1 Overview

In this lecture we will learn to know various helpful tools and techniques that are used in the analysis and processing of gridded climate data that is - for example - provided by climate models. First, we will learn to know the NetCDF file format in a few short practical demonstrations. Second, some analyses will be performed on available gridded climate data.

Topics that this lecture considers are:

- **Net**work **C**ommon **D**ata **F**orm (NetCDF): general information on the topic and how to use NetCDF files
- **C**limate **D**ata **O**perators (CDO): several examples of data reduction, analysis and transformation of NetCDF files, including operator piping

In contrast to the commonly used graphical user interface (GUI), most of the work will be done by employing the so-called shell and by formulating commands in a command line interface.

1.1 A very limited introduction to tools and methods employed in this exercise

Below some information regarding tools and methods, that are relevant for this exercise, is collected in the form of a very general overview. This information collection is intended as a supplement to the lecture. A few complementary information sources are listed in the subsection "Further Reading". This list is obviously far from being complete. There is a vast amount of freely-accessible information available on-line and the interested student is kindly invited to use search engines towards locating and exploring further documentation.

1.1.1 NetCDF

A very precise definition on the characteristic and purpose of NetCDF is given in the NetCDF FAQ, "What is netCDF?" (for a link, see section "Further Reading"):

> "NetCDF (network Common Data Form) is a set of interfaces for array-oriented data access and a freely distributed collection of data access libraries for C, Fortran, C++, Java, and other languages. The netCDF libraries support a machine-independent format for representing scientific data. Together, the interfaces, libraries, and format support the creation, access, and sharing of scientific data."

NetCDF is a data-container that has been established as a widely used file-standard in science and engineering. It has been developed for storing array-oriented values in compact and interchangeable files. The most important characteristics of NetCDF files can as well be found in the NetCDF FAQ, “What is netCDF?”. An excerpt, that highlights the advantages of NetCDF with respect to this exercise, is listed here:

- Self-Describing. A NetCDF file includes a description of the data that it contains.
- Portable. A NetCDF file can be accessed by computers that apply different formats of storing integers, characters, and floating-point numbers.
- Scalable. A small subset of a large dataset may be accessed efficiently.
- Appendable. Data may be appended to a properly structured NetCDF file without copying the dataset or redefining its structure.

These characteristics make NetCDF a perfect choice for storing any kind of array-oriented data. The data form that we will work with in this exercise is as well array-oriented - therefore, we will use NetCDF as data container for both input and output of computations.

Since NetCDF is a binary format (in contrast to ASCII-text, which can be examined and edited by means of any common text editor), reading, writing and changing of NetCDF files necessitates the use of dedicated software. Fortunately, such software is freely available and can easily be installed on any UNIX system. The following tools are of particular importance:

- ncview (lightweight but mighty explorer for NetCDF files)
- ncbrowse (a Java-based alternative to ncview for Windows)
- Panoply (a flexible Java-based generator of geographic maps of NetCDF data)
- ncdump (tool for “dumping” the contents of a NetCDF file to human-readable ASCII-text; the complete description and structure of the NetCDF file is preserved)
- ncgen (complementary to ncdump, generates a binary NetCDF file from a NetCDF ASCII-dump)
- ncks (mighty toolbox for modification of NetCDF files)
- CDO (mighty toolbox for analysis and modification of NetCDF files, strong focus on climatological data)

1.1.2 CDO

While NetCDF defines a file format (and supporting programs and routines) that can store climatological data in a practical way, the CDO are a collection of operators that allow analysis and modification of gridded binary climatological data. In climate sciences, the CDO have become a very common software tool due to the vast number of available operators and their flexibility:

- more than 400 designated operators are available
- operator-piping allows the application of complex methods on climatological data in a compact manner
- the CDO are command-line programs; in combination with shell-scripts they can be automated, and enhanced complexity of data processing and analysis may be achieved

