to "temperature":

```
ncatted -O -a long_name,T,o,c,temperature in.nc
```

Delete all existing **units** attributes:

```
ncatted -O -a units,,d,, in.nc
```

The value of *var_nm* was left blank in order to select all variables in the file. The values of *att_type* and *att_val* were left blank because they are superfluous in *Delete* mode.

Modify all existing **units** attributes to "meter second-1"

```
ncatted -O -a units,,m,c,"meter second-1" in.nc
```

Overwrite the **quanta** attribute of variable **energy** to an array of four integers.

```
ncatted -O -a quanta,energy,o,s,"010,101,111,121" in.nc
```

Tools for documenting metadata

netCDF4

netcdf4 (netcdf4-python)

Main features

- Python package
- Read, write, update NetCDF
- Add/update metadata
- Supports netCDF3 and netCDF4

Documentation

- https://unidata.github.io/netcdf4-python/netCDF4/index.html
- http://schubert.atmos.colostate.edu/~cslocum/netcdf_example.html

Tools for documenting metadata

netCDF4

netcdf4 (netcdf4-python)

Open file and inspect its structures, (some) variables and attributes

```
from netCDF4 import Dataset
from datetime import datetime
import sys
import numpy as np

filename = "https://thredds.met.no/thredds/dodsC/myocean/siw-tac/siw-metno-
svalbard/2020/05/ice_conc_svalbard_202005151500.nc"
simap = Dataset(filename, "r")
print(simap)
```

Tools for documenting metadata

netCDF

netCDF4

Output:

```
<class 'netCDF4._netCDF4.Dataset'>
root group (NETCDF3_CLASSIC data model, file format DAP2):
    title: Arctic Svalbard & Barents Ice Concentration, L4, 1km daily  (METNO-ARC-
SEAICE_CONC-L4-NRT-OBS)
    Conventions: CF-1.4
    netcdf_version_id: 3.6.3
    creation_date: 2020-05-15T13:36:20Z
    ...
    dimensions(sizes): maxStrlen64(64), time(1), xc(3812), yc(2980)
    variables(dimensions): int32 time(time), float32 yc(yc), float32 xc(xc), |S1
crs(maxStrlen64), float32 lat(yc,xc), float32 lon(yc,xc), int16
ice_concentration(time,yc,xc), int16 concentration_range(time,yc,xc)
    groups:
```

NorDataNet
Norwegian Scientific Data Network

Tools for documenting metadata

netCDF4

Inspecting metadata for one of the variables:

```
conc = simap.variables['ice_concentration']
print(conc)
```

Output:

```
<class 'netCDF4._netCDF4.Variable'>
int16 ice_concentration(time, yc, xc)
    long_name: sea ice concentration
    standard_name: sea_ice_area_fraction
    units: %
    coordinates: lon lat
    grid_mapping: crs
    source: met.no
    _FillValue: -99
    scale_factor: 1.0
    add_offset: 0.0
unlimited dimensions:
current shape = (1, 2980, 3812)
filling off
```

The dimensions (axes) of the data variable

The data variable's attributes

The actual dimensions of the data variable

Tools for documenting metadata

netCDF4

Adding a new attribute (metadata) to this variable:

```
dset = Dataset('test-newattr.nc','w', format='NETCDF3_CLASSIC')
dset.createDimension('time',size=simap.variables["ice_concentration"].shape[0])
dset.createDimension('yc',size=simap.variables["ice_concentration"].shape[1])
dset.createDimension('xc',size=simap.variables["ice_concentration"].shape[2])
newconc = dset.createVariable('sea_ice_concentration', np.int16, dimensions=('time', 'yc',
'xc'),fill_value=-99)
newconc[:]= conc[:]
for name in conc.ncattrs():
  if name != '_FillValue':
    newconc.setncattr(name, getattr(conc,name))
newconc.setncattr('source_full_name', 'Meteorological Institute of Norway')
print(newconc)
```

Copy all data

Copy all variable's attributes from original file

Create a new attribute

Tools for documenting metadata

netCDF4

Output:

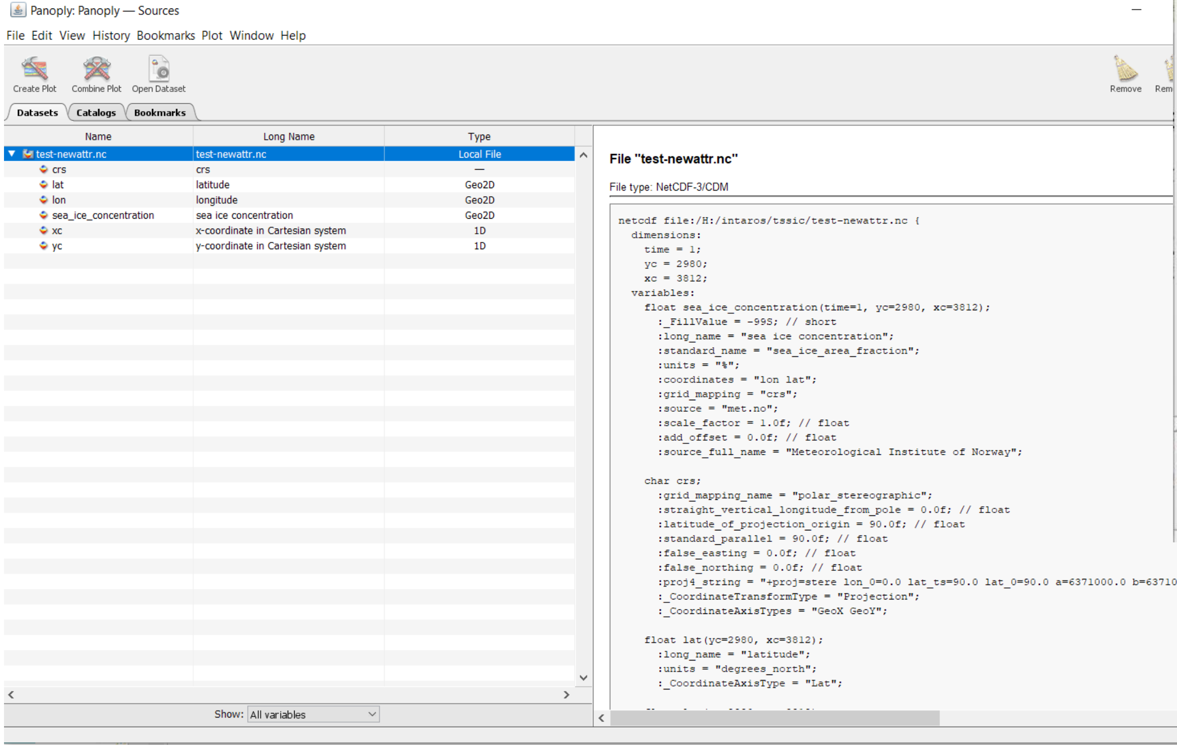
```
<class 'netCDF4._netCDF4.Variable'>
int16 sea_ice_concentration(time, yc, xc)
    _FillValue: -99
    long_name: sea ice concentration
    standard_name: sea_ice_area_fraction
    units: %
    coordinates: lon lat
    grid_mapping: crs
    source: met.no
    scale_factor: 1.0
    add_offset: 0.0
    source_full_name: Meteorological Institute of Norway
unlimited dimensions:
current shape = (1, 2980, 3812)
filling on
```

New attribute added

NorDataNet
Norwegian Scientific Data Network

Tools for documenting metadata

- After copying other variables & metadata



Name	Long Name	Type
test-newattr.nc	test-newattr.nc	Local File
crs	crs	—
lat	latitude	Geo2D
lon	longitude	Geo2D
sea_ice_concentration	sea ice concentration	Geo2D
xc	x-coordinate in Cartesian system	1D
yc	y-coordinate in Cartesian system	1D

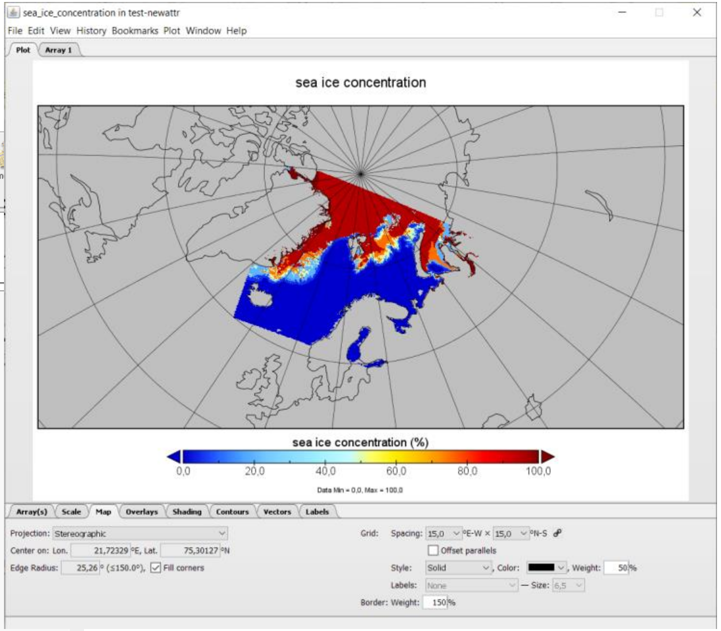
File "test-newattr.nc"

