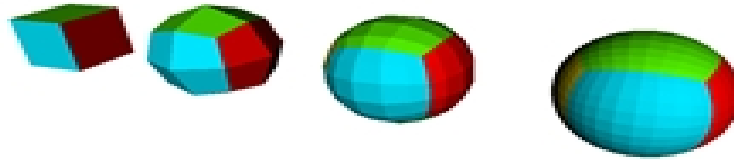# Adding mosaic grid support to LibCF

Alex Pletzer and Dave Kindig (Tech-X) - LibCF/GRIDSPEC
Ed Hartnett (UCAR) – LibCF and NetCDF
V Balaji and Zhi Liang (GFDL) – Mosaic and GRIDSPEC
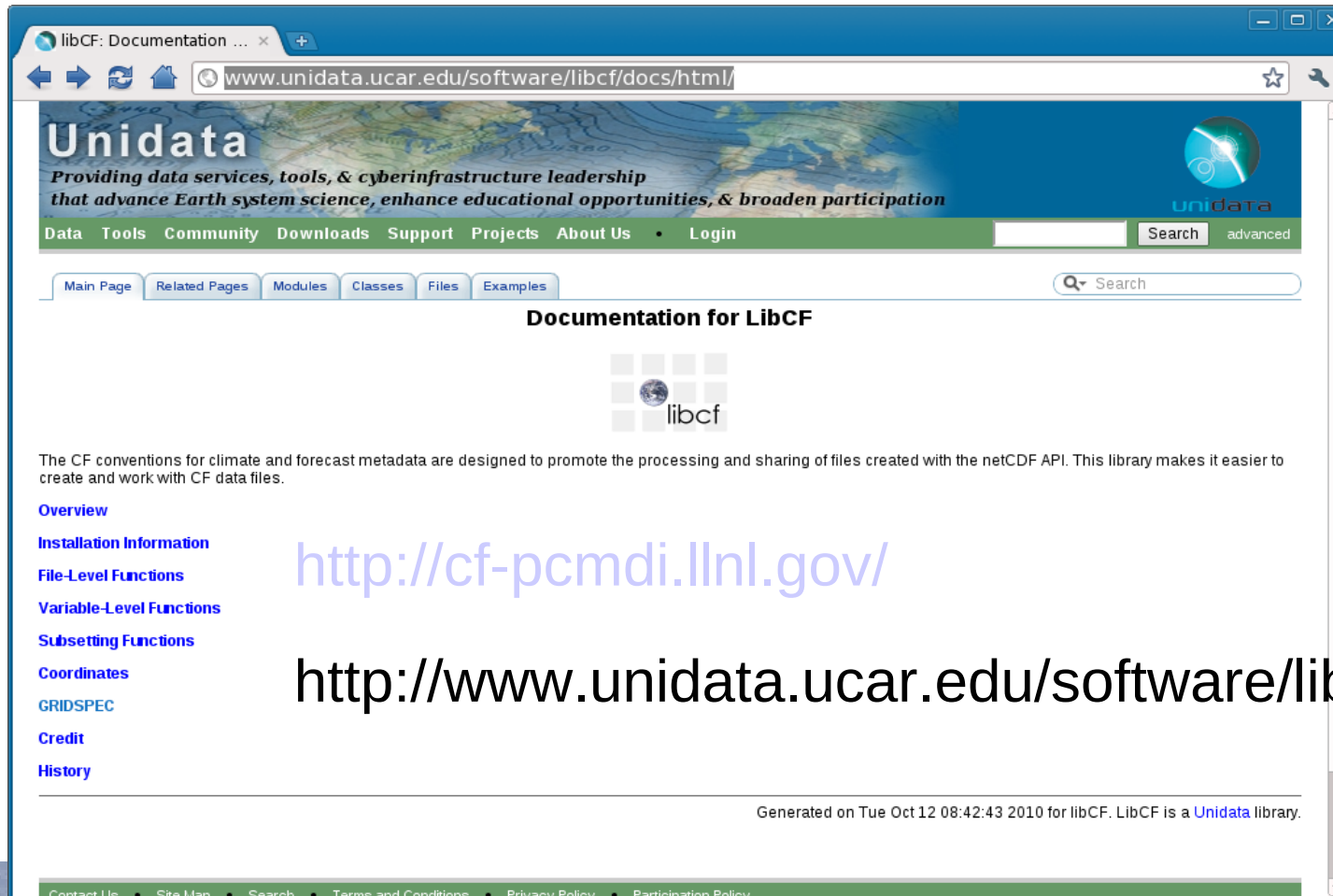**Charles Doutriaux, Jeff Painter, and Dean Williams (LLNL) – CDAT, CMIP5**
Steve Hankin and Ansley Manke (PMEL) – CF, Ferret

Oct 29 2010, Netcdf Workshop, Boulder CO

What is libCF?

