SPECS common data repository:
File content and format, data structure and metadata

September 2014
Pierre-Antoine Bretonnière, IC3

Overview

Past experience in CMIP5 and other model intercomparison projects has shown that homogeneous formatting, writing and archiving of the models output is a key aspect of the success of a project. Considering the large amount of data expected to be produced in the SPECS project, it is fundamental that before the data are produced, all partners agree on a common data convention in terms of format, data structure for the storage, file content and metadata.

This document aims at defining all the conventions and requirements for the SPECS common data repository. First, it will describe the data format and structure, then the naming and archiving conventions with all the "official definitions" of the requested keywords. The third part of this document describes the global attributes the NetCDF files have to include, followed by a short description of the experiment names planned in SPECS.

Because the SPECS data requirements have much in common with the earlier CMIP5 requirements, this document includes many paragraphs simply copied from two earlier CMIP5 documents:

Taylor, K.E., and C. Doutriaux: "CMIP5 Model Output Requirements: File Contents and Format, Data Structure and Metadata," http://cmip-pcmdi.llnl.gov/cmip5/docs/CMIP5_output_metadata_requirements.pdf, 2011.

Taylor, K.E., V. Balaji, S. Hankin, M. Juckes, B. Lawrence, and S. Pascoe: CMIP5 Data Reference Syntax (DRS) and Controlled Vocabularies," http://cmip-pcmdi.llnl.gov/cmip5/docs/cmip5_data_reference_syntax.pdf, 2012.

Anyone who has already contributed model output to the CMIP5 archive should find the font colors below to be helpful in determining what changes in post-processing are necessary to prepare data for SPECS. The colors have the following meanings:

- SPECS and CMIP5 requirements are consistent.
- SPECS requirements are inconsistent with or have been updated from CMIP5.
- SPECS calls for a new requirement.

Unless otherwise noted, all relevant CMIP5 specifications apply also to SPECS.

I Data format, data structure and file content requirements

A brief list of requirements is provided in this section. It builds upon, as most of the remaining document, the CMIP5 and CORDEX requirements, although the CHFP standards have also been taken into account.

- Data must be written in the netCDF-4 format and conform to the CF metadata standards. The output must be readable through the netCDF-4 API (application program interface).
- Each file must contain only a single field from a single simulation (i.e., a single run, a

member of a prediction). Each file will also include coordinate variables, attributes and other metadata as specified below. If the field 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, typical of coupled model simulations (decadal predictions for example), will usually be split into several files, which should neither be too large (to be unwieldy) nor too small (as to create vexing I/O performance issues).

- All fields, especially those that are a function of the vertical coordinate, should usually be reported on the native grid.

II Directory tree and names of files

1) Directory tree

The SPECS database, will comprise output from many different models and experiments sampled at different frequency (6hour, daily, monthly). Files will be stored using the following directory structure, where definitions of the different elements follows below (although the archive should handle the job of placing them in the correct directory structure):

<model_id>/<experiment_family>/<start_date>/<frequency>/<modeling_realm>/<variable_name>/<ensemble_member> /<version>/

Example: EC-Earth2/seaIceInit/S19910501/day/seaIce/sic/r1i1p1/v20100323/

Model_id: Model names should be decided by the institution in consultation by the data coordinator. They should be a string containing an acronym that identifies the model used to generate the output. It should be as short as possible, so that it can be used, for example, in labelling curves on multi- model plots. The acronym may include the acronym of the modelling centre and the model name/version separated by a hyphen (e.g., "IPSL-CM4"), although omitting the modelling centre is acceptable. Please note that a partner might in the future want to submit results from a successor to the present model, so if appropriate, the model version should be indicated, but keeping it simple e.g., CCSM4, not CCSM4.1.2. Full information about the version will appear in the "source" global attribute.

Experiment_family: Short name of the experiment family. By experiment family is meant the name of the experiment, but without the start date, to avoid hugely complicated names. The official short names of the experiments can be found in the experiment section of the *SPECS wiki*.

Frequency: A string indicating the interval between individual time-samples. The following are the only options: "mon", "day", "6hr" or "fx" (fixed, i.e., time-independent). Output frequency requirements for each variable can be found in the data section of the *SPECS wiki*.

Modeling_realm: A string that indicates the high-level modelling component that is particularly relevant to the variable encoded. For SPECS, permitted values are: "atmos", "ocean", "land", "landIce", "seaIce", "aerosol", "atmosChem", or "ocnBgchem" (ocean biogeochemical).

