# External variables in CF

# External variables in CF

- **Then:** In CF-1.6, all metadata variables must be in the same file as their associated data variable

# External variables in CF

- **Now**: Trac ticket **#145**, implemented in CF-1.7, allows cell measure variables to be in a different file

- The motivation for this was that CMIP6 is going to store cell measure variables in different files regardless, and nobody wanted petabytes of non-CF-compliant data to be archived for next *n* years

- **Jonathan Gregory**, **Balaji**, John Caron, Steve Hankin, David Hassell, Martin Juckes, Roy Lowry, Seth McGinnis, Martin Schultz, Karl Taylor

# Implementation

```
dimensions:
  lat=73;
  lon=96;
  level=20;
variables:
  float temperature(level,lat,lon);
    temperature:cell_measures="area: areacell";
    temperature:standard_name="air_temperature";
    temperature:units="degC";
// global attributes:
  :external_variables="areacell";
```

```
dimensions:
  lat=73;
  lon=96;
variables:
  float areacell(lat,lon);
    areacell:units="m2";
```

# Implementation

- Connection between files is by variable-name-indirection

- No information is provided on where to find the missing variable
  - URL, URN, filenames are all brittle, DOI less so but has a large overhead

# Objections to going further than cell measures

- **1.1** *"The purpose of the CF conventions is to require conforming datasets to contain sufficient metadata that they are* ***self-describing in the sense that*** *each variable in the file has an associated description of what it represents, including physical units if appropriate, and that* ***each value can be located in space****"*

- Coordinates and "domain ancillary" variables are essential for geolocation, but cell measures (and "field ancillary" variables) are not
  - Coordinates and "domain ancillaries" were to remain in the file for geolocation, "field ancillaries" also to remain just to keep things simple (no use case)

# Issues

- You don’t know where the external variable is
  - CMIP6 users will receive guidance outside of CF

- Even when you know where it is, diligence is required
  - e.g. a subspace of the data variable has to be carefully applied to the external variable as well
  - software could cope probably cope with this (e.g. lazy operations), but it is not trivial

# Where next? Some ideas

- Nothing - #145 is fine for now
- Modify the motivation – is it just to save space?
- Extend for auxiliary coordinate variables
- Allow 2-d lats and lons to be missing with a grid mapping?
- Extend for any metadata variables
- Extend to a whole, self-contained “domain”
    - A prerequisite for this would be to create a “domain construct” in CF
    - Use case: geometries?