



ERA4CS Joint Call on Researching and Advancing Climate Services Development – Topic B
(GRANT AGREEMENT 689029)



Deliverable Title	<i>M3.2b High-quality climate prediction data available to WP4</i>		
WP number	3		
Lead Beneficiary	<i>Barcelona Supercomputing Center</i>		
Contributors	<i>Pierre-Antoine Bretonnière, BSC Louis-Philippe Caron, BSC Nuria Perez-Zanon, BSC</i>		
Deliverable Due Date	<i>15/03/2020</i>		
Nature of the Deliverable		<i>R – Report</i>	
	SP	<i>SP – Software Package</i>	
	D	<i>D – Dataset</i>	
		<i>O - Other</i>	

Version	Date	Modified by	Comments
1.0	18/09/2020	L-P Caron P-A Bretonnière	Original version



Table of contents

Getting Started	4
Downloading Forecasts Directly from the C3S Website	4
Downloading Data Using a Script and NetCDF data	7
Downloading multiple start dates and variables in NetCDF	8





Getting Started

This document describes different strategies to obtain seasonal forecast data from the Climate Data Store (CDS) and shows how to use scripts that were designed to facilitate the interface with the MEDSCOPE toolbox CTools.

1. Register to the C3S portal [here](#).
2. The Climate Data Store Application Program Interface (CDS API) is a service providing programmatic access to CDS data. To install the CDS API on
 - a. [Windows](#)
 - b. [Mac](#)
 - c. [Linux](#)

Downloading Forecasts Directly from the C3S Website

Once the CDS API client is installed, it can be used to request data from the datasets listed in the CDS catalogue.

1. Select a seasonal forecast [product](#) from the catalogue. Data can be monthly means, daily or 6-hourly frequency (accumulations or instantaneous), 2D or 3D, etc.
2. Click on the tab “Download data” and select the model(s), variable(s), lead time(s), etc needed.
3. It is necessary to agree to the Terms of Use for every datasets that you intend to download (button near the bottom of the page).
4. The button “Show API request” at the bottom displays the python code to be used to download the data. The toolbox request is to use this data in the toolbox without downloading the file itself.
5. To obtain the data, one can either click on the green button “Submit form” or run the API request on a local machine. For the former, the API request is run as is, and it will be





possible to download the data once the request has been processed. For the latter, the API request is run (in Linux) by typing “python request.py” on the local machine. It is also possible to manually modify the request in that case (more on this below).

Example: Downloading 10-m winds for the July hindcasts of ECMWF System 5. The following API request will be produced:

```
import cdsapi
c = cdsapi.Client()
c.retrieve(
    'seasonal-monthly-single-levels',
    {
        'originating_centre': 'ecmwf',
        'system': '5',
        'variable': '10m_u_component_of_wind',
        'product_type': 'monthly_mean',
        'year': [
            '1993', '1994', '1995',
            '1996', '1997', '1998',
            '1999', '2000', '2001',
            '2002', '2003', '2004',
            '2005', '2006', '2007',
            '2008', '2009', '2010',
            '2011', '2012', '2013',
            '2014', '2015', '2016',
            '2017', '2018', '2019',
            '2020',
        ],
        'month': '07',
        'leadtime_month': [
```





```
'1', '2', '3',  
'4', '5', '6',  
,  
'format': 'grib',  
,  
'download.grib')
```

This will produce a request for

Originating centre:

ECMWF

System:

5

Variable:

10m u-component of wind

Product type:

Monthly mean

Year:

1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008,
2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020

Month:

July

Leadtime month:

1, 2, 3, 4, 5, 6

Whole available region:

Format:

GRIB



Downloading Data Using a Script and NetCDF data

The default format of the data downloaded from the C3S data is grib, which is not convenient for users in MEDSCOPE. The NetCDF format that we usually used is not explicitly offered for seasonal forecasts, although it is offered on the CDS for other types of data. The reason for this is that the current C3S NetCDF format only accommodates 1 time axis whereas the seasonal forecasts require 2: one to match the data to the actual time and one for the lead time.

