PMIP4 and the CMIP6 DRS

All the *attributes* have to be defined properly when creating data for the CMIP6 database, but you will find below details about some attributes that are especially relevant for PMIP4

Some key concepts...

- **attribute**: a *global attribute* (e.g. in a NetCDF file) used to describe the data
- **CV**: sometimes the value of a given **attribute** has to be taken from a predefined set of values, known as a *Controlled Vocabulary* (**CV**)
- **DRS** = *Data Reference Syntax*: the *DRS* is used to identify experiments, simulations, ensembles of experiments, atomic datasets and is used, for example, in file names, directory structures, the further_info_url, and in facets of some search tools

Example: the `experiment_id` **attribute** is used in the **DRS**, and its value has to be chosen from a **CV** ([*piControl*, *past1000*, *lgm*, ...])

CMIP6 official specifications

The following CMIP6 document is still *in prep* (as of July 14th 2016)

This document specifies all the global attributes that are defined for CMIP6. It also indicates how a subset of those relate to the Data Reference Syntax (DRS) and are used in file names and directory structures. Controlled vocabularies are defined for some global attributes (e.g., source_type and grid_resolution).

- **CMIP6** document: CMIP6 Global Attributes, DRS, Filenames, Directory Structure, and CV's (version 1.0)

- Legacy **CMIP5** documents:
  - CMIP5 Data Reference Syntax (DRS) and Controlled Vocabularies (Version 1.3.1 - 13 June 2012)
  - CMIP5 Model Output Requirements: File Contents and Format, Data Structure and Metadata (7 January 2010)

Project identification attributes

- `activity_id` = activity labels
- `mip_era` = activity's associated CMIP cycle
- Note: *The project_id used in CMIP5 is being replaced in CMIP6 with two global attributes: 1) an activity_id, and 2) a mip_era (a label indicating the cycle of CMIP that this activity falls under which will be set to "CMIP6" for the 6th CMIP cycle). In a few cases it may be appropriate to include multiple activities in the activity_id (with multiple activities allowed, separated by single spaces). An example of this is "LUMIP AerChemMIP" for one of the land-use change experiments.*

Project	`activity_id`	`mip_era`	Note
CMIP6	CMIP	CMIP6	
PMIP4-CMIP6	CMIP "CMIP PMIP"**??**	CMIP6	Should we use *CMIP* or "*CMIP PMIP*" for PMIP4 experiments that are part of CMIP6? This is confusing
PMIP4	PMIP	PMIP4**??** CMIP6**??**	Use this for non-CMIP6 experiments, or groups that are not part of CMIP6 Should we use *PMIP4* because it is the 4th phase of PMIP, or *CMIP6* because we will be using CMIP6 format specifications?

Experiment names

You can find all the referenced experiment names on es-doc *search* site: select `Project=CMIP6-DRAFT` and `Type=Experiment`

- `experiment_id` = root experiment identifier
- `experiment` = short expt. description

- `sub_experiment_id=none` ⇒ *needed for CMIP6 hindcast and forecast experiments to indicate "start year". For other experiments, this should be set to none*
- `sub_experiment=none` ⇒ *needed for CMIP6 hindcast and forecast experiments. For other experiments, this should be set to none*

DECK and historical experiments

Fix Me! How do we specify that an *historical* experiment is the true continuation of a *past1000* experiment?

We probably need to use

`parent_experiment_id`=*past1000* in the files' metadata, as well as `parent_activity_id`, `parent_mip_era`=*CMIP6*, `parent_source_id`, `branch_time_in_ parent` and other related `parent_*` variables. We can probably also agree on a specific `variant_label` that will appear in the file names.

experiment_id	**experiment**
amip	Atmospheric Model Intercomparison Project
piControl	Pre-Industrial Control
abrupt-4xCO2	abrupt quadrupling of CO2
1pctCO2	1 percent per year increase in CO2
historical	all-forcing simulation of the recent past

PMIP4-CMIP6 experiments

You can find specific details about the experiments by visiting the PMIP4 experimental design section, or by directly clicking on one of the experiments below

experiment_id	**experiment**
past1000	past 1000 years
mid-Holocene	mid-Holocene
lgm	last glacial maximum
lig127k	last interglacial
midPliocene-eoi400	mid-Pliocene

PMIP4 only experiments

Guidelines for creating new PMIP4 experiment_id values

- Allowed characters: CMIP6 `experiment_id` values are similar to CMIP5 (*...so the permitted characters will be: a-z, A-Z, 0-9, and "-"*), but a compound structure is allowed (segments separated by hyphens; e.g., "abrupt**-**4xCO2"); in a few cases multiple sub-experiments will be defined belonging to this root and constituting an ensemble.
- Being wise when creating/proposing a new name
    - ```
      Global attributes that label experiments are needed to construct
      file names and directories and can generally be used as search
      facets.  Together, they should have the following characteristics:

          * Uniquely label each experiment within CMIP6 and distinguish
      experiments with specified conditions that differ in any way
          * Easily be interpreted and remembered
          * Facilitate representations of groups of experiments that are
      closely related (e.g., same forecast conditions but different
      start dates, or experiment with an "offline" model driven by
      ```
      ```

```
output from various models)
```

- Planning for groups of related simulations
  - ```
    Often several simulations will be performed that satisfy the
    conditions specified for each experiment.  For example simulations
    of the historical period can branch from various points in a
    control run, and each of these will satisfy the conditions
    defining the experiment.  Together such simulations constitute a
    “conforming ensemble” with member all satisfying the same “root”
    experiment specifications.  There are also occasional cases where
    the experiment designers (MIP leaders) define a family of related
    simulations and choose to label these with a common “root”
    experiment name.  An example of this is the set of decadal
    prediction hindcasts that are all run similarly but started from
    different start dates (with each simulation identified by a
    different sub-experiment label).   Such “defined ensembles” of
    experiments will be labeled with a “root” experiment name, and a
    “sub-experiment_id” will be used to distinguish among members in
    the ensemble.
    ```
  - Ensemble of simulations usually share a common `experiment_id` and have different *ripf* variant labels.

Proposed PMIP4 experiment_id values

The following suggested PMIP4 `experiment_id` values should be considered as a *work in progress*, till they are validated!

You can find specific details about the experiments by visiting the PMIP4 experimental design section, or by directly clicking on one of the experiments below

experiment_id	**experiment**	**Status**
LDv1-LGMspin	Last Glacial Maximum spinup	Work
LDv1-transpin	Transient orbit and trace gases spinup (26-21 ka)	Work
LDv1	Transient deglaciation (21-0 ka)	Work

Handling groups of simulations

CMIP5 ensemble member

aka `r<N>i<M>p<L>` or *rip*


    ```

The definitions below have been superseded by CMIP6 specifications, but it is still useful to remember them. They have been copied from:

- a) CMIP5 Data Reference Syntax (DRS) and Controlled Vocabularies (Version 1.3.1 - 13 June 2012)
- b) CMIP5 Model Output Requirements: File Contents and Format, Data Structure and Metadata (7 January 2010)

- b) ⇒ `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 fields appearing in table “fx” in the CMIP5 Requested Output, 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. Also, each so-called RCP (future scenario) simulation should normally be assigned the same realization integer as the historical run from which it was initiated. This will allow users to easily splice together the appropriate historical and future runs. A similar convention should be followed, when appropriate, with other simulations (e.g., the decadal simulations).
  Note that the realization can be used in constructing the “ensemble member” called for by the DRS document; it is the value of **N** in **r<N>**i<M>p<L>.
  [Note that for the “Transpose AMIP” project, the “realization” number is used to distinguish among the 16 members of each of 4 suites of runs (i.e., the 4 “seasons”) generated from different observed conditions, spaced 30 hours apart. So, for example, the 16-member ensemble of runs initialized at 00Z on 15 Oct 2008, 06Z 16 Oct 2008, 12Z 17 Oct 2008, and so-on, would be assigned “r1”, “r2”, “r3”, etc.]

- b) ⇒ `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 fields appearing in table “fx” in the CMIP5 Requested Output, set initialization_method=0 (violating the general rule that it should be a positive definite integer).
  See the DRS document for guidance on assigning initialization_method. Note that the initialization_method is used in constructing the “ensemble member” called for in the DRS document; it is the value of **M** in r<N>**i<M>**p<L>.

- b) ⇒ `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. In CMIP5, one would distinguish, for example, among runs forced by different combinations of “forcing” agents (as called for under the “historicalMisc” experiment – experiment 7.3) by assigning different values to physics_version.
  For fields appearing in table “fx” in the CMIP5 Requested Output, set physics_version=0

(violating the general rule that it should be a positive definite integer). Note that the physics_version is used in constructing the “ensemble member” called for by the DRS document; it is the value of **L** in r<N>i<M>**p<L>**

- a) ⇒ Ensemble member (r<N>i<M>p<L>) =

```
This triad of integers (N, M, L), formatted as
shown above (e.g., “r3i1p21”) distinguishes among closely related
simulations by a single
model. 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. CMIP5 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). In CMIP5, 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”). For TAMIP (“the Transpose
AMIP activity), the
“realization” number is used to distinguish among the 16 members of
each of 4 ensembles (one
for each of 4 “seasons”) generated from different observed conditions,
spaced 30 hours apart.
So, for example, the 16-member ensemble of runs initialized at 00Z on
15 Oct 2008, 06Z 16 Oct
2008, 12Z 17 Oct 2008, and so-on, would be assigned “r1”, “r2”, “r3”,
etc.

