

Dynamical Core Model Intercomparison Project (DCMIP2016)

Test Case Document

Paul Aaron Ullrich
Christiane Jablonowski
Kevin A. Reed
Colin Zarzycki
Peter Hjort Lauritzen
Ramachandran D. Nair
James Kent
Antonin Verlet-Banide

DCMIP Summer School: June 2016

Version 0.3
May 11, 2016

Email questions to **dcmip@ucar.edu**

OVERVIEW OF THE TEST CASE SUITE

The set of test cases collected in this document has been developed for the 2016 Dynamical Core Model Intercomparison Project (DCMIP2016) in an effort to understand the broad treatment of the equations of motion within a variety of atmospheric General Circulation Models (GCMs). In contrast with the test cases described in the DCMIP2012 and DCMIP2008 test case documents, the suite of test cases that are described here focus on issues related to physics-dynamics coupling, non-hydrostatic scales and modeling systems that support variable resolution. To better isolate these features, the majority of tests described in this document do not include topographic forcing and emphasize features that have a relatively small spatial footprint relative to the global simulation domain. The tests proposed here have been drawn from the recent literature on global model intercomparison testing. To support the integration of these tests into existing models, a collection of initialization routines have been provided that can be used for very quickly setting up the initial conditions. Augmented Kessler physics routines have also been provided to support simple moisture feedbacks, a boundary layer parameterization and surface friction. If models provide both a shallow-atmosphere and deep-atmosphere configuration, we recommend the shallow-atmosphere setup to avoid imbalances in the initial conditions. It is further expected that most models will be run under a non-hydrostatic configuration, which will permit a correct solution to the supercell test. If models can be configured both as a hydrostatic and non-hydrostatic dynamical core, additional hydrostatic simulations might be conducted to evaluate the direct impact of the hydrostatic approximation. Table I provides an overview of all test cases described in this document.

Table I. A list of test cases described in this document.

Test Case	Description
Tier 1 Test Cases (Required)	
1-1	Moist baroclinic wave with terminator chemistry
1-2	Idealized tropical cyclone
1-3	Supercell
Tier 2 Test Cases (Optional)	
2-1	Moist Held-Suarez test

0. PRACTICAL CONSIDERATIONS

0.1. List of Symbols

Throughout this test case document we will use $\lambda \in [0, 2\pi)$ to denote longitude, $\varphi \in [-\pi/2, \pi/2]$ to represent latitude, z to represent the height with respect to the mean sea level (assumed to be zero), and p to symbolize the pressure. Table II lists the symbols used for the initialization of the model.

Table II. List of symbols for the model initialization

Symbol	Description
λ	Longitude (in radians)
φ	Latitude (in radians)
z	Height with respect to mean sea level (set to zero)
p_s	Surface pressure (p_s of moist air if $q > 0$)
Φ_s	Surface geopotential
z_s	Surface elevation with respect to mean sea level (set to zero)
u	Zonal wind
v	Meridional wind
w	Vertical velocity
ω	Vertical pressure velocity
δ	Divergence
ζ	Relative vorticity
p	Pressure (pressure of moist air if $q > 0$)
ρ	Density (density of moist air if $q > 0$)
T	Temperature
T_v	Virtual temperature
Θ	Potential temperature
Θ_v	Virtual potential temperature
q	Specific humidity
P_{ls}	Large-scale precipitation rate
q_c	Cloud water mixing ratio
q_r	Rain water mixing ratio
q_{Cl}	Singlet chlorine mixing ratio
q_{Cl2}	Chlorine gas mixing ratio

0.2. List of Physical Constants

A list of physical constants which are used throughout this document is given in Table III. Constants which are specific to each test case are similarly tabulated at the beginning of each section.

Table III. A list of physical constants used in this document.

Constant	Description	Value
a_{ref}	Radius of the Earth	$6.37122 \times 10^6 \text{ m}$
Ω_{ref}	Rotational speed of the Earth	$7.292 \times 10^{-5} \text{ s}^{-1}$
X	Reduced-size Earth reduction factor	variable (default = 1)
a	Scaled radius of the Earth	a_{ref}/X
Ω	Scaled rotational speed of the Earth	$\Omega_{\text{ref}} \cdot X$
g	Gravity	9.80616 m s^{-2}
p_0	Reference pressure	1000 hPa
c_p	Specific heat capacity of dry air at constant pressure	$1004.5 \text{ J kg}^{-1} \text{ K}^{-1}$
c_v	Specific heat capacity of dry air at constant volume	$717.5 \text{ J kg}^{-1} \text{ K}^{-1}$
R_d	Gas constant for dry air	$287.0 \text{ J kg}^{-1} \text{ K}^{-1}$
R_v	Gas constant for water vapor	$461.5 \text{ J kg}^{-1} \text{ K}^{-1}$
κ	Ratio of R_d to c_p	$2/7$
ε	Ratio of R_d to R_v	0.622
M_v	Constant for virtual temperature conversion	0.608
ρ_{water}	Density of water	1000 kg m^{-3}

0.3. Great Circle Distance

The great circle distance is used throughout the document and is given by

$$R_c(\lambda_1, \varphi_1; \lambda_2, \varphi_2) = a \arccos(\sin \varphi_1 \sin \varphi_2 + \cos \varphi_1 \cos \varphi_2 \cos(\lambda_1 - \lambda_2)). \quad (1)$$

0.4. Choice of Prognostic Variables and Equation of State

It is assumed throughout this document that the moist ideal gas law is satisfied,

$$p = \rho R_d T_v, \quad (2)$$

where the virtual temperature is defined in terms of the temperature as

$$T_v = T(1 + M_v q). \quad (3)$$

The virtual potential temperature is similarly related to the potential temperature via the relationship

$$\theta_v = \theta(1 + M_v q). \quad (4)$$

0.5. Small-Planet Experiments

The test case suite makes use of small-planet experiments that have the potential to expose the differences between hydrostatic and non-hydrostatic modeling approaches at reasonable computational cost. In particular, the small-planet setups allow the evaluation of the model behavior with physical grid spacings down to a few hundred meters. In some instances, we suggest small planets with circumferences of about 40 km and a vertical extent of 30 km which raises questions concerning the validity of the shallow-atmosphere approximation. However, since the experiments are not compared to observations, we still ask for the use of the shallow-atmosphere approach to allow for intercomparisons among the DCMIP models and to avoid imbalances of the initial conditions.

When a non-unity reduction factor X is applied in order to shrink the size of the Earth and thereby the physical grid spacing of the computational grid, a variety of model adjustments become necessary. Most prominently these include the scaling of the radius, the rotational speed, the model time step and explicit viscosity parameters (if applied). The adjustment steps for small-planet simulations are:

- Divide the radius of the Earth a_{ref} by X to obtain the rescaled radius $a = \frac{a_{\text{ref}}}{X}$.
- Divide the length of the dynamics time step Δt by X , especially if a CFL condition needs to be obeyed.
- In case of rotating planets: multiply the Earth's angular velocity Ω_{ref} by the factor X to obtain the rescaled angular speed $\Omega = \Omega_{\text{ref}} X$. This guarantees that the characteristics of Rossby waves are comparable in unscaled and scaled model experiments since the Rossby number stays constant.
- In case of explicit diffusion of type $K_{2k} \nabla^{2k}$ with a prescribed diffusion coefficient K_{2k} (and $k = 1, 2, \dots$) divide K_{2k} by the factor X^{2k-1} . This accounts for a reduction of the e-folding

time τ and the horizontal grid spacing Δx according to the relationship $\frac{(\Delta x)^{2k}}{\tau/X^{2k}}$. The K_{2k} diffusion coefficient is typically based on such a relationship. Note that some models might provide an automatic scaling of the diffusion coefficients according to the actual dynamics time step and grid spacing. If a model applies Rayleigh friction as a sponge near the model top, the friction coefficient needs to be multiplied by X. Again, this corresponds to a reduction of the e-folding frictional time scale $\frac{1}{\tau/X} = \frac{X}{\tau}$ in small-planet experiments.

- If physical forcing mechanisms are present on the right hand side of the equations of motion the strengths of the physical forcing must be increased (multiplied) by the factor X. This will not be applicable to the test cases presented here unless Rayleigh friction is considered as such a forcing as outlined above.
- Any deviations from this prescription should be documented. For example, some modeling groups may have a prescribed diffusion coefficient that depends on grid resolution that can be used in place of the scaled diffusivity described above.

0.6. Notes on the Requested Model Output

0.6.1. NetCDF

A fundamental requirement for the exchange of scientific data is the ability to precisely describe the physical quantities being represented. Therefore, special attention needs to be paid to the representation of the model data in the output files. We require data in the ‘Network Common Data Form’ (netCDF) [1] that adhere to the netCDF Climate and Forecast (CF) metadata convention (if possible to version 1.6 from Dec. 2011 [2]). All netCDF files should have the file name extension ‘.nc’. We specify details of the netCDF requirements in Appendix F and will also provide help via NCO operators before the DCMIP event to make the data CF-compliant if necessary. Please communicate your output constraints or concerns (if any) to the DCMIP organizers as soon as possible.

0.6.2. Naming convention for output file names

The naming convention for the final output file names is:

model.test_case.horizontal_resolution.levels.grid.equation.description.nc

NetCDF-CF compliant files with these naming conventions will be uploaded to the Earth System Grid Federation. The optional free-text character string at the end of the file name (“description”) might be used to denote a special setting, such as a varied non-default diffusion coefficient or the selection of a variable-resolution configuration. The standard keywords for the DCMIP models are listed in the left columns of Tables IV and V. The keyword for the number of vertical levels, e.g. “L30”, denotes the number of the full model levels (here 30). In case a model also employs interface levels, this setting indirectly implies that the model has one more interface level. Examples are

cam-se.161.r200.L60.latlon.hydro.4th_order_div_damping.nc

tempest.162.r100.L120.interp_latlon.nonhydro.nc

fv3.162.r50.L30.cubed.nonhydro.nc

cam-se.162.r50.L30.interp_latlon.hydro.double_K4.2e15.nc

cam-se.163.r25.L30.interp_latlon.hydro.variable_220_25km.nc

The last example expresses the suggested naming convention for a variable-resolution simulation. The keyword ‘r25’ denotes the finest grid spacing in the variable-resolution run which is, in this example, interpolated to a regular latitude-longitude grid with a fine grid spacing of about 28 km. The free-text ‘description’ contains the suggested keyword ‘variable’ and specifies details about the resolution range in the variable-resolution run. In addition to such an interpolated data set, the original output file of a variable-resolution simulation should also be archived. Its name should be ‘cam-se.163.r25.L30.cubed.hydro.variable_220_25km.nc’ for the example above, where the specification ‘cubed’ for the underlying base grid has been chosen.

