Dr2xml Short Tutorial: 1- The Ping File

Date: 20/01/2017

Prerequisite: We assume here that you have a working copy of dr2xml installed (see “Dr2xml User Guide”).

What is a ping file? A dr2xml *ping file* is an xml file where to write the correspondence between CMIP variable names and model variable names. As name suggests, the *ping file* acts as a bridge between field references in the *file-def* files generated by dr2xml and the model field definitions*, usually included in the native field-def* files.

When, in the *file-def file* generated by dr2xml you have (or will have, if not yet generated):

```
      <file_definition>
            <file name=… output_freq=… >
                  <field field_ref="CMIP6_sos” name=”sos” <field/>
                  <variable name="standard_name" > sea_surface_salinity
                  </variable>
            </file>
      </file_definition>
```

And in the *native field-def file*:

```
      <field_definition>
            <field id="sss" … standard_name="sea_surface_salinity"/>
      </field_definition>
```

The correspondence given in the *ping file* should be (provided that “sss” meets “sos” definition, including units):

```
      <field_definition>
            <field id="CMIP6_sos" field_ref="sss" <field/> <!-- (0.001)
      sea_surface_salinity : Sea Surface Salinity -->
      </field_definition>
```

‘CMIP6_’ is a prefix that user can freely chose. The only constraint is to use the same prefix to create the *ping file* template and to generate the *file def* with dr2xml.

Why a template of ping file? Since the list of variables requested by the CMIP6 data request is long, complex (varying across MIPs, the experiments within a given MIP, etc.), dr2xml package provides a tool (*create_ping_file* python script or notebook) to generate a template of the *ping-file* including the exhaustive list of requested variables that applies to your CMIP6 modelling plans. It is a template since the actual correspondence with the model variable names is not yet established.

A template of ping file contains a large number of *“<field />”* lines like this one:

```
<field_definition>
      <field id="CMIP6_tos" field_ref="dummy" <field/>
      ...
</field_definition>
```

One important step in the preparation of the model output for CMIP6 is to fill this ping file template. Note that the job of making the correspondence between model names and CMIP names would have to be done anyway, even if you were not using the dr2xml framework, in *file-def* files or at a post-processing stage, before or during the Cmorisation. What dr2xml does here is just to propose a file format and naming conventions (use of a prefix).

How to create the ping file template(s)?

For each XIOS context, you will have to write a *ping file* including variables for all the realms covered by this context. For NEMO users, we have one nemo context, that includes *ocean*, *seaIce* and *ocnBgchem* realms.

For that purpose you can either use the *create_ping_files.ipynb* python notebook (available in *dr2pub/* repository) or the equivalent classical Python script *create_ping_files.py* (available in *dr2pub/doc/* repository).

In *create_ping_files* script or notebook, you have to adapt a few things for your own usage. Here are some guidelines on how to use this script/notebook.

1) Provide some settings (bold highlight below in "settings" python dictionary):

- Defines the MIPs you want to take into account (*'mips'* key)
- Define the maximum priority to consider for output variables (*'max_priority'* key)
- Define the maximum tier to consider for experiments (*'max_priority'* key)
- Choose a prefix for the ping variable field id (*'ping_variable_prefix'* key). This string is free, but we recommend not to use an empty string.

```
settings={
      'mips' : {"CMIP6", "AerChemMIP", "C4MIP", "CFMIP", "DAMIP", "DCPP",
                "FAFMIP","GeoMIP", "GMMIP", "HighResMIP",
                "ISMIP6","LS3MIP","LUMIP","OMIP","PDRMIP","PMIP", "RFMIP",
                "ScenarioMIP","SolarMIP", "VolMIP", "CORDEX", "DynVar",
                "SIMIP","VIACSAB","SPECS", "CCMI", "CMIP5", "CMIP",
                "DECK"},
      'max_priority' : 3,
      'tierMax'      : 3,
      "ping_variables_prefix" : "CMIP6_",
      # We account for a file listing the variables which the lab does not
      want to produce
      # Format : MIP varname as first column, comment lines begin with '#'
      #"excluded_vars_file":"/cnrm/est/USERS/senesi/public/CMIP6/data_reque
st/cnrm/excluded_vars.txt",
      "excluded_vars_file": None,
      "excluded_vars" : None,
      }
```

> Note: It is recommended to generate an exhaustive *ping file template*, asking for all MIPs your lab is involved in (or even all CMIP6 MIPs), maximum priority level and tier you are planning so as you can share this *ping file* across several modellers in your lab. It is possible because you are not forced to provide an actual model native field reference for every variable listed in the template.

2) Provide the list of realms concerned by the XIOS context for which you want to generate the ping-file template:

```
      # List of sets of realms for which ping files must be generated
      realms=['ocean','seaIce','ocnBgchem']
```

3) Finally invoke the ping-file writer:

```
pingFileForRealmsList(realms,svars,prefix=settings['ping_variables_prefix']
,comments=True,exact=False,dummy=True,filename='ping_ocean_and_co.xml')
```

For a complete description of arguments, type *help(pingFileForRealmsList)*

The resulting *ping-file template* looks like this:

```
<field_definition>
   <field id="CMIP6_sos" field_ref="dummy" /> <!-- (0.001) sea_surface_salinity :
Sea Surface Salinity -->
   <field id="CMIP6_sosga" field_ref="dummy" /> <!-- (0.001) sea_surface_salinity
: Global Average Sea Surface Salinity -->
   <field id="CMIP6_sossq"  field_ref="dummy" /> <!-- (1e-06) sossq : Square of
Sea Surface Salinity -->
   <field id="CMIP6_spco2" field_ref="dummy" /> <!-- (Pa)
surface_partial_pressure_of_carbon_dioxide_in_sea_water : Surface Aqueous Partial
Pressure of CO2 -->
…
```

The trailing comments provides the long name, the description and units of the CMIP variable.

How to Instantiate the ping file?

For each line of the *ping template*, modellers have to provides *field_ref* entries in place of *“dummy”.*

If you don’t know yet the name of the model variable corresponding to a requested CMIP6 variable, but you have some hope to find it later, do not modify the dummy line entry in the ping file: There is an option to tell dr2xml, when generating the *file-def*, to ignore variables having a dummy reference in the *ping file*.

If you are certain your model is not able to output a given requested variable, or your lab decide not to output it, and you want to keep track of that in a single location, the proper way to proceed is to enter these variables in the *excluded_vars* list (see “settings” dictionary). **In that case, you have to regenerate the ping template before starting to fill it.**

Using excluded list of variables or keeping *“dummy”* in the ping file both enable to keep track of the requested variables you won’t output.

Obvious recommendation: **When setting a correspondence to a MIP variable, take care that the candidate variable output by your model matches the physical definition of the CMIP6 variable** (including units).

If units of the variable output by your model is different from the units asked by the Data Request, you can glue in the corresponding ping file line an XIOS operation ensuring the units conversion:

Example: to convert sea surface temperature from degrees Kelvins:

```
<field_definition>
<field id="CMIP_tos"    field_ref="sst"> sst + 273.15 <field/>
</field_definition>
```

Other important recommendation: When filling the ping file, **it is highly recommended not to modify the lines ordering.** The general ping file template syntax is now fixed, but this does not exclude some minor changes in next version. Keeping the ordering will help you merge your defined correspondences in the next version.