- **Library implementing the Climate and Forecasting (CF) Netcdf metadata conventions**

http://cf-pcmdi.llnl.gov/

http://www.unidata.ucar.edu/software/libcf/docs/

Climate models (atm and ocean) are moving away longitude-latitude grids

- **Pole singularities lead to:**
  - **Problems of numerical stability (Courant condition violated when using explicit time stepping. May need to apply filtering techniques to damp oscillations at the poles.**
  - **"Over"-resolution near the poles (waste of resources)**
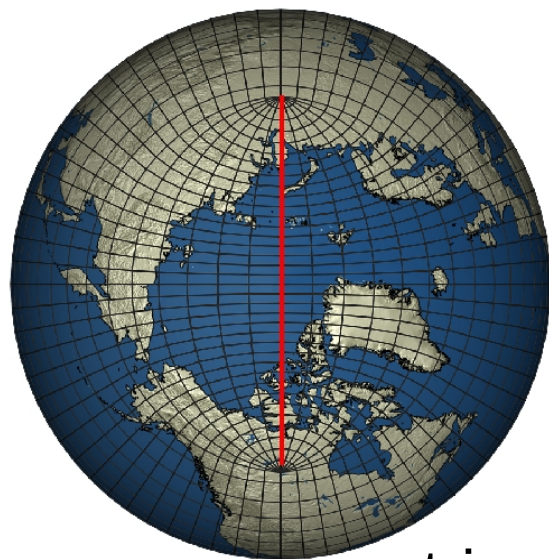
cubed-sphere
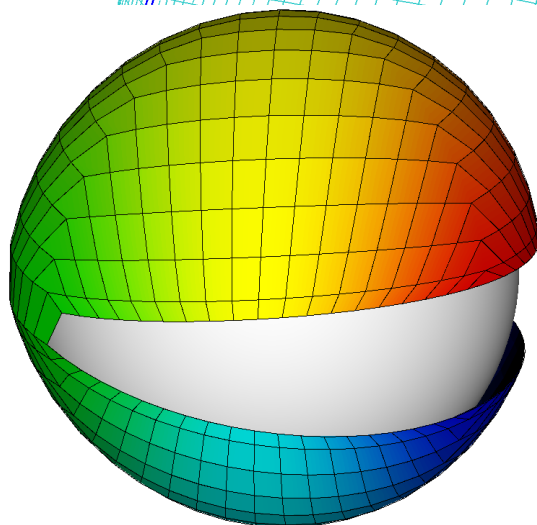
tripolar   Calhoun-Helzel-LeVeque
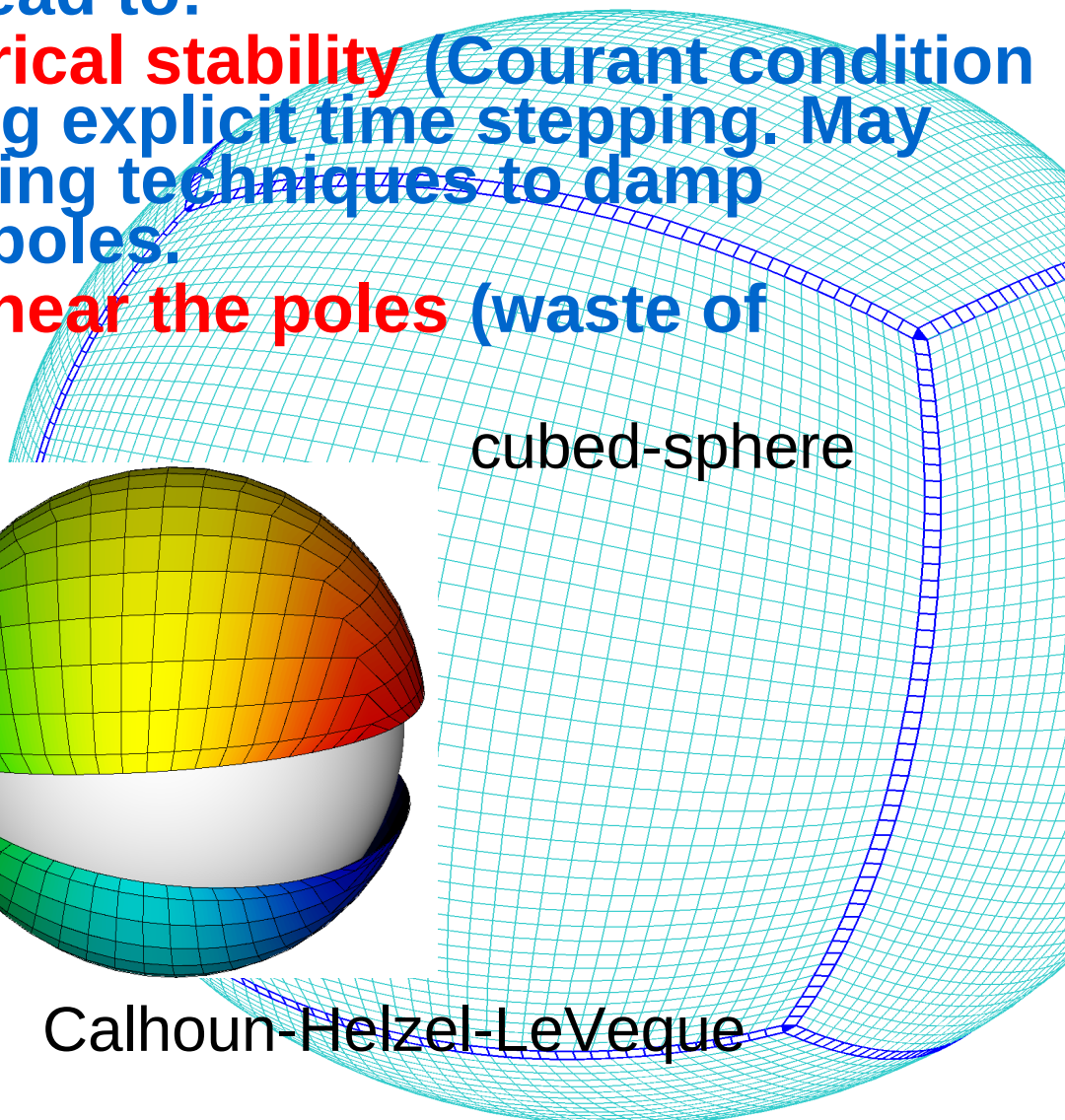
All these grids (tripolar, cubed-sphere, ...) share a common feature

