IPSL BootCamp: netCDF*

Institute Pierre Simone Laplace, IPSL BootCamp

The content of the BootCamp can be found in:
https://forge.ipsl.jussieu.fr/igcmg_doc/wiki/Train

March 25, 2016

Contents

1 Introduction **1**

2 Field characterization **1**

3 netCDF **2**
3.1 dimensions . . . 2
3.2 variables . . . 2
3.2.1 dimensional variables/coordinate variables . . . 3
3.3 attributes . . . 4

4 CMOR/CF-Compilant format **5**

5 Tools **7**
5.1 File content: cdo, nco . . . 7
5.2 ncview: Visualization tool . . . 7

6 Useful links **8**

1 Introduction

Since the beginning of the use of the computer resources in the geophysics, the management of the data has been an important issue. Geophysics scientists use and produce a large amount of data. They origins are different such as observations and model outputs and come from all over the world.

Thus it is clear that a binary, reliable, robust, standardized and easy to use file system was required. First versions were based for example on binary unformatted output from FORTRAN. Latter on, the GRIB binary data was broadly used as a uniform and flexible data format. However auto-consistency and self-description was required and the network Common Data Form (netCDF Unidata, 2015) appeared.

2 Field characterization

A generic `variable` χ consists on a set of values whith a representation of a given set of values with certain characteristics.

*Main author of this chapter: L. Fita, Laboratoire de Météorologie Dynamique (LMD). This work is licensed under Creative Commons Attribution-ShareAlike 4.0 International License

Let's take as example a 2D variable which its spatial representation is given by a pair of two indices i, j with a length $dimx$, $dimy$, with a third one representative of time t (of length $dimt$). Thus the representation of χ, can be understood as `3-dimension` variable.

$$\chi(i, j, t) \; ; \begin{cases} i & [1, dimx] \\ j & [1, dimy] \\ t & [1, dimt] \end{cases}$$

χ would have also a recognizable *name* and *units*. The *name* and *units* of the field are conventions agreed to understand the nature of the field, they consist of the meta-data or `attributes` of the variable.

χ needs to be spatially and temporally *localized*. That means to know where and when χ provides the data. Thus a mimic between the 'data' and the 'real' space has to be provided in a standardized way that everyone can understand.

3 netCDF

netCDF (`http://www.unidata.ucar.edu/software/netcdf/`) is a binary file standard developed by UNIDATA (`http://www.unidata.ucar.edu/`). It is self-describing and portable and it is use worldwide in the geophysical sciences. A full set of libraries is available to read,create and manage these kind of files. UNIDATA provides the libraries over different languages: C, Fortran, C++, Java, and others. The new standard with enhanced capabilities is called netCDF4 and some of them make necessary the use of another set of libraries called HDF5 (`https://www.hdfgroup.org/HDF5/`). These libraries are also available in order to develop own programs to manage any aspect of netCDF files, or used to develop netCDF interfaces for other languages (e.g.: a python netCDF interface `http://netcdf4-python.googlecode.com/svn/trunk/docs/netCDF4-module.html`)

These libraries provide different useful tools such as:

- `nc-config`: provides information about the environment used during the compilation of the libraries
- `ncdump`: provides content of the file (e.g.: `-h`, flag to provide the structure of the file, `-v` outputs values of a given varible)
- `nccopy`: perform a copy of the file to different netCDF versions (or also to internally compress netCDF content)
- `ncgen`: generates the code from which one could reproduce the structure of the netCDF nfile

3.1 dimensions

- Dimensions are always monotonic incremental integers.
- Usually they have a given size, but in some cases they can have an `UNLIMITED` size (e.g.: time). It is useful when one wants to concatenate files. It is also known as *record dimension*

```
$ ncdump -h test.nc
netcdf test {
dimensions:
        x = 32 ;
        y = 32 ;
        z = 32 ;
        time = UNLIMITED ; // (20 currently)
        Lstring = 50 ;
        bnds = 2 ;
variables:
```

3.2 variables

- netCDF files can content among others the standard types of variable: integer, float, character, 32/64-bit integer, 32/64-bit float

