Data Management Plan (DMP) for Clim4Energy Indicators

JIN Xia[1,2], LEVAVASSEUR Guillaume[1,3] and DENVIL Sébastien[1,4]

[1] Pierre-Simon Laplace Institute (IPSL, France)
[2] Atomic Energy Commission (CEA, France)
[3] Pierre and Marie Curie university (UPMC, France)
[4] National Centre of Scientific Research (CNRS, France)

30nd April 2017

First version published on 30nd September 2016

Revisions made to this document after 30nd September 2016 are summarized in the last Appendix.

Table of contents

Overview

This guideline document describes the life-cycle requirements for the Clim4Energy (C4E) indicators, from the data production to the final publication.

As a proof of concept, C4E will deliver 9 pan-European climate byproducts with cross-sectoral consistency, documentation and guidance. These indicators focus on 5 different energy issues: wind energy, hydropower, generation of electricity, forestry, oil and gas production.

This document is the C4E project's Data Management Plan (DMP). The DMP considers the scientific data life cycle and describes choices that will be made for the metadata standards to be used, database schema, data access policy and data access methods, long term archival and the costs associated to data management.

Project data (sectoral indicators characterizing the energy sector sensibility to climate change) have to be described and categorized (spatial and temporal coverage, standard names, units, used algorithms, inputs data, version). They will then be indexed by data catalogs compliant with OGC standards (Open Geospatial Consortium). Inputs data needed to create the project products (climate simulations, observations, reanalysis, energy production data, …) needs to be identified and tracked to enable final products traceability.

We present here a first attempt to provide (i) a metadata structure and (ii) the publication workflow for multi-sectoral C4E indicators. This document specified the metadata conventions, Data Reference Syntax (DRS) and Controlled Vocabularies (CVs) for use by the C4E data producers and publishers. The correct specification of these metadata will enable data discovery, access and support data provenance information which provides visibility to the data analytic pipeline and simplifies the tracking of errors.

All indicators must comply with the standards defined in this document to allow for faceted and free-text searching therefore ensuring data discovery via the Earth System Grid Federation (ESGF) platform.

In this document, *Section 1* specifies instructions for data producers. The data format and metadata attributes have to follow the conventions in *Section 2*. All of the facets of DRS and CVs are detailed in *Section 3*. *Section 4* describes the quality control method. *Section 5* shows how to define the version of data and to trace the data. *Section 6* describes the data access and the relationship with ESGF community.

In addition, any revisions about this document are recorded in the Appendix F.

1. Instructions for data producers

An indicator can be directly written into the appropriate format or post-processed to comply within standards detailed in *Section 2*. Each indicator should be finally provided in a NetCDF file resulting in **ONLY ONE indicator per file**.

Note that **all of the data files must be written in the NetCDF-4 format** in order to be archived and published on ESGF portal. All the datasets of indicators will be published by the Unidata's Thematic Real-time Environmental Distributed Data Services (THREDDS). With the help of the THREDDS Data Server (TDS), the datasets can be downloaded in a variety of formats (e.g. NetCDF format, text format, etc.).

The production workflow :

A. Set the variable name and dimensions in compliance with the standards detailed in *Section 2*.
B. Set the NetCDF global attributes within the file as specified in *Section 3.1*.
C. Specify a filename according to the structure presented in *Section 3.2.2.2*.
D. Send file examples of each indicator to DMP group for a previous check of the NetCDF metadata contents and the filename before the large-scale production of datasets.
E. Produce all data files.
F. Save the data files under a structured directory. It can be a simple directory self-designed or a directory as demanded by DRS in *Section 3.2.2.1*. In any way, the structure should be clear enough for the DMP workers to understand easily.
G. Transfer data files to IPSL server in several ways. Please contact one of the authors of this DMP:
    - JIN Xia (xjin@ipsl.jussieu.fr)
    - LEVAVASSEUR Guillaume (glipsl@ipsl.jussieu.fr)
    - DENVIL Sébastien (sebastien.denvil@ipsl.jussieu.fr)

The data manager will check the data files in many ways to ensure the data conformance and will notify the data producer when appropriate. **Because C4E data will be published in the ESGF system, it must comply 100% with the metadata conventions (*Section 2*), DRS and CVs (*Section 3*).** Contact us as soon as possible if you have any questions or suggestions about the DMP.

2. Clim4Energy NetCDF metadata conventions - Version 2.0

The *Copernicus Clim4Energy NetCDF metadata conventions* - Version 1.0 (CC4E-1.0) are based on the *NetCDF Climate and Forecast Metadata Conventions* - Version 1.6 (CF-1.6) [1]. CF-1.6 is a standard intended specifically for use with climate and forecast data. Comparing to CF-1.6, we have added some restrictions about metadata in CC4E-1.0 according to C4E indicators.

In this section, all the modifications about CF-1.6 are presented in details. Some important points that should be attentioned in C4E data producing are also been emphasized to help the data producers to seize rapidly the key points. In addition, examples about some specific cases are given for an intuitive comprehension of data producers. The structure of this section is similar to the structure of CF-1.6 conventions document [1], so that the users can easily find the corresponding part from CF-1.6 and understand what constraints we have added and what modifications we have done.

We will not repeat the same rules as illustrated by CF-1.6 if they are not so important to pay attention to. So it is strongly suggested that the users have a good comprehension of CF-1.6 in advance.

2.1. NetCDF Files and components

Each NetCDF data file should include dimensions, variables and their attributes, and other metadata as specified in the *Section 2.2*.

If the variable is a function of time, more than one time sample (but not necessarily all time samples) may be included in a single file. Data representing a long time-series should be split into several files of same size, which should neither be too large (to be unwieldy) nor too small (as to create vexing I/O performance issues). In that latter case, the data producer should be careful to the file naming and the time axis squareness. The start/end time has to reflect the period covered by the time axis and follows the previous chunked file according to the time frequency.

2.1.1. Naming conventions

In CF-1.6, it is allowed that the variable, dimension and attribute names should begin with a letter and be composed of letters, digits and underscores (*Section 2.3* in CF-1.6). However, the variable names are not controlled in detail by the CF-1.6 convention.

In *CC4E-1.0*, more restrictions and demands are given about the variable name to avoid a disorder of output. It should be noted that:

- The **variable name** should be a combination of only English letters (a-z) and digits (0-9) starting with a letter. **Any other characters are not allowed** (such as underscore, dash, hyphen, point, space, etc.). Note that **only lowercase letters** are allowed.
- The variable name should certainly be concise but not too simple to give too less information about this variable. At least, it must be meaningful and it should illustrate all the key information.
- For the wind power capacity factor indicator (from WP1 and WP3), it is demanded that there should have one independent name for each class of turbine. The name of wind power capacity factor should include a keyword to distinguish different turbine class. For the 4 classes of turbines suited for an average wind speed of 10, 8.5, 7.5 and 6 m/s at hub height respectively (designed by IEC-61400-1), we suggest to use `wpcf100`, `wpcf85`, `wpcf75`, and `wpcf60` to be the variable names of wind power capacity factor for the corresponding turbine class.
- For the variables of climate-projection of WP3, the details about the variable should be given in the name to be distinguished from others. Please refer to the naming conventions made specifically for the variables names of WP3 (Table A.1 in *Appendix A*).

2.1.2. Variables and attributes

Each data file must contain **ONLY ONE single indicator**. Note that it is demanded that each data file contains only one indicator but not only one variable, as it can contain several ancillary variables related to the indicator.

About the details of variable attribute, please refer to the list of variable attributes (*Appendix A: Attributes* in CF-1.6). For an C4E indicator, it is demanded that the variable attributes of `cell_methods`, `long_name`, `missing_value`, `standard_name`, and `units` must be supplied by data producers. For other variables, it is demanded that `long_name` and `standard_name` must be given. It is allowed to give more attributes in case of need.

2.2. Description of the data

2.2.1. Units

Please refer to the CF-1.6 to make sure that the units used in data file is standard (*Section 3.1* in CF-1.6).

Note that the units `percentage` (“`%`”) is not accepted. In this case, “`1`” or `<units of numerator>/<units of denominator>` can be used instead of `percentage`.

The `units` of time axis should be “`days since 1949-12-1 00:00:00`”.

2.2.2. Long name

The long name (variable attribute `long_name`) is a free descriptive text. As its type is string, a variety of characters can be used. It must be concise, meaningful and reflect the variable name properties (*Section 2.1.1*). For the long name of variables of C4E output data, it must be pointed out that:

- In all the WPs, when data that is representative of cells can be described by simple statistical methods as given by *Appendix E: Cell Methods* in CF-1.6, the statistical method name should be included in the long name of variable.
- The name of wind power capacity factor indicator (from WP1 and WP3) should include a keyword to describe the type of corresponding wind turbine.

