DART Tutorial Section 3:
DART Runtime Control and Documentation
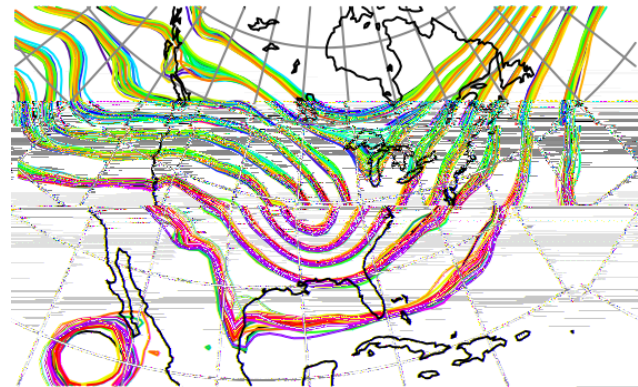
©UCAR

NCAR | National Center for
UCAR | Atmospheric Research

DART Philosophy: configurable at run-time

Use F90 namelist facility to do this.

Each F90 module can have its own associated namelist file.

All namelists combined in a single file, *input.nml*, in *work* directory.

Documentation of modules including namelists in html files.

cov_cutoff_mod.f90

Code for module **cov_cutoff_mod**

cov_cutoff_mod.html

Documentation for module.

cov_cutoff_mod.nml

Run-time control for module.

Example: Changing to a multivariate filter.

models/lorenz_63/work/

Section 1 Lorenz 63 example:
- Observed x, y, z components.
- Observation of x only impacted ensemble for x, etc.

Let's convert to a multivariate filter:
- Observations of x will impact ensembles for x, y and z.

To do this, will modify a namelist setting:
- Change will be made in file *models/lorenz_63/work/input.nml*.
- Modification to `&assim_tools_nml`.
- Namelist parameter of interest is *`cutoff.`*

Example: Changing to a multivariate filter.

Open a browser and look at file
assimilation_code/modules/assimilation/assim_tools_mod.html

Has a variety of sections:
- Overview;
- List of other modules used;
- Public interface (how to use this in another module);
- Details of public interfaces and variables;
- Namelist (what we're interested in for now).

The namelist section lists all runtime control variables for `assim_tools`.
- Gives description of each;
- *cutoff* controls distance to which observation has impact;

Originally very small: observation of x only impacts x.

Make it very big: all observations impact all state variables.

Example: Changing to a multivariate filter.

Edit *models/lorenz_63/work/input.nml* – it contains namelists for all modules used with Lorenz 63. The program **filter** uses namelists from many modules, one of which is the *assim_tools* namelist.

Modify *assim_tools_nml* namelist parameter *cutoff;* when program **filter** is run again, it will incorporate this modification.

Change to 1000000.0

Start of namelist for assim_tools module

```
&assim_tools_nml
    filter_kind                     = 1,
    cutoff                          = 0.00001,
    sort_obs_inc                    = .false.,
    spread_restoration              = .false.,
    sampling_error_correction       = .false.,
    adaptive_localization_threshold = -1,
    distribute_mean                 = .false.,
    output_localization_diagnostics = .false.,
    localization_diagnostics_file   = 'localization_diagnostics',
    print_every_nth_obs             = 0
/
```

End of namelist for assim_tools module

Example *input.nml.xxxxxx_default* files for each program are automatically constructed by compilation tool (Section 11). It is usually convenient to have **one** *input.nml* containing all the settings for the commonly-used programs.

DART Tutorial Index to Sections

1. Filtering For a One Variable System
2. The DART Directory Tree
3. DART Runtime Control and Documentation
4. How should observations of a state variable impact an unobserved state variable? Multivariate assimilation.
5. Comprehensive Filtering Theory: Non-Identity Observations and the Joint Phase Space
6. Other Updates for An Observed Variable
7. Some Additional Low-Order Models
8. Dealing with Sampling Error
9. More on Dealing with Error; Inflation
10. Regression and Nonlinear Effects
11. Creating DART Executables
12. Adaptive Inflation
13. Hierarchical Group Filters and Localization
14. Quality Control
15. DART Experiments: Control and Design
16. Diagnostic Output
17. Creating Observation Sequences
18. Lost in Phase Space: The Challenge of Not Knowing the Truth
19. DART-Compliant Models and Making Models Compliant
20. Model Parameter Estimation
21. Observation Types and Observing System Design
22. Parallel Algorithm Implementation
23. Location module design (not available)
24. Fixed lag smoother (not available)
25. A simple 1D advection model: Tracer Data Assimilation