Variable_name: Short name of the variable, following the official name collected in the table and in the data section of the SPECS wiki.

Ensemble_member: The triad of integers r<N>i<M>p<L> or (N, M, L), formatted as shown in the first option (e.g., "r3i1p21"), distinguishes among closely related simulations by a single system.

All three are required even if only a single simulation is performed. The so-called "realization" number (a positive integer value of "N") is used to distinguish among members of an ensemble typically generated by initializing a set of runs with different, but equally realistic, initial conditions.
Historical runs initialized from different times of a control run, for example, would be identified by "r1", "r2", "r3", etc.). The data supplier must assign a realization number to each atomic dataset. It is generally recommended that the numbers be assigned sequentially starting with 1 (but other recommendations, specified below, may override this recommendation). Time-independent variables (i.e., those with frequency="fx") are not expected to differ across ensemble members, so for these N should be invariably assigned the value zero ("r0").
Forecasts might be initialized from observations using different methods or different observational datasets. These should be distinguished by assigning different positive integer values of "M" in the "initialization method indicator" (i<M>). The data supplier must assign an initialization method number to each atomic dataset. It is recommended that the numbers be assigned sequentially starting with 1. Time-independent variables (i.e., those with frequency="fx") are not expected to differ across ensemble members, so for these M should invariably be assigned the value zero ("i0"). Uninitialized simulations, such as the historical runs used in decadal prediction, should also use "i0" because they are not initialized.
If there are many closely related model versions that, as a group, are generally referred to as a perturbed physics ensemble (e.g., QUMP or climateprediction.net ensembles), these should be distinguishable by a "perturbed physics" number, p<L>, where the positive integer value of L is uniquely associated with a particular set of model parameters (e.g., r3i1p78 is a third realization of the seventy-eighth version of the perturbed physics model). The standard value should be "p1".

Start_date: Start date of the experiment in the form of "Syyyymmdd". For uninitialized runs, a start date will still need to be used. This date corresponds to the beginning of the historical run from the parent experiment (e.g., the pre-industrial simulation). We refer to the date when a historical run starts from the spin up simulation or, more common in our case, the date when the forecast is initialized.

Version: The data should be version controlled with a version which relates to the date of delivery to the archive or publication. This can be managed by the archive, and does not need to be dealt with by the data producers.

2) File name convention

The filenames will follow the model below:
<variable_name>_<MIP_table>_<model_id>_<experiment_family>_<start_date>_<ensemble_member>[_<temporal subset>].nc

Example: sic_Oimon_EC-Earth2_seaIceInit_S19910501_r1i1p1_199501-199502.nc

Variable name, model_id, and ensemble_member should follow the same conventions as described above for the directory tree.
The < temporal subset> (along with the preceding underscore) is omitted for variables that are time-independent.

III Global attributes

Certain global attributes must be included to document the details of the experiment. Data producers are free to add others, but the following list of global attributes is required in all files.

Attributes followed by an asterisk (*) are those that are automatically generated in the CMORization operation without any user information, the other ones being also generated during this phase, but passed to the program by the user. The CMORization operation intends to reorganize the files to be CF-compliant and in accordance with the rules described in the present document. Those attributes will be read in the tables or directly produced by the CMOR scripts.
There is a plan in the SPECS project to adapt the CMOR2 library developed by PCMDI that would take into account all the SPECS requirements, allowing users to adapt their post-processing scripts to easily produce output files compliant with the SPECS metadata and attributes requirements.

- associated_experiment: this free text string helps to link several experiments together to compare them. The associated_experiment can be the control run (in this case, it will be seasonal or decadal. It must have the format "experiment_family (+ optionally a the rip of a particular experiment and/or model version). It can be especially useful in the case of sensitivity experiments.
- batch (*): in addition to the version attribute present in the directory tree, a batch attribute is added to identify when the CMOR processing is done. Batch attribute has the form "XXXX YYYY-MM-DD-THH:MM:SSZ ", where XXXX is a short identifier. For example, a cmorization process performed on the 01/10/2013 at IC3 should get the batch attribute IC32013-10-01-T11:25:12Z. Ideally CMOR processing logs should be kept for a while, and the batch should make possible to link particular files to a given processing log.