2.2.3. Standard name

The standard name (variable attribute `standard_name`) of one variable is very similar to the same variable’s long name but with underscore to combine all elements of `long_name`. If you have any questions about it, please refer to the *Section 3.3* in CF-1.6.

2.2.4. Ancillary Data

The ancillary data can be a variable that is associated to the indicator but is not a real indicator. It normally has the same dimension as the indicator has. For example, mask or significance about the indicator, both are related to the indicator, can be included in the same data file.

If there is ancillary variables, it is necessary to add the name of ancillary variables to the indicator as a variable attribute whose name is “ancillary_variables”. To get more details about ancillary data, please refer to the *Section 3.4* of CF-1.6.

Here is a simple example about one indicator that has two ancillary variables:

```
float indicator(...)
   .........
   indicator:ancillary_variables="indicator_mask indicator_significance" ;
float indicator_mask(...)
   indicator_mask:standard_name=...
   indicator_mask:long_name=...
   indicator_mask:units=...
  ...........
float indicator_significance(...)
   indicator_significance:standard_name=...
   indicator_significance:long_name=...
   indicator_significance:units=...
  ...........
```

2.3. Coordinate Types

2.3.1. Time Coordinate

time_bnds and climatological_bnds

time_bnds must be used when the variable represents temporal mean.

climatological statistics

climatological_bnds when the value of output_frequency include the keyword “Clim”. Eg. yrClim.

For seasonal forecast data - leadtime

For the seasonal forecast data

please add “forecast_reference_time” as global attribute.

forecast_reference_time is the start date of the forecast. The format is: YYYY-MM-DD(THH:MM:SSZ).

please add one time axis “leadtime”, which means that should have two time variables for seasonal forecast data: one being called time(time) which corresponds to the verification of the forecast and one called leadtime. Both of these variables are mandatory in the files and must have the following attributes:

```
dimensions:
        time = UNLIMITED ; // (120 currently)
        bnds = 2 ;
variables:
        double time(time) ;
                time:bounds = "time_bnds" ;
                time:units = "days since 1949-12-01 00:00:00" ;
                time:calendar = "noleap" ;
                time:axis = "T" ;
                time:long_name = "Verification time of the forecast" ;
                time:standard_name = "time" ;
        double time_bnds(time, bnds) ;
        double leadtime(time) ;
                leadtime:units = "days" ;
                leadtime:long_name = "Time elapsed since the start of the
forecast" ;
                leadtime:standard_name = "forecast_period" ;
// global attributes:
          :forecast_reference_time = "2013-01-01(T00:00:00Z)" ;
```

2.3.2. Discrete axis

station

2.3.3. Other axis

WP1: ensemble

2.4. Coordinate systems

2.4.1. Regular Latitude and Longitude Axes

If the data is a regular gridded, which means it has regular latitude axis, longitude axis, vertical axis, the coordinate system is easily to be built. You may refer to the section about independent latitude, longitude, vertical and time axis (*Section 5.1* in CF-1.6) or the section about two-dimensional latitude, longitude, coordinate variables (*Section 5.2* in CF-1.6).

As described in *Chapter 4* of CF-1.6, for the use of an external software package, two

2.4.2. Horizontal Coordinate Reference Systems

If the coordinate variables for a horizontal grid are not longitude and latitude, it is required that the true latitude and longitude coordinates be supplied via the **coordinates** attribute.

If rotated pole :

```
variables:
    double rlon(rlon) ;
      rlon:standard_name = "grid_longitude" ;
      rlon:long_name = "longitude in rotated pole grid" ;
      rlon:units = "degrees" ;
      rlon:axis = "X" ;
    double rlat(rlat) ;
      rlat:standard_name = "grid_latitude" ;
      rlat:long_name = "latitude in rotated pole grid" ;
      rlat:units = "degrees" ;
      rlat:axis = "Y" ;
    int rotated_latitude_longitude ;
              rotated_latitude_longitude:grid_mapping_name          =
"rotated_latitude_longitude" ;
      rotated_latitude_longitude:grid_north_pole_latitude = 37.55f ;
       rotated_latitude_longitude:grid_north_pole_longitude  =  177.5f
;
double hs(time, rlat, rlon) ;
                            hs:standard_name                       =
"significant_height_of_wind_and_swell_waves" ;
      hs:long_name = "mean significant wave height" ;
      hs:units = "m" ;
      hs:_FillValue = -1.e+20 ;
```

```
        hs:missing_value = -1.e+20 ;
        hs:cell_methods = "time: mean" ;
        hs:grid_mapping = "rotated_latitude_longitude" ;
```

2.4.3. Discrete data

For some indicators, they do not have continuous spatiotemporal coordinates, or they have at least one axis of the coordinates that may be discrete (e.g. WP2 and WP3). For these kinds of indicators, we can use “discrete axe” and “alternative coordinates” for the geophysical quantity which indicates either an ordered list or an unordered collection, and does not correspond to any continuous coordinate variable. The value of the discrete axis is called label here. Each label value must be unique. It can be character strings or numbers . Please refer to *Section 4.5: Discrete Axis* and *Chapter 6: Labels and Alternative Coordinates* in CF-1.6 for more information.

When the value of label is a string, it is demanded that the value should comply with the `standard_name` variable attribute. It means only English letters in lowercase, numbers, and underscores are permitted.

The value of labels of discrete axe is oftenly a list of region names. CF-1.6 has given a list of standardized region names [2]. However, those names supplied by CF-1.6 are not compatible for C4E indicators. In CC4E-1.0, we have made a new list of standardized region names including all the European country names that may be used by WP2 (Table B.1 in *Appendix B*) and a list of standardized names of the Nomenclature of Units for Territorial Statistics basic regions (NUTS-2, Table B.2 in *Appendix B*). Please refer to these tables in case you need it.

For an discrete indicator data, the first thing to make sure is how many coordinates are needed to locate each value of this indicator. Two kinds of discrete indicator files have been introduced here. They are: time-series discrete data and time-invariant statistical data with discrete axe.

2.4.3.1. Time-series of station data

Some indicators are derived from ragged distributed stations (e.g. WP2 and WP3). For this kind of indicators, at least one alternative discrete coordinate is needed to locate each indicator value. In order to dimension such a discrete axe, it is necessary to create one dimension whose value is the number of points that this discrete axe has.

Here we use `lbl` as the name of the dimension of discrete axe. The values of `lbl` can represent the identifier of each geographic location whose function is to distinguish different locations. It can be the identifier number of station, which may be a regular digital code, or a regular combination of letters and number (e.g. NUTS). It can also be the name of each geographic location (e.g. country name, region name, city name, etc.).

Suppose that `label` is the name of this discrete axe, it is important to indicate this alternative coordinate in the attribute of variable in the form of: `coordinates=”label”.`

Some examples that are designed specifically for C4E indicators are shown below. More examples about metadata of discrete data file please refer to *Appendix H.2* in CF-1.6.

Example 1: Daily load anomaly indicator (WP3)

The daily load anomaly (in WP3) is calculated by regions of European countries defined by NUTS-2. Each NUTS-2 region has a unique identifier number (`nuts2id`). So each value of indicator can be coordinated by two axes: `time` and `nuts2id`. So here `nuts2id` is a discrete axis. One dimension of `nuts2id` is `lbl`, which is the number of elements of `nuts2id` series. Another dimension of `nuts2id` is `strlen1`, which is the maximum length of any `nuts2id` value. In the attribute of indicator variable, it should be added that `coordinates = ”nuts2id”`.

Except all of the above, maybe it is also necessary to give more information to describe the region, such as region name. The region name can be supplied as another variable `nuts2region` of the same data file. `nuts2region` is the series of labels of region name. As each `nuts2region` value is corresponding to each `nuts2id`, `nuts2region` has the same dimension as `nuts2id`. Here `strlen1` and `strlen2` are used for `nuts2id` and `nuts2region` separately.

The values of `nutsid` and of `nuts2region` come from the standard name list of NUTS-2 basic regions that is shown in Table B.2 (*Appendix B*).