File type: NetCDF-3/CDM

```
netcdf file:/H:/intaros/tssic/test-newattr.nc {
  dimensions:
    time = 1;
    yc = 2980;
    xc = 3812;
  variables:
    float sea_ice_concentration(time=1, yc=2980, xc=3812);
      :_FillValue = -99S; // short
      :long_name = "sea ice concentration";
      :standard_name = "sea_ice_area_fraction";
      :units = "%";
      :coordinates = "lon lat";
      :grid_mapping = "crs";
      :source = "met.no";
      :scale_factor = 1.0f; // float
      :add_offset = 0.0f; // float
      :source_full_name = "Meteorological Institute of Norway";

    char crs;
      :grid_mapping_name = "polar_stereographic";
      :straight_vertical_longitude_from_pole = 0.0f; // float
      :latitude_of_projection_origin = 90.0f; // float
      :standard_parallel = 90.0f; // float
      :false_easting = 0.0f; // float
      :false_northing = 0.0f; // float
      :proj4_string = "+proj=stere lon_0=0.0 lat_ts=90.0 lat_0=90.0 a=6371000.0 b=6371000.0";
      :_CoordinateTransformType = "Projection";
      :_CoordinateAxisTypes = "GeoX GeoY";

    float lat(yc=2980, xc=3812);
      :long_name = "latitude";
      :units = "degrees_north";
      :_CoordinateAxisType = "Lat";
```



sea_ice_concentration in test-newattr
File Edit View History Bookmarks Plot Window Help
Plot
Array 1
sea ice concentration
sea ice concentration (%)
0,0
20,0
40,0
60,0
80,0
100,0
Data Min = 0,0, Max = 100,0
Array(s)
Scale
Map
Overlays
Shading
Contours
Vectors
Labels

Projection: Stereographic
Center on: Lon. 21,72329 °E, Lat. 75,30127 °N
Edge Radius: 25,26 ° (≤150.0°), ☑ Fill corners
Grid: Spacing: 15,0 °E-W × 15,0 °N-S
□ Offset parallels
Style: Solid, Color: ■, Weight: 50 %
Labels: None — Size: 6,5
Border: Weight: 150 %

NorDataNet
Norwegian Scientific Data Network

Tools for documenting metadata

xarray

Main features

- Python NetCDF file I/O
- Advanced analysis tools
- Multi-dimensional, labelled arrays
- Split-Apply-Combine
- Supports parallel computing

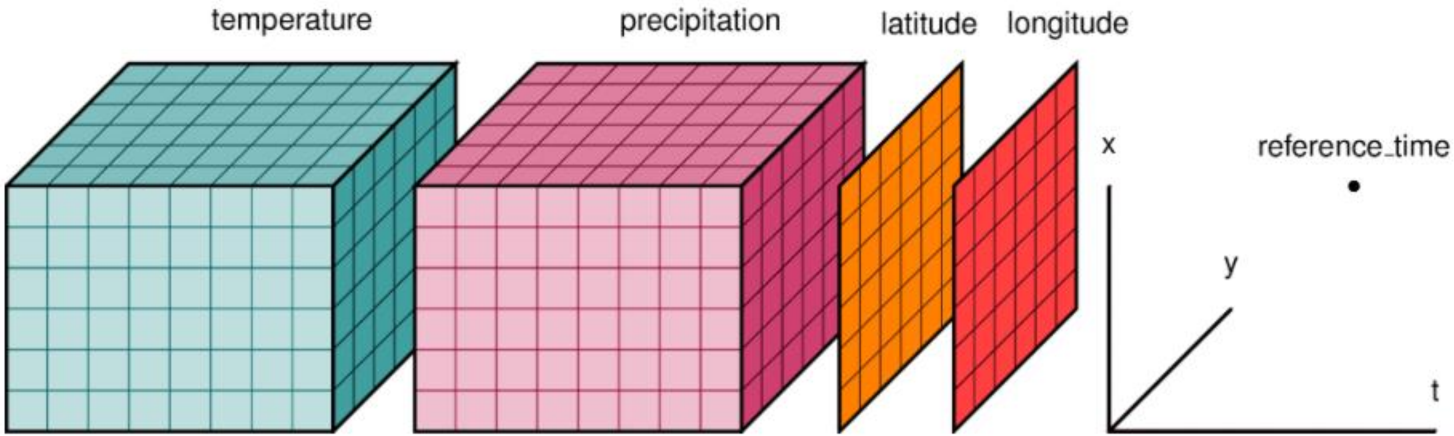
Documentation

- http://xarray.pydata.org/en/stable/

Tools for documenting metadata

xarray

Tutorial: https://rabernat.github.io/research_computing/xarray.html



Tools for documenting metadata

Install need packages and import them

```
import xarray
import netCDF4
```

Get the data file

```
curl -O http://www.ldeo.columbia.edu/~rpa/argo_float_4901412.npz
```

Load data and list names of variables (data) and coordinates

```
argo_data = np.load('argo_float_4901412.npz')
argo_data.keys()
```

Output:

```
['S', 'T', 'levels', 'lon', 'date', 'P', 'lat']
```

Then, extract the variable you want (salinity)

```
S = argo_data.f.S
```

Tools for documenting metadata

Check the dimensions of the salinity variable

```
print(S.shape)
```

Output:

```
(78, 75)
```

Make a DataArray with coordinates

```
da_salinity = xr.DataArray(S, dims=['level', 'date'],
                         coords={'level': levels,
                                 'date': date},)
```

Print (a summary) of your new array

```
da_salinity
```

Output:

Tools for documenting metadata

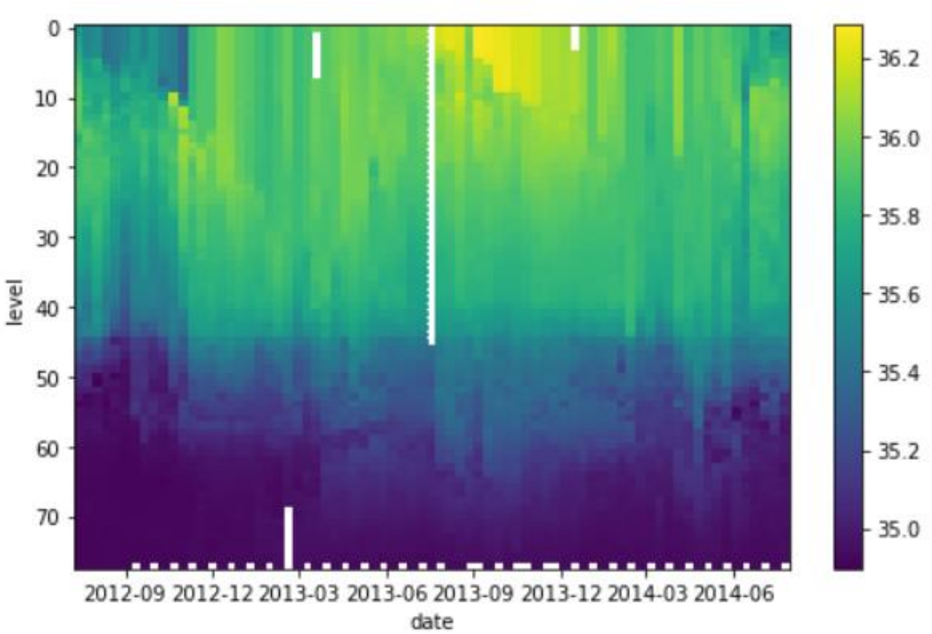
```
array([[ 35.638939,  35.514957,  35.572971, ...,  35.820938,  35.777939,
       35.668911],
     [ 35.633938,  35.521957,  35.573971, ...,  35.810932,  35.583897,
       35.667912],
     [ 35.681946,  35.525959,  35.572971, ...,  35.795929,  35.662907,
       35.665913],
     ...,
     [ 34.915859,  34.923904,  34.923904, ...,  34.934811,  34.940811,
       34.946808],
     [ 34.915859,  34.923904,  34.921906, ...,  34.932808,  34.93681 ,
       34.94381 ],
     [ 34.917858,  34.923904,  34.923904, ...,        nan,  34.93681 ,
             nan]])
Coordinates:
  * level    (level) int64 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 ...
  * date     (date) datetime64[ns] 2012-07-13T22:33:06.019200 ...
```

Tools for documenting metadata

xarray

Make a plot

```
da_salinity.plot(yincrease=False)
```



NorDataNet
Norwegian Scientific Data Network

Tools for documenting metadata

xarray

Add some attributes (required by CF)

```
da_salinity.attrs['units'] = 'PSU'
da_salinity.attrs['standard_name'] = 'sea_water_salinity'
da_salinity
```

Output:

```
array([[ 35.638939,  35.514957,  35.572971, ...,  35.820938,  35.777939,
         35.668911],
        ...
Coordinates:
  * level    (level) int64 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 ...
  * date     (date) datetime64[ns] 2012-07-13T22:33:06.019200 ...
Attributes:
    units:          PSU
    standard_name:  sea_water_salinity
```

NorDataNet
Norwegian Scientific Data Network

Validating metadata

What can be validated?

- Conformance vs Climate and Forecast (CF) Convention, e.g.
    - Inclusion of mandatory attributes
    - Use of standard names for parameters, labels from controlled vocabularies
    - Inclusion of units for each variable
    - Correct data type of each element

What is not covered by validation

- Content of your title, summary, and other text attributes
- Valid contact information (e.g. e-mail)
- Individual validator tools may differ
    - e.g. http://pumatest.nerc.ac.uk/cf-checker-status.html