- **Largely or entirely block-structured**
- **Complex folding between blocks**
- **Unstructured assembly of structured grids**
- **Many more variations beyond cubed-sphere and tripolar**
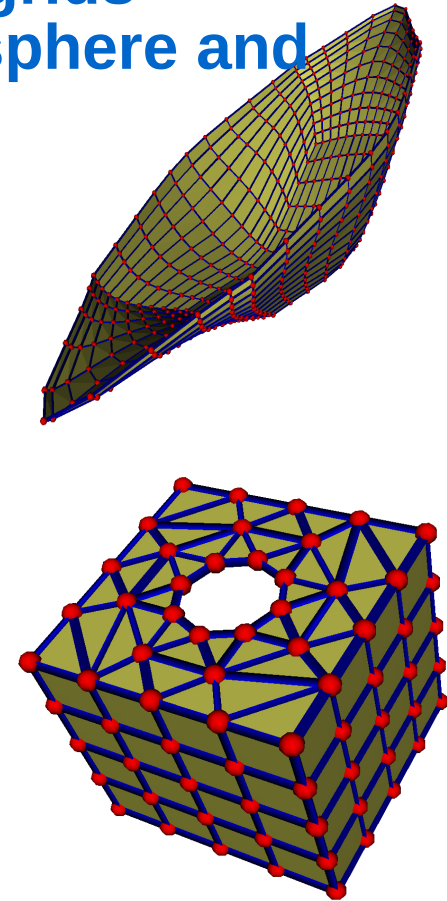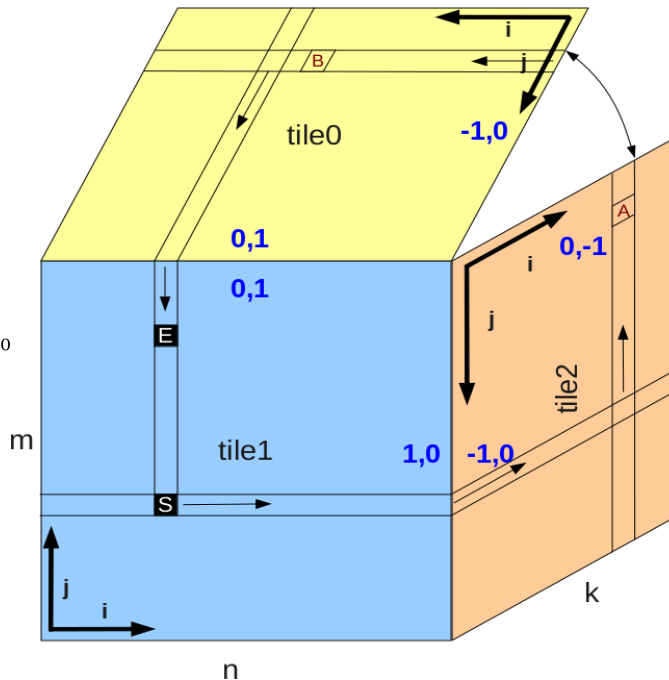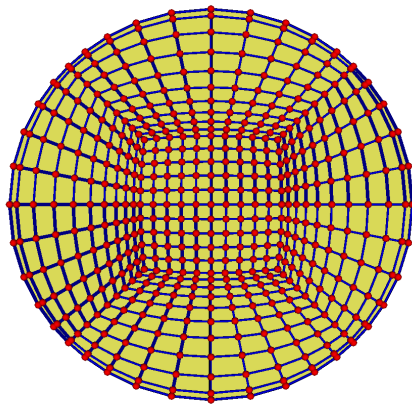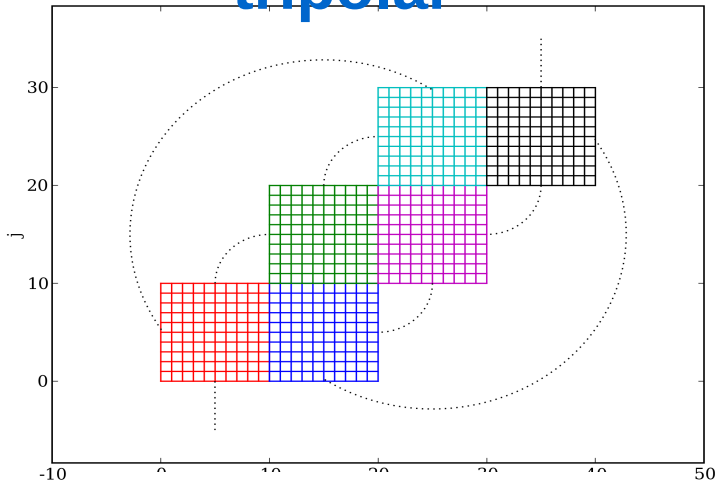
LibCF implements mosaics by adding a minimal set of extensions to the CF conventions (partial GRIDSPEC)

