Observation Processing

Nancy Collins nancy@ucar.edu

or

dart@ucar.edu

Roadmap

- What's in an Observation
- Provided tools and capabilities
- Observation sources
- Observation and representativeness error
- Types, Kinds, Preprocess
- Evaluating vs. Assimilating
- Observation output files

What's in an "Observation"

- Observation Type
  - As defined `in DART/obs_def` files listed in `input.nml : &preprocess_nml`
- Observed Value
- Expected Error
- Location
  - Generally lat/lon plus vertical
  - Height, Pressure, Surface, "undefined"
- Time
- Quality control from data source ("input qc")

What's in an "Observation Sequence File"

- List of specific obs types in this file
  - Numbers associated with each type DO NOT have to be consistent between different files
- Count of total obs in file, Count and Labels for "Copies" and for "QCs"
  - Copies => Various types of obs data
  - QCs => Quality Control values
- For more details:
  - http://www.image.ucar.edu/DAReS/DART/DART_Documentation.php#obs_seq_overview

Observation Converters

- Observation data distributed in a wide variety of formats
  - NetCDF , HDF, BUFR, compressed text, plain text
- Converters already exist for most common obs types, especially for atmospheric obs
- If you have other observations:
  - The “text” converter a good base for ascii obs
  - The “MADIS” converters for NetCDF obs
  - The “AIRS” converter for HDF-EOS

Observation Sources

- Main documentation page
  - https://proxy.subversion.ucar.edu/DAReS/DART/releases/Lanai/observation/observations.html
- Links to html pages for each of the existing converters
- Generally includes links to data repositories and comments about file formats

Observation Tools

- `obs_sequence_tool` - combines, splits, extracts, modifies obs_seq files
- `obs_seq_to_netcdf` - converts observation data into netcdf files
  - Cannot include obs-specific metadata (GPS, radar)
- Verification suite (doc obs_seq_coverage.html)
  - `obs_seq_coverage, obs_selection, obs_seq_verify`
- `obs_common_subset` - compare same obs from different experiments

Error values in the DA System

- Observation errors
  - Instrument resolution, retrieval algorithms
- Representativeness errors
  - Features the model cannot represent, e.g. convection
- We combine both of these into the 'obs error' in the obs_sequence file
- Many obs already come with error values

Generic Kinds vs Specific Types

- Model State contains various data fields, e.g. T, winds, moisture, etc
- These are identified with a GENERIC KIND code inside the DART system
  - Constants which always start with `KIND_xxx`
  - Defined in `obs_kind/obs_kind_mod.f90`
- Examples include KIND_TEMPERATURE, KIND_U_WIND_COMPONENT

Generic Kinds vs Specific Types (cont)

- Observations are identified with a SPECIFIC TYPE which allows finer control over what is assimilated and/or evaluated
  - Examples include RADIOSONDE_TEMPERATURE, AIRS_TEMPERATURE, AIRCRAFT_TEMPERATURE
- Each SPECIFIC TYPE has an associated GENERIC KIND, e.g. all above are KIND_TEMPERATURE
- These type constants are also defined in `obs_kind/obs_kind_mod.f90`
- Assimilation and evaluation namelists uses these specific types

The “preprocess” program

- The files `obs_kind/obs_kind_mod.f90` and `obs_def/obs_def_mod.f90` are generated during the build process
- Written by the `preprocess` program
- If you change the preprocess namelist, you must rebuild to regenerate these files
- Includes only the types needed by a single model
- See the documentation for how to add new types:
  - https://proxy.subversion.ucar.edu/DAReS/DART/releases/Lanai/obs_def/obs_def_mod.html
- Less common, but how to add new kinds:
  - https://proxy.subversion.ucar.edu/DAReS/DART/releases/Lanai/obs_kind/obs_kind_mod.html

Assimilating, Evaluating, Excluding

- Namelist control: `input.nml : &obs_kind_nml`, so this is a run-time choice
- `assimilate_these_obs_types = 'RADIOSONDE_TEMPERATURE', 'ACARS_TEMPERATURE', 'AIRCRAFT_TEMPERATURE',`
- `evaluate_these_obs_types = 'RADIOSONDE_SURFACE_PRESSURE'`
- Any types not on the list will be ignored

Observation Sequence output files

- In addition to original information, adds Prior and Posterior forward operator ensemble mean and ensemble spread
- By namelist control can include FO values from 1 to N ensembles
- DART QC – tells how DART processed each observation

DART QC Values

- DART QC numeric code between 0 and 7
- Indicates one of the following possible outcomes for each obs:
  - Assimilated or Evaluated successfully
  - Prior FO was ok but Posterior FO failed
  - Obs not used because:
    - Prior FO failed, Not listed in namelist, Bad incoming QC, Outlier test failed
- See: http://www.image.ucar.edu/DAReS/DART/DART_Documentation.php#obs_diagnostics