Validation tools for NetCDF/CF

NorDataNet validator: https://www.nordatanet.no/en/dataset_validation/form

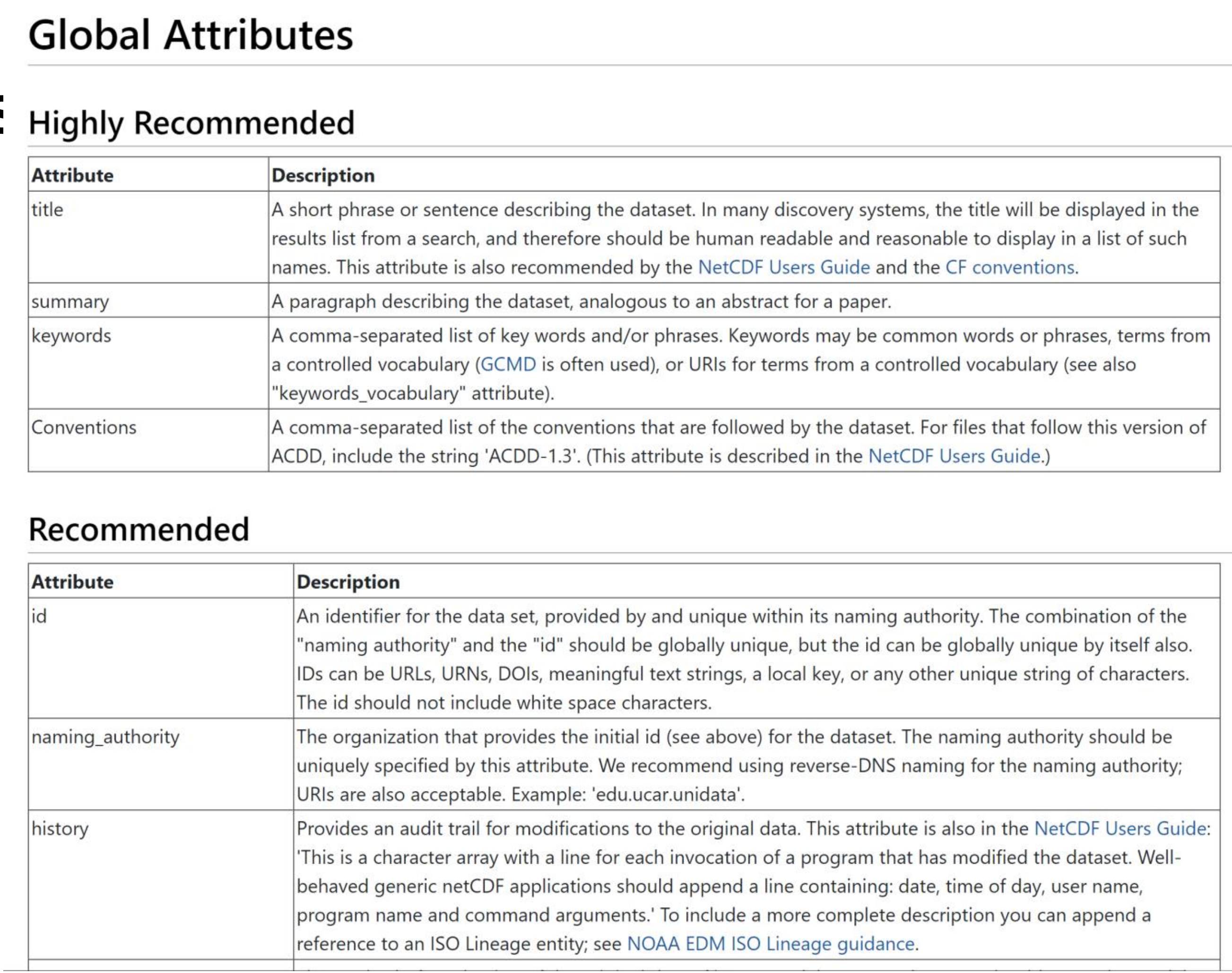
- Supports CF 1.6 - CF 1.7 + ACDD
- Builds on IOOS compliance checker
- Generates a “score” and a detailed report

PUMA validator: http://pumatest.nerc.ac.uk/cgi-bin/cf-checker.pl

- Supports CF 1.0 - CF 1.7
- Developed from scratch
- Generates a detailed report (Errors, Warnings, Information)

Validation tools

- ACDD (Attribute Conventions for Data Discovery)



Global Attributes

Highly Recommended

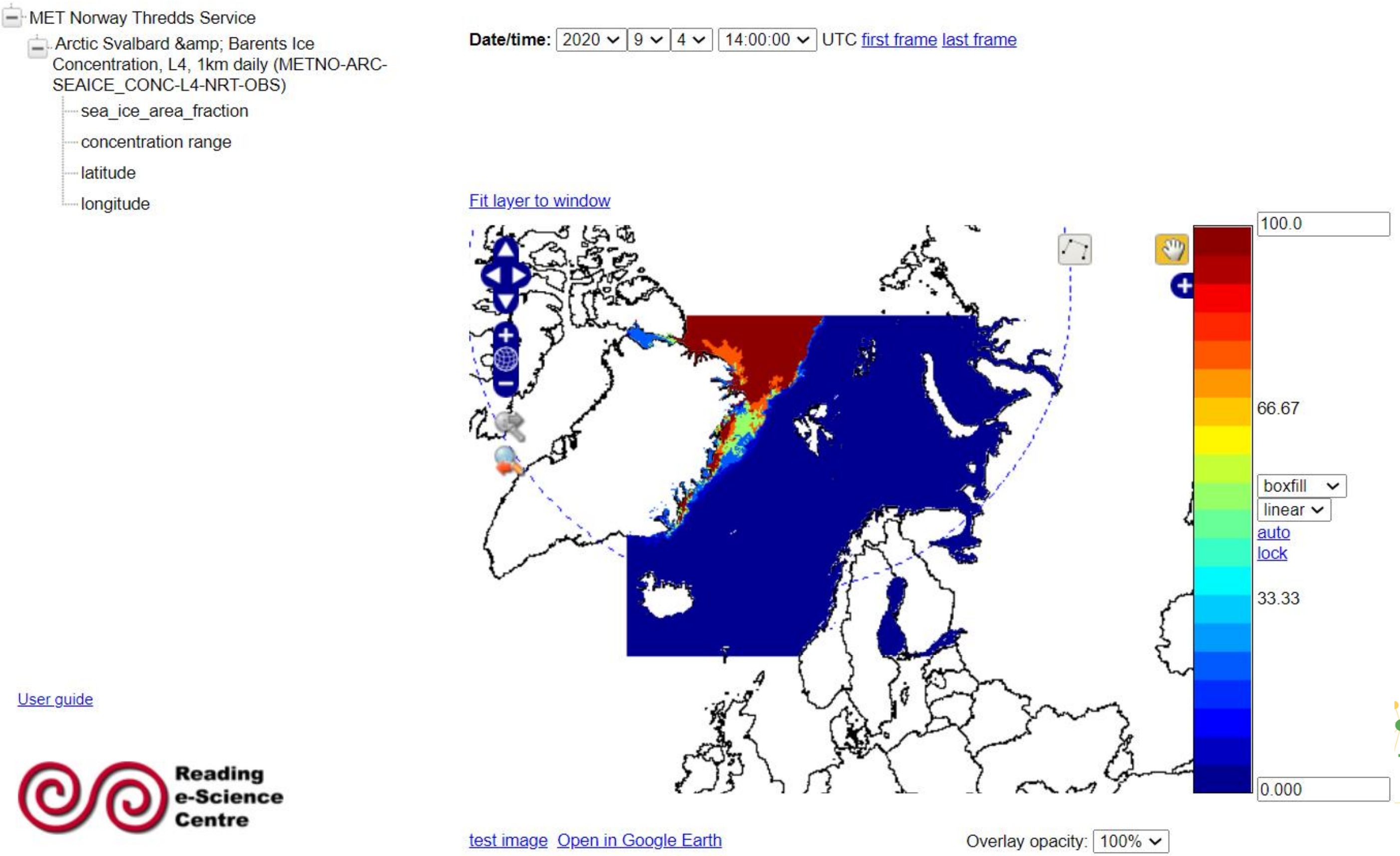
Attribute	Description
title	A short phrase or sentence describing the dataset. In many discovery systems, the title will be displayed in the results list from a search, and therefore should be human readable and reasonable to display in a list of such names. This attribute is also recommended by the NetCDF Users Guide and the CF conventions.
summary	A paragraph describing the dataset, analogous to an abstract for a paper.
keywords	A comma-separated list of key words and/or phrases. Keywords may be common words or phrases, terms from a controlled vocabulary (GCMD is often used), or URIs for terms from a controlled vocabulary (see also "keywords_vocabulary" attribute).
Conventions	A comma-separated list of the conventions that are followed by the dataset. For files that follow this version of ACDD, include the string 'ACDD-1.3'. (This attribute is described in the NetCDF Users Guide.)

Recommended

Attribute	Description
id	An identifier for the data set, provided by and unique within its naming authority. The combination of the "naming authority" and the "id" should be globally unique, but the id can be globally unique by itself also. IDs can be URLs, URNs, DOIs, meaningful text strings, a local key, or any other unique string of characters. The id should not include white space characters.
naming_authority	The organization that provides the initial id (see above) for the dataset. The naming authority should be uniquely specified by this attribute. We recommend using reverse-DNS naming for the naming authority; URIs are also acceptable. Example: 'edu.ucar.unidata'.
history	Provides an audit trail for modifications to the original data. This attribute is also in the NetCDF Users Guide: 'This is a character array with a line for each invocation of a program that has modified the dataset. Well-behaved generic netCDF applications should append a line containing: date, time of day, user name, program name and command arguments.' To include a more complete description you can append a reference to an ISO Lineage entity; see NOAA EDM ISO Lineage guidance.

