CF-2 Group Proposal
Advancing netCDF-CF for the Geosciences
ESIP Summer 2018 Meeting, Tuscon, AZ
Charlie Zender, UC Irvine
Daniel Lee, EUMETSAT
Motivation for using Groups

• netCDF Classic is “flat”

• Most data is hierarchical

IASI-NG
METimage
MWS
RO
3MI
S5
Other use cases

• Collections of related data
  • Multi-model collections
  • In-situ vs remote sensing
  • Etc.

• Ensembles
  • NWP, climate

• Discrete sampling geometries
  • Ragged arrays of observations
  • Efficient access patterns
Design principles

• Backwards compatibility with CF1

• Support “flattening” through various mechanisms

• Inherit metadata intuitively using widely adopted scoping patterns
• Data in a given group are visible to all members of child groups

• Absolute and relative paths are allowed

• Otherwise unqualified names can be found via search (first ancestor search, then in width-wise fashion down each level of hierarchy)
Best practices

• Use accepted data types

• Don’t use group attributes for variable attributes, even if they apply to all subsequent variables

• Software is not required to interpret group names
Data related by theme alone

:Conventions="CF-1.8 CF2-Group-1.0"

group: model { // Variable attributes omitted for clarity
  dimensions:
    lat=2;
    lon=3;
    time=unlimited;
  variables:
    float temperature(time,lat,lon);
    double time(time);
    double lat(lat);
    double lon(lon);
} // end model

group: measurements_remote_sensing {
  dimensions:
    lat=3;
    lon=4;
    time=unlimited;
  variables:
    float temperature(time,lat,lon);
    double time(time);
    double lat(lat);
    double lon(lon);
} // end measurements_remote_sensing

group: measurements_in_situ {
  dimensions:
    time=unlimited;
  variables:
    float temperature_10m(time);
    double time(time);
} // end measurements_in_situ
Ensembles

:Conventions="CF-1.8 CF2-Group-1.0"
group: cesm_01 {
  :Scenario = "Historical";
  :Model = "CESM";
  :Realization = "1";
  dimensions:
    time=unlimited;
  variables:
    float temperature(time);
    double time(time);
} // cesm_01

group: cesm_02 {
  :Scenario = "Historical";
  :Model = "CESM";
  :Realization = "2";
  dimensions:
    time=unlimited;
  variables:
    float temperature(time);
    double time(time);
} // cesm_02

group: cesm_03 {
  :Scenario = "Historical";
  :Model = "CESM";
  :Realization = "3";
  dimensions:
    time=unlimited;
  variables:
    float temperature(time);
    double time(time);
} // cesm_03
Discrete sampling geometries

:Conventions="CF-1.8 CF2-Group-1.0"
dimensions:
  time = unlimited;
variables:
  double time(time);
    time:standard_name = "time";
    time:units = "days since 1970-01-01 00:00:00";
group: irvine {
  variables:
    float humidity(time);
      humidity:standard_name = "specific humidity";
      humidity:coordinates = "lat lon alt station_name";
      humidity:_FillValue = -999.9f;
    float lon;
      lon:standard_name = "longitude";
      lon:units = "degrees_east";
    float lat;
      lat:standard_name = "latitude";
      lat:units = "degrees_north";
    float alt;
      alt:standard_name = "height";
      alt:units = "m";
      alt:positive = "up";
      alt:axis = "Z";
    string station_name;
      station_name:cf_role = "timeseries_id";
} // irvine

// Variables/dimensions repeated, omitted for clarity
} // boulder
Remote sensing channels with variable resolution

```plaintext
:Conventions="CF-1.8 CF2-Group-1.0"
group: data {
  dimensions:
    index = configured_value;
  variables:
    int mtg_geos_projection;
    mtg_geos_projection: grid_mapping_name = "geostationary";
    // ...
  group: vis_04 {
    variables:
      short x(x); // x coordinate variable
      x:standard_name = "projection_x_coordinate"
      x:unit = "radian"
      x:axis = "X"
      x:coordinates = "y x"
      short y(y); // y coordinate variable
      y:standard_name = "projection_y_coordinate"
      y:unit = "radian"
      y:axis = "Y"
      ushort effective_radiance(x,y);
      effective_radiance:standard_name = "effective_radiance_in_wavenumber"
      effective_radiance:units = "mW.m-2.sr-1.(cm-1)-1"
      effective_radiance:coordinates = "x y"
      effective_radiance:grid_mapping = "mtg_geos_projection"
    } // vis04
    // Other channels. All share mtg_geos_projection but define their own x and y.
  group: ir_105 {
  } // ir_105
} // data
```
9:30 AM Thursday Ventana Room...
Referencing Elements in Other Groups