- Variables can have any desired shape
- Shape of the variables are related to the dimensions provided with the file. They are usually sorted from the less changing to the most changing one (e.g.: time, z, x, y)
- There are two main kinds of variables:
  - Dimensional variables or coordinate variables: variables to provide data for a given dimension (e.g.: longitudes for the dimx)
  - Field variables: variables with any kind of data (e.g.: temperature, salinity, soil moisture)

```
$ ncdump -h test.nc
(...)
variables:
        float T(time, z, y, x) ;
```

- Character variables: netCDF does not manage string types of variables. Instead of it, strings are treated as 1-character vectors. Thus, the standard use is to define a two-dimensional variable, with a variable with a maximum length of the string.

```
$ ncdump -h test.nc
(...)
        char pftnames(Npft, Lstring) ;
```

3.2.1 dimensional variables/coordinate variables

These variables are used to provide information about the real equivalence of a given dimension in the file. Without these variables, we only will have well shaped data, but will be impossible to know they equivalence in the real world. Commonly they are float32 and use to have the same name as the dimension to they refer. There are different kinds:

- **Geographical coordinates:** the most common ones are the longitude and the latitude for the `x-axis` and `y-axis` dimensions. Commonly there are two cases:
  1. 1D variables: This is the case of regular projections, in which longitudes and latitudes only change along their respective axis (e.g.: longitudes at a given i does not change along the latitude (incrementing j)
  2. 2D variables: Non angle conservative projections, in which longitudes and latitudes change along all axes (e.g.: in a Lambert-Conformal projection, longitudes at a given i change along the latitude (incrementing j)

```
        double lon(y, x) ;
        double lat(y, x) ;
```

- **Vertical coordinates:** these variables are used to describe the vertical axis of the variables. The most common ones are the atmospheric pressure, sigma levels, vertical height, or sea depth and soil depth. In this case, although the vertical dimension index k might increase, vertical variable might decrease as it is the case on the atmospheric pressure and sigma coordinates. Even for example, in case of non-hydrostatic atmospheric models, the vertical pressure might change with time.

```
        double height(z) ;
```

- **Temporal coordinate:** these variable is used to reference the time in the file. A non recommendable way, would be to use a string variable to provide the values (e.g.: `[YYYY]-[MM]-[DD]_[HH]:[MM]:[SS]`). However, it is very common to provide the time passed since a given reference date (g.e.: *hours since 1949-12-01 00:00:00*)

```
        double time(time) ;
```

- **Additional:** there is a large variety of possible additional ones such us: `station` for the numerical station id as coordinate variable, or `ensmem` for the ensemble member id within an ensemble, etc...

3.3 attributes

Attributes are used to store all the complementary information that is not explicitly data but ”meta-data” and that is required by some tool processing the file or that might just be good to know. There are two kind of attributes:

- **variable attributes:** They are used to provide the meta-data of the variable such us: *units, standard_name, long description of the variable, coordinates of the variable, _FillValue*[1]*, offset, cell_methods, ...*

```
        float t2m(time_counter, lat, lon) ;
                t2m:units = "K" ;
                t2m:standard_name = "Temperature 2m" ;
                t2m:_FillValue = 9.96921e+36f ;
                t2m:long_name = "Temperature 2m" ;
                t2m:online_operation = "ave(X)" ;
                t2m:interval_operation = 1800.f ;
                t2m:interval_write = 2678400.f ;
                t2m:coordinates = "time_counter lat lon" ;
```

- **global attributes:** They are used to provide the information that attains to all the file. This information can be for example: institution providing the data, parameters of the projection, version of the data-set, history, etc...