Table IV. Keywords for the output file naming convention.

model	Organization
cam-se	DOE/SNL & NCAR
chombo	DOE/LBNL & NCAR
csu	Colorado State University
dynamico	IPSL
endgame	UK Met Office
gem	Environment Canada
homme-nh	DOE/SNL & CU Boulder
fv3	NOAA GFDL
icon	MPI & DWD
ifs-fv	ECMWF
neptune	NRL & NPS
nicam	RIKEN & University of Tokyo
mpas	NCAR & Los Alamos National Laboratory
olam	University of Miami
tempest	University of California, Davis

0.7. Short Note on Data Analysis and Visualization

We will provide NCAR Command Language (NCL) scripts to help visualize the model results and provide analysis functions. In addition, the DCMIP participants will have access to interactive Graphical User Interfaces (GUIs) to support the visualization and model intercomparison. Among the GUIs are the netCDF viewers Ncview and Panoply which are public domain tools and locally installed on the NCAR mirage server. In addition, we expect to provide some basic online visualization capabilities via NOAA’s Live Access Server (LAS) software. A key to the successful visualization is the adherence of the output data sets to the netCDF-CF standard (see Appendix F).

0.7.1. Computational grid

Most DCMIP models utilize non-orthogonal computational grids like cubed-sphere grids, icosahedral grids, hexagonal grid, Voronoi grids or Yin-Yang grids. Among the mix of DCMIP GCMs are even models that provide provisions for variable-resolution grids. We encourage the use of the variable-resolution configurations as an additional test option whenever possible. This raises questions concerning the desired representation of the data in the netCDF output files.

Table V. Continuation of the list of keywords for the output file naming convention: Grid specifications.

test_case	Description of the test case number
161	Moist baroclinic wave
162	Idealized tropical cyclone
163	Supercell
horizontal_resolution	Approximate grid spacing in degrees or equatorial grid spacing (unscaled)
r200	between 2°-3.0° or 200-300 km (on unscaled planet)
r100	between 1°-1.5° or 100-150 km (on unscaled planet)
r50	between 0.5°-1.0° or 50-100 km (on unscaled planet)
r25	between 0.25°-0.5° or 25-50 km (on unscaled planet)
levels	Number of vertical levels (full model levels)
L10	10 levels
L15	15 levels
L20	20 levels
L30	30 levels
L60	60 levels
L120	120 levels
grid	Type of native grid or indicator of interpolated grid
cubed	cubed sphere
hex	hexagonal grid based on an icosahedron, maybe optimized via spring dynamics
gauss	Gaussian grid
tri	triangular grid based on an icosahedron, maybe optimized via spring dynamics
interp_latlon	interpolated latitude-longitude grid
latlon	latitude-longitude
oct	octagonal
red_latlon	reduced latitude-longitude grid
red_gauss	reduced Gaussian grid
voronoi	spherical centriodal Voronoi tessellation
yinyang	Yin-Yang
equation	Indicator for hydrostatic or non-hydrostatic simulation
hydro	hydrostatic
nonhydro	non-hydrostatic

If models on non-traditional (non latitude-longitude) grids are used, we ask for two output files that represent the identical model run. The first output file should be written on the native computational grid without any interpolations. In addition, most models will likely provide built-in provisions for interpolated output to a regular (equidistant in degrees) latitude-longitude grid. We therefore also ask for a second output file that represents the data on model levels on the interpolated latitude-longitude grid. We ask for co-located (Arakawa-A type) data on the interpolated grid regardless of the GCM's staggering options. The grid spacing of the interpolated grid should be comparable to the actual resolution of the model run, which might be for example $1^\circ \times 1^\circ$. Using this example, the interpolated grid will have 180×360 horizontal grid points if the equator and pole points are not part of the interpolated grid (equally spaced in the longitudinal and latitudinal

directions). If the equator and pole points are included it yields 181×360 horizontal grid points. The 180×360 configuration is preferred for model intercomparison purposes.

Models on regular latitude-longitude or Gaussian grids should only provide a single output file using their native horizontal resolution and model levels. If reduced Gaussian grids are utilized a second file on the full Gaussian grid is requested. If models are run with variable-resolution grids, we leave the choice of the best suitable interpolation grid to the modeling group. We ask to write all output variables for each experiment to the same file.

0.8. Short Note on Data Analysis and Visualization

We will provide NCAR Command Language (NCL) scripts to help visualize the model results and provide analysis functions. In addition, the DCMIP participants will have access to interactive Graphical User Interfaces (GUIs) to support the visualization and model intercomparison. Among the GUIs are the netCDF viewers Ncview and Panoply which are public domain tools and locally installed on the NCAR mirage server. In addition, we expect to provide some basic online visualization capabilities via NOAA's Live Access Server (LAS) software. A key to the successful visualization is the adherence of the output data sets to the netCDF-CF standard (see Appendix F).

0.9. Short Note on the Fortran Templates

We have provided a set of stand-alone Fortran routines that compute the initial conditions for all test cases. They are named

- `baroclinic_wave_test.f90`
- `tropical_cyclone_test.f90`
- `supercell_test.f90`

We have also provided a set of stand-alone Fortran routines that compute tendencies associated with the physical parameterizations. They are named

- `terminator.f90`
- `kessler.f90`

1. MOIST BAROCLINIC WAVE

The baroclinic instability test of [3] considers a reference state in geostrophic and hydrostatic balance that satisfies the conditions for baroclinic instability. Although a perfect model should be able to maintain this state indefinitely, small truncation errors associated with numerical inaccuracies and grid structure will trigger the development of the wave modes associated with baroclinic development. To control the development of the baroclinic wave, a small perturbation (but one which is large compared with machine truncation) is added to the flow so as to trigger the development of a wave over a period of approximately 10 days. A moist variant of the dry dynamical test of [3] is considered here so as to understand the impact of moisture feedbacks on the development of the wave.

This test case is similar in character to the test of [4], but has a number of key differences: (1) this test is an analytical solution of the equations of motion in height (z) coordinates, (2) the bottom topography is zero throughout the domain, (3) the new test case does not have a distinct stratosphere (the presence of a stratosphere is largely irrelevant for understanding baroclinic development), and (4) the velocity field goes to zero at the model surface.

Table VI. List of constants used for the Moist Baroclinic Wave test case

Constant	Value	Description
z_{top}	44000 m	Recommended height position of the model top
p_{top}	≈ 2.26 hPa	Recommended pressure at the model top
X	1	Reduced-size planet scaling factor, see below
a	a_{ref}/X	Scaled radius of the Earth
Ω	$\Omega_{\text{ref}}X$	Scaled angular speed of the Earth
p_s	1000 hPa	Surface pressure (constant)
p_0	1000 hPa	Reference pressure (constant)
u_0	35 m s^{-1}	Maximum amplitude of the zonal wind
b	2	Half-width parameter
K	3	Power used for temperature field
T_E	310 K	Horizontal-mean temperature at the surface
T_P	240 K	Temperature at the polar surface
u_p	1 m s^{-1}	Maximum amplitude of the zonal wind perturbation
z_p	15000 m	Maximum height of the zonal wind perturbation
λ_p	$\pi/9$	Longitude of the zonal wind perturbation centerpoint (20° E)
φ_p	$2\pi/9$	Latitude of the zonal wind perturbation centerpoint (40° N)
R_p	$a/10$	Radius of the zonal wind perturbation
Γ	0.005 K m^{-1}	Temperature lapse rate
ΔT	$4.8 \times 10^5 \text{ K}$	Empirical temperature difference
φ_w	$2\pi/9$	Specific humidity latitudinal width parameter (40°)
p_w	340 hPa	Specific humidity vertical pressure width parameter
q_0	0.018 kg/kg	Maximum specific humidity amplitude
q_t	$1.0 \times 10^{-12} \text{ kg/kg}$	Specific humidity above artificial tropopause
p_t	10000 hPa	Pressure at artificial tropopause

1.1. Reference State

This section describes the analytical form of the reference state for the baroclinic wave. The test case is initialized with a constant surface pressure and with a surface geopotential equal to zero. The meridional wind in the reference state is zero.

In the reference state, the virtual temperature is given by

$$T_v(\varphi, z) = \frac{1}{\tau_1(z) - \tau_2(z)I_T(\varphi)}, \quad (5)$$

where $I_T(\varphi)$ is defined as

$$I_T(\varphi) = (\cos \varphi)^K - \frac{K}{K+2}(\cos \varphi)^{K+2}, \quad (6)$$

and $\tau_1(z)$ and $\tau_2(z)$ are defined as follows:

$$\tau_1(z) = \frac{1}{T_0} \exp\left(\frac{\Gamma z}{T_0}\right) + \left(\frac{T_0 - T_P}{T_0 T_P}\right) \left[1 - 2\left(\frac{zg}{bR_d T_0}\right)^2\right] \exp\left[-\left(\frac{zg}{bR_d T_0}\right)^2\right] \quad (7)$$

$$\tau_2(z) = \frac{(K+2)}{2} \left(\frac{T_E - T_P}{T_E T_P}\right) \left[1 - 2\left(\frac{zg}{bR_d T_0}\right)^2\right] \exp\left[-\left(\frac{zg}{bR_d T_0}\right)^2\right], \quad (8)$$

with $T_0 = \frac{1}{2}(T_E + T_P)$. To maintain hydrostatic balance, the pressure is given by:

$$p(\varphi, z) = p_0 \exp\left[-\frac{g}{R_d}(\tau_{\text{int},1}(z) - \tau_{\text{int},2}(z)I_T(\varphi))\right] \quad (9)$$

with $\tau_{\text{int},1}(z)$ and $\tau_{\text{int},2}(z)$ given by

$$\tau_{\text{int},1}(z) = \frac{1}{\Gamma} \left[\exp\left(\frac{\Gamma z}{T_0}\right) - 1\right] + z \left(\frac{T_0 - T_P}{T_0 T_P}\right) \exp\left[-\left(\frac{zg}{bR_d T_0}\right)^2\right] \quad (10)$$

$$\tau_{\text{int},2}(z) = \frac{(K+2)}{2} \left(\frac{T_E - T_P}{T_E T_P}\right) z \exp\left[-\left(\frac{zg}{bR_d T_0}\right)^2\right]. \quad (11)$$

If density is a prognostic variable, it can be obtained from p , T_v and the ideal gas law (2). Finally, the zonal velocity is

$$u_{\text{ref}}(\varphi, z) = -\Omega_{\text{ref}} a_{\text{ref}} \cos(\varphi) + \sqrt{(\Omega_{\text{ref}} a_{\text{ref}} \cos(\varphi))^2 + a_{\text{ref}} \cos(\varphi) U(z, \varphi)}, \quad (12)$$

where

$$U(z, \varphi) = \frac{gK}{a_{\text{ref}}} \tau_{\text{int},2}(z) [(\cos \varphi)^{K-1} - (\cos \varphi)^{K+1}] T_v(\varphi, z). \quad (13)$$

1.2. Perturbations

To trigger the development of the baroclinic wave, a perturbation is applied to the zonal velocity field that takes the form of a simple exponential bell with a vertical taper:

$$u'(\lambda, \varphi, z) = \begin{cases} u_p Z_p(z) \exp \left[- \left(\frac{R(\lambda, \varphi; \lambda_p, \varphi_p)}{R_p} \right)^2 \right], & \text{if } R(\lambda, \varphi; \lambda_p, \varphi_p) < R_p, \\ 0, & \text{otherwise,} \end{cases} \quad (14)$$

where

$$Z_p(z) = \begin{cases} 1 - 3 \left(\frac{z}{z_p} \right)^2 + 2 \left(\frac{z}{z_p} \right)^3, & \text{if } z \leq z_p, \\ 0, & \text{otherwise.} \end{cases} \quad (15)$$

Consequently, the perturbed velocity field takes the form

$$u(\lambda, \varphi, z) = u_{\text{ref}}(\varphi, z) + u'(\lambda, \varphi, z). \quad (16)$$

1.3. Moist initial conditions

We define the vertical η coordinate as

$$\eta(\lambda, \varphi, z) = p(\lambda, \varphi, z)/p_s. \quad (17)$$

Since the surface pressure of the moist air p_s is constant with $p_s = p_0 = 1000$ hPa the vertical coordinate η is represented by $\eta = p/p_0$. Specific humidity is specified in terms of η as

$$q(\lambda, \varphi, \eta) = \begin{cases} q_0 \exp \left[- \left(\frac{\varphi}{\varphi_w} \right)^4 \right] \exp \left[- \left(\frac{(\eta - 1)p_0}{p_w} \right)^2 \right], & \text{if } \eta > p_t/p_s, \\ q_t, & \text{otherwise.} \end{cases} \quad (18)$$

The functional form of q and its parameters were inspired by observations. This moisture fields leads to maximum relative humidities around 85% in the lower levels of the midlatitudes.