Suppose that the indicator name is `anomalyload`, the head file of the daily load anomaly indicator should be like this (here is just an example for some essential parts that should be noticed, but not a full description of head file. This is the same for all the examples of this section. Please pay attention to the content in bold style):

```
dimensions:
    times = 200;
    lbl = 274;
    strlen1 = 4;
    strlen2 = 64;
variables:
    float anomalyload(time,lbl);
        anomalyload:long_name=”anomaly national load”;
        anomalyload:standard_name=”anomaly_national_load”;
        anomalyload:units=”******”;
        anomalyload:cell_methods=”******”;
```

```
            anomalyload:missing_value=”******”;
            anomalyload:coordinates=”nuts2id”;
      char nuts2id(lbl,strlen1);
            nuts2id:long_name=”nuts identifier”;
            nuts2id:standard_name=”nuts_identifier”;
      char nuts2region(lbl,strlen2);
            nuts2region:long_name=”nuts2 region name”;
            nuts2region:standard_name=”nuts2_region_name”;
data:
      nutsid=”AT11”, “AT12”, “AT13”, ...
      nuts2region=”burgenland_a”, “lower_austria”, “vienna”, ...
       ...
```

Example 2: National degree day weighted by population (WP3)

The national degree day weighted by population (from WP3) is an indicator that is calculated by day and by country. If there is one identifier code in a regular form for each country, this indicator has the same coordinates as the daily load anomaly indicator (see Example 1). However, if there is not an identifier code for each country as in the Example 1, the series of country names should be used as a discrete axe of the indicator. `country` is the alternative coordinate for the indicator variable and it should be indicated in the attributes of indicator variable that `coordinates = ”country”`.

The values of `country` come from the standard names of European countries listed in Table B.1 (*Appendix B*).

Suppose that the indicator name is `15degday`, the head file should be as following (`lbl` and `strlen` are the same as in Example 1):

```
dimensions:
      times = 100;
      lbl = 50;
      strlen = 32;
variables:
      float 15degday(time,lbl);
            15degday:long_name=”national degree day weighted by pop”;
            15degday:standard_name=”national_degree_day_weighted_by_pop”;
            15degday:units=”******”;
```

```
        15degday:cell_methods=”******”;
        15degday:missing_value=”******”;
        15degday:coordinates=”country”;
    char country(lbl,strlen);
        country:long_name=”country name”
        country:standard_name=”country_name”
data:
    country=”austria”, “belarus”, “belgium”, ...
     ...
```

Example 3: Inflow anomalies indicator (WP2)

The inflow anomalies indicator data (from WP2), which is a time series data for each catchment over Europe, can use the same structure as designed in Example 1 (there is a regular identifier number for each catchment). Each catchment can be considered as one station or one region. Here is one example of transforming one data file from text format to NetCDF format.

An extraction from the beginning of a text file is as following in which the data is calculated by catchment:

ROWNR	SUBID	HAROID	…	CENTERX	CENTERY	LATITUDE	LONGITUDE	…
1	8801544	8801544	…	-22.5854	65.771	65.771	-22.5854	…
2	8801548	8801548	…	-24.472	65.5149	65.5149	-24.472	…
3	8000005	8000006	…	9.3096	59.1909	59.1909	9.3096	…
4	8115258	8000006	…	8.7466	59.1359	59.1359	8.7466	…
5	8115717	8000006	…	9.2398	58.9271	58.9271	9.2398	…
6	8000008	8000006	…	9.4109	59.1348	59.1348	9.4109	…
7	8102609	8000006	…	9.1371	58.9647	58.9647	9.1371	…
8	8103951	8000006	…	9.0461	59.1257	59.1257	9.0461	…
9	8000007	8000006	…	9.2676	59.1174	59.1174	9.2676	…
10	8000006	8000006	…	9.4933	59.1314	59.1314	9.4933	…
…	…	…	…	…	…	…	…	

Suppose the indicator name is `anoinflow`. It is a two-dimensional variable that are `time` and label of catchement. `ROWNR` is a regular sequence with a unique number for each catchment. So `ROWNR` can be used as one axe of `anoinflow`.

There are abundant parameters related to each catchment. All of the other parameters can be shown as subsidiary variables of indicator file. If there are too many parameters (more than 100 parameters in the example text file), a selection of parameters should be done according to their importances. Suppose that the parameters SUBID (`subid`), HAROID (`haroid`), LATITUDE (`latitude`), LONGITUDE (`longitude`) are selected. The head file should be as following (take care of the dimensions of different variables):

```
dimensions:
      times = 365;
      rownr = 34000;
variables:
      float anoinflow(time,rownr);
            anoinflow:long_name=”inflow anomalies”;
            anoinflow:standard_name=”inflow_anomalies”;
            anoinflow:units=”******”;
            anoinflow:cell_methods=”******”;
            anoinflow:missing_value=”******”;
            anoinflow:coordinates=”nuts2id”;
      int subid(rownr);
            subid:long_name=”******”;
            subid:standard_name=”******”;
      int haroid(rownr);
            haroid:long_name=”******”;
            haroid:standard_name=”******”;
      double latitude(rownr);
            latitude:long_name=”******”;
            latitude:standard_name=”******”;
      double longitude(rownr);
            longitude:long_name=”******”;
            longitude:standard_name=”******”;
data:
      subid=8801544, 8801548, 8000005, …
      haroid=8801544, 8801548, 8000006, …
       ...
```

2.4.3.2. Time-invariant statistical data with discrete axe

Except the time series data, some indicator data have time-invariant statistical field. When there are at least two statistical fields that are distinguished by characteristics or descriptions, a discrete axe should be used.

Example 4: Weather regime indicator (WP3)

The weather regime (from WP3) is a time-invariant indicator with longitude-latitude gridded distribution. There have been defined 4 types of weather regimes for winter season and 4 types for summer season. Each weather regime data are two-dimensional (longitude and latitude). For simplicity, it is suggested to write these 8 types of weather regimes in one file. So this weather regime indicator has three dimensions that are longitude (`lon`), latitude (`lat`), and one discrete axe of which the dimension value is 8 (`lbl`). Here we call this discrete axe `seasonandtype`, as it should include the information about both seasons (winter or summer) and type names of weather regime.

Suppose that the indicator name is `weareg`, the 4 weather regimes for winter are `type1`, `type2`, `type3` and `type4`, and the 4 weather regimes for summer are `type5`, `type6`, `type7` and `type8`, the head file of weather regime indicator should be as following (`lbl` and `strlen` are the same as in Example 1):

```
dimensions:
        lon = 200;
        lat = 150;
        lbl = 8;
        strlen = 32;
variables:
        float weareg(lbl,lat,lon);
                weareg:long_name=”weather regime”;
                weareg:standard_name=”weather_regime”;
                weareg:units=”******”;
                weareg:cell_methods=”******”;
                weareg:missing_value=”******”;
                weareg:coordinates=”seasonandtype”;
        char seasonandtype(lbl,strlen);
                seasonandtype:long_name=”season and weather regime type”;
                seasonandtype:standard_name=”season_and_weather_regime_type”;
```

```
data:
      seasonandtype=”winter_type1”, “winter_type2”, “winter_type3”,
      “winter_type4”, “summer_type5”, “summer_type6”, “summer_type7”,
      “summer_type8”;
       ...
```

2.5. Data representative of cells

2.5.1. Cell boundaries

For an indicator that has a regular gridded coordinate, it is necessary to provide boundary data when the indicator value represents a statistical value of one cell. It is not necessary to provide attributes for boundary variable since it is considered to be part of a coordinate variable’s metadata. Please refer to *Section 7.1: Cell Boundaries* in CF-1.6 in the need of your indicator data.

2.5.2. Cell methods

For any indicator data, it is necessary to indicate the mathematical method used to get this value, such as sum, mean, maximum, minimum, standard deviation, etc. (see all the methods listed by *Appendix E: Cell Methods* in CF-1.6). This method can be applied either to time axis or to space axis.

Please specify details of the calculation within the comment field of the global attributes if needed.

2.5.3. Climatological statistics

If an indicator represents climatological statistics (e.g. deviation in mean annual snow maximum storage in relation to a reference period of WP2, average degree day weighted by population of WP3, etc.), it should be very careful that how to describe the climatological periods. Please refer to the *Section 7.4* of CF-1.6, where it shows some examples for different climatological statistics variables.

climatology_bounds

3. Data Reference Syntax (DRS) and Controlled Vocabularies (CVs)

3.1. Global attributes

The C4E global metadata attributes follow a fixed structure and a pre-defined attribute names.

The list of global attributes can be divided into three parts:

1. Knowledge discovery in a file.
2. Metadata of input datasets used to compute an indicator.
3. Simple description of temporal and spatial characteristics of a given indicator.

For a full description of all C4E required global metadata attributes, please refer to Table C.1 that shows a list of all facets about NetCDF global metadata attributes (*Appendix C*).

Instructions to set the global attributes:

- Insert the whole list of attribute names to your NetCDF file (mandatory).
- Associate value to the attribute names. Leave empty field if not applicable for your data.
- Attribute values should be one of allowed value, or as clear and concise as possible if flexible value.

