

The *SPECS-EUPORIAS Data Portal* can be remotely accessed from R via the [?loadSystem4.R](#) function. Note that this function is part of a more comprehensive R package currently under development. This function 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 R working session, according to a particular structure described below, prior to data analysis and/or representation.

In order to explain the `loadSystem4` function, in the next lines we describe an illustrative example considering one-month lead time forecasts of minimum surface temperature for January over a window centered in Europe (0°W - 30°E and 35°S - 65°N). A more elaborated example describing a multi-model selection of a similar dataset is presented in the tutorial, which can be downloaded [?here](#), or in the section Examples.

The request is simply formulated via the `loadSystem4` function in the following way:

```
> ds <- "http://www.meteo.unican.es/tds5/dodsC/system4/System4_Seasonal_15Members.ncml";
> openDAP.query <- loadSystem4(dataset = ds, var = "tasmin", members = 1,
+   lonLim = c(0,30), latLim = c(35,65),
+   season = 1, years = 1981:2000, leadMonth = 1)
```

The arguments of the function are the next described:

- `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](#), as specified in the above example, but using the `System4_Seasonal_15Members.ncml`, `System4_Seasonal_51Members.ncml` or `System4_Annual_15Members.ncml` ending strings.
- `var`: Variable code. Argument values currently accepted are `tas`, `tasmin`, `tasmax`, `pr` or `mssl`, as internally defined in the dictionary defined for System4 following the nomenclature displayed in the table below. However, note that new variables and datasets will be progressively included. Note that System4 precipitation is internally deaccumulated by the function to return daily accumulated values.

Short Name	Dataset variable
tasmax	Maximum temperature at 2 metres since last 24 hours surface
tasmin	Minimum temperature at 2 metres since last 24 hours surface
tas	Mean temperature at 2 metres since last 24 hours surface
pr	Total precipitation surface
mssl	Mean sea level pressure surface

- `members`: Optional. Default to all members. 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=NULL` for all members, or `members=1:5` 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. `period = c(12,1,2)` for standard Boreal winter, DJF).
- `years`: Optional. Default to all available years. Vector of years to select. Note that in cases with year-crossing seasons (e.g. winter DJF, `season = c(12,1,2)`, for a particular year period `years = 1981:2000`), by convention the first season would be DJF 1980/81, if available (otherwise a warning message is given).
- `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 of the function is a data structure with all the requested information as follows.

```
> str(openDAP.query)
List of 4
 $ MemberData   :List of 1
 ..$ : num [1:930, 1:1600] 275 277 278 279 277 ...
 $ Coordinates  : num [1:1600, 1:2] 64.5 63.7 63 62.2 61.5 ...
 .. attr(*, "dimnames")=List of 2
 .. ..$ : NULL
 .. ..$ : chr [1:2] "lat" "lon"
 $ RunDates     : POSIXlt[1:30], format: "1981-12-01" "1982-12-01" "1983-12-01" ...
 $ ForecastDates: Date[1:930], format: "1982-01-01" "1982-01-02" "1982-01-03" ...
```

The output consists of a list with the following 4 elements:

- `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
- `Coordinates`: 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.
- `ForecastDates`: A vector of class `Date` of length  $i$ , corresponding to the rows of each matrix in `MemberData`, containing the verification dates.