Validation tools for NetCDF/CF

Example:
Sea ice chart
from met.no
delivered as part
of CMEMS



- MET Norway Thredds Service
  - Arctic Svalbard &amp; Barents Ice Concentration, L4, 1km daily (METNO-ARC-SEAICE_CONC-L4-NRT-OBS)
    - sea_ice_area_fraction
    - concentration range
    - latitude
    - longitude

Date/time: 2020 9 4 14:00:00 UTC first frame last frame

Fit layer to window

100.0
66.67
33.33
0.000

boxfill
linear
auto
lock

User guide

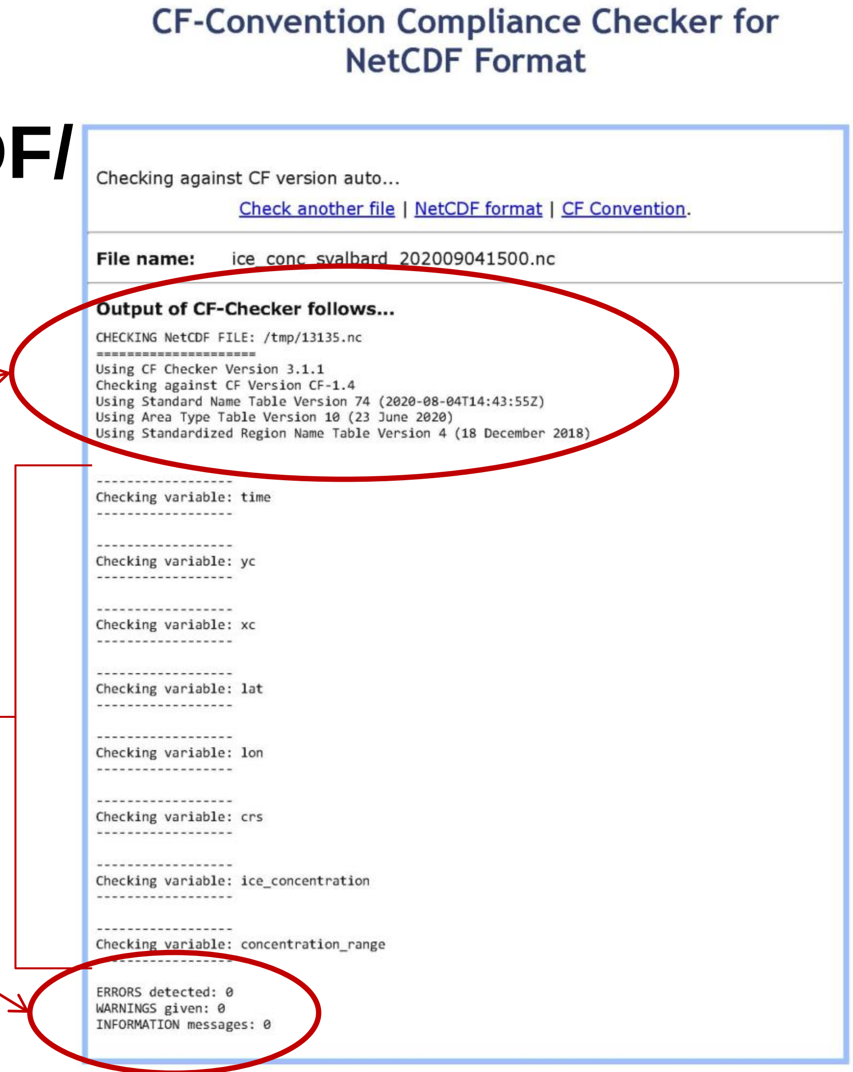
Reading e-Science Centre

test image Open in Google Earth

Overlay opacity: 100%

Validation tools for NetCDF/

Example: PUMA validator



CF-Convention Compliance Checker for NetCDF Format

Checking against CF version auto...

Check another file | NetCDF format | CF Convention.

File name: ice_conc_svalbard_202009041500.nc

Output of CF-Checker follows...

```
CHECKING NetCDF FILE: /tmp/13135.nc
=====================
Using CF Checker Version 3.1.1
Checking against CF Version CF-1.4
Using Standard Name Table Version 74 (2020-08-04T14:43:55Z)
Using Area Type Table Version 10 (23 June 2020)
Using Standardized Region Name Table Version 4 (18 December 2018)

------------------
Checking variable: time
------------------

------------------
Checking variable: yc
------------------

------------------
Checking variable: xc
------------------

------------------
Checking variable: lat
------------------

------------------
Checking variable: lon
------------------

------------------
Checking variable: crs
------------------

------------------
Checking variable: ice_concentration
------------------

------------------
Checking variable: concentration_range
------------------

ERRORS detected: 0
WARNINGS given: 0
INFORMATION messages: 0
```

Finds CF version

Checking all variables

Summary report - ALL PASS

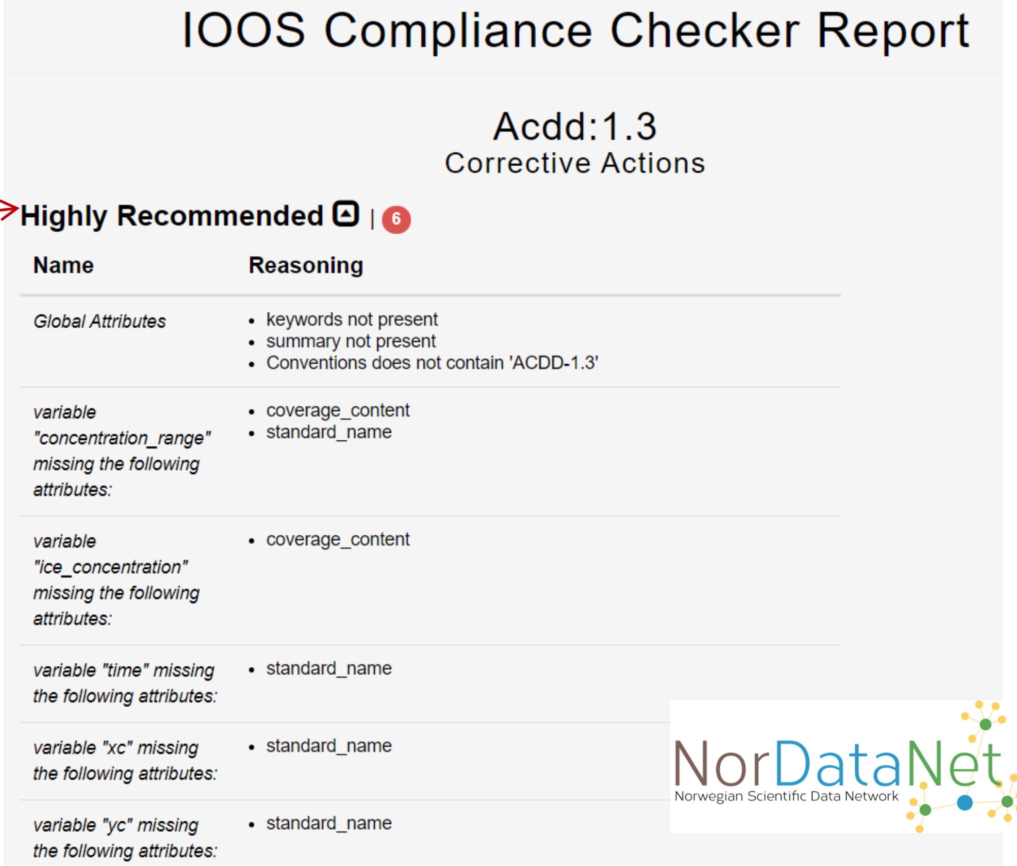
Validation tools for NetCDF/CF

Example: NorDataNet validator

Suggests improvements

Only supports CF 1.6 - 1.7

File uses CF 1.4



IOOS Compliance Checker Report

Acdd:1.3
Corrective Actions

Highly Recommended | 6

Name	Reasoning
Global Attributes	keywords not present summary not present Conventions does not contain 'ACDD-1.3'
variable "concentration_range" missing the following attributes:	coverage_content standard_name
variable "ice_concentration" missing the following attributes:	coverage_content
variable "time" missing the following attributes:	standard_name
variable "xc" missing the following attributes:	standard_name
variable "yc" missing the following attributes:	standard_name

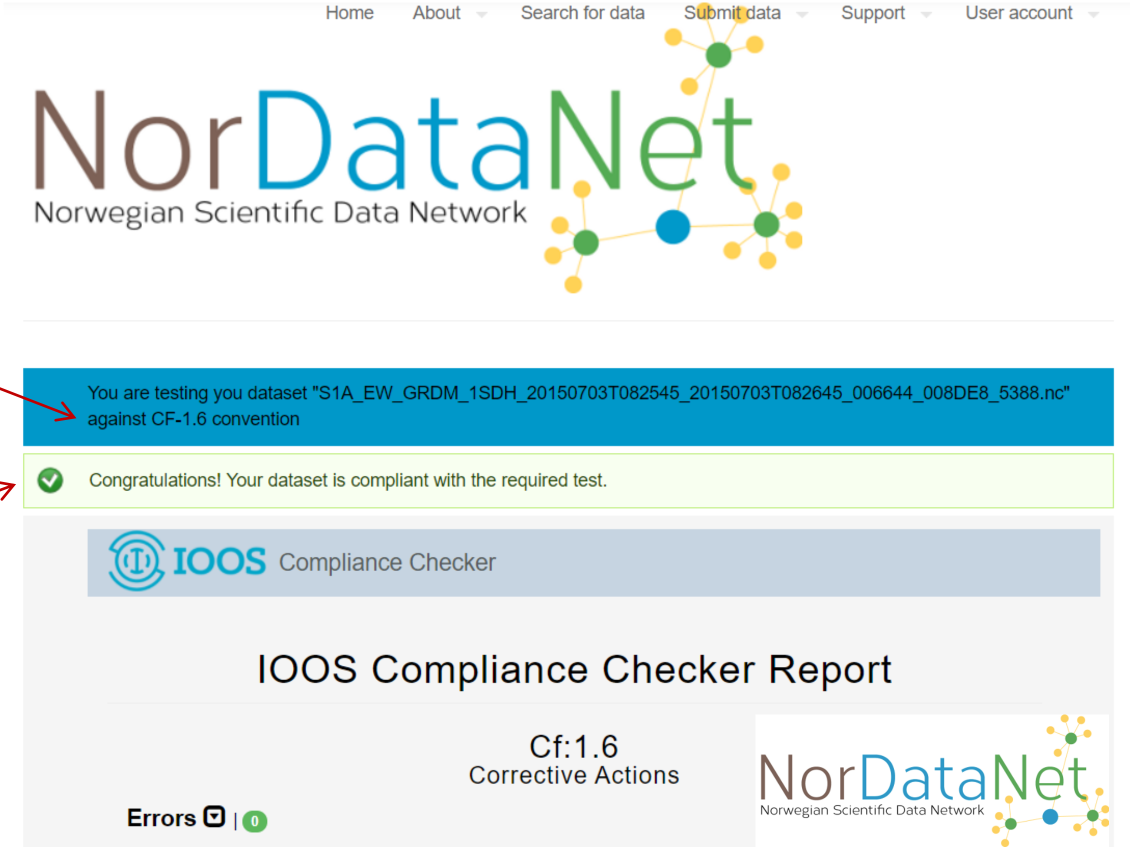
NorDataNet
Norwegian Scientific Data Network

Validation tools for NetCDF/CF

Example: NorDataNet validator

Follows CF 1.6

ALL TEST PASS



You are testing you dataset "S1A_EW_GRDM_1SDH_20150703T082545_20150703T082645_006644_008DE8_5388.nc" against CF-1.6 convention

Congratulations! Your dataset is compliant with the required test.

