Post-processing and visualizing large climate model data: radionuclide dispersion modelling and High Performance Computing (HPC)

CaSToRC EuroCC Seminar Talk

Marco Miani - Compuational Support Specialist m.miani@cyi.ac.cy

October 26th, 2021

EPD / CARE-C - The Cyprus Institute

The content of this talk

1. Introduction

2. Large climate data: structure, retrieval and processing

3. Visualizing climate data

4. High Performance Computing (HPC): the modelling of radionuclide dispersion using `FLEXPART`

Introduction

Computational Support Specialist – Climate and Atmosphere Research Center (CARE-C)

Research Topics

- Atmospheric and climate/earth modelling
- Emission inventories, modelling and analysis
- Dynamical down-scaling of climate change and weather extremes
- Air quality and dust modelling

Responsibilities

- Maintenance and management of various geophysical data-sets
- Statistical analysis for the study of temporal and spatial variations of atmospheric and climate data
- Data science and visualisation
- Presentation and interpretation of scientific results

What turns data into actual climate data?

The **short-term** state of the atmosphere is named weather (e.g., temperature, precipitation, humidity, cloudiness, wind, et al.), and it can vary from minute to minute and location to location.

Climate is a description of the **long-term** pattern of weather conditions at a location. The expression "long-term" usually means 30 years or more, believed to be a good length of time to establish the usual range of conditions at a given location throughout the year.

The difference between weather variability and long-term climate trends is like the difference between the path of a dog and the path of the person walking the dog

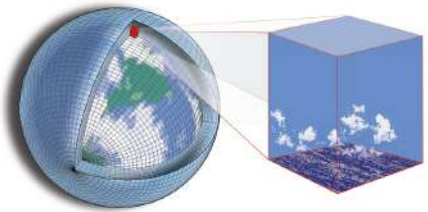
Adapted from : https://www.climate.gov/maps-data/primer/comparing-climate-and-weather
Animation : https://youtu.be/e0vj-0imOLw

Large climate data: structure, retrieval and processing

Climate data: numerical model vs. real-world observations

Numerical model

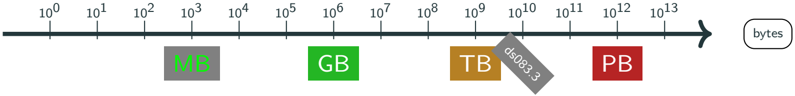
- 361×740 grid points (0.25 deg) (e.g.: ds083.3, NCEP, [1])
- 34 vertical levels
- **6-hourly** output, over **15 yrs.**
- multiple quantities (T, p, u, ...)



In-situ observations

- daily to hourly data
- increasing coverage
- smart devices are being progressively used

Extremely large data-sets (≈ peta-bytes): special binary **formats** needed.



Climate Data: Conventional Formats

Selection of common formats

`HDF4`, `HDF5`
Hierarchical **D**ata **F**ormat (NASA)

→ `GRIB1`, `GRIB2`
Gridded **B**inary
(World Meteorological Organization)

→ `netCDF`
Network **C**ommon **D**ata **F**orm
(National Center for Atmospheric Research)

Features of binary file:

- non human readable
- sequence of bytes
- memory efficient
- quickly accessible
- cross-platform
- non proprietary format

Focus of this talk: effective post-processing and visualization of large-scale, model-generated climate data. The above format(s) greatly help when post-processing!

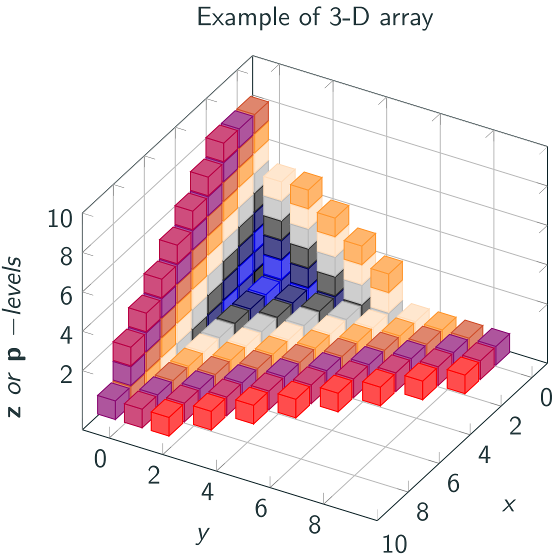
Source: `https://climatedataguide.ucar.edu/climate-data-tools-and-analysis/common-climate-data-formats-overview`

netCDF format is array-oriented:

- **Self-Describing**: a netCDF file includes information about the data it contains, i.e.: attributes or metadata. (See: [2])
- **Portable**: a netCDF file can be accessed by computers with different ways of storing integers, characters, and floating-point numbers.
- **Scalable**: small subsets of large datasets in various formats may be accessed efficiently through netCDF interfaces, *even from remote servers*.
- **Appendable**: Data may be appended to a properly structured netCDF file without copying the dataset or redefining its structure.

Adapted from: [3] (www.unidata.ucar.edu/software/netcdf/)
More sources: [2], [4], [5].

`netCDF` structure: data and metadata



data: numerical values of physical quantities (`T`, `p`, `v`,...)

metadata:

alphanumeric description;
"data describing data"

(sub)fields such as:

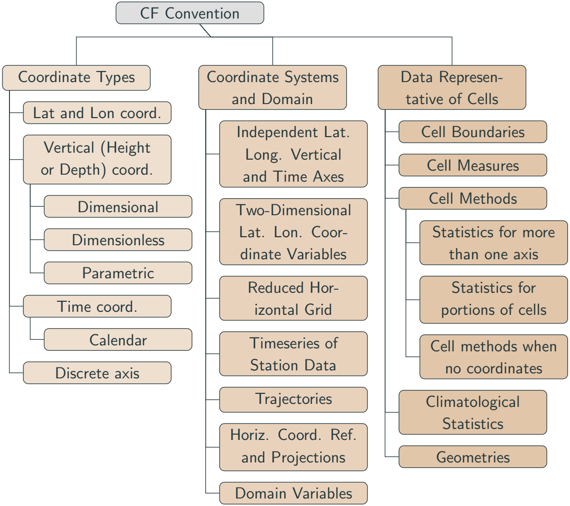
- size
- dimensions
- type
- unit & time ref
- space/time convention

collection of self-describing, portable objects (See: [4])