There is currently a hidden option to download the data in NetCDF. This can be done simply by running the previous script in python, but substituting the format to 'netcdf'. However, because of the issue about the time axis, if one were to repeat the previous example, there would be no way to differentiate between the forecast time and the start date once the data are downloaded in NetCDF and the file would be for all purposes, useless. The alternative is to download one start date at a time, as expected by our R packages. In that case, the API request can then be modified as follow and run using "python request.py":

```
import cdsapi
c = cdsapi.Client()
c.retrieve(
    'seasonal-monthly-single-levels',
    {
        'originating_centre': 'ecmwf',
        'system': '5',
        'variable': '10m_u_component_of_wind',
        'product_type': 'monthly_mean',
        'year': '1993',
        'month': '07',
        'leadtime_month': [
            '1', '2', '3',
```

```
'4', '5', '6',  
],  
'format': 'netcdf',  
},  
'uas_ECMWF_SEAS5_19930701.nc')
```

In this case, we modified the number of start dates, the format and the file name of the data that are being downloaded. By repeating this step for all the required start dates, one would obtain a series of files containing 1 variable and 1 start date.

Downloading multiple start dates and variables in NetCDF

The previous request will work well for one start date. However, if one is interested in multiple start dates and/or variables, it will be time consuming. For those working on **Linux**, we offer an efficient solution to download multiple start dates and variables at once.

It requires the following modules:

- Python/3.7.3-foss-2015a
- NCO/4.4.9-foss-2015a
- netCDF/4.3.3.1-foss-2015a
- cdsapi/0.2.3-foss-2015a-Python-3.7.3

If you don't have these modules/libraries installed on your system, installation can be done through conda. Conda allows you to install locally all the python and the necessary dependencies and libraries without needing root permissions nor conflicting with other installations

If conda is not installed on your system (the command "which conda" doesn't return anything), you can install install Miniconda on your system following the official instructions: <https://docs.conda.io/projects/continuumio-conda/en/latest/user-guide/install/>)



Create a virtual environment in which the dependencies will be installed (skip if it is not the first time you run it)

```
conda create -n cdsdownloader
```





Activate the newly created environment. From there, all what you do will be inside this environment.

```
conda activate cdsdownloader
```

(Note: It might ask you to select a shell and to log out and back in again.)

Install the 2 main libraries needed by the tool (skip if it is not the first time you run it)

```
conda install -c conda-forge/label/cf202003 pynco
```

```
conda install -c conda-forge/label/cf202003 cdsapi
```

A copy of the scripts is available on [Gitlab](#). There is a python file (.py) and a config file (.conf). Once you have retrieved these files, you can copy them to your working directory. The python file is a script similar to what one gets with a regular CDS request, however it contains loops which will allow to retrieve multiple start dates and variables in the same request. **Do not modify this file.**

The config file is where you select

- The directory where to put the data
- The model you want to download
- The variables
- The start dates

While you can download as many start dates and variables as you want, you can **only select one model at the time**. All the available options are provided in the .conf file.

To run the downloader, simply run the following command after updating the configuration file:

```
python download_seasonal_cds_frequency.py frequency.conf
```



There are currently 5 files and 2 sets of scripts: one to obtain monthly mean data and one to obtain data at the original frequency that they are produced (daily or 6-hourly). The files are:

- **README.md** - Additional information to run these scripts <https://earth.bsc.es/gitlab/external/cstools/-/tree/develop-seasonal-downloader/CDSeasonalDownloader>
- **download_seasonal_cds_6h.py** - The python script to obtain high frequency data
- **download_seasonal_cds_monthly.py** - The python script to obtain monthly mean data
- **6h.conf** - The configuration file for the 6-hourly and daily data
- **monthly.conf** - The configuration file for the monthly data

The advantage of using this script over the simpler options presented above are:

- Files are in NetCDF format.
- Can request multiple start dates and variables at once. The script will also organize the files and rename the variables/dimensions in such a way that is compatible/expected by the CTools package.
- It will automatically detect the number of lead times, so you don't have to specify it.