[1] standard attribute to mark no-value

```
$ ncdump -h tas_Amon_IPSL-CM5A-LR_historical_190001-200012.nc
(...)
// global attributes:
          :CDI = "Climate Data Interface version 1.6.8 (http://mpimet.mpg.de/cdi)" ;
          :Conventions = "CF-1.4" ;
          :source = "IPSL-CM5A-LR (2010) :  atmos :  LMDZ4 (LMDZ4_v5, 96x95x39); ocean :
ORCA2 (NEMOV2_3, 2x2L31); seaIce :  LIM2 (NEMOV2_3); ocnBgchem :
PISCES (NEMOV2_3); land :  ORCHIDEE (orchidee_1_9_4_AR5)" ;
          :institution = "IPSL (Institut Pierre Simon Laplace, Paris, France)" ;
          :institute_id = "IPSL" ;
          :experiment_id = "historical" ;
          :model_id = "IPSL-CM5A-LR" ;
          :forcing = "Nat,Ant,GHG,SA,Oz,LU,SS,Ds,BC,MD,OC,AA" ;
          :parent_experiment_id = "piControl" ;
          :parent_experiment_rip = "r1i1p1" ;
          :branch_time = 1850.  ;
          :contact = "ipsl-cmip5 _at_ ipsl.jussieu.fr Data manager :  Sebastien Denvil" ;
          :comment = "This 20th century simulation include natural and anthropogenic
forcings." ;
          :references = "Model documentation and further reference available here :
http://icmc.ipsl.fr" ;
          :initialization_method = 1 ;
          :physics_version = 1 ;
          :tracking_id = "826ee5e9-3cc9-40a6-a42b-d84c6b4aad97" ;
          :product = "output" ;
          :experiment = "historical" ;
          :frequency = "mon" ;
          :creation_date = "2011-02-23T17:52:35Z" ;
          :project_id = "CMIP5" ;
          :table_id = "Table Amon (31 January 2011) 53b766a395ac41696af40aab76a49ae5" ;
          :title = "IPSL-CM5A-LR model output prepared for CMIP5 historical" ;
          :parent_experiment = "pre-industrial control" ;
          :modeling_realm = "atmos" ;
          :realization = 1 ;
          :cmor_version = "2.5.1" ;
          :CDO = "Climate Data Operators version 1.6.8 (http://mpimet.mpg.de/cdo)" ;
          :CliMAF = "CLImate Model Assessment Framework version 0.12 (http://climaf.rtfd.org)"
;
}
```

4 CMOR/CF-Compilant format

In order to facilitate the processing and sharing of files, an internationally coordinated effort has been made to define attribute conventions `http://cfconventions.org/`. This is the format in which the data in the CMIP4-X `http://cmip-pcmdi.llnl.gov/cmip5/` has been stored in the Earth System Grid Federation (ESGF, `http://esgf.llnl.gov/`). Following these specifications, all well designed netCDF tools should be able to work with the given file. The main characteristics are:

- **dimensions:** All the dimensions must be increasing monotonic integer.
- **variables:** All variables must have at least the following three attributes: `std_name`, `long_name`, `units`, which stand for *standard name*, *long name* and *units*. A table[2] with the right names, full descriptions and units of the most common variables was coordinated.

[2] `http://cfconventions.org/standard-names.html`

```
        float nav_lon(y, x) ;
              nav_lon:standard_name = "longitude" ;
              nav_lon:units = "degrees_east" ;
              nav_lon:valid_min = -179.7507f ;
              nav_lon:valid_max = 180.f ;
              nav_lon:long_name = "Longitude" ;
              nav_lon:nav_model = "Default grid" ;
```

- time: Time coordinate is a float variable with a units of `hours since 1949-12-01 00:00:00`. A complementary attribute called `calendar` is used to provide which calendar is used (e.g.: *standard, noleap, ...*)

```
        double time_counter(time_counter) ;
              time_counter:bounds = " time_counter_bnds" ;
              time_counter:calendar = "365_day" ;
              time_counter:time_origin = "01-JAN-0000 00:00:00" ;
              time_counter:units = "days since 0000-01-01 00:00:00" ;
```