Note that the moist temperature is colder than the temperature one would obtain with $q = 0$. However, note that in the moist case the virtual temperature and moist pressure determine the strength of the pressure gradient term in the momentum equations. Since these are identical to the temperature and pressure in the dry case, the forcing by the pressure gradient term is the same in both the dry and moist variant of the baroclinic wave. The moist variant of the baroclinic wave without the temperature forcing from large-scale condensation should lead to almost identical results when compared to the dry version. Very small variations are expected since the moisture gets independently transported as a passive tracer in this case and some models utilize the moist variant of the physical constant c_p . If possible, the dry c_p should be used. Comparing the evolution of the dry baroclinic wave to its moist variant (without large-scale condensation) can serve as a first sensibility check.

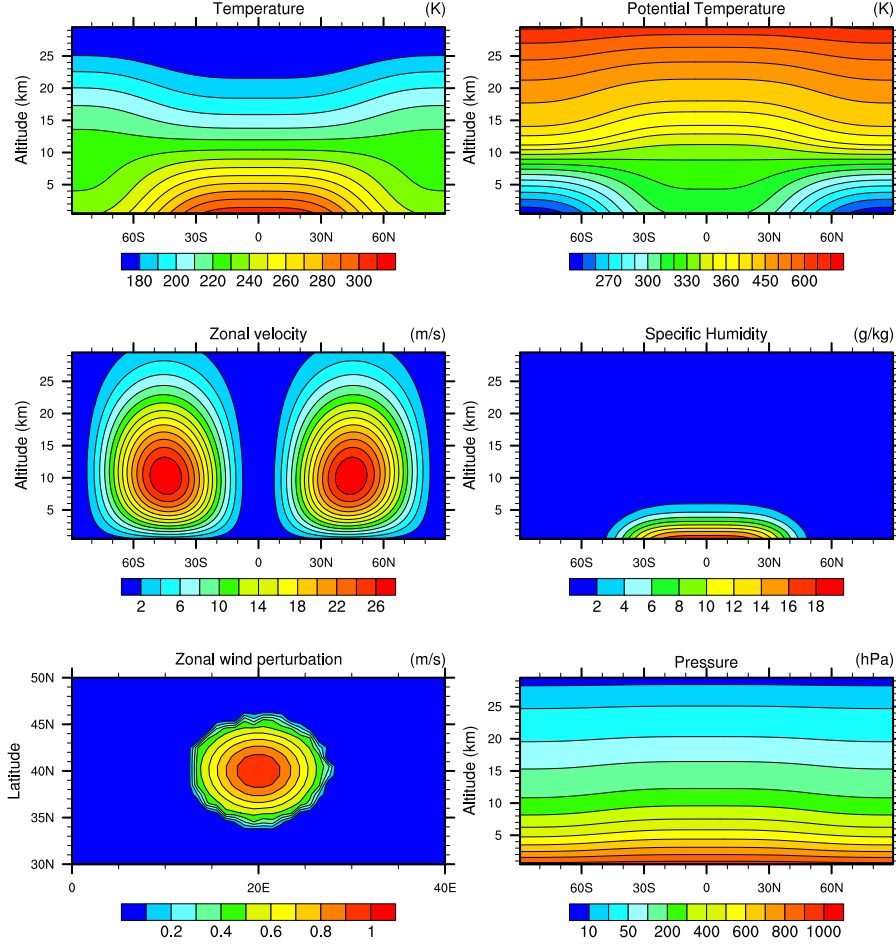


Figure 1. Initial state for the moist baroclinic wave test.

1.4. Terminator ‘toy’-chemistry

The terminator ‘toy’-chemistry is presented in [5] and mimics photolysis-driven processes near the solar terminator. Two passive* tracers, Cl and Cl₂, that are chemically reactive are transported. The sources and sinks are given by a simple, but non-linear, ‘toy’ chemistry. As a result, strong gradients in the spatial distribution of the species develop near the edge of the terminator. Despite the large spatial variations in Cl and Cl₂ the weighted sum $Cl_y = Cl + 2Cl_2$ should always be preserved in any flow field (if the initial condition for Cl_y is constant). An overview of the ‘toy’ terminator chemistry is given in Appendix B. The terminator test demonstrates how well the advection/transport scheme and/or physics-dynamics coupling preserves linear correlations.

*i.e. tracers do not feed back on the flow

HOMME

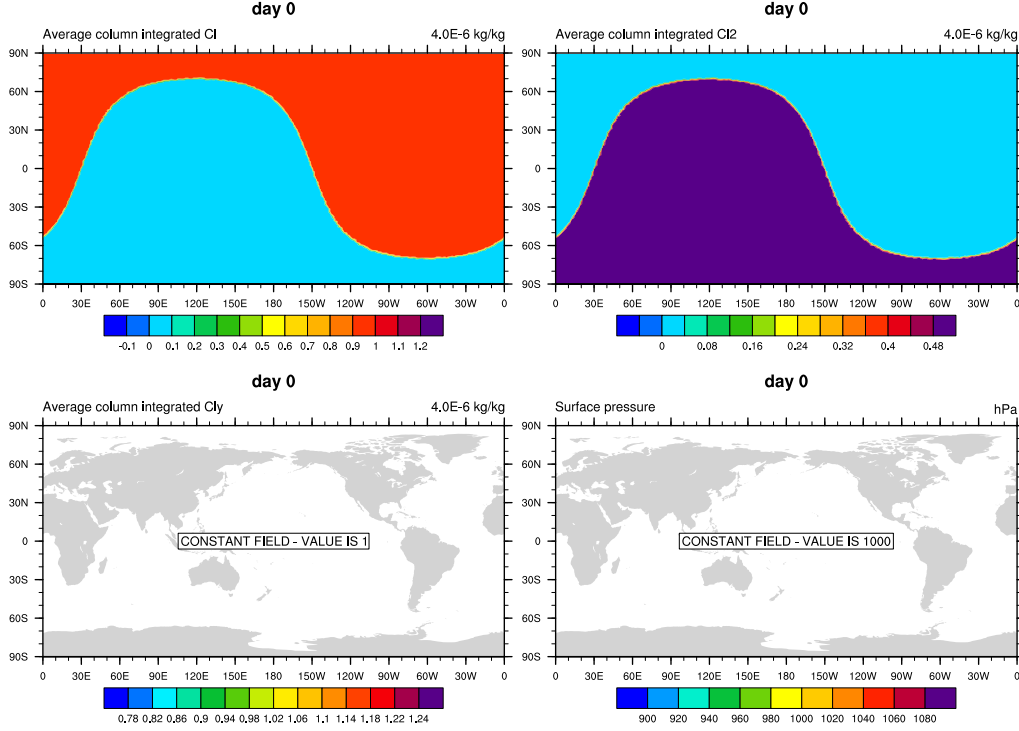


Figure 2. Contour lines of $\langle q_{Cl} \rangle / (4.0 \times 10^{-6})$, (upper left), $\langle q_{Cl_2} \rangle / (4.0 \times 10^{-6})$ (upper right), $\langle q_{Cl\gamma} \rangle$ (lower left) and surface pressure (lower right) at day 0 (initial conditions).

The initial conditions for Cl and Cl_2 (in terms of **dry** mixing ratios q_{Cl} and q_{Cl_2} , respectively) are the steady-state solutions to the terminator chemistry with no flow [5] (see Figure 2):

$$q_{Cl}(\lambda, \theta, z, t = 0) = D - r, \quad (19)$$

$$q_{Cl_2}(\lambda, \theta, z, t = 0) = \frac{1}{2} (q_{Cl\gamma} - D + r), \quad (20)$$

where $q_{Cl\gamma} = 4.0 \times 10^{-6} \text{ kg/kg}$,

$$r = \frac{k_1}{4k_2}, \quad (21)$$

$$D = \sqrt{r^2 + 2rq_{Cl\gamma}}, \quad (22)$$

and reaction coefficients k_1 and k_2 are given in appendix B (equations (100) and (101)). Please note that the mixing ratios are dry, i.e. the ratio between the density of the species and the density of dry air.

The forcing terms are computed analytically (assuming no flow) over a (physics/chemistry) time-step Δt :

$$F_{Cl}^n = -L_{\Delta t} \frac{(q_{Cl}^n - D + r)(q_{Cl}^n + D + r)}{1 + e^{-4k_2 D \Delta t} + \Delta t L_{\Delta t} (q_{Cl}^n + r)}. \quad (23)$$

where q_{Cl}^n is the value of q_{Cl} at the beginning of the n 'th time step and

$$L_{\Delta t} = \begin{cases} \frac{1-e^{-4k_2 D \Delta t}}{D \Delta t} & \text{if } D > 0 \\ 4k_2 & \text{if } D = 0. \end{cases} \quad (24)$$

and by conservation,

$$F_{Cl_2}^n = -\frac{1}{2}F_{Cl}^n. \quad (25)$$

In implementation, $L_{\Delta t}$ needs some care. As $4k_2 D \Delta t$ approaches machine precision, it is useful to simply use the formula for $D = 0$ rather than the expression for $D > 0$. The chemistry/physics updated mixing ratios are given by

$$q_{Cl}^n + \Delta t F_{Cl}^n, \quad (26)$$

$$q_{Cl_2}^n + \Delta t F_{Cl_2}^n. \quad (27)$$

In terms of Fortran code the analytical forcing is given by:

```
! dt is size of physics time step
cly = cl + 2.0*cl2

r = k1 / (4.0*k2)
d = sqrt( r*r + 2.0*r*cly )
e = exp( -4.0*k2*d*dt )

if( abs(d*k2*dt) .gt. 1e-16 )
  el = (1.0-e) / (d*dt)
else
  el = 4.0*k2
endif

f_cl  = -el * (cl-d+r) * (cl+d+r) / (1.0 + e + dt*el*(cl+r))
f_cl2 = -f_cl / 2.0
```

The reaction rates are defined by

```
! k1 and k2 are reaction rates
k1_lat_center = 20.0 ! degrees
k1_lon_center = 300.0 ! degrees
k1 = max(0.d0, sin(lat)*sin(k1_lat_center)
          + cos(lat)*cos(k1_lat_center)*cos(lon-k1_lon_center))
k2 = 1.0
```

The initial condition is defined by

```
cly = 4.0e-6
```

```

r = k1 / (4.0*k2)
d = sqrt( r*r + 2.0*cly*r )

cl = d-r
cl2 = cly / 2.0 - (d-r) / 2.0

```

Fortran subroutines that, given (λ, θ) will return the tendencies for q_{Cl} (mixing ratio) and q_{Cl2} (mixing ratio), are provided as supplemental material in [5]. Similarly for the forcing terms.