- **Each tile is a structured (but warped) grid**
- **Arbitrary number of tiles**
- **Arbitrary number of (space) dimensions**
- **Grid coordinates and data need not be in the same file**

Host file

Mosaic file

Data1   Grid1

Data2   Grid2

Data3   Grid3
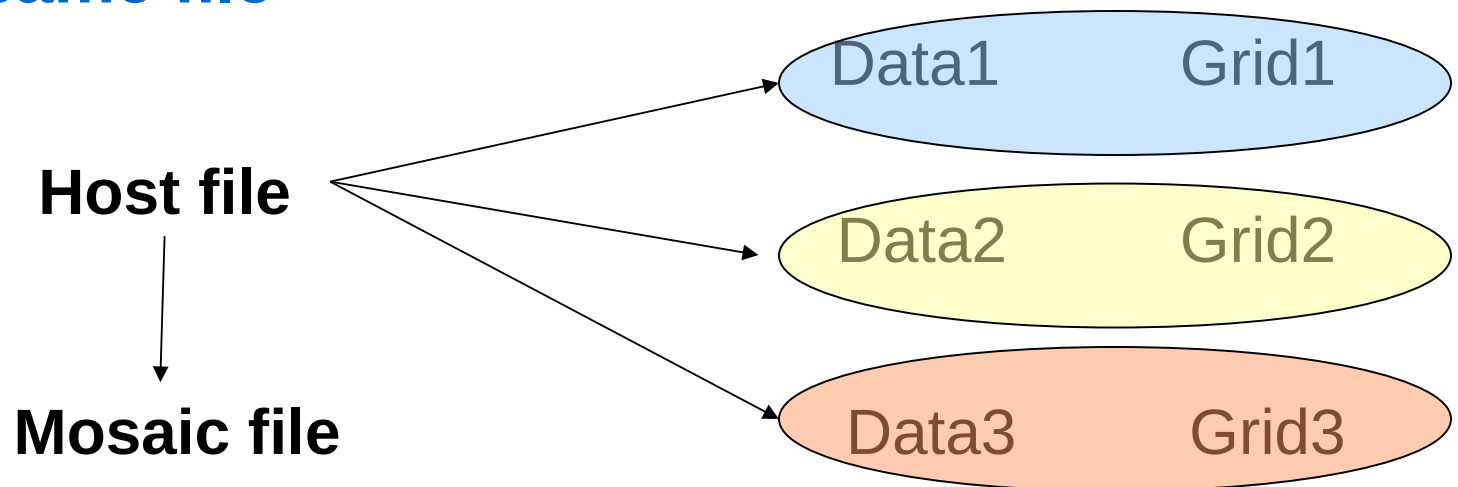
Host file aggregates all static and time dependent data files, mosaic file contains grid connectivity information

LibCF/GRIDSPEC API uses a layered approach to represent coordinates, grids, data, mosaic, and host

Each object only depends on objects below

host		regrid
mosaic	data	
grids		
coordinates		
tools		
CF		
NetCDF		

LibCF/GRIDSPEC API

- Code written in C per NetCDF standard.
  - Extensible to other languages (arguments are primitive types)

- Uses uuid (coord_system_id) to track a unified data set (grid, data, mosaic, host)

- Define, write and free
  - nccf_def_XXXX(..., &id);
  - nccf_put_XXXX(ncid, id);
  - nccf_free_XXXX(id);

- Define from file
  - nccf_def_XXXX_from_file(filename,..., &id);

```c
/* Define the Grid */
nccf_def_lon_coordinate( 2, dims, dimNames, lon, save, &coordid[0]);
nccf_def_lat_coordinate( 2, dims, dimNames, lat, save, &coordid[1]);
nccf_def_structured_grid(coordid, name_of_grid, coord_system_id, &gridid);
nccf_put_structured_grid(ncid, gridid);
nccf_add_structured_grid_global_att(mosaicid, CF_COORD_SYSTEM_ID, coord_system_id);

/* Define mosaic */
nccf_def_mosaic(ntiles, gridid, mosaic_name, periods, &mosaicid);
nccf_put_mosaic(ncid, mosaicid);
nccf_add_mosaic_global_att(mosaicid, CF_COORD_SYSTEM_ID, coord_system_id);

/* Define some data */
nccf_def_structured_data(gridid, data_var_name, var_standard_name, var_units, NULL,
&staticdataid);
nccf_def_structured_data(gridid, data_var_name, var_standard_name, var_units,
var_time_dimname, &timedataid);nccf_add_host_file(hostid, grid_filename, 0);
nccf_put_structured_data(ncid, staticdataid);
nccf_put_structured_data(ncid, timedataid);

/* Define Host */
nccf_def_host(coord_system_id, &hostid);
nccf_add_host_file(hostid, mosaic_filename,0);
nccf_add_host_file(hostid, grid_filename, 0);
ccf_add_host_file(hostid, timedata_filename, 0);
nccf_add_host_file(hostid, staticdata_filename, 0);
nccf_put_host(ncid, hostid);
```