IOOS Compliance Checker Report

Cf:1.6
Corrective Actions

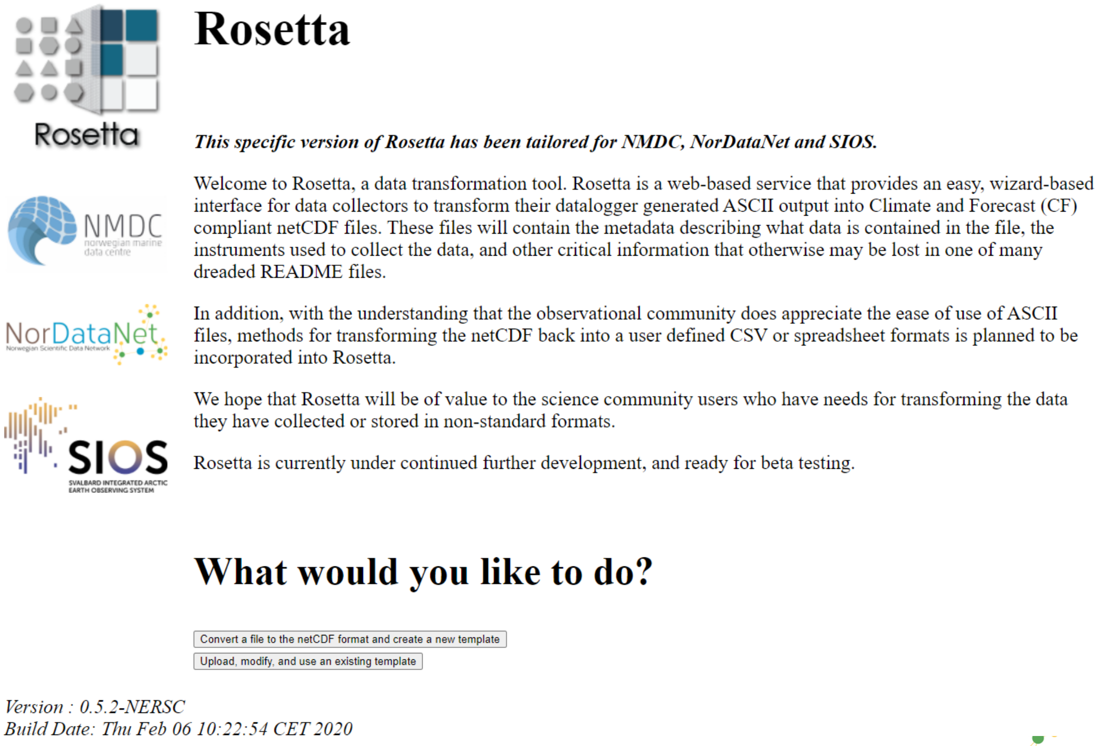
Errors | 0

Tools for documenting metadata

Rosetta

- Web-based application
- Can read metadata from header(s), and add user defined metadata
- Saves “setup” as templates for future use
- Open source tool, written in Java
- Customised version for NMDC, NorDataNet and SIOS

http://tomcat.nersc.no/rosetta/


Rosetta
NMDC
norwegian marine data centre
NorDataNet
Norwegian Scientific Data Network
SIOS
SVALBARD INTEGRATED ARCTIC EARTH OBSERVING SYSTEM

Rosetta

This specific version of Rosetta has been tailored for NMDC, NorDataNet and SIOS.

Welcome to Rosetta, a data transformation tool. Rosetta is a web-based service that provides an easy, wizard-based interface for data collectors to transform their datalogger generated ASCII output into Climate and Forecast (CF) compliant netCDF files. These files will contain the metadata describing what data is contained in the file, the instruments used to collect the data, and other critical information that otherwise may be lost in one of many dreaded README files.

In addition, with the understanding that the observational community does appreciate the ease of use of ASCII files, methods for transforming the netCDF back into a user defined CSV or spreadsheet formats is planned to be incorporated into Rosetta.

We hope that Rosetta will be of value to the science community users who have needs for transforming the data they have collected or stored in non-standard formats.

Rosetta is currently under continued further development, and ready for beta testing.

What would you like to do?

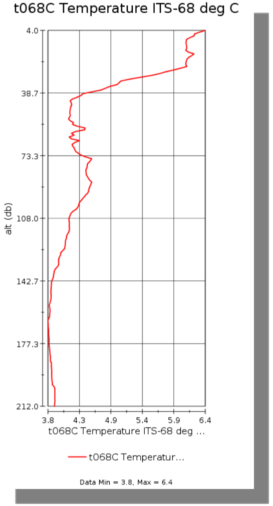
Convert a file to the netCDF format and create a new template
Upload, modify, and use an existing template

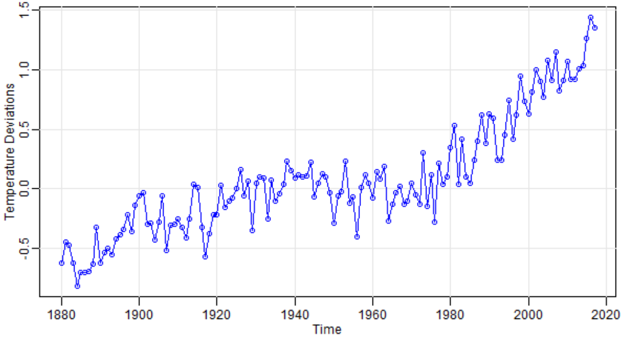
Version : 0.5.2-NERSC
Build Date: Thu Feb 06 10:22:54 CET 2020

Tools for documenting metadata

Rosetta

- Profiles, time series, trajectory


t068C Temperature ITS-68 deg C
4.0
38.7
73.3
108.0
142.7
177.3
212.0
alt (db)
3.8
4.3
4.9
5.4
5.9
6.4
t068C Temperature ITS-68 deg ...
t068C Temperatur...
Data Min = 3.8, Max = 6.4



- Template concept and benefits

Example - Rosetta

Steps

1. Start with template or build one
2. Select type of observation platform
3. Specify headerlines in the datafile
4. Specify delimiter and decimal seperator
5. Specify variable attributes
6. Specify variable attribute details
7. Specify site spesific information
8. Specify general information
9. Download resulting NetCDF file and new template

USEFUL ARCTIC UAK KNOWLEDGE

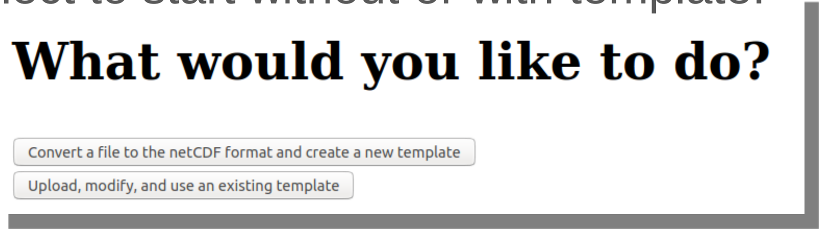
The Research Council of Norway

NERSC

NorDataNet
Norwegian Scientific Data Network

Example - Rosetta

Select to start without or with template:

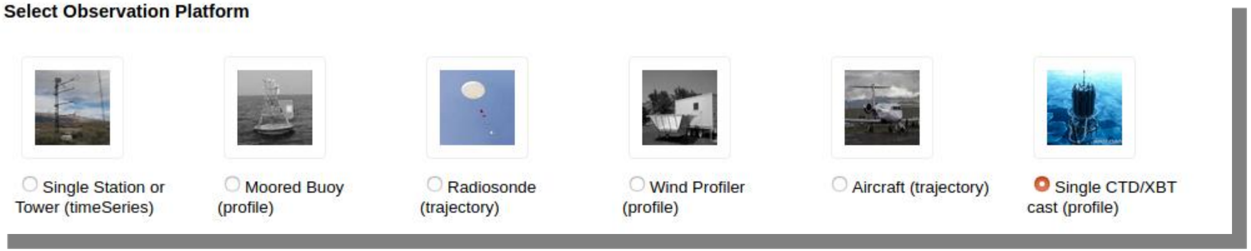


What would you like to do?