3.2. Data Reference Syntax (DRS)

This section provides a common naming system to be used in files, directories, metadata, and URLs to identify dataset wherever they might be located within the distributed data archive. It defines CVs for many of the components comprising the DRS. The purpose of the DRS is to provide a unique identifier for each dataset and file.

A “dataset” is ONE version of a dataset resulting from a single simulation/realization (i.e, characterized by a unique option of each attribute (also called “facet”) of the DRS before the version). The “dataset” is the finest granularity for publication.

For a full description of all of the DRS facets used by C4E datasets, please refer to Table D.1 that shows a list of all facets of DRS (*Appendix D*).

3.2.1. DRS facets for C4E indicators

All the facets required for indicator filename and dataset identifiers are listed in Table D.1 (*Appendix D*). The table indicates the values and supplementary descriptions of all facets.

Important notes

- The character set permitted in the components needs to be restricted in order that strings formed by concatenating components can be parsed. For the purposes of this scoping exercise, it will be assumed that the components will be used in URLs, punctuated by “/”, “=”, “:”, and “?”, and in the names of files delivered to users, punctuated by “.” and “_”. Thus, none of these characters can be permitted within the component values. Other characters will also be excluded at this time, so the permitted characters will be: a-z, A-Z, 0-9, and “-”. In constructing the “variable name” component of the DRS, it is recommended that the “-“ be avoided since hyphens cannot be imbedded in Fortran and IDL variable names, and some users would like to maintain consistency between the DRS name and the name appearing in their code.
- The `frequency` facet is the temporal frequency of the data. However, many indicators are time invariant. For time-invariant data, the value `fx` must be used. If your indicator is

climatological statistics, use e.g. `yrClim`, `seaClim` or `monClim`. When your data does not fit within these parameters given, please contact us for adding the specific value of your statistics output.

- The dataset identifier ends with the `version` facet that is the dataset version used by ESGF publication. The version will be specified by the data manager as the publishing date on ESGF (*Section 5.1*). Either way, it must always have the precise form `vYYYYMMDD`. Note that `version` facet is different from the `IndicatorRealisation` facet.
- `SourceType` describes the source type of all of the input datasets used in the calculation of indicator. There are fine values selected, which are `gcm-derived`, `rcm-derived`, `reanal-derived`, `obs-derived`, `multi-derived`. In order to find its value, it should be considered that the source type of each input dataset. For example, when one indicator has been calculated from ONLY observation dataset, the value of `SourceType` should be `obs-derived`, no matter one observation dataset has been used or several observation datasets have been used. The same for `gcm-derived`, `rcm-derived`, `reanal-derived`. When one indicator has been calculated from two or more kinds of data source, the value of `SourceType` should always be `multi-derived`.
- `SourceDataID` describes in detail about the input datasets. As the C4E indicators are derived from a variety of sources, the values of `<SourceDataID>` are very different in different cases. The value of `SourceDataID` depends on the types and the numbers of input datasets used. Table 3.1 shows the forms of different possible cases. The explanations of facets are presented in Table D.1 (*Appendix D*). Note that specific characters (e.g. point, slash, etc.) are all prohibited in `SourceDataID`, except dash.

Table 3.1 Forms and values of `SourceType` facet and `SourceDataID` facet

`SourceType` facet value	**Form of `SourceDataID` facet**	**Corresponding cases about source(s) dataset(s)**
`gcm-derived`	`<GCMName>-<ExperimentName>-<EnsembleMember>`	1 GCM
	`multiModel`	multiple GCMs
	`ens-<GCM name>-<ExperimentName>-<method>`	ensemble of several experiments of 1 GCM
	`ens-multiModel-<method>`	ensemble of multiple GCMs
`rcm-derived`	`<GCMName>-<ExperimentName>-<EnsembleMember>-<RCMName>-<RCMRealization>`	1 RCM
	`multiModel`	multiple RCMs
	`ens-<GCMName>-<ExperimentName>-<RCM name>-<RCMRealization>-<method>`	ensemble of several experiments of 1 RCM

	ens-multiModel-<method>	ensemble of multiple RCMs
bc-derived		
reanal-derived	<ReanalName>	1 reanalysis dataset
	multiReanal	multiple reanalysis datasets
	ens-mulfiReanal-<method>	ensemble of multiple reanalysis datasets
obs-derived	<ObsName>	1 observation dataset
	multiObs	multiple observation datasets
	ens-mulfiObs-<method>	ensemble of multiple observation datasets
multi-derived	multiMixed	multiple different sources datasets (at least two of obs, GCM, RCM, reanalysis)
	ens-multiMixed-<method>	ensemble of multiple sources datasets (at least two of obs, GCM, RCM, reanalysis)

Depending on the source type, some SourceDataID value refers to facet value from other DRS. In such a case, the facet value has to be unchanged in respect of the origin DRS. This ensure homogenous facets between different DRS. For instance if an indicator realization has been built from CMIP5 simulation the corresponding SourceDataID should stricly include GCMName, ExperimentName and EnsembleMember value from CMIP5 DRS.

3.2.2. Constructing the DRS

The DRS component vocabularies are used in various places within the C4E archive to identify digital objects, such as directories, file names, and file attributes. In each case there are slight variations in the encoding syntax and subset of DRS components used, reflecting the practicalities of mapping DRS concepts to different applications.

Since C4E indicators may be calculated from a variety of sources, either climate model data, reanalysis data, observational data or from multiple sources the DRS for C4E indicators are non-trivial to construct. However, we formulated a general DRS that will organize and describe all C4E indicators.

Note that the facet names are given between angle brackets “<>”. Optional facet are given between square brackets “[ ]”. Required facets must not be left blank. If you are a data producer, it is obligation for you to provide the filename using the formats in *Section 3.2.2.2*. But it is not mandatory for the data producers to use the standard directory structure (*Section 3.2.2.1*). Data producers can design a simple and proper directory for their indicator files. It is more important for data producers to assure that the names of data files have been well written. Because filename include all the necessary elements for creating its corresponding directory, and this is how the data manager archives the indicator data files.

3.2.2.1. Directory structure

The standard C4E output should be saved to a directory structure mapping DRS components to directory names as:

```
<Activity>/
     <ProjectActivity>/
          <Product>/
               <SourceType>/
                    <SourceDataID>/
                         <IndicatorRealization>/
                              <Frequency>/
                                   <VariableName>/
                                        <Version>/
                                             File1.nc
                                             File2.nc
                                             [...]
```

Note that the `<Version>` facet is set by the data publishers.

3.2.2.2. Filename encoding

The DRS for an indicator filename structure is:

```
<VariableName>_<Activity>_<ProjectActivity>_<Product>_<SourceType>_<SourceDataID>_<IndicatorRealization>_<Domain>_<Frequency>[_<StartTime>-<EndTime>].nc
```

For timeseries data, StartTime and EndTime are mandatory.

For maps data, only the middle time of the statistical period should be given.

For example, suppose one of the wind power capacity factor indicators is named `wpcf85`, this indicator data for historical scale is derived from ERA-Interim reanalysis data. The studied region is Europe. The output frequency is by every 6 hours and the output period for one file is from 1990 to 1999. Its filename should be:

```
wpcf85_cc4e_wp1_historical_reanal-derived_erainterim_r1_europe_6hr_1990010100-1999123118.nc
```

Note that only for the time-invariant data files (i.e. `frequency`="`fx`"). there is neither `<StartTime>` nor `<EndTime>`.

4. Quality control and assurance

Quality assurance and quality control are phrases used to describe activities that prevent errors from entering or staying in a data set. These activities ensure the quality of the data before it is collected, entered, or analyzed, and monitoring and maintaining the quality of data throughout the study, so that data are collected, managed, and utilized with accuracy and precision.

In general, there are two types of errors that can occur in a data set. First, errors of commission are the result of incorrect or inaccurate data being included in the data set. This may happen because of a malfunctioning instrument that produces faulty results, data that are mistyped during entry, or other problems.

Errors of omission are the second type of errors. These result from data or metadata being omitted. Situations that result in omission errors are when data are inadequately documented, when there are human errors during data collection or entry, or when there are anomalies in the field that affect the data.

For C4E data files, there are three phases approach to quality control :

- Before the data collection : data development control
- During data entry : data collection control
- After data entry: data archive management control

4.1. Before the data collection : data development control

In this section, the points that should be taken care of before data producing and during data producing have been presented.

4.1.1. For data manager

When designing the DRS and CVs, the designer should consider as many possible cases of data and the description should be clear enough, so that the data producer can easily find the format demanded and make less errors avoidable. This can reduce the chance of redoing the data.

