CDO installation and use

Climate Modelling course
Chair of Meteorology

Dánnell Quesada — 13/10/21
dquesadacr

TECHNISCHE UNIVERSITÄT DRESDEN

Europäische Union

Europa fördert Sachsen.
ESF
Europäischer Sozialfonds

Freistaat Sachsen

Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des vom Sächsischen Landtag beschlossenen Haushaltes.

1. From the CDO documentation

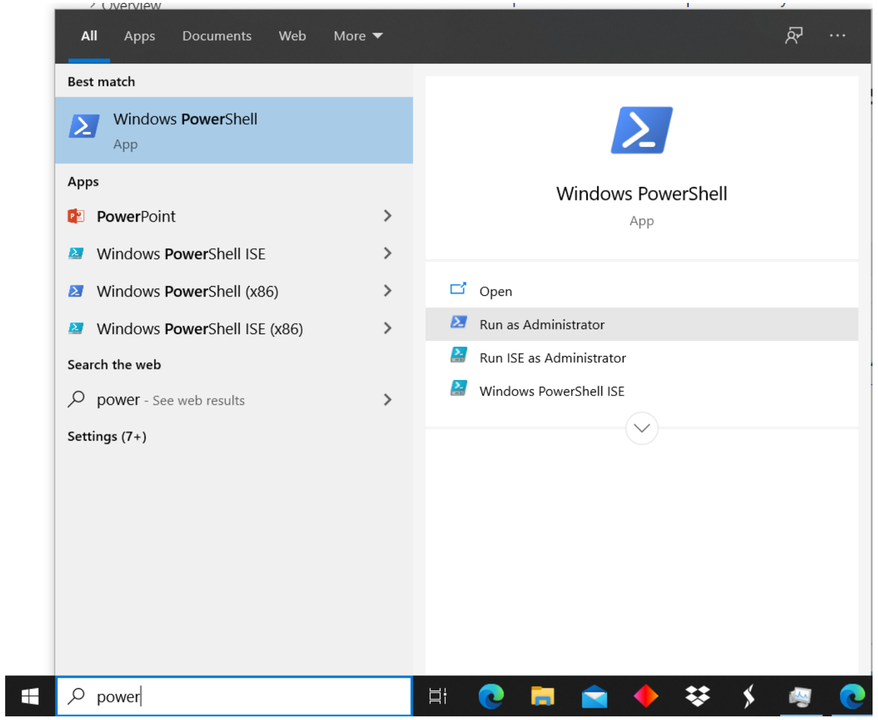
- ***CDO*** stands for Climate Data Operators
- CDO is meant for Posix compatible operating systems (like Linux)
- Nevertheless, here you can find the official recommendations to run *CDO* on *Windows*
  - The *Windows Subsystem for Linux* version 2 (wsl2) installs a Linux kernel to create a proper compatibility layer for Windows
  - The following steps correspond to option #2 on the documentation and works for *Windows 10*
  - For *Windows 8*, *wsl1* should be installed (not yet tested for our purpose)
  - *Windows S* is incompatible
- Should also work in MacOS

2. Install *wsl*

- ***Option #1:*** *Windows 10* version 2004 and higher, or *Windows 11* check here
- ***Option #2:*** older versions, check here
- Note that these steps are for *wsl* version ***2***
- Some visual aid is added

2.1. Option #1:

1. Open PowerShell as Administrator:



2. Run (type or copy — paste [right-click on *PowerShell*] and then press enter):

```
wsl --install
```

> This should already install *Ubuntu* version 20.04 (?), for other versions/distros check here

> Since these changes are quite new, these steps have not been tested

2.2. Option #2

1. Open PowerShell as Administrator (as shown before)
2. Run:

```
dism.exe /online /enable-feature /featurename:
    Microsoft-Windows-Subsystem-Linux /all /norestart
```

⚠ This should be typed all together (one line)

2.3. Update to WSL 2 (check Windows version)

- Windows 10 **version 1903+**, **build 18362+**. For most computers (x64)
- Check with **Win key + R**, type **winver** → **OK**
- Update if needed:

If you are running Windows 10 version 1903 or 1909, open "Settings" from your Windows menu, navigate to "Update & Security" and select "Check for Updates". Your Build number must be 18362.1049+ or 18363.1049+, with the minor build # over .1049.

2.4. Enable Virtual Machine feature

1. From the PowerShell as Administrator run:

```
dism.exe /online /enable-feature /featurename:VirtualMachinePlatform /all /norestart
```

2. Restart to complete WSL install and updates

2.5. Download the Linux kernel update package

- Download WSL2 Linux kernel update package for x64 machines
- Install as administrator (elevated permissions)