Convert a file to the netCDF format and create a new template

Upload, modify, and use an existing template

Select observation Platform



Select Observation Platform

○ Single Station or Tower (timeSeries)

○ Moored Buoy (profile)

○ Radiosonde (trajectory)

○ Wind Profiler (profile)

○ Aircraft (trajectory)

● Single CTD/XBT cast (profile)

Upload file(s)

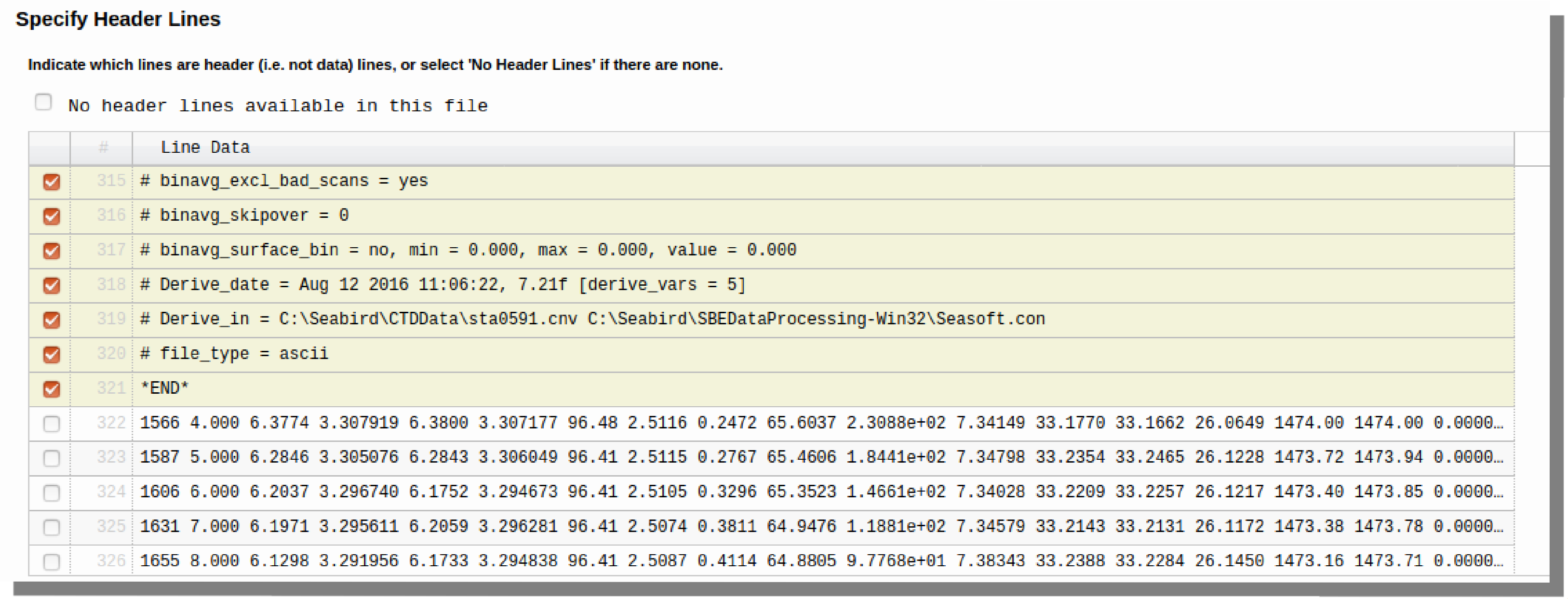
browse and upload (template and) text file

NorDataNet
Norwegian Scientific Data Network

Example - Rosetta

Specify headerlines

- All lines before data begins must be selected here



Specify Header Lines

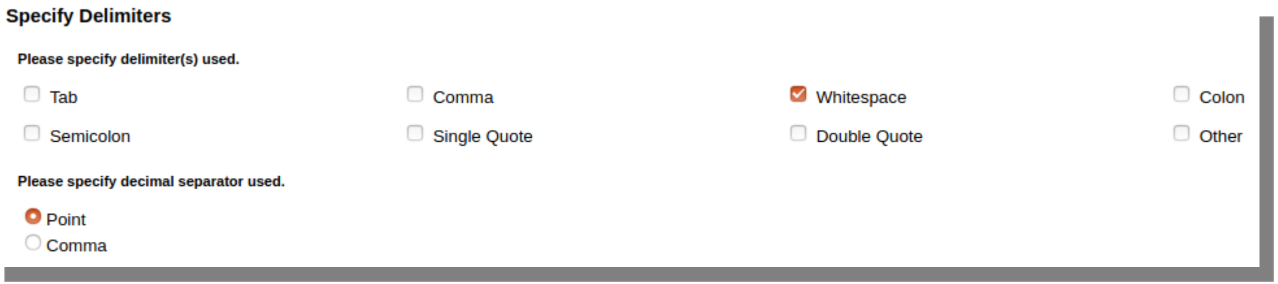
Indicate which lines are header (i.e. not data) lines, or select 'No Header Lines' if there are none.

□ No header lines available in this file

	#	Line Data
☑	315	# binavg_excl_bad_scans = yes
☑	316	# binavg_skipover = 0
☑	317	# binavg_surface_bin = no, min = 0.000, max = 0.000, value = 0.000
☑	318	# Derive_date = Aug 12 2016 11:06:22, 7.21f [derive_vars = 5]
☑	319	# Derive_in = C:\Seabird\CTDData\sta0591.cnv C:\Seabird\SBEDataProcessing-Win32\Seasoft.con
☑	320	# file_type = ascii
☑	321	*END*
□	322	1566 4.000 6.3774 3.307919 6.3800 3.307177 96.48 2.5116 0.2472 65.6037 2.3088e+02 7.34149 33.1770 33.1662 26.0649 1474.00 1474.00 0.0000…
□	323	1587 5.000 6.2846 3.305076 6.2843 3.306049 96.41 2.5115 0.2767 65.4606 1.8441e+02 7.34798 33.2354 33.2465 26.1228 1473.72 1473.94 0.0000…
□	324	1606 6.000 6.2037 3.296740 6.1752 3.294673 96.41 2.5105 0.3296 65.3523 1.4661e+02 7.34028 33.2209 33.2257 26.1217 1473.40 1473.85 0.0000…
□	325	1631 7.000 6.1971 3.295611 6.2059 3.296281 96.41 2.5074 0.3811 64.9476 1.1881e+02 7.34579 33.2143 33.2131 26.1172 1473.38 1473.78 0.0000…
□	326	1655 8.000 6.1298 3.291956 6.1733 3.294838 96.41 2.5087 0.4114 64.8805 9.7768e+01 7.38343 33.2388 33.2284 26.1450 1473.16 1473.71 0.0000…

Example - Rosetta

pecify delimiter and decimal seperator:



Specify Delimiters

Please specify delimiter(s) used.

☐ Tab ☐ Comma ☑ Whitespace ☐ Colon

☐ Semicolon ☐ Single Quote ☐ Double Quote ☐ Other

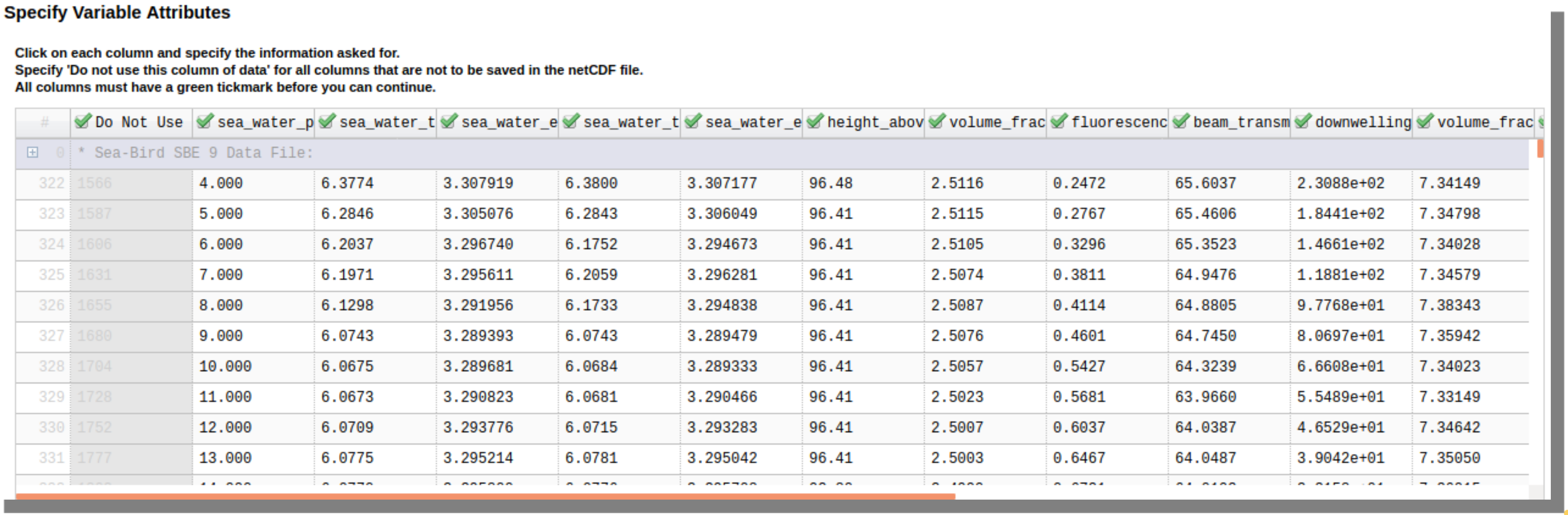
Please specify decimal separator used.

● Point

○ Comma

Example - Rosetta

Specify variable attributes



Specify Variable Attributes

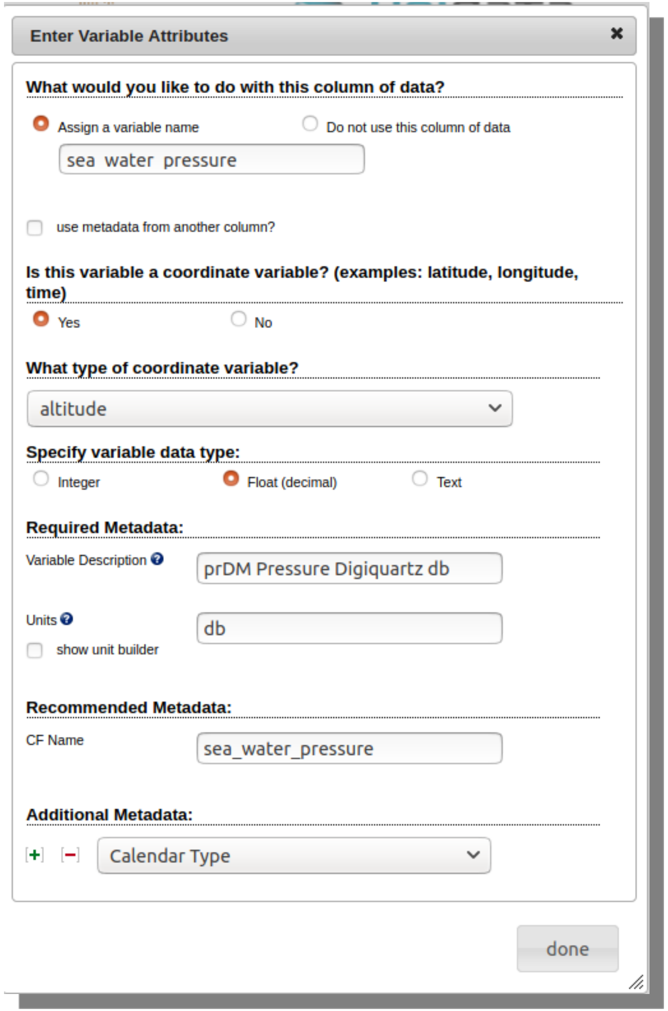
Click on each column and specify the information asked for.
Specify 'Do not use this column of data' for all columns that are not to be saved in the netCDF file.
All columns must have a green tickmark before you can continue.