- contact: name and contact information (e.g., email, address, phone number) of the person who should be contacted for more information about the data. Because it can cause some traceability issues in case of late changes in the address or name of the contact, data producers are widely encouraged to provide a re-directable address rather than a personal one.

- Conventions: CF-1.6

- creation_date (*): a string representation of the date when the file was created in the format "YYYY-MM-DD-THH:MM:SSZ" with replacement of all but "T" and "Z" by the obvious date or time indicator (e.g., "2010-03-23-T05:56:23Z").

- experiment_id: Complete experiment short name, according to the wiki list of experiments. It includes the start date (for example decadal1960).

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

- frequency (*): frequency of the output, see directory tree description.

- institute_id: standardized name of the institution that produced the data; it must belong to the official list of registered SPECS partners: CNRM-CERFACS, ECMWF, ENEA, IC3, KNMI, MOHC, MPI-M, UOXF, UREAD

- institution: name of the institution that produced the data; it can be different from the institute_id and has no character restriction.

- initialization_method: an integer (≥1) referring to the initialization method used or different observational datasets used to initialize. If only a single method and dataset was used to initialize the model, then this argument should normally be given the value 1. For time-independent fields, set initialization_method=0. Note that the

initialization_method is used in constructing the "ensemble member" as the value of M in r<N>i<M>p<L>. See the description of the ensemble member triad above.

- initialization_description: one weakness of CMIP5 data that frustrates some users is that the definition of different initialization methods and physics versions is not easily accessible: "initialization_method=2" can mean completely different things in different institutions. The data producer is encouraged to use this attribute to describe as precisely as possible the initialization technique, including the ensemble generation, given the special role that the initialization process plays in climate prediction.

- model_id: short name of the model, as described in the directory tree section above.

- associated_model: in case of experiments designed to study the impact of sensitivity to important changes in a forecast system, such as the resolution (experiment_families C3 "horizontalResolutionImpact" and C4 "standardStratosphereVerticalResolution improvedStratosphereVerticalResolution"), the only difference between two experiments would be the resolution of the model. Filling this attribute suggests the name of the model used to perform the same experiment at a different resolution, so that they can be related. For example, if resolution sensitivity experiments are done using IPSL-CM5A-LR/IPSL-CM5A-MR, the global attribute "associatedModel" will take the value "IPSL-CM5A-MR [medium resolution]" in the IPSL-CM5A-LR files and the value "IPSL-CM5A-LR [low resolution]" in the IPSL-CM5A-MR file, this attribute linking both experiments. This will reduce the uncertainty associated with the similarity of model names. It can contain several models if necessary.

- modeling_realm (*): a string that indicates the high level modeling component which is particularly relevant. For SPECS, permitted values are: "atmos", "ocean", "land", "landIce", "seaIce", "aerosol", "atmosChem", or "ocnBgchem" (ocean biogeochemical).

- physics_version: an integer (≥1) referring to the physics version used by the model. If there is only one physics version of the model, then this argument should be normally given the value 1. Note that model versions that are substantially different should be given a different "model_id"; assigning a different "physics_version" should be reserved for closely-related model versions (e.g., as in a "perturbed physics" ensemble) or for the same model, but with different forcing or feedbacks active. For time-independent fields, set physics_version=0. Note that the physics_version is used in constructing the "ensemble member" as the value of L in r<N>i<M>p<L>. See the description of the ensemble member triad above.

- physics_description: one weakness of CMIP5 data that frustrates some users is that the definition of different initialization methods and physics versions is not easily accessible "physics_version=2" can mean completely different things in different institutions. Data producers are encouraged to use this attribute to describe as precisely as possible the physics version.

- project_id (*): SPECS

- realization: an integer (≥1) distinguishing among members of an ensemble of simulations (e.g., 1, 2, 3, etc.). If only a single simulation was performed, then it is recommended that realization=1. For time-independent fields, set realization=0 (violating the

general rule that it should be a positive definite integer). Note that if two different simulations were started from the same initial conditions, the same realization number should be used for both simulations. For example, if a historical run with "natural forcing" only and another historical run that includes anthropogenic forcing were initiated from the same point in a control run, both should be assigned the same realization. Note that the realization can be used in constructing the "ensemble member" as the value of N in r<N>i<M>p<L>. See the description of the ensemble member triad above.