The ‘toy’ terminator chemistry test here uses the baroclinic wave initialization (described in section 1) based on the moist setup and the flow will transport the two species that interact non-linearly with each other through the toy chemistry. A physics time-step of 30 minutes and 15 minutes, respectively, is used. Note that if the user has the baroclinic wave setup the only additional work is to initialize two tracers and implement the chemistry. Example solutions are shown for HOMME (High-Order Method Modeling Environment) [6] spectral elements and HOMME-CSLAM [7]. The latter model is based on CSLAM [8] transport of tracers consistently coupled with spectral element dynamics.

1.4.1. Diagnostics

If the initial conditions for q_{Cl} and $2q_{Cl2}$ add up to a constant (as is the case in this setup) then no matter how the individual species evolve the weighted sum $q_{Cly} = q_{Cl} + 2q_{Cl2}$ should be constant in space and time. Hence the analytical solution for q_{Cly} is known. The terminator chemistry preserves the linear relationship between q_{Cl} and q_{Cl2} so the only causes for this relationship to break are:

- the transport operator (usually the limiter/filter) does not exactly preserve linear relations, and/or,
- physics-dynamics coupling breaks the relationship (see, e.g., Figure 3).

The following diagnostics are used in this test case:

- Average column integrated mixing ratio (two-dimensional variable):

$$\langle q \rangle = \frac{\int_{z=0}^{z_{top}} q \, dz}{\int_{z=0}^{z_{top}} dz}. \quad (28)$$

where $q = q_{Cl}, q_{Cl2}$. The global integrals should be computed consistently with the numerical method (preferably ‘inline’ in the source code on the native grid and not using interpolated data).

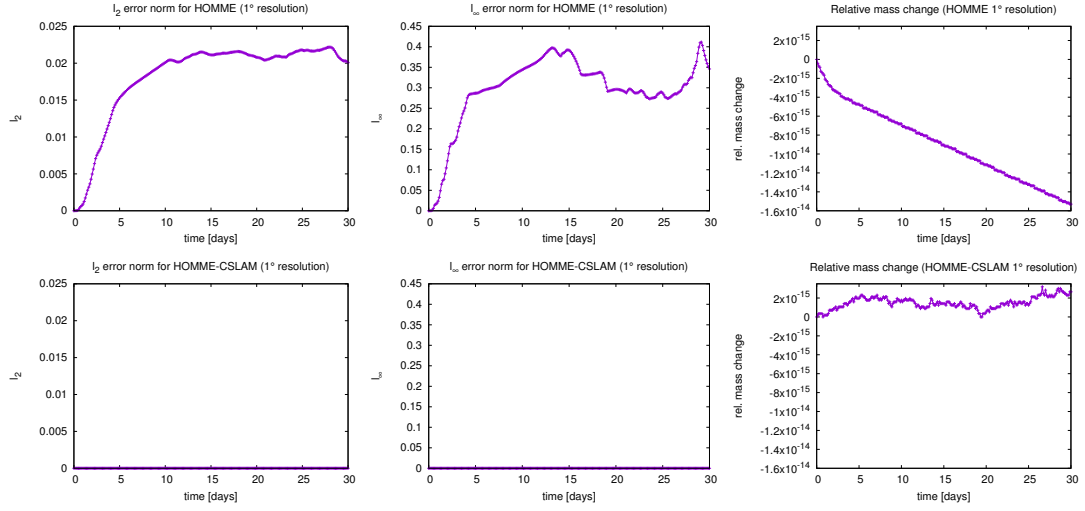
- $\ell_2(t)$, $\ell_\infty(t)$ and relative mass change $\Delta M(t)$ error norms for Cl_y :

$$\ell_2(t) = \frac{\sqrt{\int_{z=0}^{z_{top}} (\langle q_{Cly} \rangle - 4.0 \times 10^{-6})^2 \, dz}}{\sqrt{\int_{z=0}^{z_{top}} (4.0 \times 10^{-6})^2 \, dz}}, \quad (29)$$

$$\ell_\infty(t) = \frac{\max_{\text{all } \lambda, \theta} |\langle q_{Cly} \rangle - 4.0 \times 10^{-6}|}{4.0 \times 10^{-6}}, \quad (30)$$

$$\Delta M(t) = \frac{\int_{z=0}^{z_{top}} q_{Cly} \, dz - M_0}{M_0} \quad (31)$$

Diagnostics for terminator test: chemistry time-step = 900s



Diagnostics for terminator test: chemistry time-step = 1800s

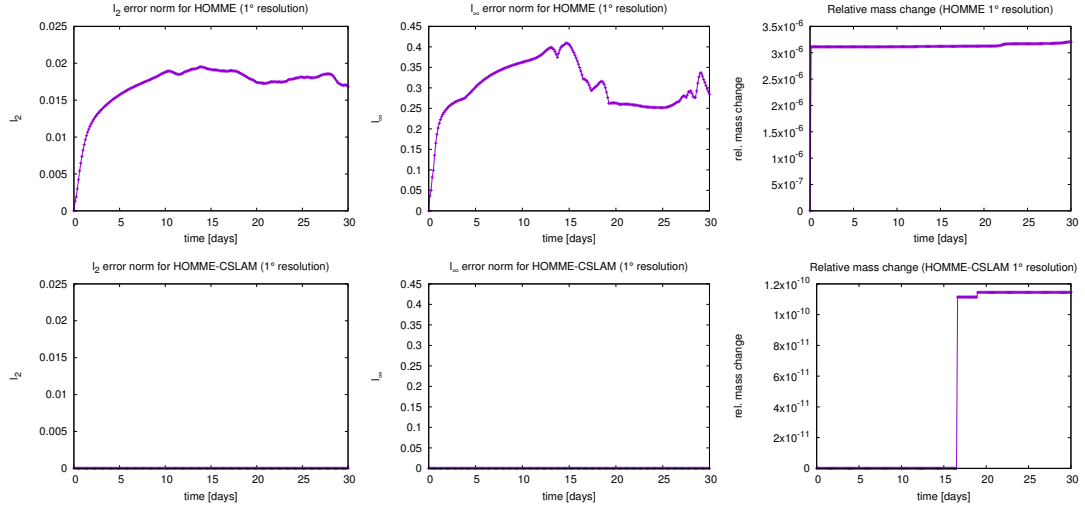


Figure 3. Global error norms $\ell_2(t)$ (column 1), $\ell_\infty(t)$ (column 2) and relative mass change $\Delta M(t)$ (column 3) for q_{Cl_y} for HOMME (row 1 and 3) and HOMME-CSLAM (row 2 and 4) with physics/chemistry time-step of 15 minutes (row 1 and 2) and 30 minutes (row 3 and 4), respectively. The non-conservation of Cl_y mass ($\Delta M(t)$) for physics time-step of 30 minutes is due to physics-dynamics coupling in which the tendencies are altered if they result in negative mixing ratios. For HOMME-CSLAM this happens at one point around day 16 and 18, respectively, whereas it happens frequently (in time and space) for HOMME.

respectively, where

$$\langle q_{Cl_y} \rangle = \langle q_{Cl} + 2q_{Cl_2} \rangle. \quad (32)$$

and M_0 is the initial mass of Cl_y

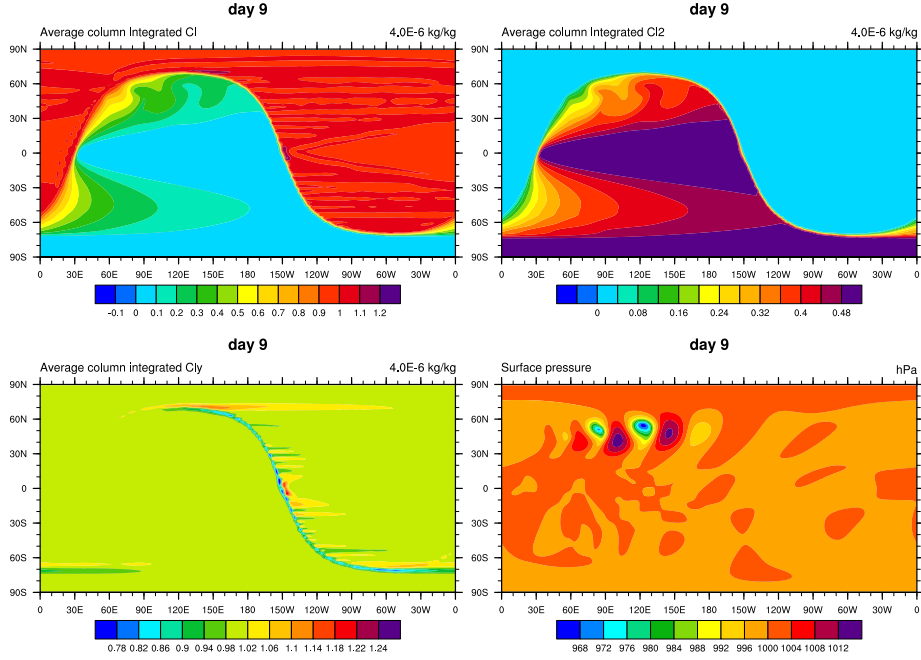
$$M_0 = \int_{z=0}^{z_{top}} 4.0 \times 10^{-6} dz. \quad (33)$$

Note that if the physics-dynamics coupling procedure breaks Cl_y conservation it can be very subtle but shows in the $\Delta M(t)$ diagnostic (see, e.g., Figure 3).