CDO operator piping in combination with shell-programming can be demonstrated shortly in the following very short bash-script, where two input files are interpolated to a common resolution, the resulting fields are added, the sum is time-averaged, and the result being stored in a new file. Shell-programming allows for the diagnostic output of additional information to the screen, here the spatial-average of the field resulting from the CDO-operator-chain.

```bash
#!/bin/bash

#select level 6, interpolate to 1x1 degree, and convert from deg. C to Kelvin
cdo timmean -addc,273.15 -remapcon,r360x180 -sellevel,6 input.nc output.nc
#note: the rightmost command is executed first!

#compute global mean
spat_avg=$(cdo output -fldmean output.nc)

#print result to screen
echo
echo
echo "spatial average of global ocean surface temperature is ${spat_avg} K."

#clean up
rm output.nc
```

1.1.3 The Bash, a popular UNIX-Shell

Due to their scripting ability Shells are a very powerful tool of UNIX-systems. This and the fact, that a large number of (free) software tools and programs are available on UNIX systems, makes UNIX-computers the choice for tackling complex scientific problems that involve the analysis and processing of large amounts of data. Particularly the bash-shell is a famous tool for scientists and commonly used during their daily work routine. Yet, giving a comprehensive overview on the use and ability of the bash clearly exceeds the scope of this course. Please refer to a bash-scripting guide referenced below if you would like to gain further insights into the topic and gain abilities in shell-scripting - and note: "... the only way to really learn scripting is to write scripts" (Advanced Bash-Scripting Guide).

1.2 Further Reading

The following resources provide an introduction to tools and methods considered in this exercise:

- NetCDF: The NetCDF FAQ
  (`www.unidata.ucar.edu/software/netcdf/docs/faq.html`)
  The NetCDF Fact Sheet
  (`http://www.unidata.ucar.edu/publications/factsheets/current/netcdf_factsheet.pdf`)
- CDO: The CDO User's Guide
  (`https://code.zmaw.de/projects/cdo/embedded/1.6.3/cdo.html`)
  The CDO Reference Card
  (`http://www.iac.ethz.ch/edu/courses/master/modules/radiation_and_climate_change/download/cdo_refcard.pdf`)
- Bash: Bash Guide for Beginners
  (`http://www.tldp.org/LDP/Bash-Beginners-Guide/html/`)
  Advanced Bash-Scripting Guide
  (`http://www.tldp.org/LDP/abs/html/`)

Dynamics 2 — Exercise 2, Summer semester 2019
Lecturer: Prof. Dr. G. Lohmann — Dr. Christian Stepanek
Due date: 29.04.2019 (discussion: 27.05.2019, AWI) — 08.04.2019

2 Exercise

2.1 A (subjective) remark: Windows vs. UNIX

For those of you who consider to work in a scientific field with a strong focus on programming or the analysis of large amounts of data: Consider to gain experience with a UNIX-environment, e.g. Ubuntu. In many scientific fields, definitely in climate sciences, UNIX-like operating-systems are the computing environments of choice. Many tools that are necessary for efficiently working in such scientific fields are not natively available on Windows-systems; even if ports are available, they may still suffer from limitations or incompatibilities. It definitely makes sense for you to setup an own Linux partition on your computer (after backuping your data, do not forget that), and to learn how to use and program the available software tools. For every free UNIX-environment (e.g. Ubuntu) and the included software tools, a vast amount of detailed - and free - documentation can be found on-line, that addresses both novices and experts.

2.2 CDO tasks

In the following tasks we will process NetCDF files using the CDO. In order to fulfil these tasks, you may use the documentation (https://code.zmaw.de/projects/cdo/embedded/index.html). First, please download the data files via the links that have been distributed.

File INIOM_PD_echam5_main_mm_3901-4000_climatological_mean.nc represents the climatological average over 100 years of a recent climate as simulated with a coupled atmosphere-ocean general circulation model. File LGM-W_echam5_6100-6200_climatological_mean.nc represents similar data, but for the Last Glacial Maximum (i.e. the coldest part of the last ice age, about 21,000 years ago). During the Last Glacial Maximum the global average temperatures were lower than today, mostly due to a combination of different insolation conditions and greenhouse gas concentrations. Due to differences in albedo and in the circulation of the atmosphere with respect to today, some regions were extremely cold.