4.1.2. For data producers

It is suggested strongly that the data producers should communicate with the data manager to discuss the variable name and data file format and attributes before they begin to produce all of the data files, so that the data manager can help to check the data format and the content of data file, and no variable short names are repeated.

It is also suggested that the data producer can send few examples of indicator files to the data manager for an intuitive check of the data format.

4.2. During data entry : data collection control

4.2.1. Manual of operations

After the data files have been created, the data files will be transferred to the IPSL server. The DMP group will be responsible for taking care of the reception of data files. It is necessary to make a specific data collection specifications and procedures in advance. The purpose is to insure a standard way to collect data for the data receiver so that no important step would be forgotten in the data collection process for any indicator. The standard way of data collection should generally include these steps:

1) Record the reception progress (e.g. which indicator, how many files, etc.).
2) Technical check of each data file (*Section 4.2.3*).
3) Documentation of the result of technical check.
4) Correction of data file if any error found.
5) Save the data files under security directory.

4.2.2. Technical check

It is important to check technically the quality of each data files (including the directory structure and file name, metadata of NetCDF, and value of dimensions and variables, etc.). Some scripts and programs are needed by technical check. NetCDF packages and CMOR may be used to check these values.

4.2.2.1. Directory structure and file name

For checking the directory structure and file names, the quality checker can follow the next steps:

1) Read in detail the latest version of specification of indicator for a good comprehension of the values of all of the DRS facets.
2) Check the directory structure that if the value of each element is right or not.
3) Check the data file name that if it has used the good DRS structure, and if the value of each element is right.
4) If there are too many files, scripts are needed for efficiency.

4.2.2.2. NetCDF file metadata

In order to check the metadata of NetCDF file, scripts are necessary and CMOR can be used as an efficient tool. Following the *CC4E-1.0* conventions and DRS facets (Tables C.1 and D.1), the quality checker should be careful that these following questions are important:

1) Is there any global attribute omitted in the data file?
2) Do the values of all of the global attributes are good?
3) Does the indicator variable has good dimension as specified in its specification?
4) Do the values of all of the dimensions, variables, auxiliary variables or subsidiary variable are right?

All the above points should comply well with the *CC4E-1.0* conventions and DRS as described in *Sections 2 and 3*.

4.2.2.3. Axis values

About the axis values, the possible errors might be:

- for time axis: repeated time; wrong time interval, wrong time format, etc.
- for a discrete axis: wrong format of string (e.g. non-lowercase of letter, forbidden characters, etc.), not attached as a coordinate to the indicator, miss of `cell_measure` variable, etc.

4.2.2.4. Statistic check for variables

It may be a heavy job if we check the value of each grid of all of the variables. So it could be much more efficient by using some statistical methods, such as:

- maximum, minimum, median
- mean
- distribution of probability
- temporal variation

4.2.3. Documentation

4.2.3.1. Documentation of technique check

It is demanded to document all of the methods, processes, and conclusions of technique check. The document of technique check should be well written and easily be readable. In addition, all of these documents should be well organised and be archived together.

4.2.3.2. Logfile record

Using simple logfile to record all the processes that have done for quality control.

4.2.4. Data correction process

If an error appears in the directory structure, the file names or the metadata attributes of NetCDF file, the data manager should inform the data producers to confirm this error. After the confirmation of error from the data producer, data manager and data producers can decide together that who will correct this error. If it is a simple error, data manager can directly finish the correction.

However, if an error appears in the values of dimensions or variables, it is necessary to ask data producer to correct the error.

Note that after correction, the version of data file should be changed (*Section 5.1*).

4.3. After data entry: data archive management control

4.3.1. Security

After the collection of data files, it is important to assure that the files are saved under a security directory. It means that it is necessary to control who can do what with data within the database.

The IPSL data manager only will have write access to the data files.

4.3.2. Distribution of responsibility

The responsibilities and authorities of DMP group members should be clear declared so that all of the steps during data collection and about data management can be assured completely.

For example,

- Data manager:
  - responsibility: reception of data; quality checks.
  - authority: controls global data flow; ability to add and delete records.
- Data publisher:
  - responsibility: ESGF publication; errata updating.
  - authority: versioning management.

5. Versioning and data traceability

5.1. Versioning details

Versioning is the basement of data traceability. Due to the inherent complexity of the C4E protocol, it is important to **record and track the reasons of a new version.** The dataset version is an important metadata to relate with data modifications/corrections that might impact the scientific analysis. **Any kind of error or mistake or data modification leads to a version change of a dataset and, conversely, any change of a dataset (values, metadata, etc.) is justified by an issue.** The goal is to define and to establish a stable and coordinated C4E procedure to collect and give access to errata information related to datasets hosted by ESGF.

A data modification/correction can be suggested by anyone use the data archive. This modification has to be reported to the data provider using one of the following alternatives:

- The email of the appropriate data producer,
- An esg-issue mailing list (e.g., esgf-issues@lists.llnl.gov).

Consequently, all data providers have to be clearly identified and updated into the file metadata. The data provider evaluates the relevance of the issue by directly investigating the pointed data and checks if no issue exists with the same topic.

Then, the data provider has to perform a similar procedure as *Section 4.2.4:*

A. Identify the corresponding publication unit(s) to revise,
B. Document the issue by building a JSON issue template compliant with the ES-DOC Errata Service (see sample into *Table E.1*),
C. Produce corrected data files (DRS-formatted) according to the issue,
D. Request the data manager for publication by sending the new publication unit(s) and the issue JSON file. Note that **the dataset level is the finest publication units that can be received.**

The data manager will then apply the versioning procedure as follows:

A. Check the existing version history of the submitted unit(s). If no previous version exists, the versioning is initialized. Otherwise, the data manager ensures that this exact collection of files has not already been published under a version of this dataset anywhere in ESGF (i.e., master + replicas). **The same collection of files should not be published with two different versions.**
B. Manage the storage on the IPSL server following the directory format (*Section* *3.2.2.1*). The data manager determines the new version string in agreed format and builds the corresponding directory structure.

5.2. File traceability

Each file metadata includes a `tracking_id` attribute. This tracking ID is a UUID auto-generated during DRS formatting. A unique tracking ID will be attached to each file and will remain unchanged for the whole data lifecycle.

In addition, for each dataset to be published, the ESGF publisher generates a dataset Persistent IDentifier (PID) and initiates the PID registration. The dataset PID are sent and definitely persisted with metadata (dataset ID(s), checksums, version(s), issue(s) ID(s), etc.) into the ESGF Handle Service hosted by DKRZ. The PID registration is decoupled from the publication process (i.e., the publication process will not be stopped even if the Handle Service is unreachable. In this case the PID will be registered later).

This PID Service implies that each file tracking IDs are superseded by PIDs. Consequently, the use of PIDs instead of tracking IDs should be discussed to benefit the whole PID Service and interactions (such as errata see *Section 5.3*).

5.3. Errata

The IPSL is finalizing an new ESGF Errata Service through the Earth System Documentation (ES-DOC - http://es-doc.org/) in order to:

- Provide timely information about known issues. Within the ES-DOC ecosystem, the errata web-service front-end display the whole list of known issues. The list can be filtered by several useful parameter as the issue severity or status. Three tabs describe each issue providing (i) the information details, (ii) graphics or pictures to illustrate the issue, and (iii) the list of the affected dataset.
- Allow identified and authorized actors to create, update and close an issue. We developed a piece of software that enables the interaction with the errata service. It can be used to create, update, close and retrieve issues. The client is basically aimed to be used by publishing teams, so that they can directly describe problems when they are discovered.
- Enable users to query about modifications and/or corrections applied to the data in different ways. The errata web-service provide an API to query the issue database. The end users can submit one or several files or datasets identifiers to get back all annotations related to each corresponding issue. This search API is also able to retrieve

the issues that affect a MIP variable or experiment.

To succeed the Errata Service exploits the Persistent IDentifier (PID) attached to each dataset during the ESGF publication process. The PIDs enable to request the version history of a (set of) file/dataset(s) even for unpublished data. Consequently, the use of PIDs should be discussed to benefit all the search features provided by the Errata Service.

5.4. Tools and guidelines

- Do not publish dataset without version number,
- Do not publish the same file with two version numbers,
- **The version format must be vYYYYMMDD** (as the publication date of the dataset),
- **No version update will be published without the corresponding a issue JSON template,**
- The use of the `esgprep drs` command-line is recommended to manage directory contents between versions using symlinks to save disk space. A symlink can be set to point to the "latest" version and clarify the data management.