- time_bnds: this temporal variable has two values for each time. It is used to define the range of a period, begin the first value the beginning, and the second the end of each period. This variable is used when one would for example refer to a variable which is the daily mean of a variable. The pair of values would be: [YYYY]-[MM]-[DD] 00:00:00, [YYYY]-[MM]-[DD] 23:59:59. It is also common to use an additional centered coordinate variable, that in this case might be [YYYY]-[MM]-[DD] 12:00:00

```
netcdf tas_Amon_IPSL-CM5A-LR_historical_190001-200012 {
dimensions:
        lon = 96 ;
        bnds = 2 ;
        lat = 96 ;
        time = UNLIMITED ; // (1212 currently)

variables:
(...)
        double time(time) ;
              time:standard_name = "time" ;
              time:long_name = "time" ;
              time:bounds = "time_bnds" ;
              time:units = "days since 1850-01-01 00:00:00" ;
              time:calendar = "365_day" ;

        double time_bnds(time, bnds) ;

        float tas(time, lat, lon) ;
              tas:standard_name = "air_temperature" ;
              tas:long_name = "Near-Surface Air Temperature" ;
              tas:units = "K" ;
              tas:_FillValue = 1.e+20f ;
              tas:missing_value = 1.e+20f ;
              tas:original_name = "t2m" ;
              tas:cell_methods = "time:  mean (interval:  30 minutes)" ;
(...)
```

5 Tools

There are a lot of different tools to read, create and manage netCDF.

5.1 File content: cdo, nco

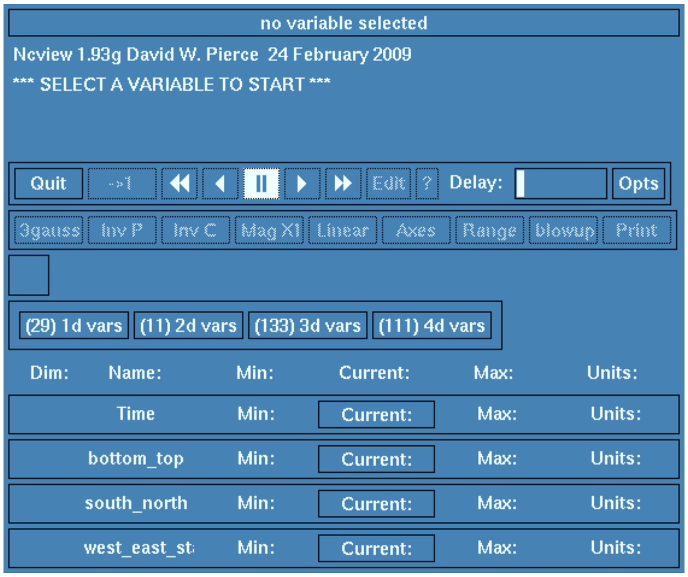
These are the two most used tools to manage netCDF. They will be introduced in the corresponding 'Boot Camp' session.

5.2 ncview: Visualization tool

There are multiple tools to browse netCDF content, however, `ncview` (http://meteora.ucsd.edu/~pierce/ncview_home_page.html) is the most common one due to its versatility and easy use.

```
$ ncview wrfout_d01_2001-11-11_00:00:00 &
```

It opens the file providing an initial panel with a list of the variables. They are usually sorted by shape (see figure below).


no variable selected

Ncview 1.93g David W. Pierce 24 February 2009

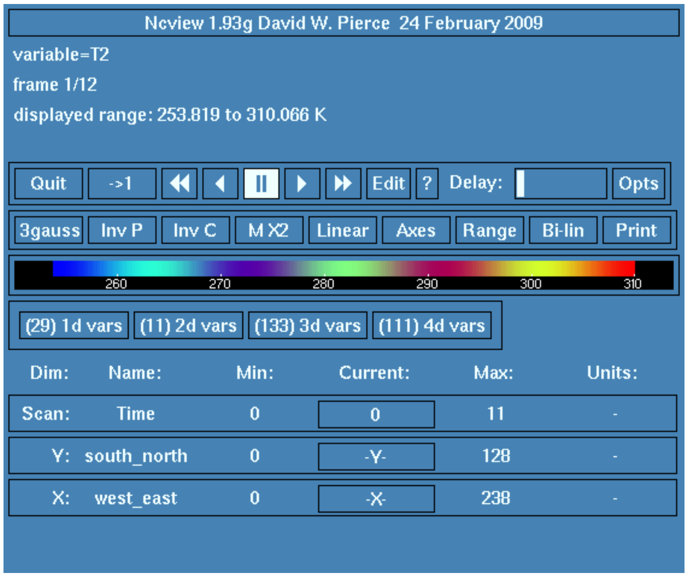
*** SELECT A VARIABLE TO START ***

Dim:	Name:	Min:	Current:	Max:	Units:
	Time	Min:	Current:	Max:	Units:
	bottom_top	Min:	Current:	Max:	Units:
	south_north	Min:	Current:	Max:	Units:
	west_east_st	Min:	Current:	Max:	Units:

When one variable is selected, a new window is opened showing the values along the 2-right most dimensions (if it is possible showing a 2D-map, see figure below). All the additional features are selected from the initial window:

- Navigation along the rest of the possible dimensions
- Time loop, movie player
- `3gauss`: choice of color palettes
- `Inv P`: inverse of y-axis
- `Inc C`: inverse color palette
- `M X2`: Size of the map plot
- `Linear`: Changes of values to logarithmic (High or Low values based)
- `Axes`: Change of axes used for the 2D plot
- `Range`: Selection of range to plot
- `Bi-lin`: Deactivate/Activate 2D map interpolation

- `Print`: print figure on file/printer



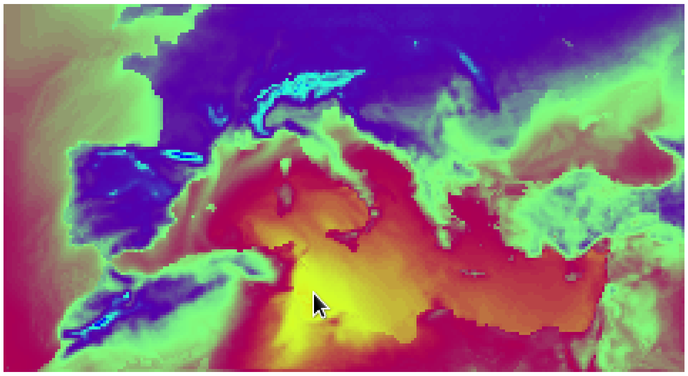
Ncview 1.93g David W. Pierce 24 February 2009

variable=T2

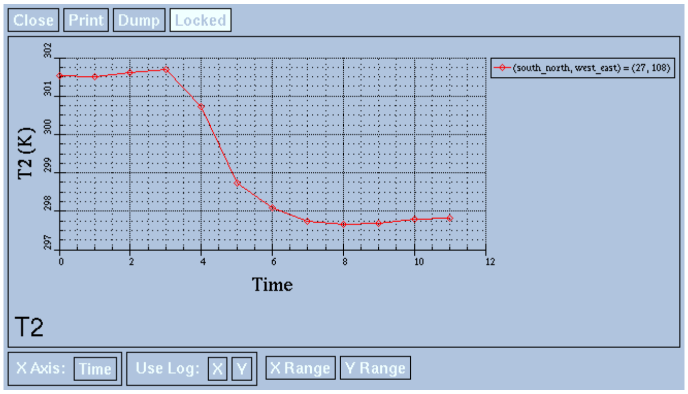
frame 1/12

displayed range: 253.819 to 310.066 K

Dim:	Name:	Min:	Current:	Max:	Units:
Scan:	Time	0	0	11	-
Y:	south_north	0	-Y-	128	-
X:	west_east	0	-X-	238	-



- 'variable evolution': On a left-mouse button click over a grid point on the secondary window, one obtains a time-series of the values at the selected point in a third new window (see figure below).
  - `Print`: print plot on file/printer
  - `Dump`: ASCII file retrieval of grid-point evolution
  - `X Axis`: Axes used to draw the plot can be also changed
  - `Use Log`: use logarithmic scale
  - `X Range`: Selection of x-range to plot
  - `Y Range`: Selection of y-range to plot



6 Useful links

- netCDF: `http://www.unidata.ucar.edu/software/netcdf/`
- ncview: `http://meteora.ucsd.edu/~pierce/ncview_home_page.html`

References

Unidata (2015). Network common data form (netcdf) version 4.3.3.1 [software]. *Boulder, CO: UCAR/Unidata.*