1.5. Grid spacings, simulation time, output and diagnostics

- Reference simulations (dry and moist) should be performed at 1° resolution with 30 vertical levels for 15 days. Plots should be produced for the moist simulation and the anomaly between moist and dry simulations at day 9, 12 and 15.
- Plots of minimum surface pressure over the duration of the simulation for both dry and moist configurations.
- Experiments could address the coupling frequency between the dynamics and physics.
- A variable resolution simulation should be performed that (a) studies the effect of the baroclinic wave transitioning from coarse resolution to fine resolution and (b) studies the effect of enhanced resolution near the front.
- Terminator chemistry (use physics time-step of 30 minutes and 15 minutes):
 - Please plot contour lines for $\langle q_{Cl} \rangle / (4.0 \times 10^{-6})$ at day 9. Contour interval must be 0.1 with zero contour. Please offset the zero contour $-1.0\text{E-}12$ to avoid contouring round-off undershoots. Contour levels used in Figure 4 are $(-0.1, -1.0\text{E-}12, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.2)$
 - Please plot contour lines for $\langle q_{Cl2} \rangle / (4.0 \times 10^{-6})$ at day 9. Contour interval must be 0.04 with round-off offset zero contour. Contour levels used in Figure 4 are $(-0.04, -1.0\text{E-}12, 0.04, 0.08, 0.12, 0.16, 0.20, 0.24, 0.28, 0.32, 0.36, 0.40, 0.44, 0.48)$
 - Please plot contour lines for $\langle q_{Cl_y} \rangle / (4.0 \times 10^{-6})$ at day 9. Contour interval must be 0.04 (or 0.02 or 0.01 depending on your data) excluding the 1.0 contour but symmetric about 1.0. Contour levels used in Figure 4 are $(0.78, 0.82, 0.86, 0.90, 0.94, 0.98, 1.02, 1.06, 1.1, 1.14, 1.18, 1.22, 1.24, 1.08, 1.12, 1.16, 1.20)$
 - Please plot global error norms $\ell_2(t)$, $\ell_\infty(t)$ and relative mass change $\Delta M(t)$ for q_{Cl_y} as a function of time from day 0 to 30 preferably with 3 hourly time spacing. Use vertical axis adjusted to your data. See example on Figure 3.

HOMME



HOMME-CSLAM

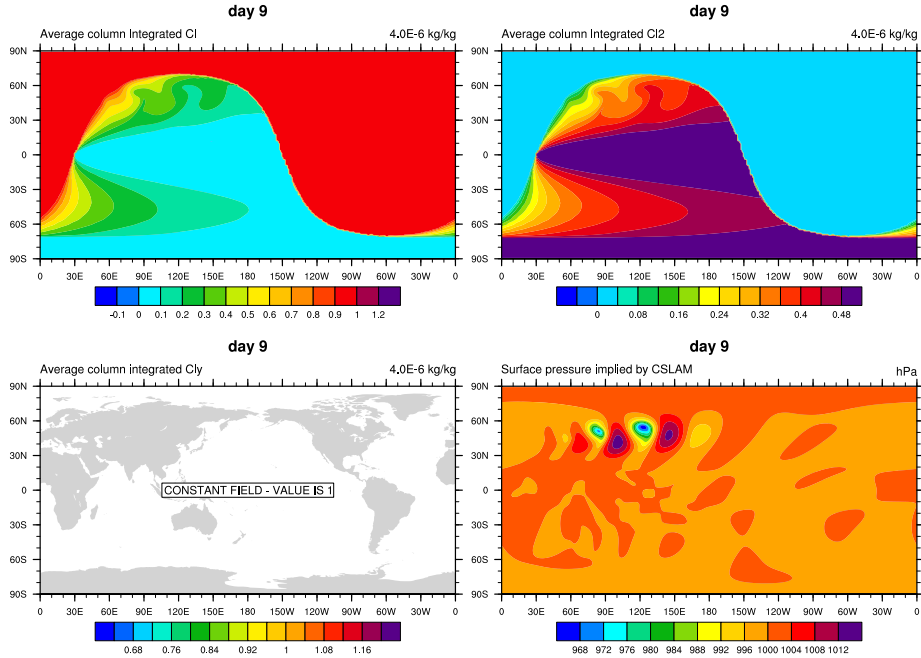


Figure 4. Upper panel of 4 plots is contour lines of $\langle q_{Cl} \rangle / (4.0 \times 10^{-6})$, (upper left), $\langle q_{Cl2} \rangle / (4.0 \times 10^{-6})$ (upper right), $\langle q_{Cly} \rangle$ (lower left) and surface pressure (lower right) at day 9 simulated with HOMME at approximately 1° resolution (30x30 elements on each cubed-sphere panel with 4x4 quadrature points in each element). Lower panel of 4 plots is the same as the upper panel but simulated with HOMME-CSLAM with 3x3 control volumes within each element. The results are based on the moist baroclinic wave setup (with no moist processes) and a physics-coupling time-step of 15 minutes.

2. TROPICAL CYCLONE

The simplified tropical cyclone test case on a regular-size Earth is based on the work of [9, 10, 11, 12]. In this test an analytic vortex is initialized in a background environment which is tractable to a rapid intensification of tropical cyclones.

Table VII. List of constants used for the Idealized Tropical Cyclone test

Constant	Value	Description
X	1	small-planet scaling factor (regular-size Earth)
z_t	15000 m	Tropopause height
q_0	0.021 kg/kg	Maximum specific humidity amplitude
q_t	10^{-11} kg/kg	Specific humidity in the upper atmosphere
T_0	302.15 K	Surface temperature of the air
T_s	302.15 K	Sea surface temperature (SST), 29 C°
z_{q1}	3000 m	Height related to the linear decrease of q with height
z_{q2}	8000 m	Height related to the quadratic decrease of q with height
Γ	0.007 K m^{-1}	Virtual temperature lapse rate
p_b	1015 hPa	Background surface pressure
φ_c	$\pi/18$	Initial latitude of vortex center (radians)
λ_c	π	Initial longitude of vortex center (radians)
Δp	11.15 hPa	Pressure perturbation at vortex center
r_p	282000 m	Horizontal half-width of pressure perturbation
z_p	7000 m	Height related to the vertical decay rate of p perturbation
ϵ	10^{-25}	Small threshold value

2.1. Initialization

The background state consists of a prescribed specific humidity profile, virtual temperature and pressure profile. The initial profile is defined to be in approximate gradient wind balance. The vertical sounding is chosen to roughly match an observed tropical sounding documented in [13]. The background specific humidity profile $\bar{q}(z)$ as a function of height z is

$$\bar{q}(z) = q_0 \exp\left(-\frac{z}{z_{q1}}\right) \exp\left[-\left(\frac{z}{z_{q2}}\right)^2\right] \quad \text{for } 0 \leq z \leq z_t \quad (34)$$

$$\bar{q}(z) = q_t \quad \text{for } z_t \leq z$$

The background virtual temperature sounding $\bar{T}_v(z)$ is split into two different representations for the lower and upper atmosphere. It is given by

$$\begin{aligned} \bar{T}_v(z) &= T_{v0} - \Gamma z & \text{for } 0 \leq z \leq z_t, \\ \bar{T}_v(z) &= T_{vt} = T_{v0} - \Gamma z_t & \text{for } z_t < z, \end{aligned} \quad (35)$$

with the virtual temperature at the surface $T_{v0} = T_0(1 + 0.608 q_0)$ and the virtual temperature at the tropopause level $T_{vt} = T_{v0} - \Gamma z_t$. The background temperature profile can be obtained from (3).

The background vertical pressure profile $\bar{p}(z)$ of the moist air is computed using the hydrostatic balance and (35). The profile is given by:

$$\begin{aligned}\bar{p}(z) &= p_b \left(\frac{T_{v0} - \Gamma z}{T_{v0}} \right)^{g/R_d\Gamma} & \text{for } 0 \leq z \leq z_t, \\ \bar{p}(z) &= p_t \exp \left(\frac{g(z_t - z)}{R_d T_{vt}} \right) & \text{for } z_t < z.\end{aligned}\quad (36)$$

The pressure at the tropopause level z_t is continuous and given by

$$p_t = p_b \left(\frac{T_{vt}}{T_{v0}} \right)^{\frac{g}{R_d\Gamma}}, \quad (37)$$

which, for the given set of parameters, is approximately 130.5 hPa.

2.1.1. Axisymmetric Vortex

The pressure equation $p(r, z)$ for the moist air is comprised of the background pressure profile (36) plus a 2D pressure perturbation $p'(r, z)$,

$$p(r, z) = \bar{p}(z) + p'(r, z), \quad (38)$$

where r symbolizes the radial distance (or radius) to the center of the prescribed vortex. On the sphere r is defined using the great circle distance

$$r = a \arccos(\sin \varphi_c \sin \varphi + \cos \varphi_c \cos \varphi \cos(\lambda - \lambda_c)). \quad (39)$$

The perturbation pressure is defined as

$$\begin{aligned}p'(r, z) &= -\Delta p \exp \left[-\left(\frac{r}{r_p} \right)^{3/2} - \left(\frac{z}{z_p} \right)^2 \right] \left(\frac{T_{v0} - \Gamma z}{T_{v0}} \right)^{\frac{g}{R_d\Gamma}} & \text{for } 0 \leq z \leq z_t, \\ p'(r, z) &= 0 & \text{for } z_t < z.\end{aligned}\quad (40)$$

The pressure perturbation depends on the pressure difference Δp between the background surface pressure p_b and the pressure at the center of the initial vortex, the pressure change in the radial direction r_p and the pressure decay with height within the vortex z_p . The moist surface pressure $p_s(r)$ is computed by setting $z = 0$ m in (38), which gives

$$p_s(r) = p_b - \Delta p \exp \left[-\left(\frac{r}{r_p} \right)^{3/2} \right]. \quad (41)$$

The axisymmetric virtual temperature $T_v(r, z)$ is computed using the hydrostatic equation and ideal gas law

$$T_v(r, z) = -\frac{gp(r, z)}{R_d} \left(\frac{\partial p(r, z)}{\partial z} \right)^{-1}. \quad (42)$$

Again it can be written as a sum of the background state and a perturbation,

$$T_v(r, z) = \bar{T}_v(z) + T'_v(r, z), \quad (43)$$

where the virtual temperature perturbation is defined as

$$T'_v(r, z) = (T_{v0} - \Gamma z) \left\{ \left[1 + \frac{2R_d(T_{v0} - \Gamma z)z}{gz_p^2 \left[1 - \frac{p_b}{\Delta p} \exp \left(\left(\frac{r}{r_p} \right)^{3/2} + \left(\frac{z}{z_p} \right)^2 \right) \right]} \right]^{-1} - 1 \right\} \quad \text{for } 0 \leq z \leq z_t,$$

$$T'_v(r, z) = 0 \quad \text{for } z_t < z. \quad (44)$$

The axisymmetric specific humidity $q(r, z)$ is set to the background profile everywhere

$$q(r, z) = \bar{q}(z). \quad (45)$$

Consequently, the temperature can be written as

$$T(r, z) = \bar{T}(z) + T'(r, z), \quad (46)$$

with the temperature perturbation

$$T'(r, z) = \frac{T_{v0} - \Gamma z}{1 + 0.608\bar{q}(z)} \left\{ \left[1 + \frac{2R_d(T_{v0} - \Gamma z)z}{gz_p^2 \left[1 - \frac{p_b}{\Delta p} \exp \left(\left(\frac{r}{r_p} \right)^{3/2} + \left(\frac{z}{z_p} \right)^2 \right) \right]} \right]^{-1} - 1 \right\} \quad \text{for } 0 \leq z \leq z_t,$$

$$T'(r, z) = 0 \quad \text{for } z_t < z. \quad (47)$$

Due to the small specific humidity value in the upper atmosphere (10^{-11} kg/kg for $z > z_t$) the virtual temperature equals the temperature to a very good approximation in this region. The formulation presented here is equivalent to the one presented in [9].