6. Data dissemination and ESGF publication

6.1. ESGF publication

To provide access to the C4E indicators, the IPSL will handle the publication of data on the Earth System Grid Federation (ESGF). The IPSL hosts an ESGF node able to serve C4E data. Such a node lists all available data (and metadata) for the platform through metadata catalogs that defines access protocols to data endpoints. These catalogs are indexed and broadcasted to the whole federation. The user can then reach any ESGF front-end to get the data using the search interface.

6.1.1. Publication stakeholders

The ESGF publication workflow includes many steps. It is useful to define who does what from the indicators data production to the availability of NetCDF files on ESGF front-ends. It is intended that:

A. **The data producers** deals with the data production: i.e., raw production, DRS formatting, corrections, issues, transfer.
B. **The data manager (IPSL)** ensures the readiness of data files for publication with several checks (i.e., controlled vocabulary, data quality, etc.); stores the formatted data on the file system using the appropriate directory tree and versioning; is in charge of publication actions through the ESGF publisher.

6.1.2. Publication workflow

The publication workflow should follow the best practices recommended by the ESGF Publication Sprint Report[4].

A. All data files included into a dataset requested to be publish has to be CC4E-1.0 compliant. **The data manager will notifies the readiness for publication to the data producer if all quality checks have succeeded** (i.e., controlled vocabulary, axis squareness, etc.).
B. At the end of the versioning process (*Section 5.1*), the data node Manager generates the *mapfile* using the `esgprep mapfile` command-line tool. The mapfile has to embed the version number(s) and becomes the key by-product of the publication process. We will only mapfiles as input to the ESGF publisher. **We will produce one mapfile per publication unit.**
C. The data node manager runs the publication process following the ESGF publication best practices[5]. The first publication step ingests the dataset and files metadata into a PostgreSQL database of the ESGF node and generates the XML THREDDS catalogs reflecting the version change (initial version, new version, version retraction or removal). Finally the Solr index ingests the corresponding metadata by pulling and parsing the THREDDS catalogs.

6.1.3. Unpublication

Several reasons could lead to a dataset removal or retraction from ESGF (e.g., data locally deleted, storage limit, critical issue avoiding the use of indicator data, etc.). In respect of the CC4E traceability policy (*Section 5*):

- Dataset unpublication from ESGF may be conducted only by the data manager,
- Local data files removal may be conducted only by the data manager,
- The metadata of removed of retracted dataset versions will be persisted into SolR and the Handle Service (if the PIDs are finally used).

6.1.4. INI configuration file

Most of the versioning and publication steps requires the same configuration file called `esg.<project>.ini`. This INI file declares the facet sequences used by the directory format, the dataset identifier and all the allowed values for each facet. It allows us to ensure an homogeneous DRS management at each step of the publication workflow.

The data manager will provide such a file for CC4E, called `esg.cc4e.ini`, following the ESGF INI anatomy[6].

6.1.5. CoG configuration

The IPSL ESGF node will serve C4E datasets through a front-end called “CoG” (https://esgf-node.ipsl.upmc.fr/). A specific project webpage will be build to present a summary for C4E project, contacts, references and links to the Climate Data Store, the Copernicus Climate Change Service and the C4E website. This page will be reachable at https://esgf-node.ipsl.upmc.fr/projects/cc4e-ipsl/.

The corresponding Data Search interface will be configured according to the DRS (Section 2.2). The IPSL will provide a `search.cfg` which joins each DRS facet with its CoG label.

6.2. Data dissemination outside of ESGF

The THREDDS Data Server used by IPSL node can be used to serve other websites or to directly provide data access URLs. The access protocols initially available are:

- HTTP
- OpenDAP
- NetCDFSubset
- WCS (Web Coverage Service)

In the same way as CoG front-end, the IPSL THREDDS server can be used to serve other entry points for C4E data access as the C4E website and its visualization system and the Climate Data Store.

Appendix A: Naming conventions for WP3

Table A.1 Naming conventions for climate-projection of WP3

	Prefix		Root	Suffix	
	Absolute / Relative change / Anomaly	Nuts0 / Nuts2	Keyword for each indicator	Full winter / Cold events	Statistics

Maps	if absolute: []
	if relative change: [r]
	if anomaly: [a]
		if Nuts0: [n]
		if Nuts2: [r]
			if Demande estimated with degree day: [whdd]
			if Demande estimated with consumption models: [eca]
			if Wind capacity factor: [wcf1] [wcf2] [wcf3]
			if Solar capacity factor: [pvcf1] [pvcf2] [pvcf3]
				if full winter: []
				if cold events: [ce]
					if mean: [avg]
					if P5: [p5]
					if P95: [p95]
					if P25: [p25]
					if P75: [p75]

Timeseries				-	-

Appendix B: Standardized Region Names

Table B.1 was based on the NASA GCMD keyword list for locations - Version 8.4.1 [3] which was revised on 11 May 2016. We retained only the names of European regions from the GCMD list, changing them to lowercase and replacing the separators within the names by **dash** to be consistent with the style used for C4E standard names. We have selected the keywords which are possibly needed by C4E indicators. Our intention is to keep this list consistent with GCMD.

Please contact to us if the keywords that needed by your indicator files are not included in these lists (please find our contacts in *Section 1*).

Table B.1 Keywords about countries and geographic regions in Europe

aland-islands
albania
andorra
austria
belarus
belgium
black-sea
bosnia-and-herzegovina
british-isles
bulgaria
byelorussian-ssr
central-europe
channel-islands
croatia
cyprus
czech-republic
denmark
eastern-europe
estonia
europe
finland
france
germany
gibraltar
greece
hungary
iceland
ireland
irish-sea
italy
kosovo
latvia
liechtenstein
lithuania
luxembourg
macedonia
malta
mediterranean-sea
moldova
monaco
montenegro
netherlands
northern-europe
norwegian-sea
north-sea
norway
poland
portugal
romania
russian-federation
san-marino
scandinavia
serbia
slovakia
slovenia
southern-europe
spain
sweden
switzerland
ukraine
united-kingdom
vatican-city
western-europe
northwest-european-shelf

Table B.2 is the list of NUTS-2 basic regions for the application of basic policies. We have translated all the NUTS-2 region names from their own language to English, changing them to lowercase and replacing the separators within the names by underscores. The purpose is to make sure that there will be no problem in the display by any system, and to make the format be consistent with the style used for C4E standard names. Most of the English name of regions are found from the internet. Some regions do not have a corresponding English name. In this case, the original names are used by only replacing their non-English letters with English letters (a-z) and dash ("-") (e.g. Tübingen - tubingen, Åland - aland, Provence-Alpes-Côte d'Azur - provence-alpes-cote-d-azur, Småland and the islands - smaland-and-the-islands, etc.). Please contact us if you have any suggestions about the English name of these regions.

Table B.2 Label names of NUTS-2 basic regions

ID of NUTS-2 and standard region Name	Label (nutsid)	Lable (nuts2_region)
AT11 - Burgenland (A)	AT11	burgenland-a
AT12 - Niederösterreich	AT12	lower-austria
AT13 - Wien	AT13	vienna
AT21 - Kärnten	AT21	carinthia
AT22 - Steiermark	AT22	styria
AT31 - Oberösterreich	AT31	upper-austria
AT32 - Salzburg	AT32	salzburg
AT33 - Tirol	AT33	tirol
AT34 - Vorarlberg	AT34	vorarlberg
BE10 - Région de Bruxelles-Capitale / Brussels Hoofdstedelijk Gewest	BE10	brussels-capital-region
BE21 - Prov. Antwerpen	BE21	prov-antwerp
BE22 - Prov. Limburg (B)	BE22	prov-limburg-b
BE23 - Prov. Oost-Vlaanderen	BE23	prov-east-flanders
BE24 - Prov. Vlaams-Brabant	BE24	prov-vlaams-brabant
BE25 - Prov. West-Vlaanderen	BE25	prov-west-flanders
BE31 - Prov. Brabant Wallon	BE31	prov-brabant-wallon
BE32 - Prov. Hainaut	BE32	prov-hainaut
BE33 - Prov. Liège	BE33	prov-cork
BE34 - Prov. Luxembourg (B)	BE34	prov-luxembourg-b
BE35 - Prov. Namur	BE35	prov-namur
BG31 - Северозападен / Severozapaden	BG31	severozapaden
BG32 - Северен централен / Severen tsentralen	BG32	severen-tsentralen
BG33 - Североизточен / Severoiztochen	BG33	severoiztochen
BG34 - Югоизточен / Yugoiztochen	BG34	yugoiztochen
BG41 - Югозападен / Yugozapaden	BG41	yugozapaden
BG42 - Южен централен / Yuzhen tsentralen	BG42	yuzhen-tsentralen
CH01 - Région lémanique	CH01	lake-geneva-region
CH02 - Espace Mittelland	CH02	espace-mittelland
CH03 - Nordwestschweiz	CH03	northwestern-switzerland
CH04 - Zürich	CH04	zurich
CH05 - Ostschweiz	CH05	eastern-switzerland
CH06 - Zentralschweiz	CH06	central-switzerland
CH07 - Ticino	CH07	ticino