2.2.1 Identifying the contents and properties of the contained data

Consider the file INIOM_PD_echam5_main_mm_3901-4000_climatological_mean.nc:

- Question 1: What is the physical unit of the field **srad0**, code 178? (1/2 Point)
- Question 2: What is the physical quantity stored in the field **precip**, code 4? (1/2 Point)
- Question 3: What is the number of levels for field **q**, code 133? Provide them. (1/2 Point)
- Question 4: On how many levels is field **aps**, code 134, given? What reason is there for the number of levels? Hint: consider the physical meaning of this field. (1 Point)
- Question 5: Which months are contained in the file, which years? (1/2 Point)
- Question 6: What temporal resolution does the file have? (1/2 Point)
- Question 7: What is the distance between the centers of two grid cells along longitudes and latitudes? (1/2 Point)
- Question 8: How many longitudes are available per time step / variable? Note: A test for one 2D-/3D-variable and time step is sufficient. (1/2 Point)
- Question 9: What is the southernmost latitude of grid cell centers in the file? What is the latitude of the grid cell's center that is closest to the equator in the Northern Hemisphere? (1/2 Point)
- Question 10: How many climate variables are present in provided data set? (1/2 Point)

Exercise continues on the next page!!!

2.2.2 Climatological analysis of two climate states

Now we will analyse two different modelled climate states that are representative for the Pre-Industrial (PI, file INIOM_PD_echam5_main_mm_3901-4000_climatological_mean.nc) and the Last Glacial Maximum (LGM, file LGM-W_echam5_6100-6200_climatological_mean.nc):

- Question 11: What is the global and annual average total precipitation (variable **precip**, code 4) for PI? (1/2 Point)
- Question 12: Find the physical unit of field **precip**, code 4. Based on this information, what is the result derived in Question 10 in units of mm/year? (1/2 Point)
- Question 13: Extract the annual cycle of surface temperature (variable **tsurf**, code 169) for climate state INIOM_PD_echam5_main_mm_3901-4000_climatological_mean.nc at the location of Bremen (geographical coordinates: 53.0793°N, 8.8017°E) using nearest neighbour interpolation (CDO operator remapnn). Which values do you get in units of °C, using an appropriate conversion from the physical unit of the model output? (1 Point)
- Question 14: What is the minimum, maximum, and mean surface temperature (variable **tsurf**, code 169) over a year in the spatial average over a rectangular area of $4° \times 4°$ around Bremen (geographical coordinates: 53.0793°N, 8.8017°E) for the PI climate state provided in file INIOM_PD_echam5_main_mm_3901-4000_climatological_mean.nc? (1 Point)
- Question 15: Compute the spatial standard deviation of the annual mean surface temperature (variable **tsurf**, code 169) in a rectangular area of $5° \times 5°$ around Bremen (geographical coordinates: 53.0793°N, 8.8017°E). Perform this computation for both the PI climate state, given by file INIOM_PD_echam5_main_mm_3901-4000_climatological_mean.nc, and the LGM climate state, given by file LGM-W_echam5_6100-6200_climatological_mean.nc. Which climate state has the higher spatial variability? (1/2 Point)
- Question 16: How large is the meridional gradient of annual and zonal average surface temperature (i.e. the range of values across latitudes of variable **tsurf**, code 169) in the Northern Hemisphere for both climate states PI and LGM? For which climate state is it larger? (1 Point)
- Question 17 (bonus question, you will earn up to one additional point): If you plot (e.g. with ncview or panoply) and compare the annual mean surface temperature (variable **tsurf**, code 169) for both climate states: Which geographical region appears to have the largest influence on the difference between meridional temperature gradients in the Northern Hemisphere for both climate states? (up to 1 Point)

Notes on submission of **this (and only this)** exercise's solutions: Working in study groups is encouraged. You may hand in one solution for your group (2-3 students). Students' names must be clearly stated on the solution. The answers to the questions can be send until the due date to Christian Stepanek (Christian.Stepanek@awi.de) or handed in on paper during the tutorium on the 29$^{\text{th}}$ of April.