Data describing data – ncdump

```
# Output file generated with FLEXPART dispersion model
ncdump -h my_netCDF_FileName.nc # "-h" stands for "header only"

dimensions:
    time        = UNLIMITED ; // (336 currently)
    longitude   = 200 ; # nr. of longitudes (gridded)
    latitude    = 280 ; # nr. of latitudes  (gridded)
    height      = 2   ; # nr. of vertical levels
    numspec     = 1   ; # number of chem. species (i.e. tracers)
    pointspec   = 1   ; # nr. of release point(s)
    nchar       = 45  ; # string lenght
    numpoint    = 1   ; # number of receptor(s)

variables:
    int time(time) ;
        time:units                 = "seconds since 2020-12-01 00:00" ;
        time:calendar              = "proleptic_gregorian" ;

    float longitude(longitude) ;
        longitude:long_name        = "longitude in degree east" ;
        longitude:axis             = "Lon" ;
        longitude:units            = "degrees_east" ;
        longitude:standard_name = "grid_longitude" ;
        longitude:description   = "grid cell centers" ;
```

Climate and Forecast (CF) Convention



List of "Software that 'understand' CF" – `https://cfconventions.org/software.html`
Online "CF checker" available – `https://pumatest.nerc.ac.uk/cgi-bin/cf-checker.pl`

More available reference: [6], [7], [8]

FAIR (climate) data

Findable

Machine-readable metadata are essential for automatic discovery of data-sets and services

Accessible

How can data be accessed, possibly including authentication and authorisation

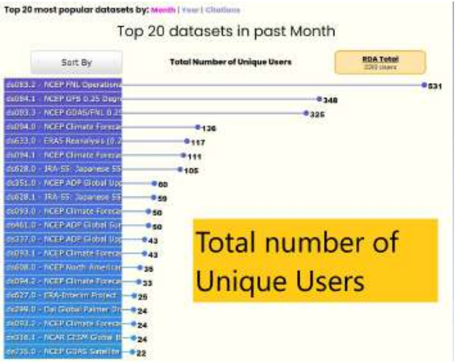
Interoperable

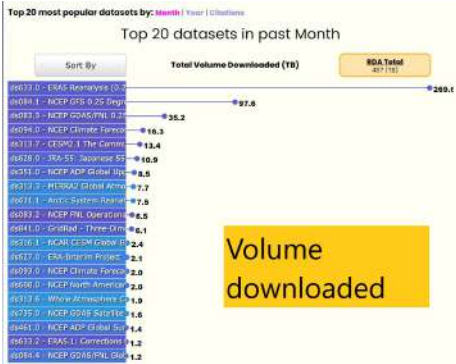
the data need to interoperate with applications or workflows for analysis, storage, and processing

Reusable

Metadata and data should be well-described so that they can be replicated and/or combined in different settings

The Research Data Archive (RDA) – free access big data center





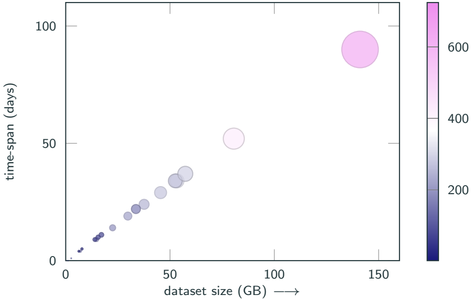

```
import rdams_client as rc          #import the rdams_client

#control file manually, with custom dates, variables, levels and products
idx = rc.submit('controlFile.ctl') #submit a request, and store index
rc.get_status()                    #check if ready, or more can be submitted
rc.download(idx)                   #download that index, when ready
```

Eg. datasets: Atmosphere • Ocean • Land cover • Ice sheet cover • Agriculture

NCAR–UCAR: https://rda.ucar.edu/ – Access requires creation of a user profile (e.g., affiliation and role)

The Research Data Archive (RDA) – the actual content

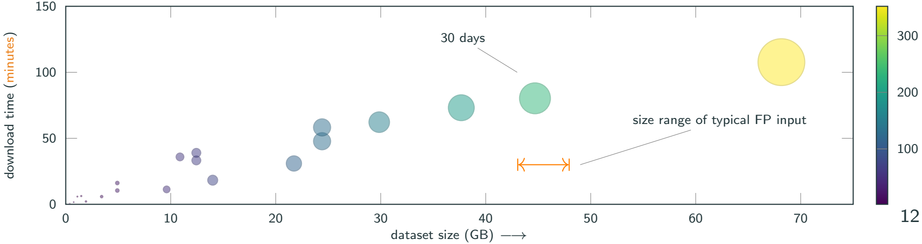
File size vs. its time span - marker color and size is proportional to number of files composing each dataset



Data content, ds083.3 (see [1]):

- Nr. of variables : 18
- Spatial Coverage : global, @ 0.25 deg
- Vertical levels : 34 (isobaric)
- Products:
  - → Analysis3-hour
  - → Accumulation (initial+0 to initial+3)
  - → 3-hour Average (initial+0 to initial+3)
  - → 3-hour Forecast

ds083.3 - NCEP FNL (Final) operational global analysis and forecast data 0.25-degree grid
Marker color and size is proportional to number of files composing each dataset



The fastest ~~pistol~~ climate operator in the west

- cdo* – Climate Data Operator
  - origin: MPI-M, Hamburg, Germany
  - type of data: `grib`, `netCDF`
- eccodes*:
  - origin: ECMWF
  - type of data: `GRIB 1&2; BUFR 1&2`
- wgrib2*:
  - origin: NOAA, Climate Prediction Center
  - type of data: `grib2`

Encode and decode. Open source – *Python interface

E.g. of other tools:

- `nco`: netCDF opeartors – manipulates & analyzes netCDF formats.
- `GeoCAT` Geoscience Community Analysis Toolkit: collection of Python tools related to NCAR Command Language – "swiss knife"

The spirit of open source & open science! More references: [9], [10]

Visualizing climate data

NCL - Seasonal mean for radionucleide dispersion

Extracting the near-surface field from initial volume of data:

$$\left[^{131}I\right]_{(t_i,x,y,z_0)} = \left[^{131}I\right]_{(t_i,x,y,z=0)}$$

Definition of Relative Risk Index (RRi), by normalising the field:

$$RRi_i = \frac{\left[^{131}I\right]_{(t_i,x,y,z_0)}}{max(\left[^{131}I\right]_{t_i})_{(x,y,z_0)}}$$