If the density of the moist air needs to be initialized its formulation is based on the ideal gas law

$$\rho(r, z) = \frac{p(r, z)}{R_d T_v(r, z)} \quad (48)$$

which utilizes the moist pressure (38) and virtual temperature (43). The surface elevation z_s and thereby the surface geopotential $\Phi_s = gz_s$ are set to zero.

Finally, the tangential velocity field $v_T(r, z)$ of the axisymmetric vortex is defined by utilizing the gradient-wind balance, which depends on the pressure (38) and the virtual temperature (44). The tangential velocity is given by

$$v_T(r, z) = -\frac{f_c r}{2} + \sqrt{\frac{f_c^2 r^2}{4} + \frac{R_d T_v(r, z) r}{p(r, z)} \frac{\partial p(r, z)}{\partial r}}, \quad (49)$$

where $f_c = 2\Omega \sin(\varphi_c)$ is the Coriolis parameter at the constant latitude φ_c . Substituting $T_v(r, z)$ and $p(r, z)$ into (49) gives

$$v_T(r, z) = -\frac{f_c r}{2} + \sqrt{\frac{f_c^2 r^2}{4} - \frac{\frac{3}{2} \left(\frac{r}{r_p}\right)^{3/2} (T_{v0} - \Gamma z) R_d}{1 + \frac{2R_d(T_{v0} - \Gamma z)z}{gz_p^2} - \frac{p_b}{\Delta p} \exp\left(\left(\frac{r}{r_p}\right)^{3/2} + \left(\frac{z}{z_p}\right)^2\right)}} \quad \text{for } 0 \leq z \leq z_t,$$

$$v_T(r, z) = 0 \quad \text{for } z_t < z. \quad (50)$$

The last step is to split the tangential velocity (50) into its zonal and meridional wind components $u(\lambda, \varphi, z)$ and $v(\lambda, \varphi, z)$. Similar to [14] these are computed using the following expressions,

$$d_1 = \sin \varphi_c \cos \varphi - \cos \varphi_c \sin \varphi \cos(\lambda - \lambda_c) \quad (51)$$

$$d_2 = \cos \varphi_c \sin(\lambda - \lambda_c) \quad (52)$$

$$d = \max(\epsilon, \sqrt{d_1^2 + d_2^2}), \quad (53)$$

which are utilized in the projections

$$u(\lambda, \varphi, z) = \frac{v_T(\lambda, \varphi, z) d_1}{d} \quad (54)$$

$$v(\lambda, \varphi, z) = \frac{v_T(\lambda, \varphi, z) d_2}{d}. \quad (55)$$

A small $\epsilon = 10^{-25}$ value avoids divisions by zero. The vertical velocity is set to zero.

Note: We are currently investigating a test case specification that places the idealized tropical cyclone on an f -plane. This configuration would remove issues associated with β drift of the cyclone and allow for a more direct intercomparison of the simulated storm.

2.2. Grid spacings, simulation time, output and diagnostics

- Moist simulations should be performed at 0.5° resolution with 30 vertical levels for 10 days.
- Plots of minimum surface pressure over the duration of the simulation.
- Experiments could address the coupling frequency between the dynamics and physics.
- A variable resolution simulation should be performed that (a) studies the effect of the tropical cyclone transitioning from fine resolution to coarse resolution and (b) high resolution simulations down to 0.125° over the tropical cyclone.

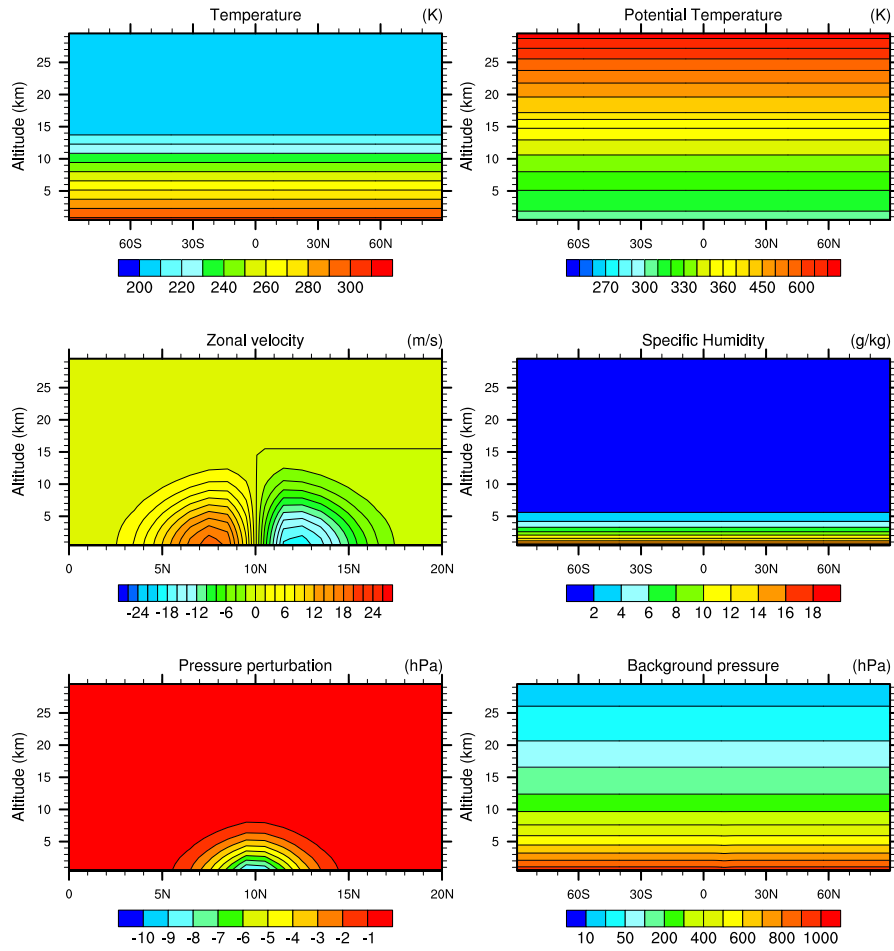


Figure 5. Initial state for the tropical cyclone test.

3. SUPERCELL

The supercell test permits the study of a non-hydrostatic moist feature with strong vertical velocities and associated precipitation and is based on the work of [15].

Table VIII. List of constants used for the Supercell test

Constant	Value	Description
X	120	Small-planet scaling factor (reduced Earth)
θ_{tr}	343 K	Temperature at the tropopause
θ_0	300 K	Temperature at the equatorial surface
z_{tr}	12000 m	Altitude of the tropopause
T_{tr}	213 K	Temperature at the tropopause
U_s	30 m/s	Maximum zonal wind velocity
U_c	15 m/s	Coordinate reference velocity
z_s	5000 m	Lower altitude of maximum velocity
Δz_u	1000 m	Transition distance of velocity
$\Delta\theta$	3 K	Thermal perturbation magnitude
λ_p	0	Thermal perturbation longitude
φ_p	0	Thermal perturbation latitude
r_p	$X \times 10000$ m	Perturbation horizontal half-width
z_c	1500 m	Perturbation center altitude
z_p	1500 m	Perturbation vertical half-width

It is assumed that the saturation mixing ratio is given by

$$q_{vs}(p, T) = \left(\frac{380.0}{p} \right) \exp \left(17.27 \times \frac{T - 273.0}{T - 36.0} \right) \quad (56)$$

The definition of this test case relies on hydrostatic and gradient wind balance, written in terms of Exner pressure π and virtual potential temperature θ_v as

$$\frac{\partial \pi}{\partial z} = -\frac{g}{c_p \theta_v}, \quad \text{and} \quad u^2 \tan \varphi = -c_p \theta_v \frac{\partial \pi}{\partial \varphi}. \quad (57)$$

These equations can be combined to eliminate π , leading to

$$\frac{\partial \theta_v}{\partial \varphi} = \frac{\sin(2\varphi)}{2g} \left(u^2 \frac{\partial \theta_v}{\partial z} - \theta_v \frac{\partial u^2}{\partial z} \right). \quad (58)$$

The wind velocity is analytically defined throughout the domain. Meridional and vertical wind is initially set to zero. The zonal wind is obtained from

$$\bar{u}(\varphi, z) = \begin{cases} \left(U_s \frac{z}{z_t} - U_c \right) \cos(\varphi) & \text{for } z < z_s - \Delta z_u, \\ \left(-\frac{4}{5} + 3 \frac{z}{z_s} - \frac{5}{4} \frac{z^2}{z_s^2} \right) U_s - U_c & \text{for } |z - z_s| \leq \Delta z_u \\ (U_s - U_c) \cos(\varphi) & \text{for } z > z_s + \Delta z_u \end{cases} \quad (59)$$

The equatorial profile is determined through numerical iteration. Potential temperature at the equator is specified via

$$\theta_{\text{eq}}(z) = \begin{cases} \theta_0 + (\theta_{tr} - \theta_0) \left(\frac{z}{z_{tr}} \right)^{\frac{5}{4}} & \text{for } 0 \leq z \leq z_{tr}, \\ \theta_{tr} \exp \left(-\frac{g(z - z_{tr})}{c_p T_{tr}} \right) & \text{for } z_{tr} \leq z \end{cases} \quad (60)$$

And relative humidity is given by

$$\bar{H}(z) = \begin{cases} 1 + \frac{3}{4} \left(\frac{z}{z_{tr}} \right)^{5/4} & \text{for } 0 \leq z \leq z_{tr}, \\ \frac{1}{4} & \text{for } z_{tr} \leq z. \end{cases} \quad (61)$$

Pressure and temperature at the equator are obtained by iterating on hydrostatic balance with initial state

$$\theta_{v,\text{eq}}^{(0)}(z) = \theta_{\text{eq}}(z), \quad (62)$$

and iteration procedure

$$\pi_{\text{eq}}^{(i)} = 1 - \int_0^z \frac{g}{c_p \theta_{v,\text{eq}}^{(i)}} dz \quad (63)$$

$$p_{\text{eq}}^{(i)} = p_0 (\pi_{\text{eq}}^{(i)})^{c_p/R_d} \quad (64)$$

$$T_{\text{eq}}^{(i)} = \theta_{\text{eq}}(z) \pi_{\text{eq}}^{(i)} \quad (65)$$

$$q_{\text{eq}}^{(i)} = H(z) q_{vs}(p_{\text{eq}}^{(i)}, T_{\text{eq}}^{(i)}) \quad (66)$$

$$\theta_{v,\text{eq}}^{(i+1)} = \theta_{\text{eq}}(z) (1 + M_v q_{\text{eq}}^{(i)}) \quad (67)$$

This iteration procedure appears to converge to machine epsilon after approximately 10 iterations. The equatorial moisture profile is then extended through the entire domain,

$$q(z, \varphi) = q_{\text{eq}}(z). \quad (68)$$

Once the equatorial profile has been constructed, the virtual potential temperature through the remainder of the domain can be computed by iterating on (58),

$$\theta_v^{(i+1)}(z, \varphi) = \theta_{v,\text{eq}}(z) + \int_0^\varphi \frac{\sin(2\phi)}{2g} \left(\bar{u}^2 \frac{\partial \theta_v^{(i)}}{\partial z} - \theta_v^{(i)} \frac{\partial \bar{u}^2}{\partial z} \right) d\varphi. \quad (69)$$

Again, approximately 10 iterations are needed for convergence to machine epsilon. Once virtual potential temperature has been computed throughout the domain, Exner pressure throughout the domain can be obtained from (57),

$$\pi(z, \varphi) = \pi_{\text{eq}}(z) - \int_0^\varphi \frac{u^2 \tan \varphi}{c_p \theta_v} d\varphi, \quad (70)$$

and so

$$p(z, \varphi) = p_0 \pi(z, \varphi)^{c_p/R_d}, \quad (71)$$

$$T_v(z, \varphi) = \theta_v(z, \varphi)(p/p_0)^{R_d/c_p}. \quad (72)$$

3.1. Potential temperature perturbation

To initiate convection, a thermal perturbation is introduced in the initial potential temperature field:

$$\theta'(\lambda, \phi, z) = \begin{cases} \Delta\theta \cos^2\left(\frac{\pi}{2} R_\theta(\lambda, \varphi, z)\right) & \text{for } R_\theta(\lambda, \varphi, z) < 1, \\ 0 & \text{for } R_\theta(\lambda, \varphi, z) \geq 1, \end{cases} \quad (73)$$

where

$$R_\theta(\lambda, \varphi, z) = \left[\left(\frac{R_c(\lambda, \varphi; \lambda_p, \varphi_p)}{r_p} \right)^2 + \left(\frac{z - z_c}{z_p} \right)^2 \right]^{1/2}. \quad (74)$$

Note: An additional iterative step will be required here to bring the potential temperature perturbation into hydrostatic balance. Without this additional iteration, large vertical velocities will be generated as the model rapidly adjusts to hydrostatic balance.