TECH

```c
/* Read host file */
nccf_def_host_from_file(host_filename, hostid);

nccf_get_timedata_from_host(hostid,timedata_filename);
nccf_inq_host_ngrids(hostid, &ngrids);
nccf_inq_host_gridids(hostid, gridids);

nccf_get_staticdata_from_host(hostid, staticdata_filename);
nccf_inq_host_nstaticdata(hostid, &nstaticdata);
nccf_inq_host_staticdatas(hostid, staticdataids);

nccf_get_structured_data_from_host(hostid, grid_filename);
nccf_inq_host_ntimedata(hostid, &ntimedata);
nccf_inq_host_timedata(hostid, timedataids);

nccf_get_mosaic_from_host(hostid, mosaic_filename);
nccf_inq_host_mosaicid(hostid, mosaicid);

/* Do stuff */

/* Free memory */
nccf_free_sturctured_grid(grididid);
nccf_free_mosaic(mosaicidid);
nccf_free_structured_data( timedataid);
nccf_free_structured_data( staticdataidid);
nccf_free_host( hostid);
```

TECH

Regridding/interpolation is work in progress

- **Linear interpolation using nearest neighbors only**
  - **No over-shooting**
  - **Straightforward to parallelize**
- **Pseudo-Newton search of position in index space**
  - **Only one iteration required for uniform, rectilinear grids**
- **Line search to improve convergence**
- **Use previous index location as initial guess when regridding from structured to structured grid**
- **"Snake" iterator to navigate coordinate data, hopping to nearest from one vertex to neighbor vertex**
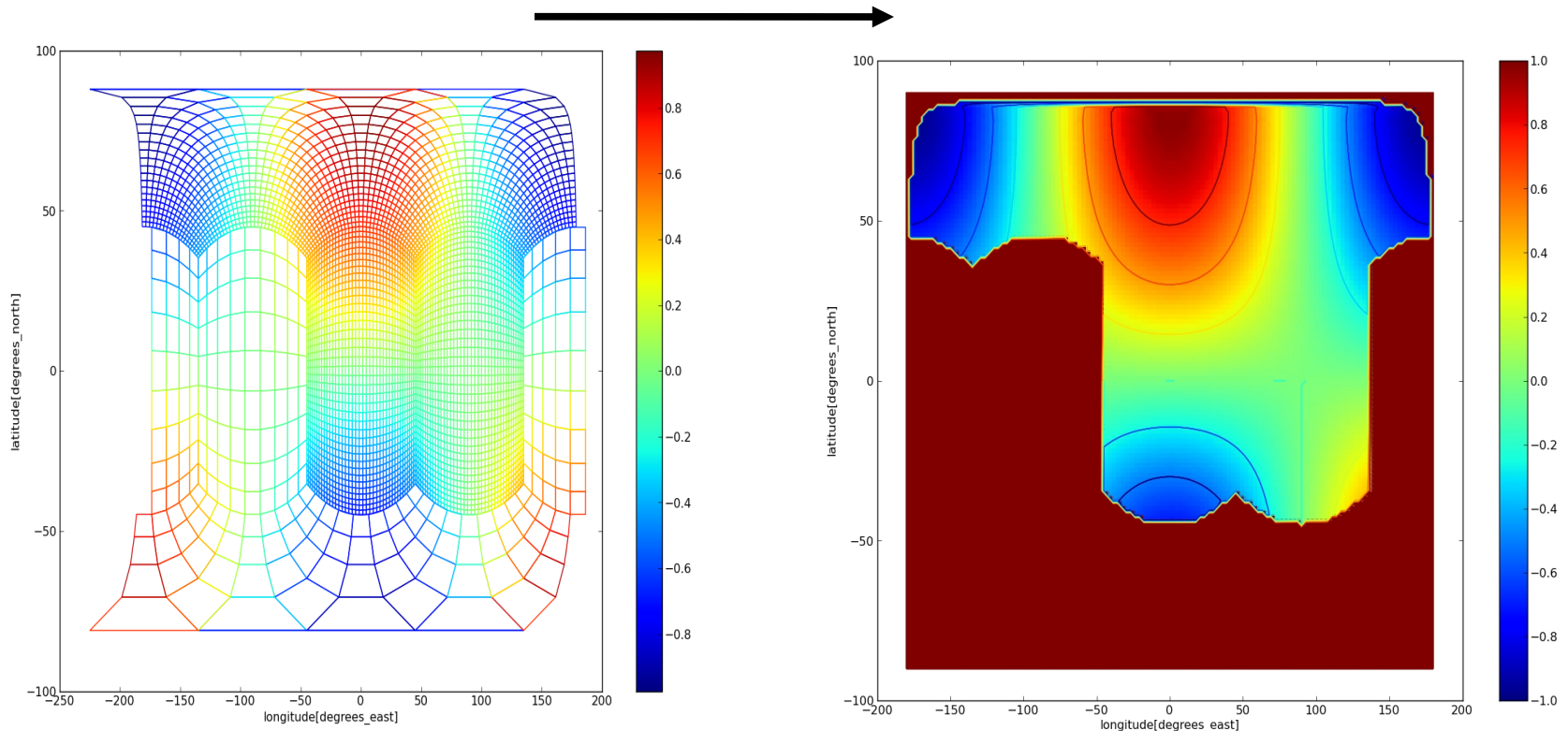
TECH