Extraction of monthly mean:

$$< \left[^{131}I\right] >_{montly} = \sum_{i=1}^{nx}\sum_{j=1}^{ny}\sum_{k=1}^{nt}\cdots \qquad if\ \ j\ \in\ [1,2,3]$$

NCL - Seasonal mean for radionucleide dispersion

Extracting the near-surface field from initial volume of data:

$$[{}^{131}I]_{(t_i,x,y,z_0)} = [{}^{131}I]_{(t_i,x,y,z=0)}$$

Definition of Relative Risk Index (RRi), by normalising the field:

$$RRi_i = \frac{[{}^{131}I]_{(t_i,x,y,z_0)}}{max([{}^{131}I]_{t_i})_{(x,y,z_0)}}$$

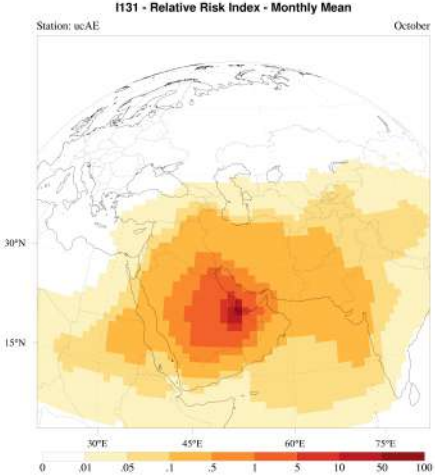
Extraction of monthly mean:

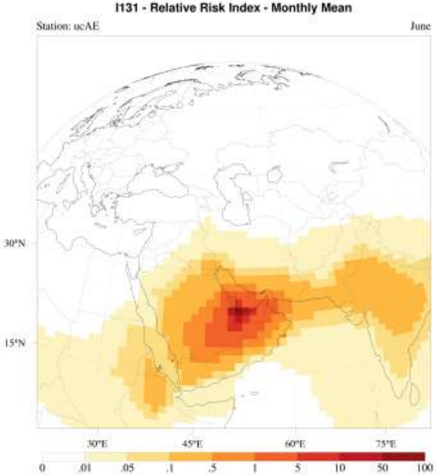
$$< [{}^{131}I] >_{montly} = \sum_{i=1}^{nx}\sum_{j=1}^{ny}\sum_{k=1}^{nt} \cdots \quad if\ \ j \in [1,2,3]$$

Meanwhile in `cdo` (either command line or Python, ≈ 2 sec/step):

1. `cdo -sellevel,34 -selname,I131 infile outfile`
2. `cdo.expr('norm = I131 / fldmax(I131)'), infile outfile)`
3. `cdo monmean infile outfile`
4. `cdo -O -P 4 -f nc4 -z zip_5 -copy infile outfile`

High-quality (publication ready) maps





Full control over all possible parameters: land boundaries, color-scale, general appearance, output format, etc... – Python compatible!

Animating results – Paraview

PARAVIEW ([11]) is an open source software for creating high quality animations of scientific data. Link: https://youtu.be/Ks15emMSh7E

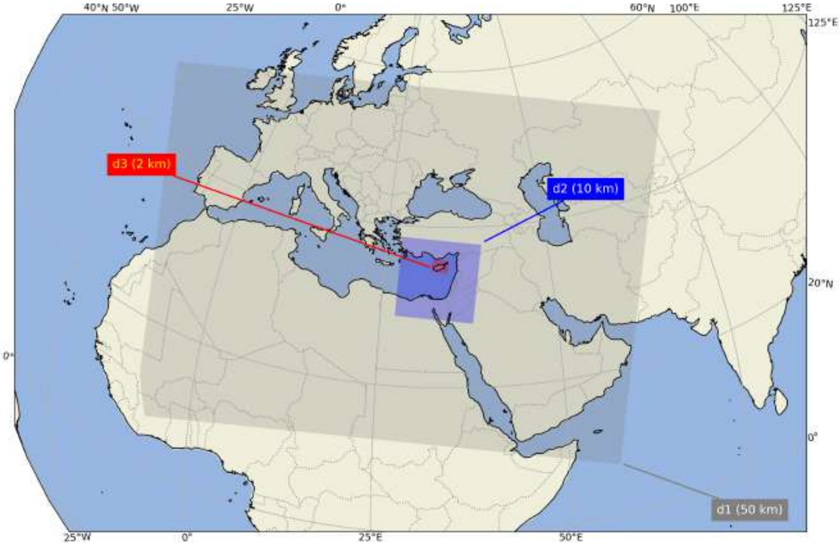
Iodine – [^{131}I]

- **gas**
- Half life-time: $\approx$ **8 days**
- residing in the atmosphere

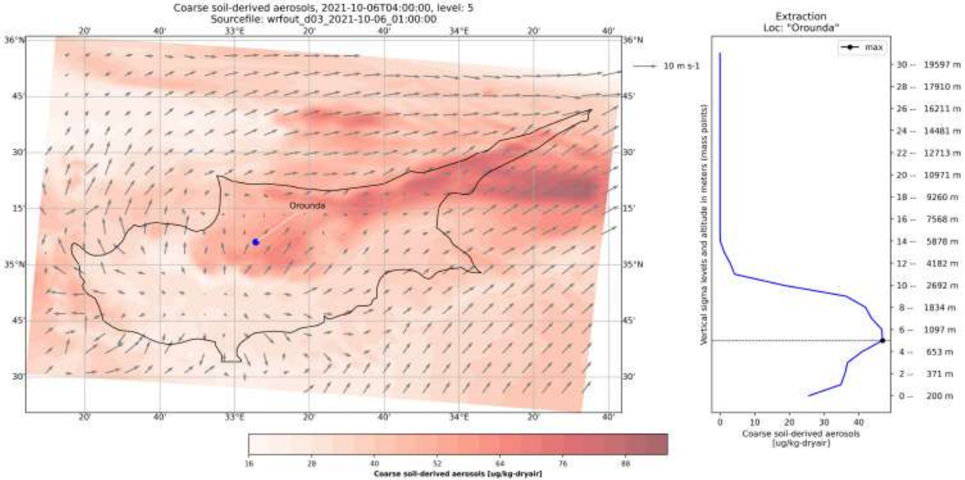
Cesium – [^{137}Cs]

- **aerosol**, accumulation (20 yrs)
- Half life-time: $\approx$ 30 yrs decay
- deposited on the ground

WRF model – nested domains (Δx: 50→10→2)



WRF model – dust plumes – maps and vertical profiles



Aim – use high-resolution forecast over Cyprus to optimize UAV flight plan with USRL ⟶ **Altitude? Trend? Time-window?**

High Performance Computing (HPC): the modelling of radionuclide dispersion using FLEXPART

"Release from a hypothetical accident in Akkuyu Nuclear P.P."

