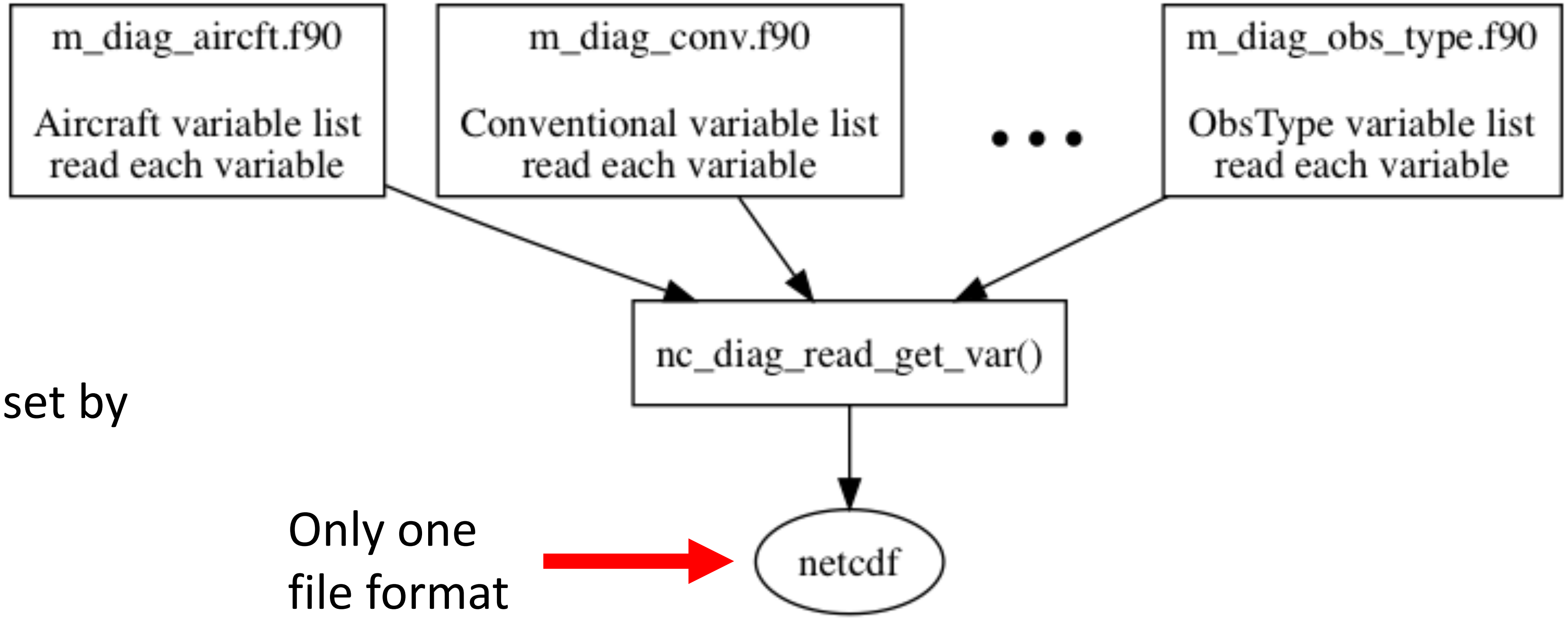
ObsSpace Reorganization

Stephen Herbener

JEDI Core Team

5/3/18

IODA: File IO, Current State



m_diag_aircft.f90

Aircraft variable list
read each variable

m_diag_conv.f90

Conventional variable list
read each variable

• • •

m_diag_obs_type.f90

ObsType variable list
read each variable

nc_diag_read_get_var()

netcdf

Duplicated code
Names of variables set by reader

Only one file format

IODA: File IO: Introduce OO structure

- Motivation for this work
  - Eliminate duplicate code
  - Be able to handle multiple file formats with a uniform interface
- Add OO class structure
  - Client instantiates a file object
    - Constructor
      - Potential arguments
        - File name
        - File open mode (read, write, append)
  - Client calls read method
    - Potential arguments to read method
      - List of variables
      - Memory object to hold variable values

ObsSpace: Work plan

- Two nearly independent efforts
  - Reorganize ObsSpace class hierarchy
  - Introduce OO structure to IODA Reader/Writer
  - Separate class hierarchies between ObsSpace and IODA Reader/Writer
    - However, these two structures are associated so the work has to be done together
- Limit this work to UFO and IODA repositories
  - No modifications to the OOPS ObsSpace classes
- Steps
  - Design new class structures, data, methods
    - Get a reasonable design set first before writing code
  - Implement new class structures
    - Create feature branches for UFO and IODA repositories
    - Two of us work in parallel
      - ObsSpace tasks
      - IODA reader/writer tasks