CZ01 - Praha	CZ01	prague
CZ02 - Střední Čechy	CZ02	middle-bohemia
CZ03 - Jihozápad	CZ03	southwest
CZ04 - Severozápad	CZ04	northwest
CZ05 - Severovýchod	CZ05	northeast
CZ06 - Jihovýchod	CZ06	southeast
CZ07 - Střední Morava	CZ07	central-moravia
CZ08 - Moravskoslezsko	CZ08	silesia
DE11 - Stuttgart	DE11	stuttgart
DE12 - Karlsruhe	DE12	karlsruhe
DE13 - Freiburg	DE13	freiburg
DE14 - Tübingen	DE14	tubingen
DE21 - Oberbayern	DE21	upper-bavaria
DE22 - Niederbayern	DE22	lower-bavaria
DE23 - Oberpfalz	DE23	upper-palatinate
DE24 - Oberfranken	DE24	upper-franconia
DE25 - Mittelfranken	DE25	middle-franconia
DE26 - Unterfranken	DE26	lower-franconia
DE27 - Schwaben	DE27	swabia
DE30 - Berlin	DE30	berlin
DE40 - Brandenburg	DE40	brandenburg
DE50 - Bremen	DE50	bremen
DE60 - Hamburg	DE60	hamburg
DE71 - Darmstadt	DE71	darmstadt
DE72 - Gießen	DE72	giessen
DE73 - Kassel	DE73	kassel
DE80 - Mecklenburg-Vorpommern	DE80	mecklenburg-vorpommern
DE91 - Braunschweig	DE91	braunschweig
DE92 - Hannover	DE92	hannover
DE93 - Lüneburg	DE93	lunenburg
DE94 - Weser-Ems	DE94	weser-ems
DEA1 - Düsseldorf	DEA1	duesseldorf
DEA2 - Köln	DEA2	cologne
DEA3 - Münster	DEA3	muenster
DEA4 - Detmold	DEA4	detmold
DEA5 - Arnsberg	DEA5	arnsberg
DEB1 - Koblenz	DEB1	koblenz
DEB2 - Trier	DEB2	trier
DEB3 - Rheinhessen-Pfalz	DEB3	rheinhessen-pfalz

DEC0 - Saarland	DEC0	saarland
DED2 - Dresden	DED2	dresden
DED4 - Chemnitz	DED4	chemnitz
DED5 - Leipzig	DED5	leipzig
DEE0 - Sachsen-Anhalt	DEE0	saxony-anhalt
DEF0 - Schleswig-Holstein	DEF0	schleswig-holstein
DEG0 - Thüringen	DEG0	thuringia
DK01 - Hovedstaden	DK01	capital
DK02 - Sjælland	DK02	zealand
DK03 - Syddanmark	DK03	denmark
DK04 - Midtjylland	DK04	jutland
DK05 - Nordjylland	DK05	north-jutland
EL30 - Αττική / Attiki	EL30	attica
EL41 - Βόρειο Αιγαίο / Voreio Aigaio	EL41	north-aegean
EL42 - Νότιο Αιγαίο / Notio Aigaio	EL42	south-aegean
EL43 - Κρήτη / Kriti	EL43	crete
EL51 - Anatoliki Makedonia	EL51	eastern-macedonia-and-thrace
EL52 - Kentriki Makedonia	EL52	central-macedonia
EL53 - Dytiki Makedonia	EL53	western-macedonia
EL54 - Ipeiros	EL54	epirus
EL61 - Thessalia	EL61	thessaly
EL62 - Ionia Nisia	EL62	ionian-islands
EL63 - Dytiki Ellada	EL63	western-greece
EL64 - Sterea Ellada	EL64	central-greece
EL65 - Peloponnisos	EL65	peloponnese
ES11 - Galicia	ES11	galicia
ES12 - Principado de Asturias	ES12	asturias
ES13 - Cantabria	ES13	cantabria
ES21 - País Vasco	ES21	basque-country
ES22 - Comunidad Foral de Navarra	ES22	navarre
ES23 - La Rioja	ES23	la-rioja
ES24 - Aragón	ES24	aragon
ES30 - Comunidad de Madrid	ES30	madrid
ES41 - Castilla y León	ES41	castilla-y-leon
ES42 - Castilla-La Mancha	ES42	castilla-la-mancha
ES43 - Extremadura	ES43	extremadura
ES51 - Cataluña	ES51	catalonia
ES52 - Comunidad Valenciana	ES52	valencia
ES53 - Illes Balears	ES53	illes-balears

ES61 - Andalucía	ES61	andalusia
ES62 - Región de Murcia	ES62	region-of-murcia
FI19 - Länsi-Suomi	FI19	western-finland
FI1B - Helsinki-Uusimaa	FI1B	helsinki-uusimaa
FI1C - Etelä-Suomi	FI1C	southern-finland
FI1D - Pohjois- ja Itä-Suomi	FI1D	northern-and-eastern-finland
FI20 - Åland	FI20	aland
FR10 - Île de France	FR10	ile-de-france
FR21 - Champagne-Ardenne	FR21	champagne-ardenne
FR22 - Picardie	FR22	picardie
FR23 - Haute-Normandie	FR23	upper-normandy
FR24 - Centre	FR24	center
FR25 - Basse-Normandie	FR25	lower-normandy
FR26 - Bourgogne	FR26	burgundy
FR30 - Nord - Pas-de-Calais	FR30	nord-pas-de-calais
FR41 - Lorraine	FR41	lorraine
FR42 - Alsace	FR42	alsace
FR43 - Franche-Comté	FR43	franche-comte
FR51 - Pays de la Loire	FR51	pays-de-la-loire
FR52 - Bretagne	FR52	brittany
FR53 - Poitou-Charentes	FR53	poitou-charentes
FR61 - Aquitaine	FR61	aquitaine
FR62 - Midi-Pyrénées	FR62	midi-pyrenees
FR63 - Limousin	FR63	limousin
FR71 - Rhône-Alpes	FR71	rhone-alpes
FR72 - Auvergne	FR72	auvergne
FR81 - Languedoc-Roussillon	FR81	languedoc-roussillon
FR82 - Provence-Alpes-Côte d'Azur	FR82	provence-alpes-cote-d’azur
FR83 - Corse	FR83	corsica
HR03 - Jadranska Hrvatska	HR03	adriatic-croatia
HR04 - Kontinentalna Hrvatska	HR04	continental-croatia
HU10 - Közép-Magyarország	HU10	central-hungary
HU21 - Közép-Dunántúl	HU21	central-transdanubia
HU22 - Nyugat-Dunántúl	HU22	western-transdanubia
HU23 - Dél-Dunántúl	HU23	southern-transdanubia
HU31 - Észak-Magyarország	HU31	northern-hungary
HU32 - Észak-Alföld	HU32	northern-great-plain
HU33 - Dél-Alföld	HU33	southern-great-plain
IE01 - Border, Midland and Western	IE01	border-midland-and-western

IE02 - Southern and Eastern	IE02	southern-and-eastern
ITC1 - Piemonte	ITC1	piemonte
ITC2 - Valle d'Aosta/Vallée d'Aoste	ITC2	aosta-valley
ITC3 - Liguria	ITC3	liguria
ITC4 - Lombardia	ITC4	lombardy
ITF1 - Abruzzo	ITF1	abruzzo
ITF2 - Molise	ITF2	molise
ITF3 - Campania	ITF3	campania
ITF4 - Puglia	ITF4	puglia
ITF5 - Basilicata	ITF5	basilicata
ITF6 - Calabria	ITF6	calabria
ITG1 - Sicilia	ITG1	sicily
ITG2 - Sardegna	ITG2	sardinia
ITH1 - Provincia Autonoma di Bolzano/Bozen	ITH1	bolzano
ITH2 - Provincia Autonoma di Trento	ITH2	trento
ITH3 - Veneto	ITH3	veneto
ITH4 - Friuli-Venezia Giulia	ITH4	friuli-venezia-giulia
ITH5 - Emilia-Romagna	ITH5	emilia-romagna
ITI1 - Toscana	ITI1	tuscany
ITI2 - Umbria	ITI2	umbria
ITI3 - Marche	ITI3	marche
ITI4 - Lazio	ITI4	lazio
NL11 - Groningen	NL11	groningen
NL12 - Friesland (NL)	NL12	friesland-nl
NL13 - Drenthe	NL13	drenthe
NL21 - Overijssel	NL21	overijssel
NL22 - Gelderland	NL22	gelderland
NL23 - Flevoland	NL23	flevoland
NL31 - Utrecht	NL31	utrecht
NL32 - Noord-Holland	NL32	north-holland
NL33 - Zuid-Holland	NL33	south-holland
NL34 - Zeeland	NL34	zeeland
NL41 - Noord-Brabant	NL41	north-brabant
NL42 - Limburg (NL)	NL42	limburg-nl
NO01 - Oslo og Akershus	NO01	oslo-and-akershus
NO02 - Hedmark og Oppland	NO02	hedmark-and-oppland
NO03 - Sør-Østlandet	NO03	south-eastern-norway
NO04 - Agder og Rogaland	NO04	agder-and-rogaland
NO05 - Vestlandet	NO05	vestlandet