Test case 1: bipolar cap data regridded onto uniform lon-lat grid

- Bi-polar cap defined for lat >= 60 deg north
- 2 poles at lat = 60 deg N, lon = 0, +/- 180 deg
- Coordinate cut at -/+ 180 deg
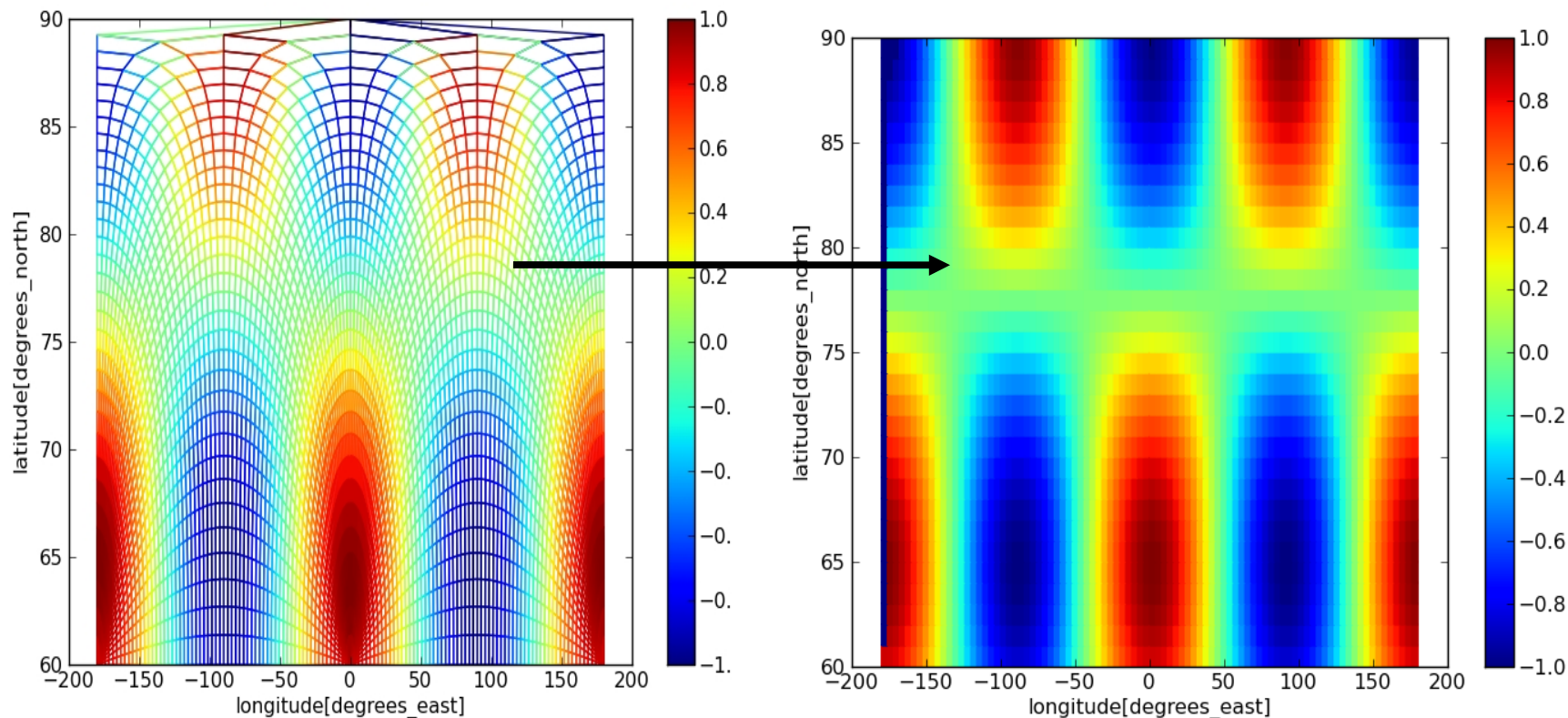
Test case 2: cubed-sphere grid to lon-lat