Conclusions: Topographic factor was found to play a crucial role:
"*different meteorological data sets are likely to reveal different results even if all other parameters are kept constant*" -

Bilgiç and Gunduz, Journal of Environmental Radioactivity, 2020 ([12]).

Improved & more detailed data-set:

High-resolution input (`ds083.3`)

$0.5^\circ \rightarrow 0.25^\circ$

6 $\rightarrow$ 3 hrs

`FlexPy` package

old, improved

"Release from a hypothetical accident in Akkuyu Nuclear P.P."

Conclusions: Topographic factor was found to play a crucial role:
"*different meteorological data sets are likely to reveal different results even if all other parameters are kept constant*" -

Bilgiç and Gunduz, Journal of Environmental Radioactivity, 2020 ([12]).

Improved & more detailed data-set:

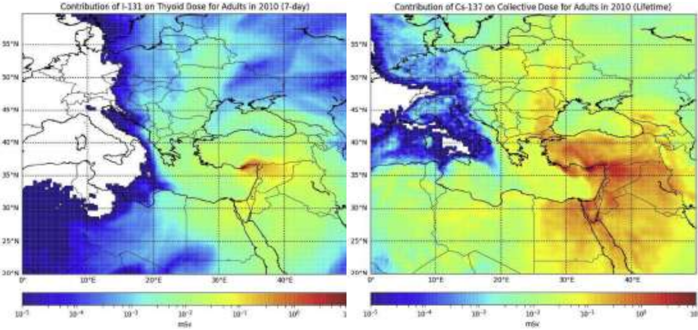
High-resolution input (`ds083.3`)

$0.5^\circ \rightarrow 0.25^\circ$

$6 \rightarrow 3$ hrs

`FlexPy` package

old, improved



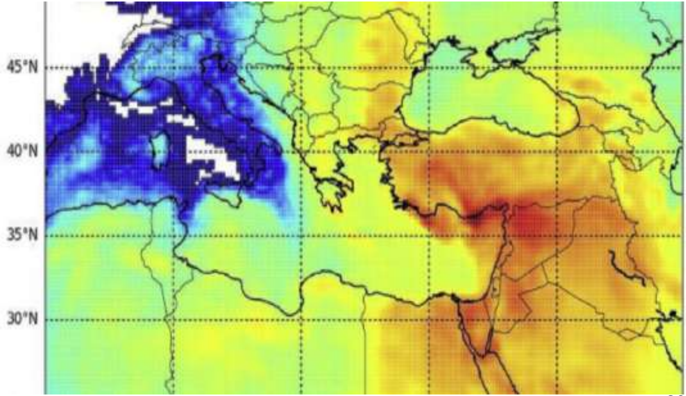
"Release from a hypothetical accident in Akkuyu Nuclear P.P."

Conclusions: Topographic factor was found to play a crucial role: "*different meteorological data sets are likely to reveal different results even if all other parameters are kept constant*" -

Bilgiç and Gunduz, Journal of Environmental Radioactivity, 2020 ([12]).

Expansion:

- High-resolution input (`ds083.3`)
- 0.5° → 0.25°
- 6 → 3 hrs
- `FlexPy` package



Future objectives – Dispersion modelling @ Akkuyu N.P.P.

Objective 1

Create a daily/weekly/seasonal climatology based on multi-year input data. Quickly grab available results/data from the shelf and hand over to policy makers and decision takers...

Objective 2

Reference runs to be compared with available bibliography. Pure research-related speculations...

Objective 3

Reference climatology for future risk assessment – web based services? Dissemination etc...

FLEXPART Lagrangian Dispersion model - v 10.4

– Lagrangian particle dispersion models (LPDM) simulate atmospheric transport & turbulent mixing of gases & aerosols, as well as dust!

– Loss processes (radioactive decay, chemical loss, dry/wet deposition) may affet particles and their mass.

– Importance of LPDMs: ability to run backward in time: trajectories. Inverse modelling and identification of sources!

– Computational time scales linearly with number of particles used.

Particle trajectory

$\mathbf{X}, \mathbf{v} \rightarrow$ position & wind vector

$t, \Delta t \rightarrow$ time and its increment

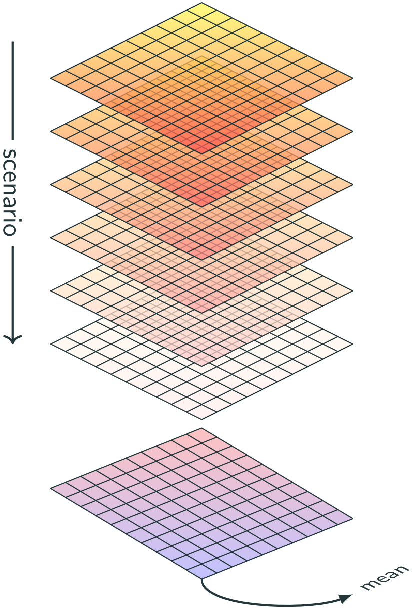
$$\mathbf{X}(t + \Delta t) = \mathbf{X}(t) + \mathbf{v}(x, t) \cdot \Delta t$$

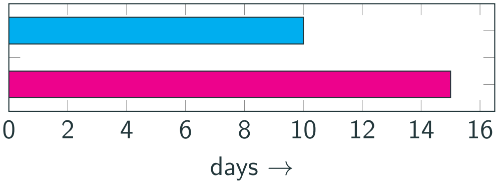
Wet deposition

$$m(t + \Delta t) = m(t) exp(-\Delta t / \beta)$$

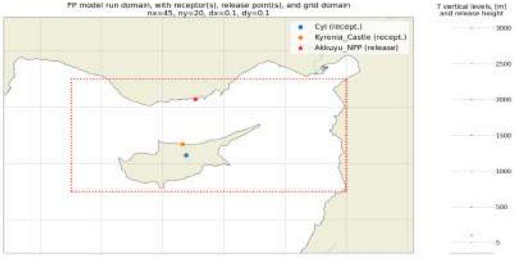
References: [13], [14], [15].