- source: character string fully identifying the model and version used to generate the output. The first portion of the string should be a copy of the global attribute "model_id". Additionally, this attribute must include the year (i.e., model vintage) when this model version was first used in a scientific application. Finally, it should include information concerning the component models. The following template should be followed in constructing this string: '<model_id><year>atmosphere:<model_name> (<technical_name>, <resolution_and_levels>); ocean:<model_name> (<technical_name> , <resolution_and_levels>); sea ice:<model_name> (<technical_name>); land: <model_name> (<technical_name>)". For some models, it may not make much sense to include all these components, and nothing following "<year>" is absolutely mandatory. Additional explanatory information may follow the required information, in particular URLs where full information describing the model can be found.

- startdate: start date of the experiment in the form of "Syyyymmdd". For uninitialized runs, a start date will still need to be used. This date corresponds to the beginning of the historical run from the parent experiment (e.g., the pre-industrial simulation). We refer to the date when a historical run starts from the spin up simulation or, more common in our case, the date when the forecast is initialized.

- table_id (*): it should be assigned a string of the form "Table <table name>" as in cell F1 of the spreadsheet in "CMIP5 Requested Output", followed by "(<date of table>", where the date is the date of the requested output table (e.g., "table_id=Table Amon (10 June 2010)").These tables are ASCII files that can be read by CMOR and are typically made available from MIP web sites. Because these tables contain much of the metadata that is useful in the MIP context, they are the key to reducing the programming burden imposed on the individual users contributing data to a MIP. Additional tables can be created as new MIPs are born.

- title (*): it should follow the template "<model_id> model output prepared for SPECS <experiment>" where <model_id> should be replaced by the contributing model's acronym or name (see description above) and <experiment> should be replaced by one of the experiment id.

- tracking_id (*): a string that is almost certainly unique to this file and must be generated using the OSSP utility that supports a number of different DCE 1.1 variant UUID options. The tracking_id might look something like 02d9e6d5-9467-382e-8f9b-9300a64ac3cd.

IV Time axis

One of the novelties of the SPECS conventions is the requirement for two time variables: one being

called time(time) which corresponds to the verification time of the forecast and one called leadtime. Both of these variables are mandatory in the files and must have the following attributes:

```
double time(time) ;
        time:units = "days since 1850-01-01" ;
        time:bounds = "time_bnds" ;
        time:long_name = "Verification time of the forecast" ;
        time:standard_name = "time" ;
        time:axis = "T"
 double leadtime(time) ;
        leadtime:units = "days" ;
        leadtime:long_name = "Time elapsed since the start of the forecast" ;
        leadtime:standard_name = "forecast_period" ;
```

The forecast reference time is not explicitly provided as a variable but it can be obtained by the global attributes or as forecast_reference_time=time-leadtime.

V Experiment description

A more complete description of the experiments below can be found in the SPECS wiki experiment section.
The list of names of the SPECS experiments appears below.

Core experiments:

C1: Experiment family: soilMoistureInit
Description: Soil moisture initial conditions impact with the best possible and climatological initialization.
Associated workpackages: 3.1

C2: Experiment family: seaIceInit
Description: sea-ice initialization impact with the best possible and climatological initialization
Associated workpackages: 3.1

C3: Experiment family: horizlResImpact
Description: impact of increased horizontal resolution
Associated workpackages: 4.1

C4: Experimentfamily: improvedStratVertRes
Description: impact of improved stratosphere vertical resolution
Associated workpackages: 4.3

C5: Experiment family: decadal
Description: 1960-2012, one start date per year (1st November), 5-year forecast length
Associated workpackages:

C6: Experiment family: seasonal
Description: NMME (National Multi Model Ensemble) operative forecasts + control runs for IC3 simulations, experiments associated with C1 and C3 experiment families

Tier1 experiments:

T1.1: Experiment family: snowInit

Description: hindcasts over 2004-2012, ten-member ensemble, 15th October, 1st November, 15th November, 1st December, 1st February, 1st March; forecast length until end of February or May according to the start date

Associated workpackages: 3.1

T1.2: Experiment family: phenology

Description: hindcasts over 1991-2012, ten-member, one start date per year (first of May), seven-month forecast length

Associated workpackages: 4.2

T1.3: Experiment family: aerosols

Description: Assessment of the sensitivity to aerosol (natural and anthropogenic) specification

Associated workpackages: 4.3

T1.4: Experiment family: solarIrradiance

Description: seasonal hindcasts and decadal hindcasts

Associated workpackages: 4.3