#	✓ Do Not Use	✓ sea_water_p	✓ sea_water_t	✓ sea_water_e	✓ sea_water_t	✓ sea_water_e	✓ height_abov	✓ volume_frac	✓ fluorescenc	✓ beam_transm	✓ downwelling	✓ volume_frac
⊞ 0	* Sea-Bird SBE 9 Data File:											
322	1566	4.000	6.3774	3.307919	6.3800	3.307177	96.48	2.5116	0.2472	65.6037	2.3088e+02	7.34149
323	1587	5.000	6.2846	3.305076	6.2843	3.306049	96.41	2.5115	0.2767	65.4606	1.8441e+02	7.34798
324	1606	6.000	6.2037	3.296740	6.1752	3.294673	96.41	2.5105	0.3296	65.3523	1.4661e+02	7.34028
325	1631	7.000	6.1971	3.295611	6.2059	3.296281	96.41	2.5074	0.3811	64.9476	1.1881e+02	7.34579
326	1655	8.000	6.1298	3.291956	6.1733	3.294838	96.41	2.5087	0.4114	64.8805	9.7768e+01	7.38343
327	1680	9.000	6.0743	3.289393	6.0743	3.289479	96.41	2.5076	0.4601	64.7450	8.0697e+01	7.35942
328	1704	10.000	6.0675	3.289681	6.0684	3.289333	96.41	2.5057	0.5427	64.3239	6.6608e+01	7.34023
329	1728	11.000	6.0673	3.290823	6.0681	3.290466	96.41	2.5023	0.5681	63.9660	5.5489e+01	7.33149
330	1752	12.000	6.0709	3.293776	6.0715	3.293283	96.41	2.5007	0.6037	64.0387	4.6529e+01	7.34642
331	1777	13.000	6.0775	3.295214	6.0781	3.295042	96.41	2.5003	0.6467	64.0487	3.9042e+01	7.35050

NorDataNet
Norwegian Scientific Data Network

Example - Rosetta

Specify variable attribute details



What would you like to do with this column of data?

● Assign a variable name ○ Do not use this column of data

sea water pressure

□ use metadata from another column?

Is this variable a coordinate variable? (examples: latitude, longitude, time)

● Yes ○ No

What type of coordinate variable?

altitude

Specify variable data type:

○ Integer ● Float (decimal) ○ Text

Required Metadata:

Variable Description: prDM Pressure Digiquartz db

Units: db

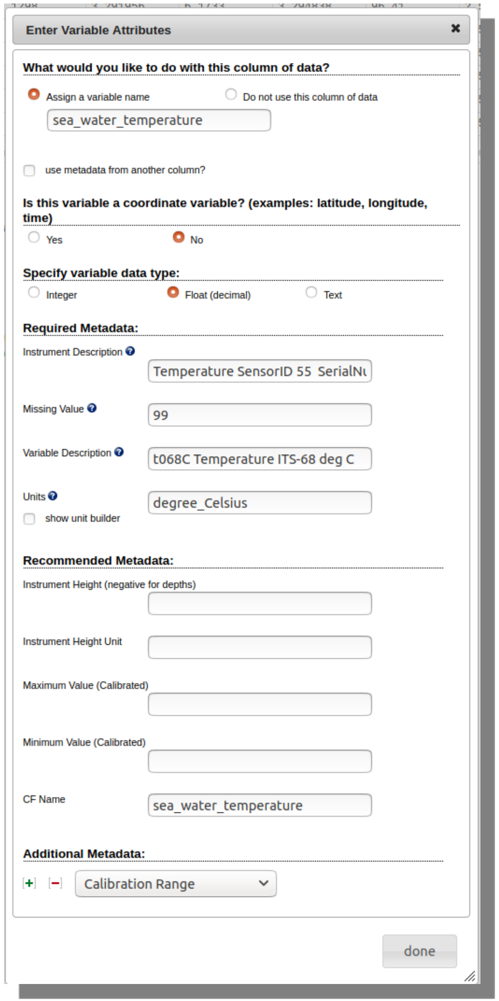
□ show unit builder

Recommended Metadata:

CF Name: sea_water_pressure

Additional Metadata:

[+] [−] Calendar Type



What would you like to do with this column of data?

● Assign a variable name ○ Do not use this column of data

sea_water_temperature

□ use metadata from another column?

Is this variable a coordinate variable? (examples: latitude, longitude, time)

○ Yes ● No

Specify variable data type:

○ Integer ● Float (decimal) ○ Text

Required Metadata:

Instrument Description: Temperature SensorID 55 SerialNu

Missing Value: 99

Variable Description: t068C Temperature ITS-68 deg C

Units: degree_Celsius

□ show unit builder

Recommended Metadata:

Instrument Height (negative for depths):

Instrument Height Unit:

Maximum Value (Calibrated):

Minimum Value (Calibrated):

CF Name: sea_water_temperature

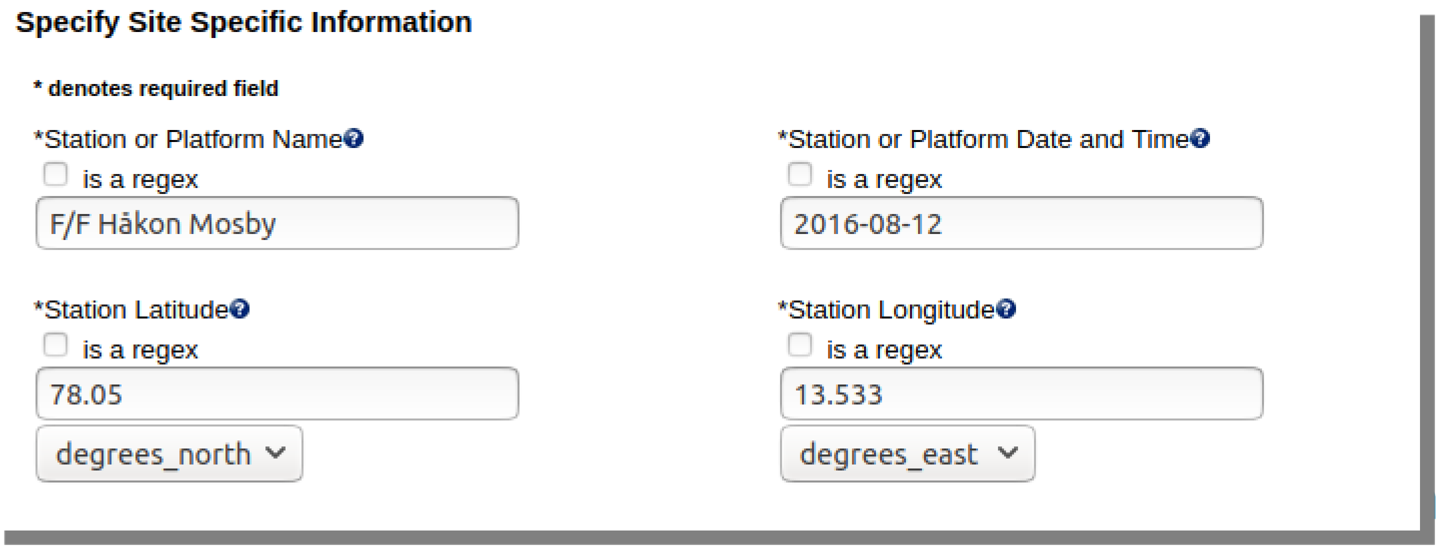
Additional Metadata:

[+] [−] Calibration Range

Example - Rosetta

Specify site spesific information

- what information that appears here is dependent on type of observation platform