FLEXPART – Climatology based on multi-year input: set-up





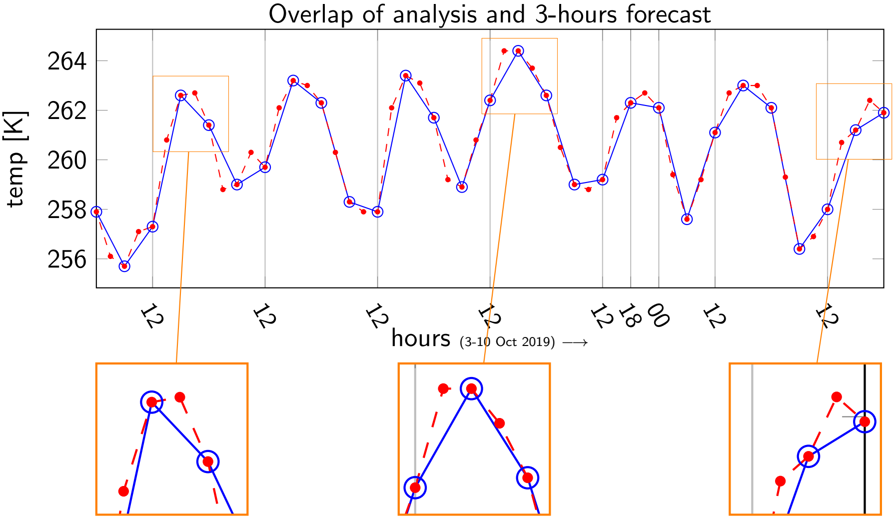
Emission @ Akkuyu (■) and run duration (■)



;

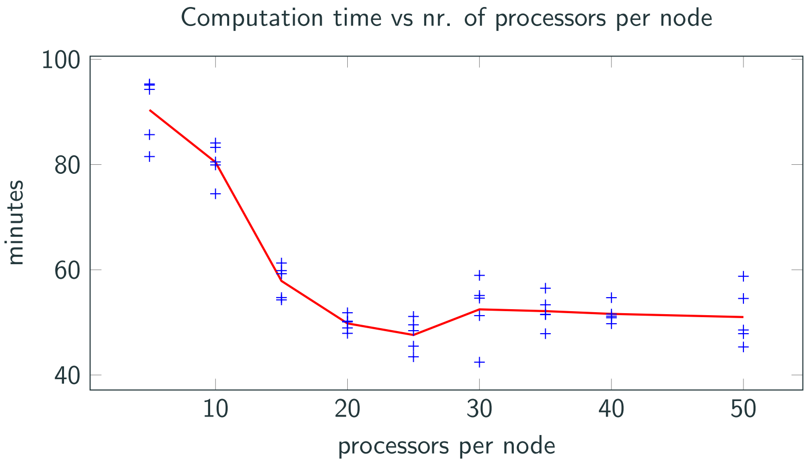
Automatized model set-up – climatology

Improving temporal resolution with `ds083.3`: inclusion of f+03



Based on an improved version of Global Forecast System (gfs)

Performance and CPU resources



CyI HPC facility – using AMD "*Epyc*" cluster.

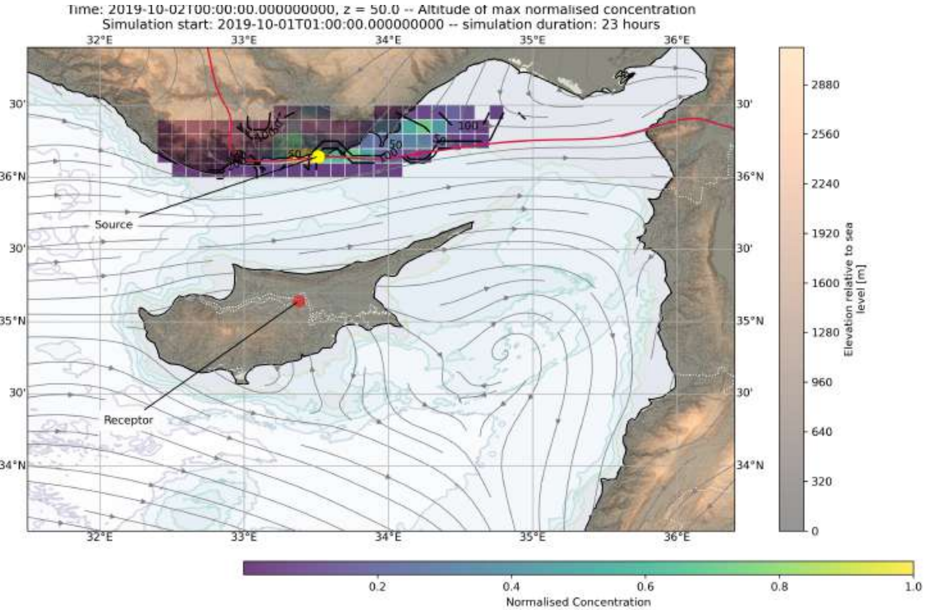
Inputs – COMMAND file [truncated for brevity]

```
*****************************************************************
* Input file for the Lagrangian particle dispersion model FLEXPART
*                  Generated with FlexPy
*****************************************************************
&COMMAND
 LDIRECT=                  1, ! Sim. direction  1 (forward), -1 (backward)
 IBDATE=            20191001, ! Start date of the simulation   ;
 IBTIME=              000000, ! Start time of the simulation   ;
 IEDATE=            20191015, ! End date of the simulation     ;
 IETIME=              230000, ! End  time of the simulation    ;
 LOUTSTEP=              3600, ! average concentr. calculated every LOUTSTEP (s
 LOUTAVER=              3600, ! Interval of output averaging (s)
 LOUTSAMPLE=             900, ! Interval of output sampling  (s),
 ITSPLIT=           99999999, ! Interval of particle splitting (s)
 LSYNCTIME=              900, ! All processes synchronized to this
 IOUT=                     9, ! Output type: [1]mass 2]pptv 3]1&2 4]plume 5]1&4
 IPOUT=                    1, ! Particle position output: 0]no 1]every output 
 LSUBGRID=                 0, ! Increase of ABL heights due to sub-grid scale 
 LCONVECTION=              1, ! Switch for convection parameterization;
 LAGESPECTRA=              0, ! Switch for calculation of age spectra
 IFLUX=                    0, ! Output of mass fluxes through output grid box 
 MDOMAINFILL=              0, ! Switch for domain-filling, if limited-area par
 IND_SOURCE=               1, ! Unit for source  : [1]mass 2]mass mixing ratio
 IND_RECEPTOR=             1, ! Unit for receptor: [1]mass 2]mass mixing ratio
 MQUASILAG=                0, ! Quasi-Lagrangian mode to track individual numb
```