Models used for forecasts that depend on the initial conditions 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>). For CMIP5 this indicator might in some cases be
needed to distinguish
among runs in the decadal-prediction suite of experiments (1.1-1.6).
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. In CMIP5, time-independent variables (i.e.,
6
those with frequency=”fx”) are not expected to differ across ensemble members, so for these M
should invariably be assigned the value zero (“i0”). A key (i.e., a table) should be made
available that associates each value of M with a particular initialization method and/or
observational dataset.

If there are many closely related model versions, which, as a group, are generally referred to as a
perturbed physics ensemble (e.g., QUMP or climateprediction.net ensembles), then 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). If there are different
“forcing” combinations prescribed in experiment 7.3 in CMIP5 (the “historicalMisc” runs), then
each of these different runs are also assigned different values of L (in “p<L>”). Note that the
data supplier must assign a physics version number to each atomic dataset. It is recommended
that the numbers be assigned sequentially starting with 1. In CMIP5, time-independent variables
(i.e., those with frequency=”fx”) are not expected to differ across ensemble members, so for
these L should always be assigned the value zero (“p0”). A key (i.e., a table) should be made
available that associates each value of L with a particular set of model parameter values and/or,
in the case of the “historicalMisc” experiment, a particular suite of “forcing” agents.

Note that for a single model and experiment N, M, and L should be interpretable independently;
for all members of the ensemble, the correspondence between the values of N, M, and L and the
simulation characteristics they represent should be consistent. For example the two different
ensemble members, r3i1p7 and r3i1p8, should both be initialized from exactly the same initial
conditions using the same method (because the “r” and “i” values are identical) although the
subsequent evolution of the simulations will presumably differ since they were produced by two
different “perturbed physics” versions of the same model. Note that

```
there may be cases where
“gaps” could occur in the list of ensemble members. If, for example,
two different initialization
procedures were used, but the second procedure was tested with only a
subset of the initial
condition cases of the first procedure (say, every other case). Then
the list of ensemble members
would look like: r1i1p1, r2i1p1 r3i1p1, r4i1p1, r5i1p1, r6i1p1, …,
r1i2p1, r3i2p1, r5i2p1, …

A recommendation for CMIP5 is that each so-called RCP (future scenario)
simulation should
when possible be assigned the same realization integer as the
historical run from which it was
initiated. This will allow users to easily splice together the
appropriate historical and future runs.
Thus, for example, suppose a 3-member ensemble of historical runs of a
model exists, and a
single rcp45 simulation was produced, initialized from the third member
of the historical
ensemble. The recp45 simulation would be designated “r3” (rather than
“r1”), even though it is
the only existing ensemble member, in order to indicate that it was
spawned from member 3 of
the historical ensemble. A similar convention should be followed, when
appropriate, with other
simulations (e.g., the decadal simulations).
```

CMIP6 variant_label

aka r<k>i<l>p<m>f<n> or *ripf*

- realization_index = realization number (integer >0)
- initialization_index = index for variant of initialization method (integer >0)
- physics_index = index for model physics variant (integer >0)
- forcing_index = index for variant of forcing (integer >0)
  - Note: the information stored in the *forcing* attribute in CMIP5 may in CMIP6 appear in the variant_info attribute
- variant_label = a label constructed from 4 indices stored as global attributes

  ```
  r<k>i<l>p<m>f<n>
                    where
                       k = realization_index
                       l = initialization_index
                       m = physics_index
                       n = forcing_index
  ```

- variant_info = brief description of what is unique about this *ripf* variant
  - Example: *“forcing: black carbon aerosol only”*, *“realization 1”*, *“realization 1; initialized*

using anomaly approach (method 2)"

PMIP4 and variant_label notes

Reminder: each option in `r<k>i<l>p<m>f<n>` has to be a strictly positive integer

realization_index r<k>

The long PMIP4 simulations are going to require both a lot of processing power and a lot of storage. It is quite likely that there will be **only one realization**, for a given set of *i<l>p<m>f<n>* and that the variant label will always start with `r1`

forcing_index f<n>

Depending on available resources, the PMIP4 groups may choose to perform several simulations for the same experiment, using different combinations of forcings. **The forcings used will have to be carefully described** in the documentation (and in the metadata inside each NetCDF file) and be *encoded* in the integer value of the forcing_index.