- Attributes that reference variables include: \texttt{ancillary\_variables}, \texttt{bounds}, \texttt{coordinates}, \texttt{cell\_measures}, \texttt{formula\_terms}, \texttt{grid\_mapping}

- In-Group references same as CF1

- Out-of-Group reference options:
  - Ancestor-then-Lateral Search ('lat')
  - Relative Paths ('../../geo/lat')
  - Absolute Paths ('/geo/lat')
:Conventions="CF-1.8 CF2-Group-1.0"
dimensions:
  time=unlimited;
  lat=180;
  lon=360;
group: sci {
  variables:
    time(time);
    lat(lat);
    lon(lon);
    float temperature(time,lat,lon);
    temperature:coordinates="time lat lon";
} // sci
Out-of-Group References I: Absolute Paths

:Conventions="CF-1.8 CF2-Group-1.0"

dimensions:
  time=unlimited;
  lat=180;
  lon=360;

group: geo {
  group: g1 {
    variables:
      time(time);
      lat(lat);
      lon(lon);
  } // geo/g1
} // geo

group: sci {
  group: g1 {
    variables:
      float flux(time, lat, lon);
      flux:coordinates="/geo/g1/time /geo/g1/lat /geo/g1/lon";
  } // sci/g1
} // sci
Out-of-Group References II: Relative Paths

:Conventions="CF-1.8 CF2-Group-1.0"
dimensions:
  time=unlimited;
  lat=180;
  lon=360;
group: geo {
  group: g1 {
    variables:
      time(time);
      lat(lat);
      lon(lon);
  } // geo/g1
} // geo
group: sci {
  group: g1 {
    variables:
      float flux(time, lat, lon);
      flux:coordinates="../../geo/g1/time ../../geo/g1/lat ../../geo/g1/lon";
  } // sci/g1
} // sci
:Conventions="CF-1.8 CF2-Group-1.0"
dimensions:
  time=unlimited;
  lat=180;
  lon=360;

group: geo {
  group: g1 {
    variables:
      time(time);
      lat(lat);
      lon(lon);
  } // geo/g1
} // geo

group: sci {
  group: g1 {
    variables:
      float flux(time, lat, lon);
      flux:coordinates="time lat lon";
  } // sci/g1
} // sci
:Conventions="CF-1.8 CF2-Group-1.0"
dimensions:
  time=unlimited;
  lat=180;
  lon=360;
group: geo {
  group: g1 {
    variables:
      time(time);
      lat(lat);
      lon(lon);
  } // geo/g1
} // geo

group: sci {
  group: g1 {
    variables:
      float flux(time, lat, lon);
      flux:coordinates="time lat lon"; <--Same Syntax as CF1!
  } // sci/g1
} // sci
Search for out-of-group items

1. Search in group
2. Search direct ancestors
3. Search levels width-wise

/geo
/meta
/sci
/other

/geo/g1
/geo/g2
/meta/g1
/sci/g1
/sci/g2

/geo/g1/g1
/meta/g1/g1
/meta/g1/g2
/sci/g1/g1
/sci/g1/g2
The way here and the way forward

The work so far...

- **September 2013:**
  - CF Mail list
  - “Towards recognizing and exploiting hierarchical groups”
  - Discussion leading to creation of guiding principals

- **June 2017:**
  - Google Doc
  - “CF-2 Group: Draft Extension for Files with Groups”
  - Additional contributions from community

- **September 2017:**
  - EarthCube netCDF-CF Workshop 2
  - “CF-2 Group: Hierarchical data and Metadata Extensions to Climate/Forecast Conventions”
  - Presentation garners further input

- **April 2018:**
  - netCDF Operators
  - Support for using CF2-Group proposal implemented as of v4.7.4.

- **May 2018:**
  - Github
  - NASA Dataset Interoperability Working Group Frameworks
  - Iterations through issues and pull requests for greater traceability

...And now?

- **June 2018:**
  - EarthCube netCDF-CF Workshop 3
  - Convergence?
1. CF2-Groups is ready to use

2. Feedback welcome as GitHub issues and/or PRs

3. Will propose to CF this summer