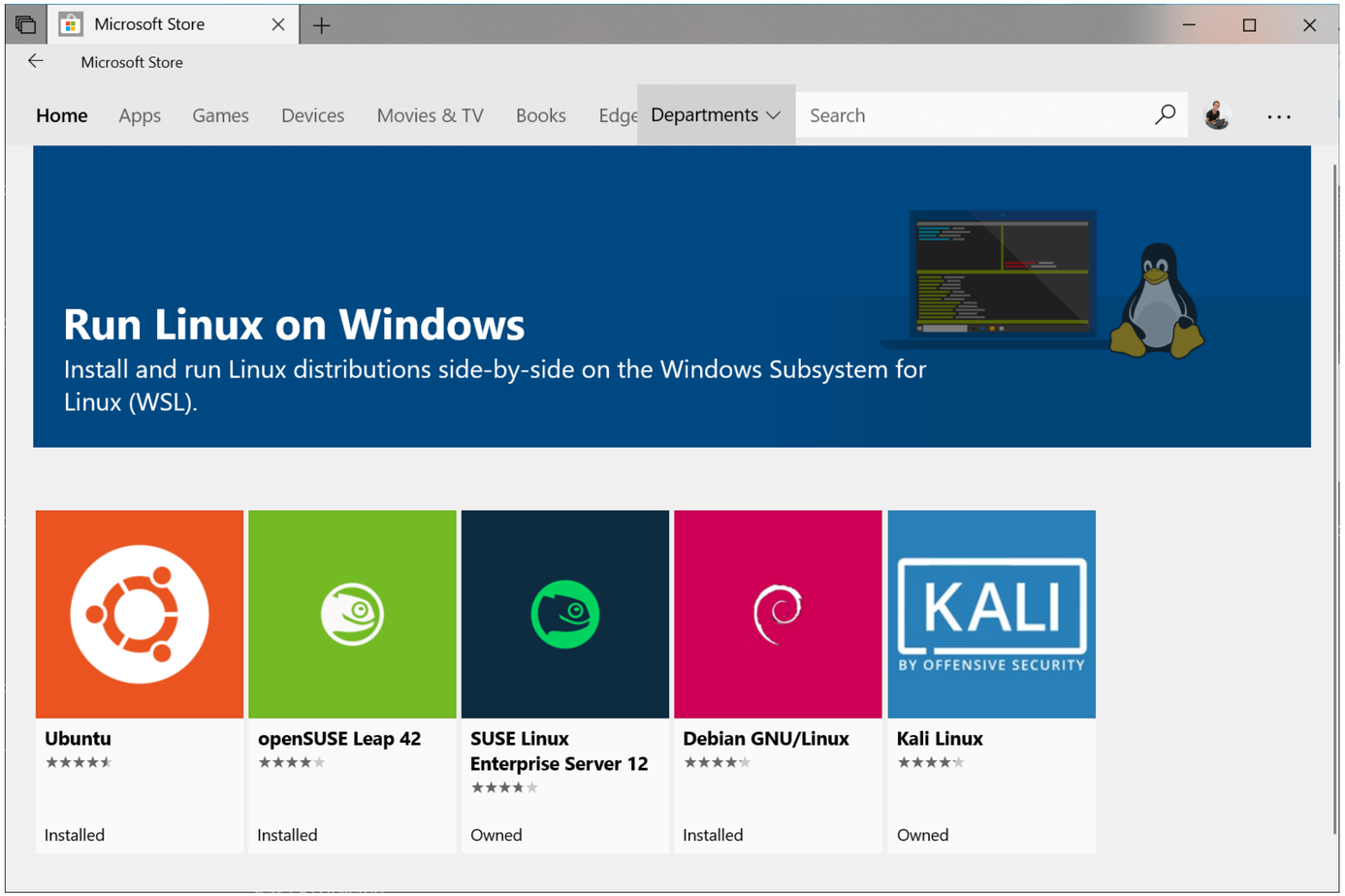
2.6. Set WSL 2 as your default version

- From the PowerShell as Administrator run:

```
wsl --set-default-version 2
```

2.7. Install Linux distro

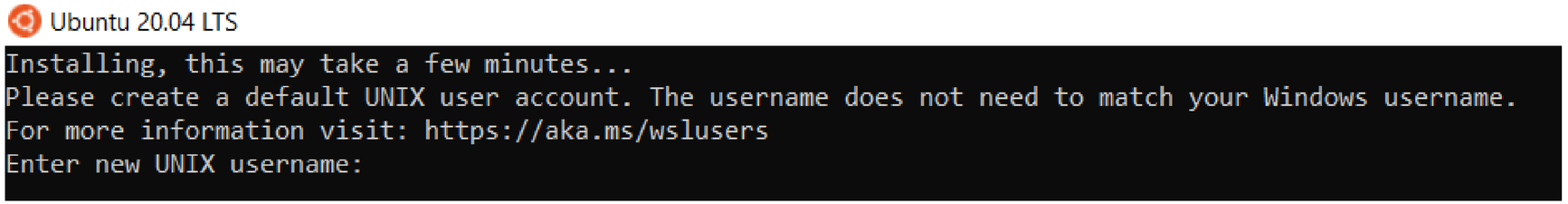
1. Open the Microsoft Store and select a Linux distribution (distro)



2. Ubuntu 18.04 and 20.04 were tested for CDO. After this, *Ubuntu* should be a searchable program in Windows

3. Setting up Ubuntu

1. Open the Ubuntu app
2. On the first lunch, it will show this:



3. Set credentials, user and password (twice)
4. Output should look like this:

dqc@[redacted]: ~

```
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: dqc
New password:
Retype new password:
passwd: password updated successfully
Installation successful!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 4.19.128-microsoft-standard x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sat Oct 17 17:22:44 CEST 2020

  System load:  0.1               Processes:             8
  Usage of /:   0.4% of 250.98GB  Users logged in:       0
  Memory usage: 0%                IPv4 address for eth0: 192.168.115.158
  Swap usage:   0%

1 update can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update


This message is shown once once a day. To disable it please create the
/home/dqc/.hushlogin file.
dqc@[redacted]:~$
```

3.2. Update and upgrade

- Run from the Linux terminal:

```
sudo apt-get update
sudo apt-get upgrade -y
```

- To run `sudo` you will be asked for the password set in the previous step
- This might take a while, you should see something like this:

```
Fetched 17.4 MB in 4s (4629 kB/s)
Reading package lists... Done
dqc@                    :~$ sudo apt-get upgrade -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages have been kept back:
  base-files sosreport ubuntu-server
The following packages will be upgraded:
  alsa-ucm-conf apport bcache-tools bind9-dnsutils bind9-host bind9-libs bolt bsdutils busybox-initramfs busybox-st
  cryptsetup-run curl fdisk finalrd gcc-10-base gir1.2-packagekitglib-1.0 initramfs-tools initramfs-tools-bin initr
  libc-bin libc6 libcryptsetup12 libcurl3-gnutls libcurl4 libdns-export1109 libfdisk1 libgcc-s1 libgl1 libglvnd0 li
  libpam-modules-bin libpam-runtime libpam0g libproxy1v5 libpulse0 libpulsedsp libpython3.8 libpython3.8-minimal li
  locales mdadm mount open-vm-tools packagekit packagekit-tools pulseaudio-utils python3-apport python3-commandnotf
  python3-software-properties python3-urllib3 python3.8 python3.8-minimal rsyslog secureboot-db show-motd snapd sof
  ubuntu-wsl unattended-upgrades update-motd util-linux uuid-runtime xz-utils zlib1g
97 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
Need to get 55.9 MB of archives.
After this operation, 3645 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 bsdutils amd64 1:2.34-0.1ubuntu9.1 [63.1 kB]
Get:2 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 gcc-10-base amd64 10.2.0-5ubuntu1~20.04 [19.7 kB]
```