NO06 - Trøndelag	NO06	trondelag
NO07 - Nord-Norge	NO07	northern-norway
PL11 - Łódzkie	PL11	lodz
PL12 - Mazowieckie	PL12	mazowieckie
PL21 - Małopolskie	PL21	malopolska
PL22 - Śląskie	PL22	silesian
PL31 - Lubelskie	PL31	lublin
PL32 - Podkarpackie	PL32	podkarpackie
PL33 - Świętokrzyskie	PL33	swietokrzyskie
PL34 - Podlaskie	PL34	podlaskie
PL41 - Wielkopolskie	PL41	greater-poland-voivodeship
PL42 - Zachodniopomorskie	PL42	west-pomeranian-voivodeship
PL43 - Lubuskie	PL43	lubuskie
PL51 - Dolnośląskie	PL51	lower-silesia
PL52 - Opolskie	PL52	opole
PL61 - Kujawsko-Pomorskie	PL61	kujawsko-pomorskie
PL62 - Warmińsko-Mazurskie	PL62	warmia-mazury
PL63 - Pomorskie	PL63	pomorskie
PT11 - Norte	PT11	north
PT15 - Algarve	PT15	algarve
PT16 - Centro (P)	PT16	center-p
PT17 - Lisboa	PT17	lisbon
PT18 - Alentejo	PT18	alentejo
RO11 - Nord-Vest	RO11	northwest
RO12 - Centru	RO12	centre
RO21 - Nord-Est	RO21	northeast
RO22 - Sud-Est	RO22	southeast
RO31 - Sud - Muntenia	RO31	sud-muntenia
RO32 - Bucureşti - Ilfov	RO32	bucharest-ilfov
RO41 - Sud-Vest Oltenia	RO41	south-west-oltenia
RO42 - Vest	RO42	vest
SE11 - Stockholm	SE11	stockholm
SE12 - Östra Mellansverige	SE12	east-central-sweden
SE21 - Småland med öarna	SE21	smaland-and-the-islands
SE22 - Sydsverige	SE22	southern-sweden
SE23 - Västsverige	SE23	western-sweden
SE31 - Norra Mellansverige	SE31	north-central-sweden
SE32 - Mellersta Norrland	SE32	central-norrland
SE33 - Övre Norrland	SE33	upper-norrland

SI03 - Vzhodna Slovenija	SI03	eastern-slovenia
SI04 - Zahodna Slovenija	SI04	western-slovenia
SK01 - Bratislavský kraj	SK01	bratislava-region
SK02 - Západné Slovensko	SK02	western-slovakia
SK03 - Stredné Slovensko	SK03	central-slovakia
SK04 - Východné Slovensko	SK04	eastern-slovakia
UKC1 - Tees Valley and Durham	UKC1	tees-valley-and-durham
UKC2 - Northumberland and Tyne and Wear	UKC2	northumberland-and-tyne-and-wear
UKD1 - Cumbria	UKD1	cumbria
UKD3 - Greater Manchester	UKD3	greater-manchester
UKD4 - Lancashire	UKD4	lancashire
UKD6 - Cheshire	UKD6	cheshire
UKD7 - Merseyside	UKD7	merseyside
UKE1 - East Yorkshire and Northern Lincolnshire	UKE1	east-yorkshire-and-northern-lincolnshire
UKE2 - North Yorkshire	UKE2	north-yorkshire
UKE3 - South Yorkshire	UKE3	south-yorkshire
UKE4 - West Yorkshire	UKE4	west-yorkshire
UKF1 - Derbyshire and Nottinghamshire	UKF1	derbyshire-and-nottinghamshire
UKF2 - Leicestershire, Rutland and Northamptonshire	UKF2	leicestershire-rutland-and-northamptonshire
UKF3 - Lincolnshire	UKF3	lincolnshire
UKG1 - Herefordshire, Worcestershire and Warwickshire	UKG1	herefordshire-worcestershire-and-warwickshire
UKG2 - Shropshire and Staffordshire	UKG2	shropshire-and-staffordshire
UKG3 - West Midlands	UKG3	west-midlands
UKH1 - East Anglia	UKH1	east-anglia
UKH2 - Bedfordshire and Hertfordshire	UKH2	bedfordshire-and-hertfordshire
UKH3 - Essex	UKH3	essex
UKI3 - Inner London - West	UKI3	inner-london-west
UKI4 - Inner London - East	UKI4	inner-london-east
UKI5 - Outer London - East and North East	UKI5	outer-london-east-and-north-east
UKI6 - Outer London - South	UKI6	outer-london -south
UKI7 - Outer London - West and North West	UKI7	outer-london-west-and-north-west
UKJ1 - Berkshire, Buckinghamshire and Oxfordshire	UKJ1	berkshire-buckinghamshire-and-oxfordshire
UKJ2 - Surrey, East and West Sussex	UKJ2	surrey-east-and-west-sussex
UKJ3 - Hampshire and Isle of Wight	UKJ3	hampshire-and-Isle-of-wight
UKJ4 - Kent	UKJ4	kent
UKK1 - Gloucestershire, Wiltshire and Bristol/Bath area	UKK1	gloucestershire-wiltshire-and-bristol
UKK2 - Dorset and Somerset	UKK2	dorset-and-somerset

UKK3 - Cornwall and Isles of Scilly	UKK3	cornwall-and-isles-of-scilly
UKK4 - Devon	UKK4	devon
UKL1 - West Wales and The Valleys	UKL1	west-wales-and-the-valleys
UKL2 - East Wales	UKL2	east-wales
UKM2 - Eastern Scotland	UKM2	eastern-scotland
UKM3 - South Western Scotland	UKM3	south-western-scotland
UKM5 - North Eastern Scotland	UKM5	north-eastern-scotland
UKM6 - Highlands and Islands	UKM6	highlands-and-islands
UKN0 - Northern Ireland	UKN0	northern-ireland

Appendix C: Global Metadata Attributes for C4E indicators

Table C.1. NetCDF Global Metadata Attributes for the data files published by Clim4Energy

Appendix D: C4E indicators DRS facets

Table D.1. Clim4Energy Data Reference Syntax facets

Appendix E: Versioning

Table E.1. Issue JSON sample

```
{
    "description": "This is a test description, void of meaning.",
    "institute": "wp3",
    "materials": [
        "http://www.sample.fr/images_errata/1.jpg",
        "http://www.sample.fr/images_errata/2.jpg",
    ],
    "project": "cc4e",
    "severity": "medium",
    "title": "Test issue title",
    "url": "http://websitetest.com/"
}
```

Appendix F: Revisions to this document

30/04/2017 : Revision about 2.1.1 Naming conventions.
30/04/2017 : Revision about Table C.1 and Table D1.

References

[1] CF Conventions (http://cfconventions.org/cf-conventions/v1.6.0/cf-conventions.pdf)
[2] CF standardized region names (http://cfconventions.org/Data/cf-standard-names/docs/standardized-region-names.html)
[3] GCMD's Science Keywords and Associated Directory Keywords: Locations (http://gcmdservices.gsfc.nasa.gov/static/kms/locations/locations.csv)
[4] Publication Sprint Report (https://docs.google.com/document/d/1YrEJD0chwk_jKgeL6lYKuHmWHl8Vjf9dhVjsX_lRCEA/edit?usp=sharing)
[5] ESGF Publication Best Practices (https://acme-climate.atlassian.net/wiki/display/ESGF/Guide+to+ESGF+Publishing+and+Best+Practices)
[6] ESGF esg.ini anatomy (https://acme-climate.atlassian.net/wiki/display/ESGF/More+on+ESGF+publishing)