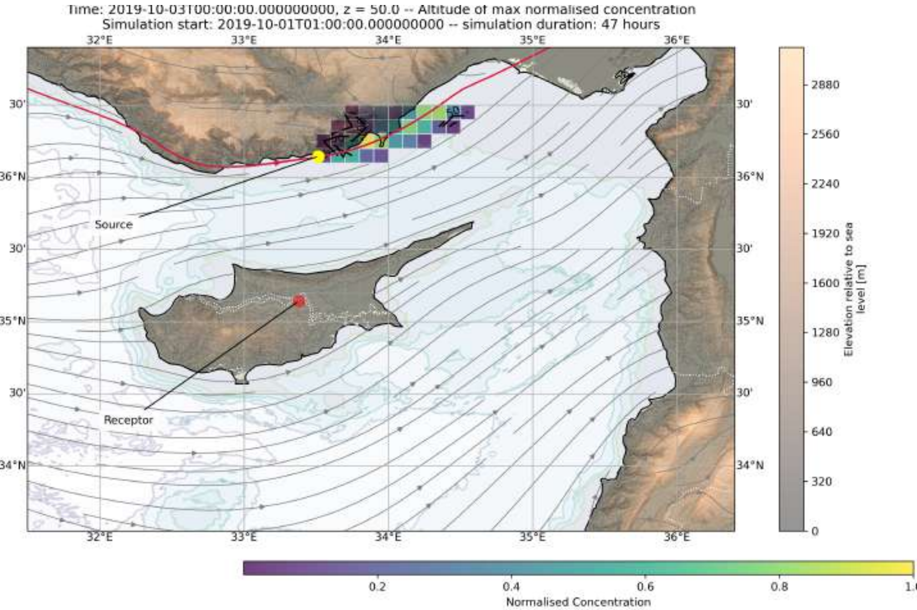
OUTPUTs – OUTGRID file [truncated for brevity]

```
31.50000       GEOGRAFICAL LONG. OF LOWER LEFT CORNER OF OUTPUT GRID
OUTLONLEFT     (left boundary of first grid cell - non its centre)

34.50000       GEOGRAFICAL LAT. OF LOWER LEFT CORNER OF OUTPUT GRID
OUTLATLOWER    (lower boundary of first grid cell - non its centre)

45             NUMBER OF GRID Pts IN X DIRECTION (=No. of cells+1)
NUMXGRID

20             NUMBER OF GRID Pts IN Y DIRECTION (=No. of cells+1)
NUMYGRID

0.10000        GRID DISTANCE IN X DIRECTION
DXOUTLON
0.10000        GRID DISTANCE IN Y DIRECTION
DYOUTLAT
5              NUMBER OF LEVELS
LEVEL 1        HEIGHT OF LEV (UPPER BOUNDARY)
500
LEVEL 2        HEIGHT OF LEV (UPPER BOUNDARY)
1000
LEVEL 3        HEIGHT OF LEV (UPPER BOUNDARY)
1500
LEVEL 4        HEIGHT OF LEV (UPPER BOUNDARY)
```

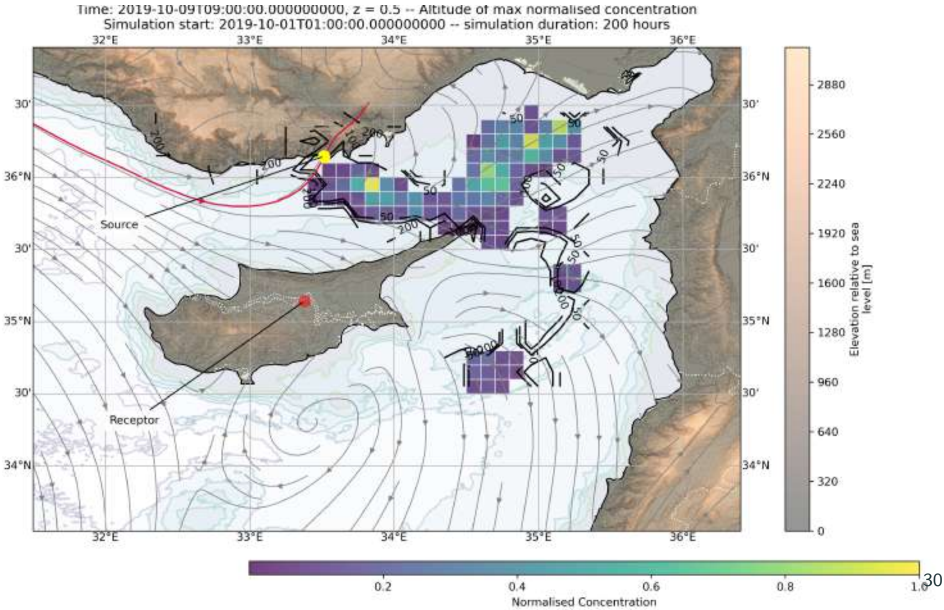
FLEXPART results – Iodine – normalised concentration 1day



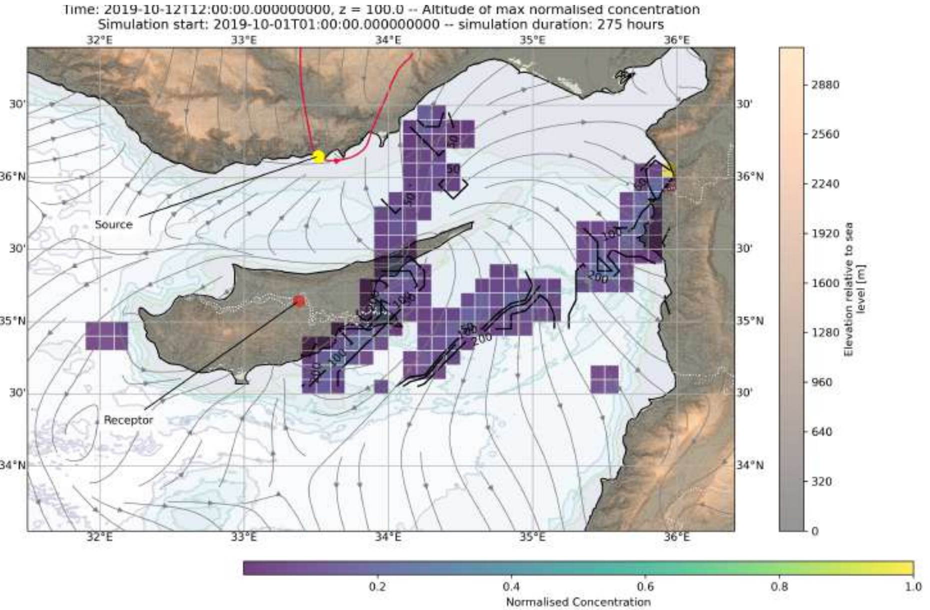
FLEXPART results – Iodine – normalised concentration 2 days