4. Install and check CDO

- To install, simply run from the Ubuntu terminal:

```
sudo apt-get install cdo -y
```

- The last step works also for other *Ubuntu* based distros
- It is also available for other distros, e.g. *Arch* based

4.1. CDO version

- Check that *CDO* is working with:

```
cdo --version
```

```
dqc@[redacted]:/mnt/e/HSE/Thesis/Data/Taurus$ cdo --version
Climate Data Operators version 1.9.3 (http://mpimet.mpg.de/cdo)
CXX Compiler: g++ -g -O2 -fdebug-prefix-map=/build/cdo-g3Qjnd/cdo-1.9.3+dfsg.1=. -fstack-protector-stron
g -Wformat -Werror=format-security -fopenmp
CXX version : g++ (Ubuntu 7.3.0-1ubuntu1) 7.3.0
C Compiler: gcc -g -O2 -fdebug-prefix-map=/build/cdo-g3Qjnd/cdo-1.9.3+dfsg.1=. -fstack-protector-strong
-Wformat -Werror=format-security -Wall -pedantic -fPIC -fopenmp
C version : gcc (Ubuntu 7.3.0-1ubuntu1) 7.3.0
F77 Compiler: f77 -g -O2 -fdebug-prefix-map=/build/cdo-g3Qjnd/cdo-1.9.3+dfsg.1=. -fstack-protector-stron
g
F77 version : unknown
Features: 12GB C++14 Fortran DATA PTHREADS OpenMP45 HDF5 NC4/HDF5/threadsafe OPeNDAP SZ UDUNITS2 PROJ.4
MAGICS CURL FFTW3 SSE2
Libraries: HDF5/1.10.0 proj/4.93 curl/7.58.0
Filetypes: srv ext ieg grb1 grb2 nc1 nc2 nc4 nc4c nc5
     CDI library version : 1.9.3
GRIB_API library version : 2.6.0
  NetCDF library version : 4.6.0 of Feb  9 2018 19:21:24 $
    HDF5 library version : library undefined
    EXSE library version : 1.4.0
    FILE library version : 1.8.3
```

5. CDO installation in MacOS

- The following solutions have not been tested by us but they reportedly work for MacOS. This information was taken from the CDO wiki

  1. With MacPorts simply run from the terminal:

```
port install cdo +grib_api +magicspp +szip
```

  2. With homebrew:

```
brew tap moffat/sciencebits
brew install cdo
```

- If needed, to install `homebrew`:

  1. To install with `ruby`, run this:

```
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

  2. If the last command doesn't work, try:

```
curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install > \
brew_install_script
ruby brew_install_script
rehash
```

6. Basic shell commands

- To work from the terminal, some basic commands are needed:

1. `mkdir` → creates directory. The following creates the folder `data` in the current directory

```
mkdir data
```

2. `pwd` → print working directory. To check the current working directory
3. `cd` → change directory. To access the created folder:

```
cd data
```

4. `ls` → list contents of current directory
5. `cp` → copy file. This copies `file1` to `file2`

```
cp file1 file2
```

6. `mv` → move files or directories (also to rename). The following moves `file1` into the created folder while renaming it to `file2`

```
mv file1 data/file2
```

7. `cat` → prints the content of a file (text)

- More info here and here

7. Accessing Windows files from Linux

1. Download the `test.nc` file (here, under `nc_files`)
2. If the file was downloaded to `C:\User\Downloads`, to access the downloads folder do:

```
cd /mnt/c/User/Downloads
```

3. Try some *CDO* commands as:

```
cdo sinfo test.nc # short description of the file
cdo griddes test.nc # grid description, output on next slide
cdo graph,device="png" -selgridcell,1 test.nc plot # quick plot
```

4. The *CDO* operators are here (explained) or type:

```
cdo --operators
```

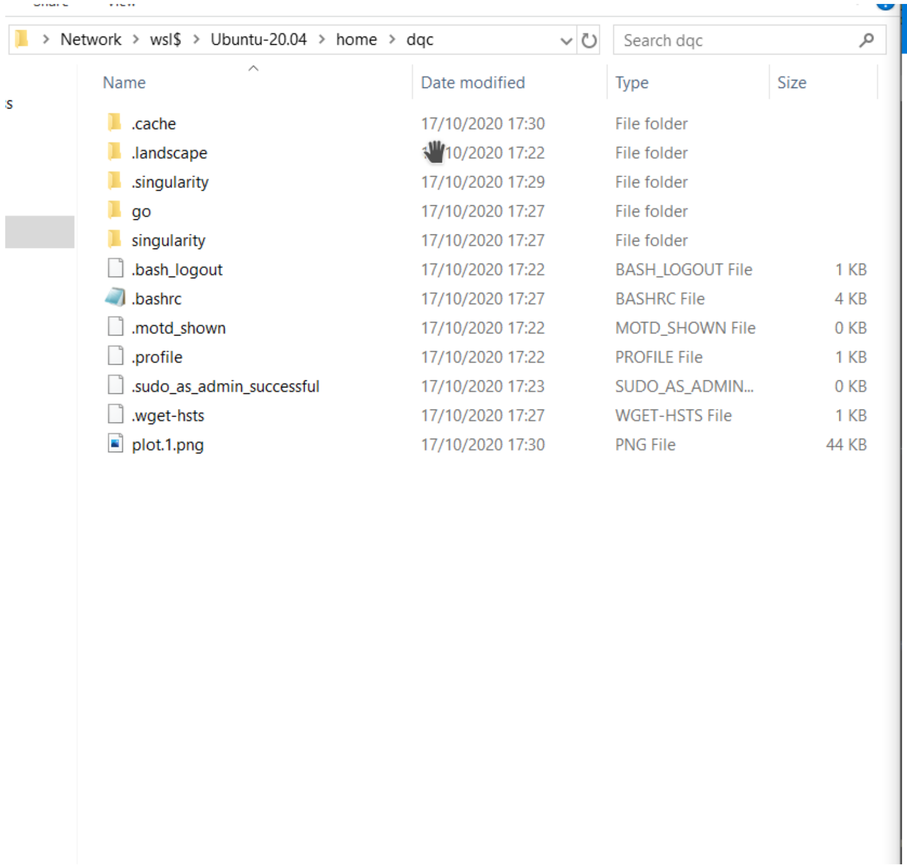
5. If you encounter issues opening or accessing the file do:

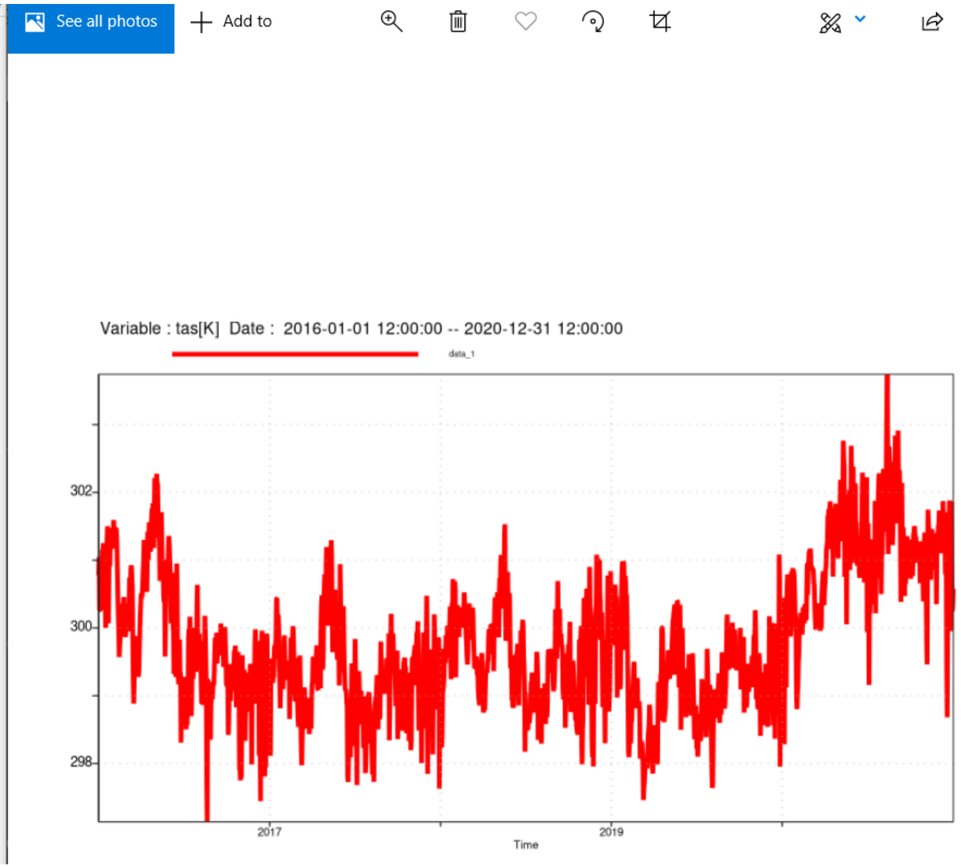
```
sudo chown $USER test.nc
```