- Three tiles, some target points not in original grid
- Lon-lat grid covers entire globe

To do list/future directions

- More required to improve robustness of interpolation
- Better integration between LibCF/GRIDSPEC and the remainder of LibCF needed

Will need support for virtual files (in memory data access) in NetCDF

- Proposing GRIDSPEC extensions to CF
  - Multi-file aggregation
  - Mosaic file connectivity
  - Support for staggered data
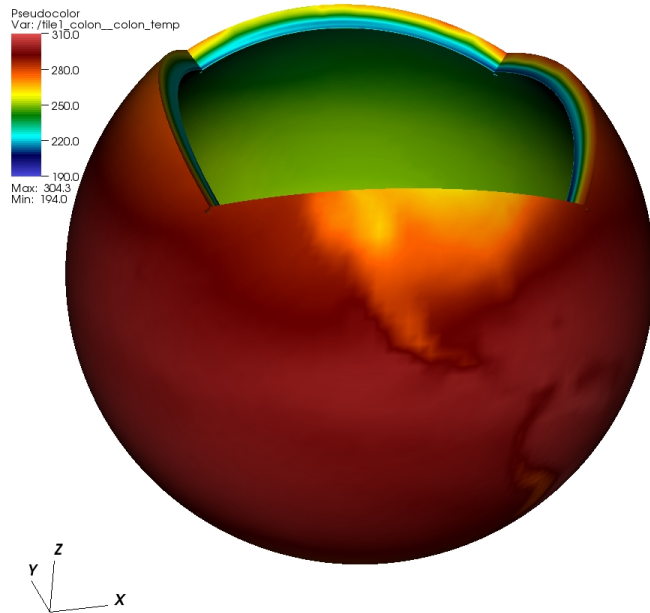   - Need to support face and edge centered data (Arakawa C/D)

Supergrids?

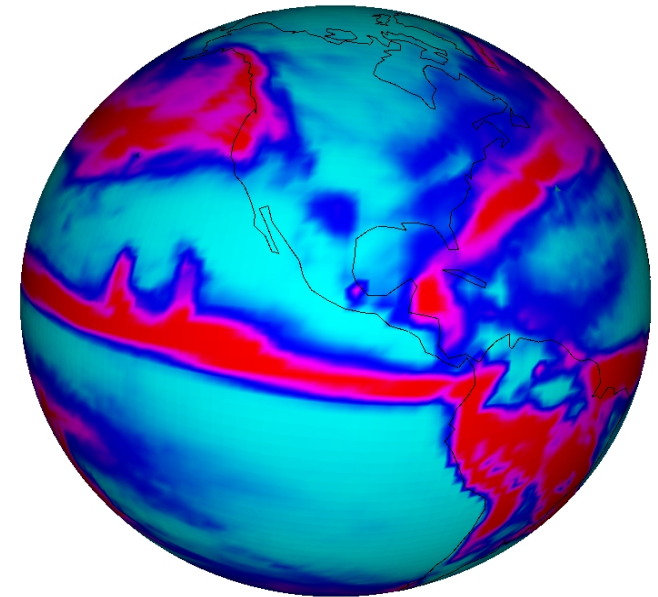Integration of libCF/GRIDSPEC into data analysis tools (CDAT, Ferret, NCL, ...)

- Become an integral part of the NetCDF library

TECH

Would need help

- Need alpha users to provide use case, feedback, set priorities...

precip @ 0 .

MoDAVE