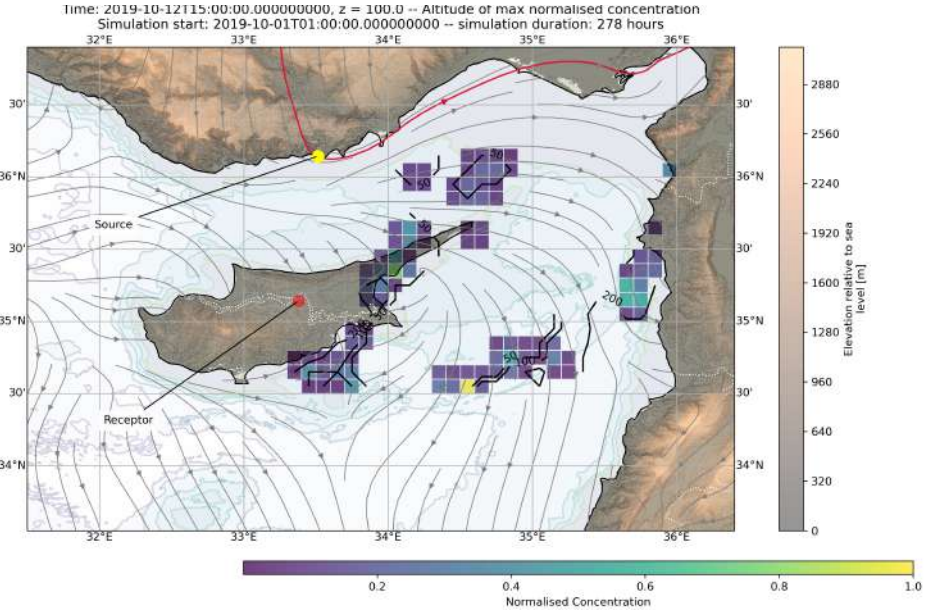
FLEXPART results – Iodine – normalised concentration 1/3



FLEXPART results – Iodine – normalised concentration 2/3



FLEXPART results – Iodine – normalised concentration 3/3



End of Presentation

Questions are welcome

Backup slides on following pages

Marco Miani — Computational Support Specialist

Climate and Atmosphere Research Center (CARE-C)

+357 22 397 561 — m.miani@cyi.ac.cy

Web: `cyi.ac.cy` — `emme-care.cyi.ac.cy`

References i

[1] National Centers for Environmental Prediction, National Weather Service, NOAA, U.S. Department of Commerce, "NCEP GDAS/FNL 0.25 Degree Global Tropospheric Analyses and Forecast Grids," Boulder CO, 2015. [Online]. Available: https://doi.org/10.5065/D65Q4T4Z

[2] NOAA, "World Ocean Database ragged array netCDF format." [Online]. Available: https://www.nodc.noaa.gov/OC5/WOD/netcdf_descr.html

[3] UniData.ucar.edu, "Network common data form (netcdf)." [Online]. Available: https://www.unidata.ucar.edu/software/netcdf/

[4] netCDF, "An introduction to netcdf." [Online]. Available: https://www.unidata.ucar.edu/software/netcdf/docs/

References ii

[5] NOAA, "What is netcdf?" [Online]. Available: https://www.esrl.noaa.gov/psd/data/gridded/whatsnetCDF.html

[6] CF Community, "CF conventions and metadata." [Online]. Available: http://cfconventions.org/

[7] B.O.D.C., "Climate and forecast (CF) netcdf format." [Online]. Available: https://www.bodc.ac.uk/resources/delivery_formats/cfnetcdf_format/

[8] N.A.S.A., "Climate and forecast (CF) metadata conventions." [Online]. Available: https://earthdata.nasa.gov/esdis/eso/standards-and-references/climate-and-forecast-cf-metadata-conventions

[9] U. Schulzweida, "C.D.O. user guide," Oct. 2019. [Online]. Available: https://doi.org/10.5281/zenodo.3539275

[10] "ecCodes," 2017. [Online]. Available: https://www.ecmwf.int/node/18132

[11] J. Ahrens, B. Geveci, and C. Law, "Paraview: An end-user tool for large-data visualization," in The Visualization Handbook, 2005.

[12] E. Bilgiç and O. Gunduz, "Dose and risk estimation of Cs-137 and I-131 released from a hypothetical accident in Akkuyu Nuclear Power Plant," Journal of Environmental Radioactivity, vol. 211, p. 106082, 01 2020.

[13] M. Cassiani, A. Stohl, D. Olivié, Ø. Seland, I. Bethke, I. Pisso, and T. Iversen, "The offline lagrangian particle model flexpart–noresm/cam (v1): model description and comparisons with the online noresm transport scheme and with the reference flexpart model," Geoscientific Model Development, vol. 9, no. 11, pp.

4029–4048, 2016. [Online]. Available:
https://gmd.copernicus.org/articles/9/4029/2016/

[14] A. Stohl, G. Wotawa, P. Seibert, and H. Kromp-Kolb, "Interpolation errors in wind fields as a function of spatial and temporal resolution and their impact on different types of kinematic trajectories," J. Appl. Meteorol., vol. 34, pp. 2149–2165, 1995.

[15] A. Stohl and P. Seibert, "Accuracy of trajectories as determined from the conservation of meteorological tracers," Q. J. Roy. Meteor. Soc., vol. 125, pp. 1465–1584, 1998.