```
dqc@[redacted]:/mnt/e/HSE/Thesis/Data/Taurus$ cdo griddes test1.nc4
#
# gridID 1
#
gridtype  = lonlat
gridsize  = 196
xsize     = 14
ysize     = 14
xname     = lon
xlongname = "longitude"
xunits    = "degrees_east"
yname     = lat
ylongname = "latitude"
yunits    = "degrees_north"
xfirst    = 270
xinc      = 0.75
yfirst    = 15
yinc      = -0.75
scanningMode = 64
cdo griddes: Processed 1 variable [0.01s 57MB]
```

8. Accessing Linux files from Windows

1. Type `\\wsl$` on the file explorer path (on the top), enter
2. Then click *Ubuntu* → *home* → *user*
3. Should look something like this (note the plot created before):





9. Exercise

1. Exploration of *NetCDF* files
2. Visualization of the files
    - A set of example files are in this repo, `nc_files` folder
3. Manipulation of files
4. Selection of desired coordinates
5. Export data as text for further manipulation in other software
6. Download CORDEX files

10. Exploration

- To have an idea of the contents of the file we can run some commands from the terminal that will print some information to the screen
- Commands to try:
  - `cdo sinfo file` → short description of the contents
  - `cdo griddes file` → description of the grid
  - `ncdump file` → metadata description, not CDO but useful
    - Try with `-h` (header) and `-c` (coordinates)
- See more options with `cdo --operators`
- Check help of a operator with e.g. `cdo --help sellonlatbox` or here

11. Visualization

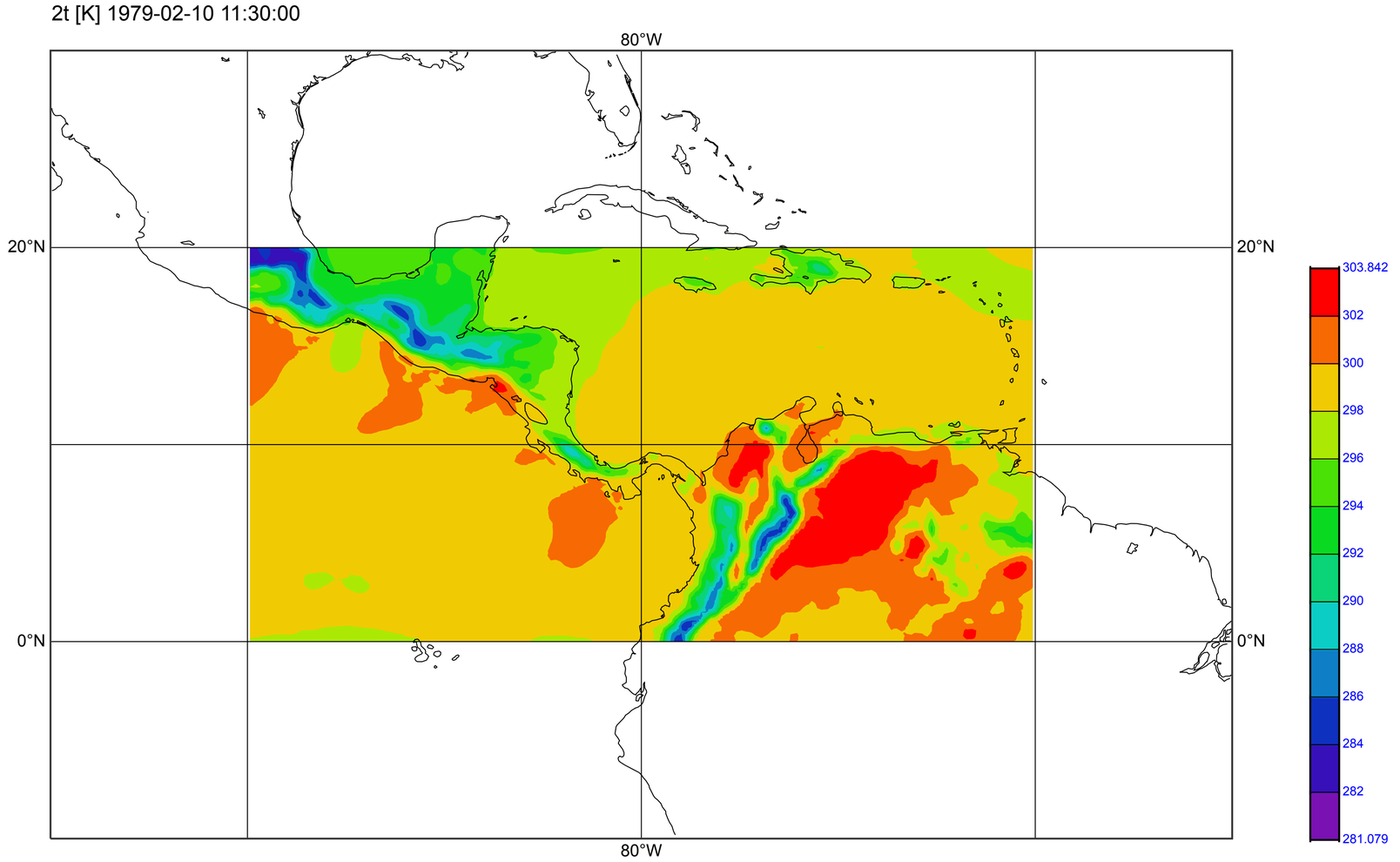
- It is possible to create high-quality plots from CDO by using the Magics software
  - Already included in the installation process explained before
- The documentation of the use of `CDO+Magics` is here
- There are 3 types of plots with this combination
  - 2D Lon-Lat plots: `shaded`, `grfill` and `contour`
  - 2D `vector` plots: plots vectors (e.g. wind fields) on 2D maps
  - Line `graph` plots: generates time-series plots
- Due to the *rotated pole* rotation embedded in ***CORDEX*** projections, 2D plotting options within CDO are not available
  - They only accept *rectilinear* grids and CORDEX is *curvilinear*
  - Reprojection needed to plot them
    - It can be done with CDO
    - Or check this repo for an *R* solution

11.1. 2D Lon-Lat

- From the terminal, access the folder where the `2m_temperature.nc` file was downloaded
  - This file is a small example of a *rectangular* grid taken from ERA5
- Run the following command