There are several ways to proceed. The easiest way is to let each group choose its own way of numbering the forcings combinations (and document it!), but **all groups should try to use a common scheme** for and associate the same combination of forcings with the same integer

Sequential numbering scheme

The contact people for each experiment determine which forcings combinations are most likely to be used and associate them with a predefined number. If necessary, a group can later ask for a new forcing combination to be registered

Forcings	**fforcing_index**
Recommended default, or most likely combination, or mandatory simulation	`f1`
forcing1='on', forcing2='off', etc	`f2`
Some other combination	`fN`

Hierarchical numbering scheme

The following scheme will create bigger integers, but the values will be more meaningful

If there are 10 or less options for each type of forcing, we can assign a power of 10 to each type, multiply it with the forcing option and add everything

Tentative example for the lgm experiment:

Power	Forcing	Options
2	Ice sheet	1=*ICE-6G-C* 2=*GLAC-1D*
1	Aerosols	1=*Hopcroft et al* 2=*Albani et al*
0	Vegetation	1=*interactive vegetation* 2=*interactive carbon cycle* 3=*prescribed*

Example: *GLAC-1D* + *Hopcroft et al* + *interactive vegetation* = `2 * 100 + 1 * 10 + 1` ⇒ `f211`

variant_label constraints for PMIP4 experiments

historical

`historical` simulations that are the continuation of a past1000 experiment should use `1000` for the `initialization_method` ⇒ `i1000`

past1000

mid-Holocene

lgm

lig127k

midPliocene-eoi400

LD-LGMspin

LD-transpin

LD

PMIP4-CMIP6 directory structure and file names

The *DRS* defines (among other things) how the different attributes will be combined to generate unambiguous directories and file names, in the ESGF distributed database

```
Directory structure = <mip_era>/
                        <activity_id>/
                          <institution_id>/
                            <source_id>/
                              <experiment_id>/
                                <member_id>/          <== variant_label
                                  <table_id>/
                                    <variable_id>/
                                      <grid_label>/
                                        <version>/

file name = <variable_id>_<table_id>_<experiment_id
>_<source_id>_<member_id>_<grid_label>[_<time_range>].nc
```

For PMIP4, we have `sub_experiment_id == none` (because we don't use forecast and hindcast), and therefore `member_id == variant_label`

Used in dir?	Used in file?	Attribute name	Value for PMIP4-CMIP6
Y	N	`mip_era`	CMIP6 PMIP4 ?
Y	N	`activity_id`	CMIP PMIP Note: “CMIP PMIP” becomes CMIP (*If multiple activities are listed in the global attribute, the first one is used in the directory structure*)

Used in dir?	Used in file?	Attribute name	Value for PMIP4-CMIP6
Y	N	`institution_id`	institution label (*IPSL*, ...)
Y	N	`version`	vYYYYMMDD (e.g., `v20160218`), indicating a representative date for the version Note: the *version* is **not** stored in the NetCDF files and **not** used in the file names, because it is only specified when *publishing* (eg storing the data in ESGF) the NetCDF files
Y	Y	`source_id`	source label (e.g. the model name/version using only authorized characters)
Y	Y	`experiment_id`	See the Experiment names section
Y	Y	`member_id`	PMIP4 does not use `sub_experiment_id`, so the value of `member_id` is equal to the `variant_label`: r<k>i<l>p<m>f<n> (see the CMIP6 variant label section)
Y	Y	`table_id`	CMOR table label (`Amon`, ...)
Y	Y	`variable_id`	variable identifier (`tas`, `pr`, ...)
Y	Y	`grid_label`	`gn`: output is reported on the native grid `gr`: output is regridded by the modeling group to a “primary grid” of its choosing `gr1`, `gr2`, ...: output is regridded on another grid than the *primary grid* (that was already different from the *native grid*)
N	Y	`time_range`	the last segment of the file name indicates the time-range spanned by the data in the file, and is omitted when inappropriate. The format for this segment is the same as in CMIP5

Examples:

- Directory = `CMIP6/CMIP/NCAR/CCSM2-1/1pctCO2/r1i1p1f1/Amon/tas/gn/v20150320/`
  File = `tas_Amon_CCSM2-1_1pctCO2_r1i1p1f1_gn_202001-202912.nc`
- Directory = `CMIP6/DCPP/NCAR/CCSM2-1/dcppA-hindcast/s1960-r1i2p1f1/Amon/tas/gr/v20150320/`
  File = `tas_Amon_CCSM2-1_hindcast_s1960-r1i2p1f1_gn_198001-198412.nc`