Specify Site Specific Information

*** denotes required field**

*Station or Platform Name
□ is a regex
F/F Håkon Mosby

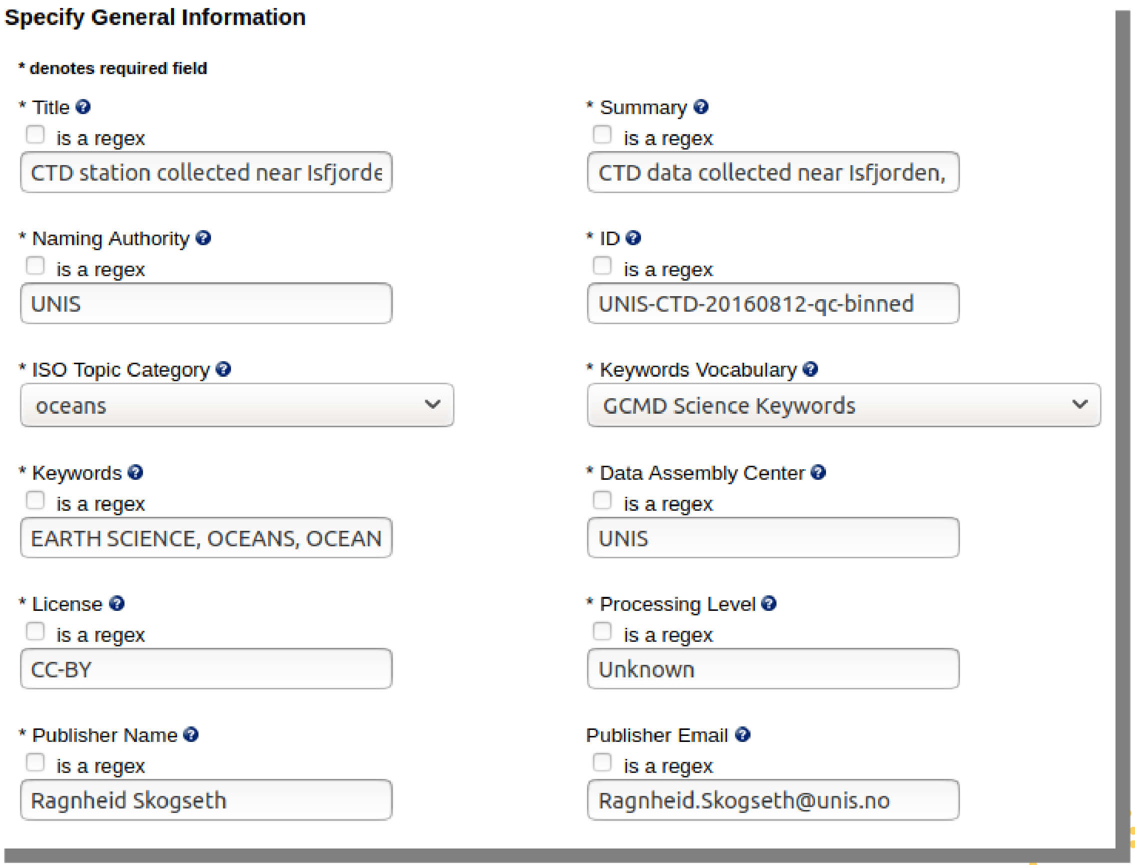
*Station or Platform Date and Time
□ is a regex
2016-08-12

*Station Latitude
□ is a regex
78.05
degrees_north

*Station Longitude
□ is a regex
13.533
degrees_east

Example - Rosetta

Specify general information



Specify General Information

*** denotes required field**

* Title

☐ is a regex

CTD station collected near Isfjorde

* Summary

☐ is a regex

CTD data collected near Isfjorden,

* Naming Authority

☐ is a regex

UNIS

* ID

☐ is a regex

UNIS-CTD-20160812-qc-binned

* ISO Topic Category

oceans

* Keywords Vocabulary

GCMD Science Keywords

* Keywords

☐ is a regex

EARTH SCIENCE, OCEANS, OCEAN

* Data Assembly Center

☐ is a regex

UNIS

* License

☐ is a regex

CC-BY

* Processing Level

☐ is a regex

Unknown

* Publisher Name

☐ is a regex

Ragnheid Skogseth

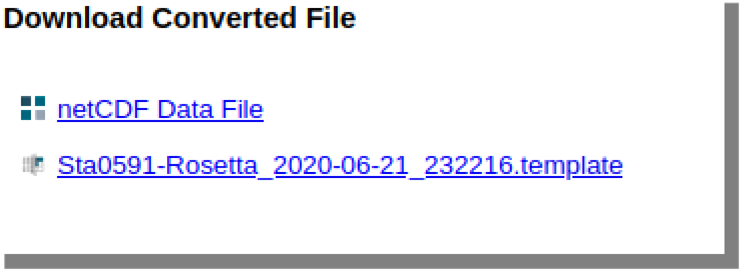
Publisher Email

☐ is a regex

Ragnheid.Skogseth@unis.no

Example - Rosetta

Download resulting files.



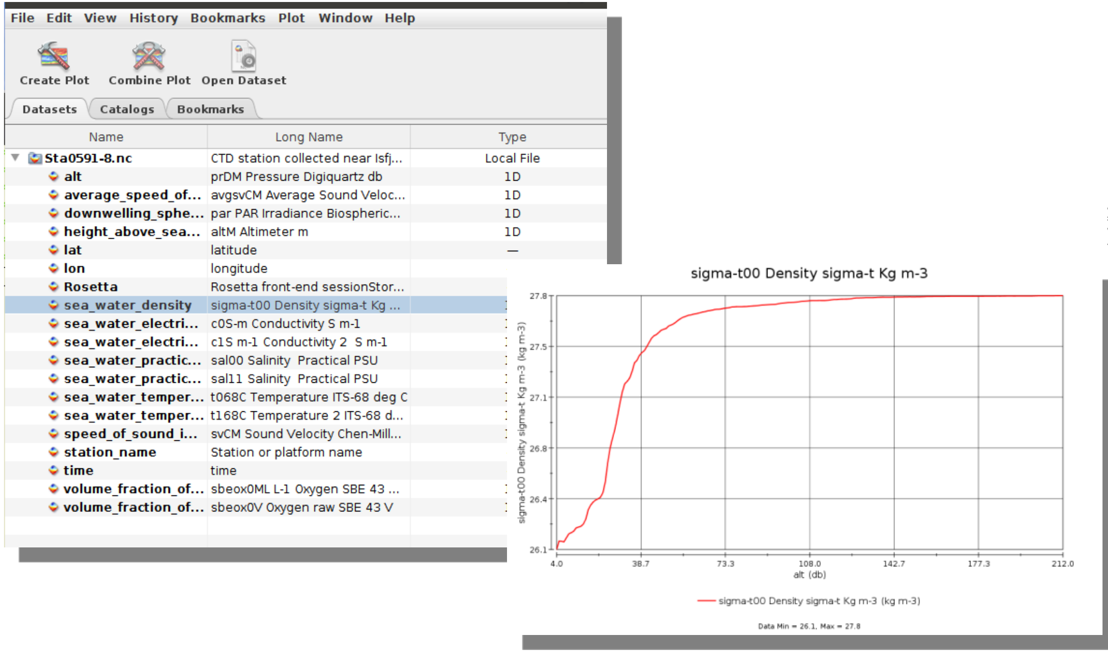
Download Converted File

- netCDF Data File
- Sta0591-Rosetta_2020-06-21_232216.template

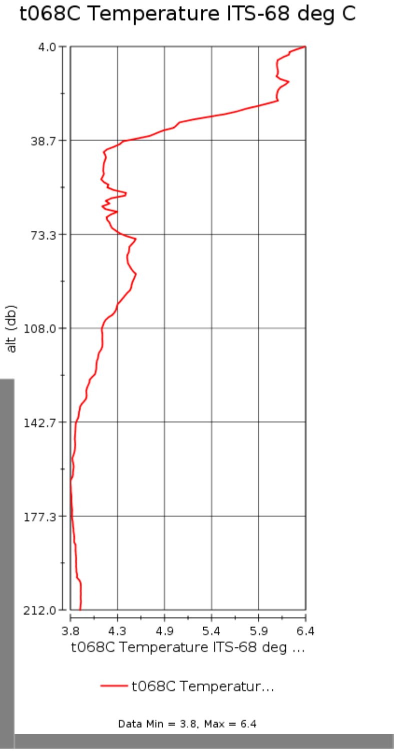
Example - Rosetta

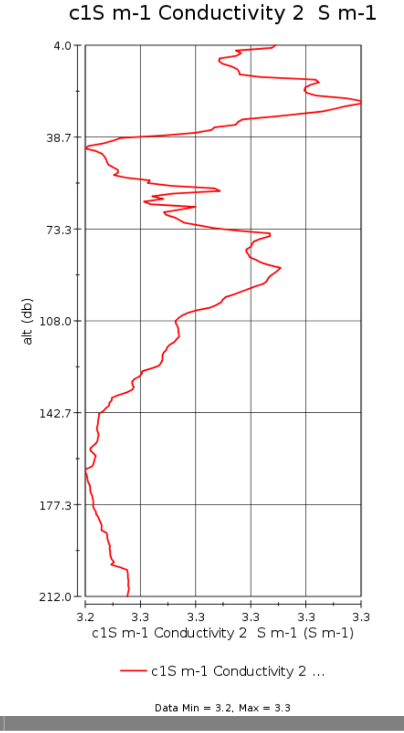
Inspect your new NetCDF file, e.g. download Panoply:

https://www.giss.nasa.gov/tools/panoply/download/



Name	Long Name	Type
Sta0591-8.nc	CTD station collected near Isfj...	Local File
alt	prDM Pressure Digiquartz db	1D
average_speed_of...	avgsvCM Average Sound Veloc...	1D
downwelling_sphe...	par PAR Irradiance Biospheric...	1D
height_above_sea...	altM Altimeter m	1D
lat	latitude	—
lon	longitude	
Rosetta	Rosetta front-end sessionStor...	
sea_water_density	sigma-t00 Density sigma-t Kg ...	
sea_water_electri...	c0S-m Conductivity S m-1	
sea_water_electri...	c1S m-1 Conductivity 2 S m-1	
sea_water_practic...	sal00 Salinity Practical PSU	
sea_water_practic...	sal11 Salinity Practical PSU	
sea_water_temper...	t068C Temperature ITS-68 deg C	
sea_water_temper...	t168C Temperature 2 ITS-68 d...	
speed_of_sound_i...	svCM Sound Velocity Chen-Mill...	
station_name	Station or platform name	
time	time	
volume_fraction_of...	sbeox0ML L-1 Oxygen SBE 43 ...	
volume_fraction_of...	sbeox0V Oxygen raw SBE 43 V	





Benefits of standardisation

- Metadata
  - Document all important aspects of your data so you can reuse it! ...and know what you are working with...
  - Become visible as a scientist producing valueable data
  - Support reproducible science - by enabling other scientist to reuse your data
- Formats
  - Large collections of libraries and tools you can use
  - Eases collaboration with other scientists - share data *and* code
  - Metadata can be included in the same file as data (e.g. NetCDF)

NorDataNet
Norwegian Scientific Data Network

Resources

Rosetta: http://tomcat.nersc.no/rosetta/ ; https://github.com/Unidata/rosetta

NCO/CDO: http://nco.sourceforge.net/

netCDF4: https://unidata.github.io/netcdf4-python/netCDF4/index.html

xarray: http://xarray.pydata.org/en/stable/

DataOne: https://old.dataone.org/education-modules

ESIP: https://commons.esipfed.org/node/1422 ; Data management training