```
cdo -shaded,device=pdf,lon_max=-50,lon_min=-110,lat_max=30,lat_min=-10,\
interval=2,colour_triad=cw,colour_min=violet,colour_max=red \
-seltimestep,42 2m_temperature.nc shaded
```

- Resulting file `shaded_2t.pdf` should be created, shown on next slide
- Note the arguments passed to CDO
  - Two functions, `shaded` and `seltimestep`, are passed with corresponding parameters
  - The `-` allows concatenation of several functions
  - Here we selected only the 42nd step.
  - The `\` are only to break the line, this command could be written in one line
- Try adding `,step_freq=10` after `red` (no space) and removing `-seltimestep,42`
  - Check resulting file
- Try the `contour` and `grfill` commands

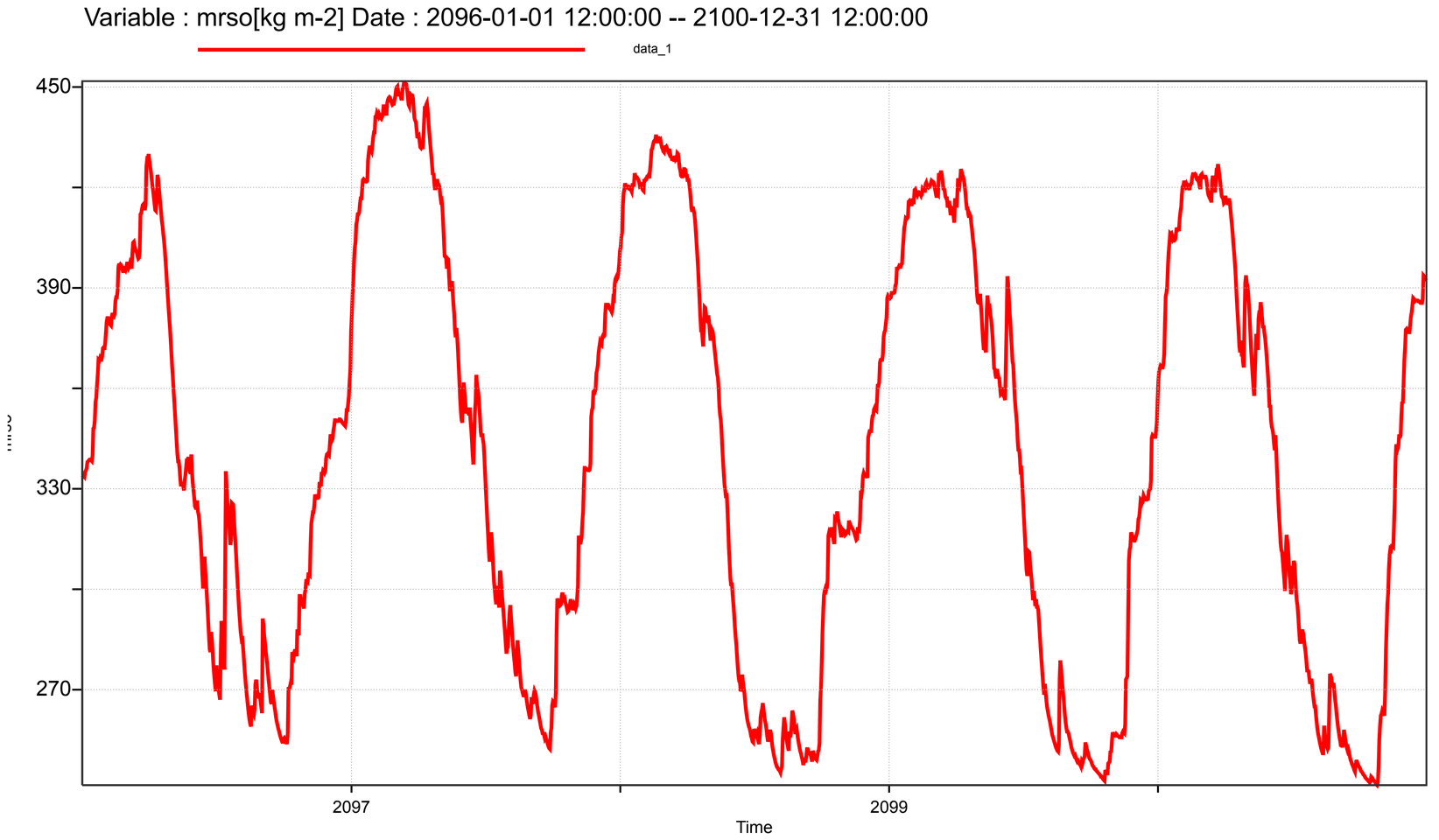

2t [K] 1979-02-10 11:30:00
80°W
20°N
0°N
80°W
303.842
302
300
298
296
294
292
290
288
286
284
282
281.079

11.3. Line graph plots

- These type of plots are possible with CORDEX files (no need for rectangular grids)
- Let’s plot a time series for the city of Dresden
- Run this command from the terminal:

```
cdo -graph,device=pdf,linewidth=4 -remapnn,lon=13.73/lat=51.05 \
mrso_EUR-44_ICHEC-EC-EARTH_rcp85_r12i1p1_SMHI-RCA4_v1_day_20960101-21001231.nc graph
```

- The file `graph.pdf` should have been created in the working directory
- Note the `remapnn` function → remap to the nearest neighbour
  - Returns the time series related to the pixel in which the given coordinate pair is
- Plot in next slide


Variable : mrso[kg m-2] Date : 2096-01-01 12:00:00 -- 2100-12-31 12:00:00
data_1
450
390
330
270
2097
2099
Time

12. Manipulation of files

12.1. Merge

- Merge all the provided historical run files into one and save into new file

```
cdo -z zip -mergetime \
mrso_EUR-44_ICHEC-EC-EARTH_historical_r12i1p1_SMHI-RCA4_v1_day_19* \
historical_1961-1990.nc
```

- Notes
  - `-z zip` to compress the file
  - `-mergetime` is the main function
  - `19*` wildcard to give as input all the historical files instead of typing each
  - `historical_1961-1990.nc` is the name of the output file
- Explore resulting file

12.2. Cropping

- To select only a pixel where our location of interest is, again, for Dresden:

```
cdo -remapnn,lon=13.73/lat=51.05 \
mrso_EUR-44_ICHEC-EC-EARTH_rcp85_r12i1p1_SMHI-RCA4_v1_day_20960101-21001231.nc \
mrso_dresden.nc
```

- To select the pixels inside a boundary box, Saxony (approx.) for example:

```
cdo -sellonlatbox,11.5,15.5,49.7,52.2 \
mrso_EUR-44_ICHEC-EC-EARTH_rcp85_r12i1p1_SMHI-RCA4_v1_day_20960101-21001231.nc \
mrso_saxony.nc
```

- Check the files with `griddes` and `sinfo`

12.3. Exporting to text

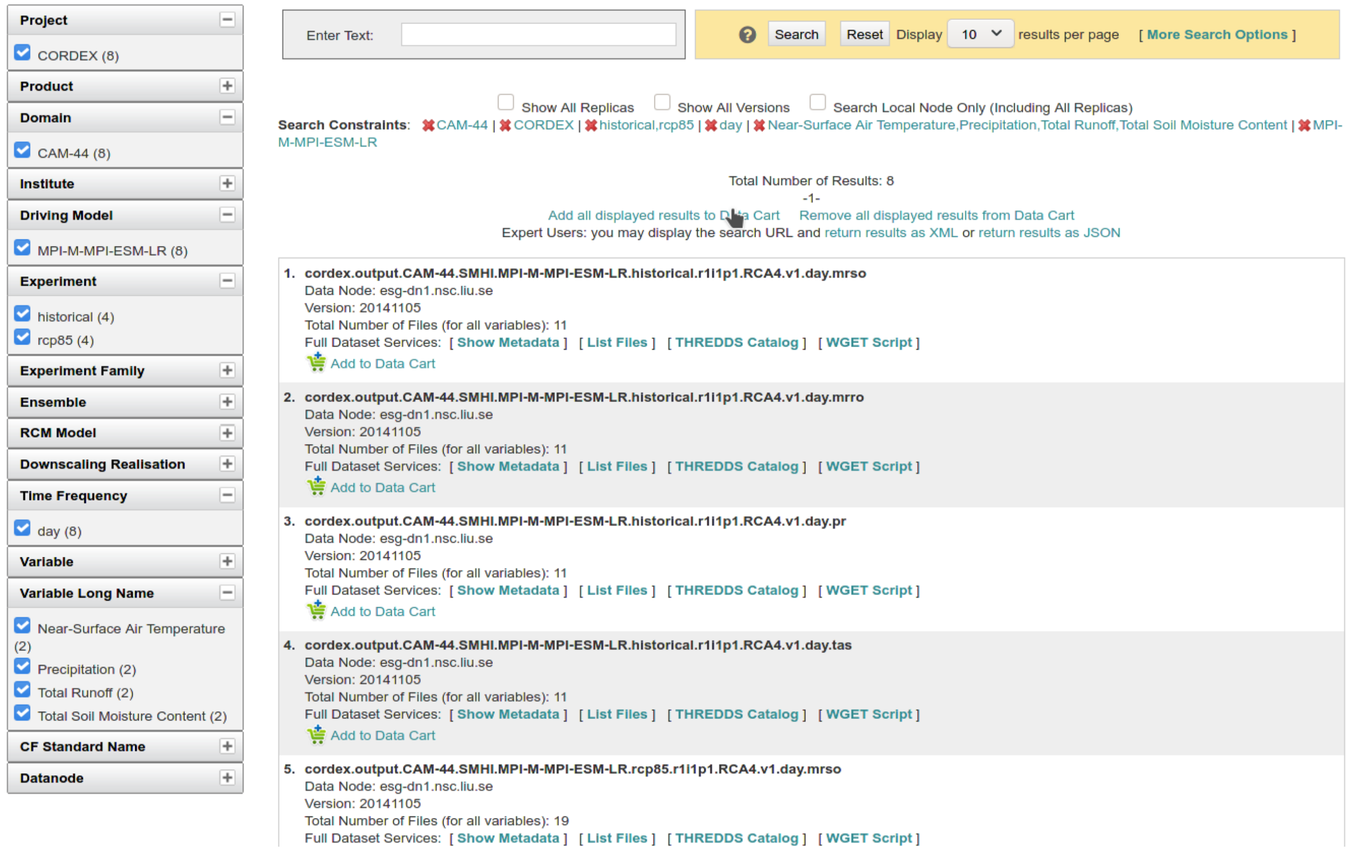
- For further analysis and plots, one might want to export some information in text format
- For example, for the previous file for Dresden:

```
cdo -outputkey,year,month,day,lon,lat,value mrso_dresden.nc > table.txt
```

- The file `table.txt` will be created and can be parsed with e.g. R, Python or Excel to do:
  - Further statistical analysis
  - Boxplots comparing historical vs future conditions
  - Day of the year plots, etc.

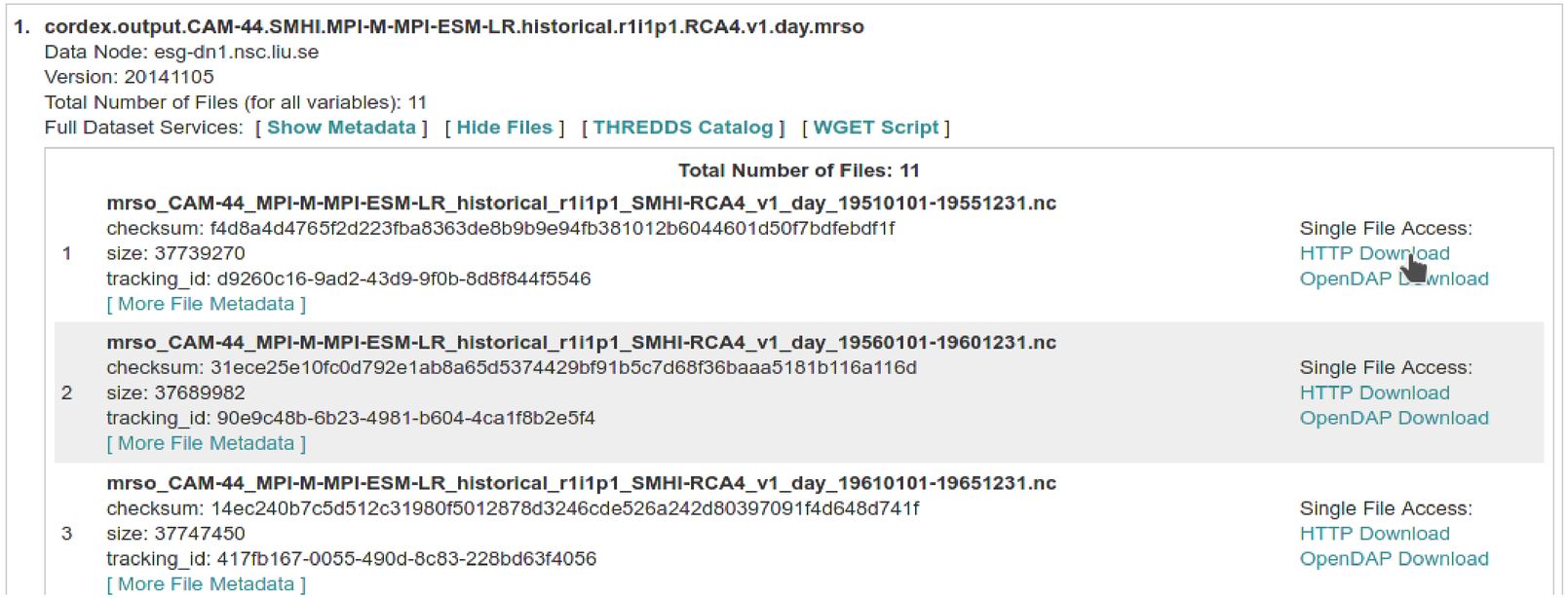
13. Downloading CORDEX files

1. Create account for ESGF node
2. Login with your `OpenID` after receiving confirmation email
3. Register to the CORDEX group
4. Explore and select desired data (checked boxes, note for later, where the pointer is)



13.1. File by file

1. To download file by file through your browser click on `list files`
2. Then click on `HTTP Download`, this will open a prompt



1. **cordex.output.CAM-44.SMHI.MPI-M-MPI-ESM-LR.historical.r1i1p1.RCA4.v1.day.mrso**
Data Node: esg-dn1.nsc.liu.se
Version: 20141105
Total Number of Files (for all variables): 11
Full Dataset Services: [Show Metadata] [Hide Files] [THREDDS Catalog] [WGET Script]

Total Number of Files: 11

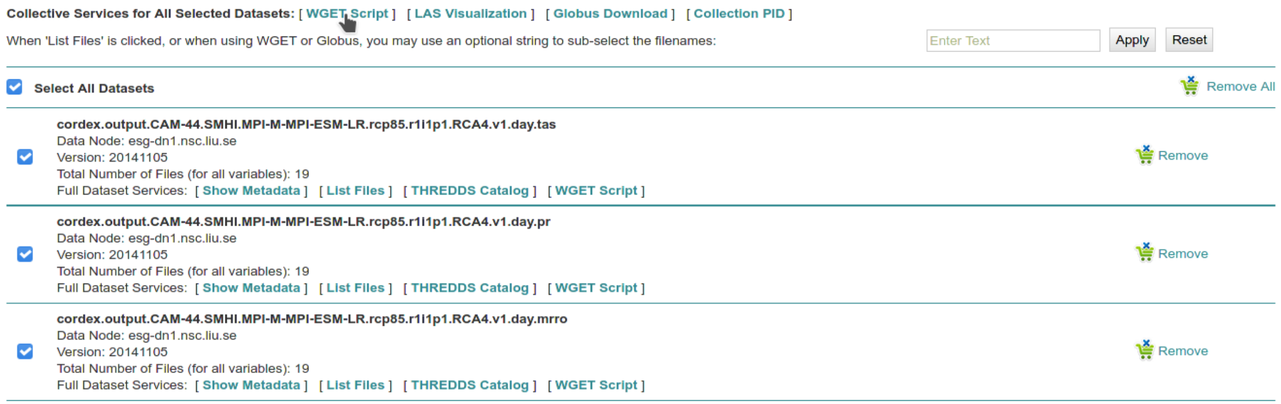
1	**mrso_CAM-44_MPI-M-MPI-ESM-LR_historical_r1i1p1_SMHI-RCA4_v1_day_19510101-19551231.nc** checksum: f4d8a4d4765f2d223fba8363de8b9b9e94fb381012b6044601d50f7bdfebdf1f size: 37739270 tracking_id: d9260c16-9ad2-43d9-9f0b-8d8f844f5546 [More File Metadata]	Single File Access: HTTP Download OpenDAP Download
2	**mrso_CAM-44_MPI-M-MPI-ESM-LR_historical_r1i1p1_SMHI-RCA4_v1_day_19560101-19601231.nc** checksum: 31ece25e10fc0d792e1ab8a65d5374429bf91b5c7d68f36baaa5181b116a116d size: 37689982 tracking_id: 90e9c48b-6b23-4981-b604-4ca1f8b2e5f4 [More File Metadata]	Single File Access: HTTP Download OpenDAP Download
3	**mrso_CAM-44_MPI-M-MPI-ESM-LR_historical_r1i1p1_SMHI-RCA4_v1_day_19610101-19651231.nc** checksum: 14ec240b7c5d512c31980f5012878d3246cde526a242d80397091f4d648d741f size: 37747450 tracking_id: 417fb167-0055-490d-8c83-228bd63f4056 [More File Metadata]	Single File Access: HTTP Download OpenDAP Download

3. Save file and repeat
4. Note that CORDEX files are usually on a 5 years basis (daily)

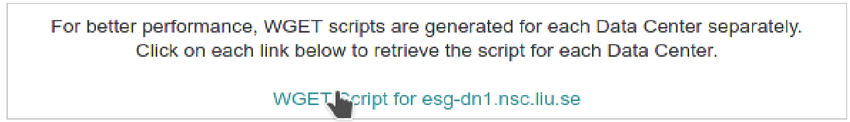
13.2. The automated way

1. Click on `Add all displayed results to Data Cart` (where the pointer was in the penultimate image)
2. Go to your `data cart`

3. Select datasets to download. When ready, click on `WGET Script`



4. Depending on your dataset, there might be several `WGET Scripts`. Click and save them.



13.3. Run script

14. Homework

1. Download CORDEX files for your home region
    - If not available, choose Lisbon or Delft, check here for domains
    - Historical period: 1961-1990
    - Projected: RCP 8.5 2070-2100
    - Variables: temperature, precipitation and either run off or soil moisture content
2. Merge and crop files for your country and city
3. Perform statistical analysis for the 3 variables in another software
    - Box plots of historical vs projected for your city
4. Plot time series for your home city
    - Both periods and all variables
5. Extra points for:
    - Probability density function plot for both periods
    - Average *Day of the year* (1-365) comparison plot
    - Monthly comparison boxplots
    - 2D map of average difference between the periods in your home country
        - Tips: `sellatlonbox`, this repo, or reprojection with CDO (`remapbil`)

15. Questions — ideas for the extra points?

You can contact me!