Backup slides

cdo – some more insight with examples

`cdo -ntime <infile>` – Display number of timesteps of a file

`cdo -selname,tas <in> <out>` – Select variable tas from file

`cdo -seltimestep,1/12 <in> <out>` – Select a time range ($1 \rightarrow 12$)

`cdo -invertlat <in> <out>` – Invert latitudes from N-S to S-N

Interpolate model levels to pressure levels:

```
cdo -ml2pl,92500,85000,50000,20000 <in> <out>
```

Create a file with masked ocean:

```
cdo -setrtomiss,-20000,0 -topo topo-Land.grb
```
⟵

Interpolate 6-hourly data to 1-hourly data:

```
cdo -inttime,6 <infile> <outfile>
```

→ operators can be combined and concatenated!

```
cdo -timmean -yearsum -setrtoc,0,15,0 -selname,depth ...
    -selmon,1,2,3 -selyear,1960/1969 <infile> <outfile>
```

Source: https://code.mpimet.mpg.de/projects/cdo/wiki/Tutorial – more examples+descriptions available

wgrib2 - slice and dice grib2 format (encode and decode!)

Question: I have a grib2 file that contains information at global scale. Using wgrib2, I want to extract a subset from it based on user-provided latitude(s) and longitude(s). **How can I do that**?

```
wgrib2 INput.grb 30:35:.2 31:34:.2 OUTput.grb2 grib
```

INput.grb	- original input file name
30:35:.2	- lat_i, lat_e, Δ_{lat}
31:34:.2	- lon_i, lon_e, Δ_{lon}
OUTput.grb	- output file name
grib	- format to be encoded in

Confront with:

cdo -sellonlatbox,lon_i,lon_e,lat_i,lat_e INput.grb OUTput.grb

Adapted from: https://stackoverflow.com/questions/11048920/extracting-data-from-grib-file-based-on-latitude-and-longitude

ecCodes – effective use to compare two suspicious `.grb` files

Question: From **R**esearch **D**ata **A**rchive, download *exact* same input climate data, via (a) GUI and (b) control file. Model would accept only one of them. – **Why**?

`"OUTPUT grid outside mother domain"` – FLEXPART

```
long [totalLength]:                           [945255] != [1297979]
long [latitudeOfFirstGridPoint]:              [90000000] != [-90000000]
long [longitudeOfFirstGridPoint]:             [0] != [180000000]
long [latitudeOfLastGridPoint]:               [-90000000] != [90000000]
long [longitudeOfLastGridPoint]:              [359750000] != [179750000]
long [scanningMode]: [0] != [64]              ([00000000] != [01000000])
long [parameterCategory]:                     [2]   != [0]
long [parameterNumber]:                       [2]   != [0]
string [typeOfFirstFixedSurface]:             [sfc] != [pl]
long [scaledValueOfFirstFixedSurface]:        [10]  != [200]
long [dataRepresentationTemplateNumber]:      [3]   != [0]
[missingValueManagementUsed] not found in 2nd field
```

Beside coordinate grid, more metada being inspected (here, only a few shown...)

Parameter identification – grib v.1 vs grib v.2 definition tables

GRIB2 - GRIB1
Local Parameter Conversion Table

Revised 02/27/2009
Red text depicts changes made since 03/25/2008

The following table contains a list of GRIB1 locally defined NCEP parameters and their GRIB2 equivalent.

GRIB2			Parameter Name	GRIB1	
Product Discipline	Category	Parameter Number		Parameter Table #	Parameter Number
Sect 0 Octet 7	Sect 4 Octet 10	Sect 4 Octet 11		Sect 1 Octet 4	Sect 1 Octet 9
0	0	0	Temperature	3 (SST)	11
0	0	15	Virtual potential temperature	2	189
0	0	192	Snow Phase Change Heat Flux	2	229
0	0	193	Temperature tendency by all radiation	2	216
0	1	22	Cloud Water Mixing Ratio	2	153
0	1	23	Ice mixing ratio	2	178
0	1	24	Rain Water Mixing Ratio	2	170
0	1	25	Snow Water Mixing Ratio	2	171
0	1	32	Graupel mixing ratio	2	179
0	1	192	Categorical Rain	2	140
0	1	193	Categorical Freezing Rain	2	141
0	1	194	Categorical Ice Pellets	2	142
0	1	195	Categorical Snow	2	143
0	1	196	Convective Precipitation Rate	2	214
0	1	197	Horizontal Moisture Divergence	2	135
0	1	199	Potential Evaporation	2	228
0	1	200	Potential Evaporation Rate	2	145
0	1	201	Snow Cover	2	238
0	1	202	Rain Fraction of Total Liquid Water	129	131
0	1	203	Rime Factor	129	133
0	1	204	Total Column Integrated Rain	129	138
0	1	205	Total Column Integrated Snow	129	139
0	1	206	Total Icing Potential Diagnostic	2	186
0	1	207	Number concentration for ice particles	2	198

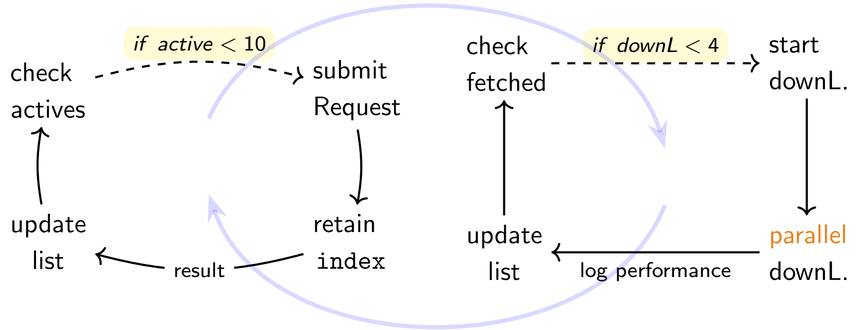
GRIB1: Element coding by `element` and `table` – Eg: temperature is `element`=11 in `table`=3 (WMO)

GRIB2: Element coding by a **triplet**: `discipline`, `category`, `parameter` – Eg: temperature is (0,0,0)

The Python API client – 2 steps workflow

1- Submitting requests to RDA

2- Bringing completed request home



Submission constrain: max number of active requests per user being processed (RDA side) **equals 10**.

Download constrain: max number of active datasets being downloaded in parallel, **should** ~~not exceed~~ **equal 3**.

```
from rda_client import submit, checkStatus, download, authenticate
```

https://github.com/NCAR/rda-apps-clients/blob/main/src/python/rdams_client.py