Some nagging doubts about the CF `standard_name` attribute

GO-ESSP Workshop 2007
Jussieu, Paris

V. Balaji

Princeton University

NOAA/GFDL

11 June 2007

Outline

How the CF `standard_name` is produced

- It is an *attribute* that is easily added to existing model output.
- Modeling frameworks such as FMS, ESMF, and PRISM recognize the `standard_name` as an *optional* attribute of a physical field: it is held in the "container class" of a variable and automatically output.
- NCO tools (`http://nco.sourceforge.net`) such as `ncatted` can be used to add it *post facto*.
- Tools such as CMOR also add it by hand.

How the `standard_name` is consumed ...

It isn't really, at this point ... what actually happens is this:

- CMOR adds the `standard_name`, but also modifies the variable *name*: for example, the GFDL variable `slp` bears the standard name `air_pressure_at_sea_level`, and the "PCMDI standard name" `psl`.
- It is the string `psl` that users actually store in their `ferret` or `Matlab` scripts, or pass to the `-v` flag of the NCO utilities like `ncbo` and so on.
- By "standardizing" the name `psl`, you enabled users to write analysis packages that worked for any model in the AR4 archive.
- This PCMDI or AR4 standard actually carries over into other projects, such as TFSP (e.g see Paco Doblas-Reyes' TFSP Data Management planning document).

Could we shift “variable recognition” over to `standard_name`?

Maybe, but there are some difficulties:

- The `standard_name` is too long to type: it is human-readable, but not human-writable.
- There is no mechanism or rule in place to ensure that two variables in a dataset not bear the same `standard_name`: in fact it is necessary in some cases, e.g high, middle and low cloud variables are all `cloud_area_fraction_in_atmosphere_layer`. You may need many attributes to “uniquify” a variable, something the netCDF name does cleanly.
- At best, I see the `standard_name` being used (along with other attributes) by analysis tools to generate a lookup table from which you pick out the variable name.
- If you asked data consumers, they’d vastly prefer if all experiments standardized the short name, if indeed that were practical.

Could we shift “variable recognition” over to `standard_name`?

Maybe, but there are some difficulties:

- The `standard_name` is too long to type: it is human-readable, but not human-writable.
- There is no mechanism or rule in place to ensure that two variables in a dataset not bear the same `standard_name`: in fact it is necessary in some cases, e.g high, middle and low cloud variables are all `cloud_area_fraction_in_atmosphere_layer`. You may need many attributes to “uniquify” a variable, something the netCDF name does cleanly.
- At best, I see the `standard_name` being used (along with other attributes) by analysis tools to generate a lookup table from which you pick out the variable name.
- If you asked data consumers, they’d vastly prefer if all experiments standardized the short name, if indeed that were practical.

Could we shift “variable recognition” over to `standard_name`?

Maybe, but there are some difficulties:

- The `standard_name` is too long to type: it is human-readable, but not human-writable.
- There is no mechanism or rule in place to ensure that two variables in a dataset not bear the same `standard_name`: in fact it is necessary in some cases, e.g high, middle and low cloud variables are all `cloud_area_fraction_in_atmosphere_layer`. You may need many attributes to “uniquify” a variable, something the netCDF name does cleanly.
- At best, I see the `standard_name` being used (along with other attributes) by analysis tools to generate a lookup table from which you pick out the variable name.
- If you asked data consumers, they’d vastly prefer if all experiments standardized the short name, if indeed that were practical.

Could we shift “variable recognition” over to `standard_name`?

Maybe, but there are some difficulties:

- The `standard_name` is too long to type: it is human-readable, but not human-writable.
- There is no mechanism or rule in place to ensure that two variables in a dataset not bear the same `standard_name`: in fact it is necessary in some cases, e.g high, middle and low cloud variables are all `cloud_area_fraction_in_atmosphere_layer`. You may need many attributes to “uniquify” a variable, something the netCDF name does cleanly.
- At best, I see the `standard_name` being used (along with other attributes) by analysis tools to generate a lookup table from which you pick out the variable name.
- If you asked data consumers, they’d vastly prefer if all experiments standardized the short name, if indeed that were practical.

Use case or thought experiment

- User `mike` wants to compare “high cloud amount” between two models.
- Define a procedure for doing this on the basis of the CF conventions alone.
- `cloud_area_fraction_in_atmosphere_layer` + auxiliary coordinate representing layer bounds in pressure coordinates.

Use case or thought experiment

- User `mike` wants to compare “high cloud amount” between two models.
- Define a procedure for doing this on the basis of the CF conventions alone.
- `cloud_area_fraction_in_atmosphere_layer` + auxiliary coordinate representing layer bounds in pressure coordinates.

Use case or thought experiment

- User `mike` wants to compare “high cloud amount” between two models.
- Define a procedure for doing this on the basis of the CF conventions alone.
- `cloud_area_fraction_in_atmosphere_layer` + auxiliary coordinate representing layer bounds in pressure coordinates.