3.2. Uniform Diffusion

The supercell test case requires uniform diffusion to be continuously applied to all prognostic variables. This diffusion takes the form

$$\frac{\partial \psi}{\partial t} = \dots + \nu_s \nabla^2 \psi, \quad (75)$$

or

$$\frac{\partial}{\partial t}(\rho\psi) = \dots + \nu_s \nabla \cdot (\rho \nabla \psi), \quad (76)$$

for scalar quantities ψ and q , and

$$\frac{\partial \mathbf{u}}{\partial t} = \dots + \nu_v \nabla^2 \mathbf{u}, \quad (77)$$

for the vector velocity \mathbf{u} . The diffusion coefficients used in this test are $\nu_s = 1500 \text{ m}^2 \text{ s}^{-1}$ and $\nu_v = 500 \text{ m}^2 \text{ s}^{-1}$.

3.3. Physical Parameterizations

For initial verification, all physical parameterizations should be disabled (except for uniform diffusion, as described in section 3.2).

3.4. Grid spacings, simulation time, output and diagnostics

- Moist simulations should be performed at 0.5° , 1° , 2° and 4° resolution with 40 uniformly spaced vertical levels for 120 minutes (7200 seconds).

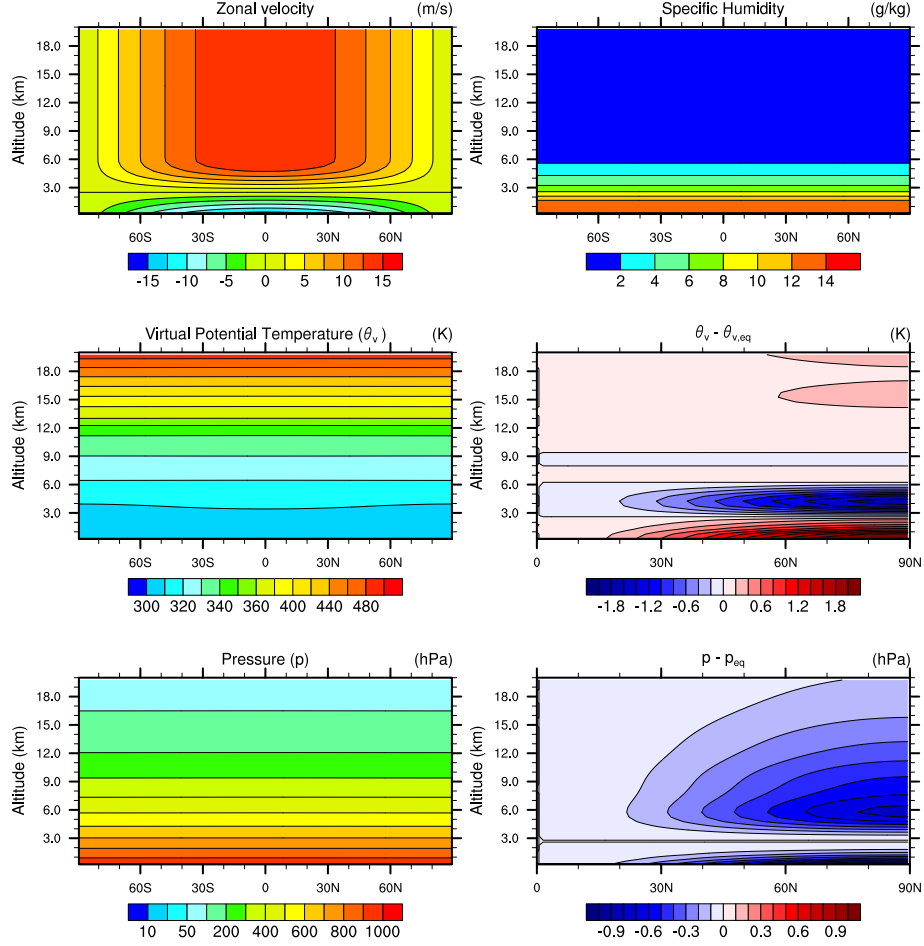


Figure 6. Initial state for the supercell test.

- Maximum vertical velocity and instantaneous precipitation rate should be reported at 60 second increments.
- Plots of vertical velocity and rainwater should be produced at 5 km altitude after 30, 60, 90 and 120 minutes over the domain $[30W, 30E] \times [40S, 40N]$ (script provided).
- Experiments could address the coupling frequency between the dynamics and physics.
- A variable resolution simulation should be performed that (a) studies the effect of the supercell transitioning from fine resolution to coarse resolution and (b) high resolution simulations down to 0.125° over the supercell.

REFERENCES

1. NetCDF. NetCDF data format. Online 2016. URL: <http://www.unidata.ucar.edu/software/netcdf/>.

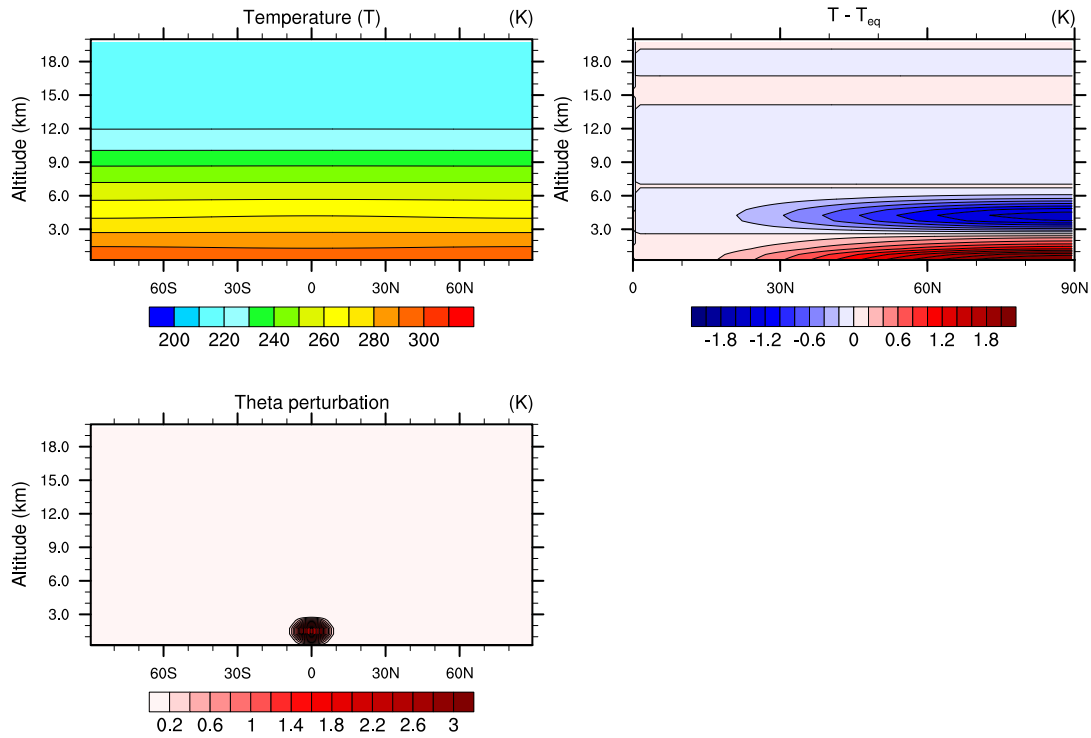


Figure 7. Initial state for the supercell test (cont'd).

2. NetCDF-CF. CF Conventions and Metadata. Online 2016. URL: <http://cfconventions.org/>.
3. Ullrich PA, Melvin T, Jablonowski C, Staniforth A. A proposed baroclinic wave test case for deep and shallow-atmosphere dynamical cores. *Quart. J. Royal Meteor. Soc.* 2014; **140**(682):1590–1602.
4. Jablonowski C, Williamson DL. A baroclinic instability test case for atmospheric model dynamical cores. *Quart. J. Roy. Meteor. Soc.* 2006; **132**(621C):2943–2975.
5. Lauritzen PH, Conley AJ, Lamarque JF, Vitt F, Taylor MA. The terminator "toy" chemistry test: a simple tool to assess errors in transport schemes. *Geoscientific Model Development* 2015; **8**(5):1299–1313, doi:10.5194/gmd-8-1299-2015.
6. Dennis JM, Edwards J, Evans KJ, Guba O, Lauritzen PH, Mirin AA, St-Cyr A, Taylor MA, Worley PH. CAM-SE: A scalable spectral element dynamical core for the Community Atmosphere Model. *Int. J. High. Perform. C.* 2012; **26**(1):74–89, doi:10.1177/1094342011428142. URL <http://hpc.sagepub.com/content/26/1/74.abstract>.
7. Lauritzen PH, Taylor MA, Overfelt J, amd S Goldhaber PAU. CAM-SE-CSLAM: Semi-Lagrangian finite-volume transport with spectral elements dynamics. *Mon. Wea. Rev.* 2016; In prep.
8. Lauritzen PH, Nair RD, Ullrich PA. A conservative semi-Lagrangian multi-tracer transport scheme (CSLAM) on the cubed-sphere grid. *J. Comput. Phys.* 2010; **229**:1401–1424, doi:10.1016/j.jcp.2009.10.036.

