

Wikiprint Book

Title: `udg/ecom/dataserver/interfaces/python`

Subject: TracMeteo - `udg/ecom/dataserver/interfaces/python`

Version: 17

Date: 07/05/2022 10:48:27 AM

Table of Contents

Placeholder for table of contents

0

A python function has been created in order to access the *SPECS-EUPORIAS Data Portal* in a user-friendly way, allowing the retrieval of dimensional slices of selected simulation members from the ECMWF's SYSTEM4 model. This function ([?load_system4.py](#)) automatically cares about the proper location of the right indices for data sub-setting across the different variable dimensions, given a few simple arguments for subset definition. In addition, instead of retrieving a NetCDF file that needs to be opened and read, the requested data is directly loaded into the current python working session, according to a particular structure described below, prior to data analysis and/or representation.

The request is simply formulated via the `load_system4` function:

```
>>> load_system4(dataset, var, season, leadMonth, lonLim, latLim, year, members=[])
```

The arguments of the function are described below:

- dataset:** A character string indicating the full URL path to the OPeNDAP dataset. Currently, the accepted values correspond to the System4 datasets described in Section [?Datasets](#), for instance `http://www.meteo.unican.es/tds5/dodsC/system4/System4_Seasonal_15Members.ncml`, but using the `System4_Seasonal_15Members.ncml`, `System4_Seasonal_51Members.ncml` or `System4_Annual_15Members.ncml` ending strings depending on the dataset of choice.
- var:** Variable code. Argument values currently accepted are `tas`, `tasmin`, `tasmax`, `pr` or `mslp`, as internally defined in the vocabulary for System4 following the nomenclature displayed in the table below. However, note that new variables and datasets will be progressively included. Note that depending on the time step of the variable the units might be referred to different time aggregations. For instance, currently `mslp` is 6-hourly, and thus the 6-hourly mean value is returned for each time step. Similarly, 24-h accumulated values are returned for `pr`, and so on. Note that the *instantaneous* and *aggregated* fields in table below refer to the potential time step values that the variables may take, which does not mean that the resolution provided by the System4 model is necessarily that.

Short Name	Long name	Units	Instantaneous	Aggregated
tasmax	Maximum temperature at 2 metres	degC	No	Yes
tasmin	Minimum temperature at 2 metres	degC	No	Yes
tas	Mean temperature at 2 metres	degC	Yes	Yes
pr	Total precipitation accumulated	mm	No	Yes
mslp	Mean sea level pressure	Pa	Yes	Yes

- members:** List of members to select. In the above case, a single member (the first) of the System4 ensemble is loaded, but additional members could be also specified (e.g. `members=[0,1,2,3,4]` for the first five members).
- lonLim:** Vector of length = 2, with minimum and maximum longitude coordinates, in decimal degrees, of the bounding box selected.
- latLim:** Vector of length = 2, with minimum and maximum latitude coordinates, in decimal degrees, of the bounding box selected.
- season:** A vector of integers specifying the desired season (in months, January=1, etc.) of analysis. Options include a single month (as in the above example) or a standard season (e.g. `season = [12,1,2]` for standard Boreal winter, DJF).
- year:** List of years to select. Note that in cases with year-crossing seasons (e.g. winter DJF, `season = [12,1,2]`), for a particular year period `year = [1981,1982,1983]`, by convention the first season would be DJF 1981/82.
- leadMonth:** Lead month forecast time corresponding to the first month of the specified season. Note that `leadMonth = 1` for `season = [1]` (January) corresponds to the December initialization forecasts. In this way the effect of the lead time forecast in the analysis of a particular season can be analyzed by just changing this parameter.

The output returned by the function consists of a list with the following elements providing the necessary information for data representation and analysis:

- VarName:** Character string indicating the variable long name, as defined in the vocabulary (see Table above)
- VarUnits:** Character string. Units of the variable, as returned in `MemberData`
- TimeStep:** A `diffTime` class object. Indicates the time span of each forecast time
- MemberData:** This is a list of length n , where n = number of members of the ensemble selected by the `member` argument. Each element of the dataset is a 2-D matrix of i rows \times j columns, of i forecast times and j grid-points
- LatLonCoords:** A 2-D matrix of j rows (where j = number of grid points selected) and two columns corresponding to the latitude and longitude coordinates respectively.
- RunDates:** A `POSIXlt` time object corresponding to the initialization times selected. There is an initialization time associated to each forecast time.

ForecastDates: A list with two POSIXlt time elements of length i , corresponding to the rows of each matrix in `MemberData`. The list contain tow elements:

- **Start:** Starting times of the verification period of the variable
- **End:** End time of the verification period of the variable