Climate Change

Copernicus Regional Reanalysis for Europe

Data access

Ludvig Isaksson, SMHI

Climate Change

Main sections

1. **Introduction**
   - MARS/CDS
2. **MARS retrieval system**
   - Getting started
   - Web API scripts
3. **Post-processing**
   - CDO
   - ncview

Introduction - MARS/CDS

MARS

- ECMWF's **M**eteorological **A**rchival and **R**etrieval **S**ystem.
- Petabytes of data, mainly using GRIB format.

CDS

- **C**limate **D**ata **S**tore - cornerstone of the C3S infrastructure.
- Soon hosting the UERRA data, now only available via MARS.
- UERRA data will remain available via MARS web API, but be retrieved from CDS.

CDS toolbox, ERA5

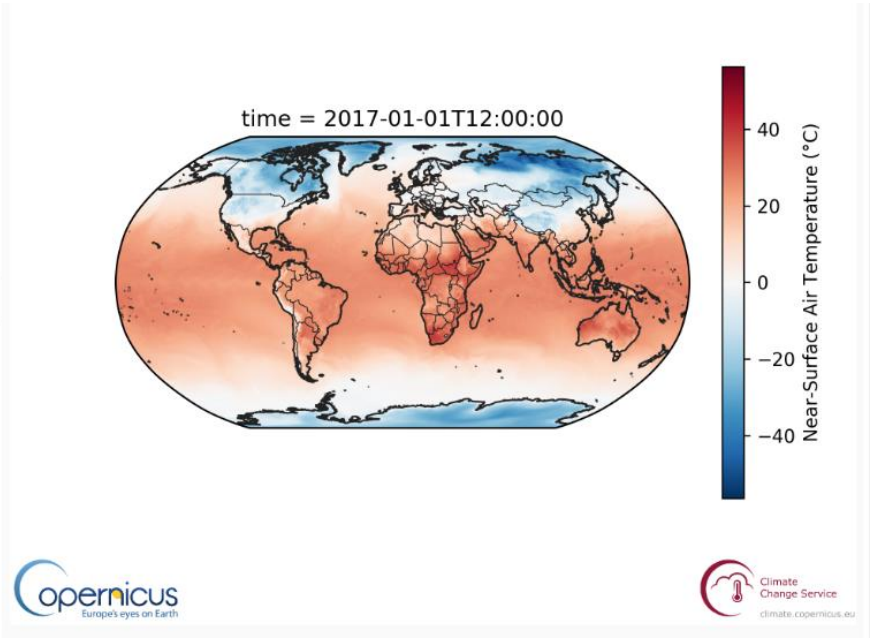
```
import cdstoolbox as ct


@ct.application(title='My first application!')
@ct.output.figure()
def application():
    data = ct.catalogue.retrieve(
        'reanalysis-era5-single-levels',
        {
            'variable': '2m_temperature',
            'product_type': 'reanalysis',
            'year': '2017',
            'month': '01',
            'day': '01',
            'time': '12:00'
        }
    )

    print(data)

    fig = ct.cdsplot.geomap(data)

    return fig
```



MARS - Getting started

Without registration you can:

- Look at available public data.
- Check out all the available parameters.

Data are free to download after you register.

- Access data via the MARS Catalogue.
- Retrieve data via the Web API.

MARS - Web API

- Download larger amounts of data.
- Some installation is needed, all well described on the page Access ECMWF Public Datasets
  (note: accept the UERRA license)
- Use python scripts to retrieve data.

MARS web API - scripts

```
def uerra_eswi_request(requestDates, target):
    server.retrieve({
        "class": "ur",
        "stream": "oper",
        "type": "fc",
        "dataset": "uerra",
        "origin" : "eswi",
        "date": requestDates,
        "expver": "prod",
        "levtype": "sfc",
        "param": "201/202",
        "target": target,
        "time": "00/12",
        "step": "4/5/6/9/12/15",
    })
```

```
def retrieve_uerra_eswi():
    yearStart  = 2017
    yearEnd    = 2017
    monthStart = 1
    monthEnd   = 12
    for year in list(range(yearStart, yearEnd + 1)):
        for month in list(range(monthStart, monthEnd + 1)):
            startDate = '%04d%02d%02d' % (year, month, 1)
            numDays = calendar.monthrange(year, month)[1]
            lastDate = '%04d%02d%02d' % (year, month, numDays)
            target = "T2m_maxmin_%04d%02d_fc.grb" % (year, month)
            requestDates = (startDate + "/TO/" + lastDate)
            uerra_eswi_request(requestDates, target)
```

Data access

Save the script to disk, then execute it in a command prompt:

```
python retrieve_T2m_maxmin.py
ls
T2m_maxmin_UERRA_201701_fc.grb  T2m_maxmin_UERRA_201704_fc.grb
T2m_maxmin_UERRA_201707_fc.grb  T2m_maxmin_UERRA_201710_fc.grb
T2m_maxmin_UERRA_201702_fc.grb  T2m_maxmin_UERRA_201705_fc.grb
T2m_maxmin_UERRA_201708_fc.grb  T2m_maxmin_UERRA_201711_fc.grb
T2m_maxmin_UERRA_201703_fc.grb  T2m_maxmin_UERRA_201706_fc.grb
T2m_maxmin_UERRA_201709_fc.grb  T2m_maxmin_UERRA_201712_fc.grb
```

GRIB

- The result of the retrieval are binary GRIB files.
- One GRIB file can contain many data fields, metadata is included.
- To process the GRIB files a tool is needed, for example CDO, Climate Data Operators.

Post processing - CDO

```
cdo -sinfov T2m_maxmin_UERRA_201712_fc.grb
   File format : GRIB2
    -1 : Institut Source   Steptype Levels Num    Points Num Dtype : Parameter name
     1 : unknown  unknown  max           1   1    319225   1  P16  : 2t
     2 : unknown  unknown  min           1   1    319225   1  P16  : 2t
   Grid coordinates :
     1 : projection                : points=319225 (565x565)
                          mapping : lambert_conformal_conic
                                x : 0 to 6204000 by 11000 m
                                y : 0 to 6204000 by 11000 m
   Vertical coordinates :
     1 : height                    : levels=1
                           height : 2 m
   Time coordinate :  unlimited steps
     RefTime =  2017-12-01 00:00:00  Units = hours  Calendar = proleptic_gregorian
  YYYY-MM-DD hh:mm:ss  YYYY-MM-DD hh:mm:ss  YYYY-MM-DD hh:mm:ss  YYYY-MM-DD hh:mm:ss
  2017-12-01 04:00:00  2017-12-01 05:00:00  2017-12-01 06:00:00  2017-12-01 09:00:00
    ..........................................................................
    ..................
  2017-12-31 18:00:00  2017-12-31 21:00:00  2018-01-01 00:00:00  2018-01-01 03:00:00
```

ncview

```
cdo -f nc copy T2m_maxmin_UERRA_201712_fc.grb T2m_maxmin_UERRA_201712_fc.nc
ncview T2m_maxmin_UERRA_201712_fc.nc
```