9. Reed KA, Jablonowski C. Idealized tropical cyclone simulations of intermediate complexity: a test case for AGCMs. *J. Adv. Model. Earth Syst.* 2012; **4**:M04 001, doi:10.1029/2011MS000099.
10. Reed KA, Jablonowski C. An analytic vortex initialization technique for idealized tropical cyclone studies in AGCMs. *Mon. Wea. Rev.* 2011; **139**:689–710.
11. Reed KA, Jablonowski C. Impact of physical parameterizations on idealized tropical cyclones in the Community Atmosphere Model. *Geophys. Res. Lett.* 2011; **38**:L04 805, doi:10.1029/2010GL046297.
12. Reed KA, Jablonowski C. Assessing the Uncertainty of Tropical Cyclone Simulations in NCAR’s Community Atmosphere Model. *J. Adv. Model. Earth Syst.* 2011; **3**:Art. 2011MS000 076, 16 pp.
13. Jordan CL. Mean soundings for the west indies area. *Journal of Meteorology* 1958; **15**(1):91–97.
14. Nair RD, Jablonowski C. Moving vortices on the sphere: A test case for horizontal advection problems. *Monthly Weather Review* 2008; **136**(2):699–711.
15. Klemp JB, Wilhelmson RB. The simulation of three-dimensional convective storm dynamics. *Journal of the Atmospheric Sciences* 1978; **35**(6):1070–1096.
16. Ogura Y, Takahashi T. Numerical simulation of the life cycle of a thunderstorm cell. *Monthly Weather Review* 1971; **99**(12):895–911.
17. Soong ST, Ogura Y. A comparison between axisymmetric and slab-symmetric cumulus cloud models. *Journal of the Atmospheric Sciences* 1973; **30**(5):879–893.
18. Kessler E. On the distribution and continuity of water substance in atmospheric circulation 1969; .
19. NCO. NetCDF Operator (NCO). Online 2016. URL: <https://github.com/nco/nco>.

A. KESSLER PHYSICS

The cloud microphysics update according to the following equation set:

$$\frac{\Delta\theta}{\Delta t} = -\frac{L}{c_p\pi} \left(\frac{\Delta q_{vs}}{\Delta t} + E_r \right) \quad (78)$$

$$\frac{\Delta q_v}{\Delta t} = \frac{\Delta q_{vs}}{\Delta t} + E_r \quad (79)$$

$$\frac{\Delta q_c}{\Delta t} = -\frac{\Delta q_{vs}}{\Delta t} - A_r - C_r \quad (80)$$

$$\frac{\Delta q_r}{\Delta t} = -E_r + A_r + C_r - V_r \frac{\partial q_r}{\partial z}, \quad (81)$$

where L is the latent heat of condensation, A_r is the autoconversion rate of cloud water to rain water, C_r is the collection rate of rain water, E_r is the rain water evaporation rate, and V_r is the rain water terminal velocity.

The pressure follows from the equation of state

$$p = \rho R_d T (1 + 0.61 q_v) \quad (82)$$

with p the pressure, ρ the density of moist air, R_d the gas constant for dry air, T the temperature and q_v the mixing ratio of water vapor. The equation is rewritten as a nondimensional pressure Π equation.

$$\pi = \left(\frac{p}{p_0} \right)^{\frac{R_d T}{c_p}} \quad (83)$$

To determine the saturation vapor mixing ratio the Tetten’s formula is used,

$$q_{vs}(p, T) = \left(\frac{380.0}{p} \right) \exp \left(17.27 \times \frac{T - 273.0}{T - 36.0} \right) \quad (84)$$

The autoconvection rate (A_r) and collection rate (C_r) follow Kessler parametrization and are defined by:

$$A_r = k_1(q_c - a) \quad (85)$$

$$C_r = k_2 q_c q_r^{0.875} \quad (86)$$

With $k_1 = 0.001\text{s}^{-1}$, $a = 0.001\text{g.g}^{-1}$ and $k_2 = 2.2\text{s}^{-1}$

Deriving from [15] description of cloud water, rain water and water vapor mixing ratios. they are define as followed:

$$q_c^{n+1} = \max(q_c^r - \Delta q_r, 0) \quad (87)$$

$$q_r^{n+1} = \max(q_r^r - \Delta q_r + S, 0) \quad (88)$$

where S is the sedimentation term and Δq_r is defined as

$$\Delta q_r = q_c^n - \frac{q_c^n - \Delta t \max(A_r, 0)}{1 + \Delta t C_r} \quad (89)$$

The Rain evaporation equation is defined similarly to [16] description:

$$E_r = \frac{1}{\rho} \frac{\left(1 - \frac{q_v}{q_{vs}}\right) C(\rho q_r)^{0.525}}{5.4 \times 10^5 + \frac{2.55 \times 10^6}{\rho q_{vs}}} \quad (90)$$

With ventilation factor C define as

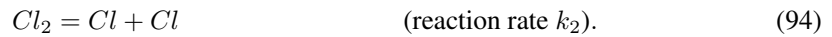
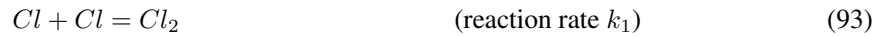
$$C_r = 1.6 + 124.9(\rho q_r)^{0.2046} \quad (91)$$

The liquid water terminal velocity is similar to [17] description with a mean density adjustment as suggested by [18]:

$$V_r = 36349(\rho q_r)^{0.1346} \left(\frac{\rho}{\rho_0}\right)^{-\frac{1}{2}} \quad (92)$$

B. ‘TOY’ CHEMISTRY

The toy chemistry module represents a simple photolysis-driven chemical reaction that incorporates combination and the dissociation of a chemical species:



Observe that the total number of molecules of the chemical species are conserved in this reaction,

$$Cl_T = Cl + 2Cl_2. \quad (95)$$

Representing the mixing ratios of these species as q_{Cl} and q_{Cl_2} , we can define the total mixing ratio of Chlorine atoms as

$$q_{Cl_y} = q_{Cl} + 2q_{Cl_2}. \quad (96)$$

The differential equations describing the evolution of Cl and Cl_2 under this reaction take the form

$$\frac{Dq_{Cl}}{Dt} = 2k_1q_{Cl_2} - 2k_2q_{Cl}^2, \quad (97)$$

$$\frac{Dq_{Cl_2}}{Dt} = -k_1q_{Cl_2} + k_2q_{Cl}^2, \quad (98)$$

where D/Dt denotes the Lagrangian derivative. Observe that the total mixing ratio of Chlorine atoms then satisfies

$$\frac{Dq_{Cl_y}}{Dt} = \frac{Dq_{Cl}}{Dt} + 2\frac{Dq_{Cl_2}}{Dt} = 0, \quad (99)$$

and so the total mixing ratio of Chlorine is held constant.

The two reaction rate coefficient k_1 and k_2 , representing the the photolytic dissociation and recombination of Chlorine gas are defined as

$$k_1(\lambda, \theta) = \max[0, \sin \theta \sin \theta_c + \cos \theta \cos \theta_c, \cos(\lambda - \lambda_c)] \quad (100)$$

$$k_2(\lambda, \theta) = 1, \quad (101)$$

where $(\lambda_c, \theta_c) = (20^\circ N, 300^\circ E)$ denote the sub-solar point on the Earth's surface.

C. SURFACE FLUXES ON AN AQUA-PLANET WITH PRESCRIBED SEA SURFACE TEMPERATURES

The forcing by surface fluxes from an idealized ocean is described in [9] and is partly reproduced here. We use a model configuration which corresponds to an aqua-planet setup with prescribed sea surface temperatures (SSTs). This forcing by the surface fluxes is applied to the state variables in the lowermost model level using a partially implicit formulation to avoid numerical instabilities. Throughout this section we use the subscript a to denote variables defined on the lowermost model level.

The surface fluxes depend on the *drag coefficient* C_d , defined as

$$\begin{aligned} C_d &= C_{d0} + C_{d1}|\vec{v}_a| & \text{for } |\vec{v}_a| < 20 \text{ m s}^{-1} \\ C_d &= 0.002 & \text{for } |\vec{v}_a| \geq 20 \text{ m s}^{-1}, \end{aligned} \quad (102)$$

where C_{d0} and C_{d1} are 7.0×10^{-4} (unitless) and $6.5 \times 10^{-5} \text{ s m}^{-1}$, respectively, and $|\vec{v}_a|$ is the magnitude of the horizontal wind at the lowermost model level. In terms of the zonal wind u_a and meridional wind v_a , it is defined as

$$|\vec{v}_a| = \sqrt{u_a^2 + v_a^2}. \quad (103)$$

For both evaporation and sensible heat the bulk coefficient is set to

$$C_E = C_H = 0.0011. \quad (104)$$

The formulation of the surface fluxes makes use of the height of the lowermost full model level z_a (in m). For pressure-based models, z_a can be expressed with the help of the hydrostatic equation in terms of pressure

$$z_a = \frac{R_d T_{\nu,a}}{g} \frac{(\ln p_s - \ln p_-)}{2}, \quad (105)$$

where $T_{\nu,a} = T_a(1 + 0.608q_a)$ is the virtual temperature at the lowermost full model level and p_- is the edge pressure at the model level interface between the lowest and second lowest full model levels. This notation and all following equations assume that the temperature, horizontal wind components and the specific humidity in the physical parameterization package are co-located in both the vertical and horizontal directions, as is the case for the Lorenz grid. The height of the lowest full model level should ideally lie between 60-70m above the ground to make the results comparable to those in the literature.

As described in [9], the surface fluxes can be written as

$$\frac{\partial \vec{v}_a}{\partial t} = - \frac{C_d |\vec{v}_a| \vec{v}_a}{z_a} \quad (106)$$

$$\frac{\partial T_a}{\partial t} = \frac{C_H |\vec{v}_a| (T_s - T_a)}{z_a} \quad (107)$$

$$\frac{\partial q_a}{\partial t} = \frac{C_E |\vec{v}_a| (q_{sat,s} - q_a)}{z_a}. \quad (108)$$

We note that the wind at the surface is taken to be zero and therefore does not appear explicitly in (106). In these equations T_s denotes the prescribed sea surface temperature (SST) and $q_{sat,s}$ is the saturation specific humidity defined by the Clausius-Clapeyron equation

$$q_{sat}(p) \approx \varepsilon \frac{e_s(T_s)}{p} \approx \frac{\varepsilon}{p} e_0^* e^{-(L/R_\nu)[(1/T_s) - (1/T_0)]}, \quad (109)$$

where e_0^* ($= 610.78$ Pa) is the saturation vapor pressure at $T_0 = 273.16$ K.

The final form of the surface fluxes will vary for models with other choices of prognostic variables. For example, if potential temperature Θ_a is used (107) takes the form

$$\frac{\partial \Theta_a}{\partial t} = \frac{C_H |\vec{v}_a| (T_s - T_a)}{z_a} \left(\frac{p_0}{p_a} \right)^{R_d/c_p} \quad (110)$$

where $p_0 = 1000$ hPa is a reference pressure. This conversion uses the assumption that the pressure is time-invariant when individual physics parameterizations are applied. For other choices of prognostic variables like $(\rho u)_a$, $(\rho v)_a$, $(\rho \Theta)_a$ and $(\rho q)_a$ the right-hand-side of (106), (110) and (108) would need to be multiplied by the density of the air ρ .

In order to ensure numerical stability, each of the aforementioned surface fluxes are applied via a semi-implicit operator. We demonstrate this procedure on the temperature evolution equation (107). First, the time derivative is expanded using a backward Euler operator,

$$\frac{T_a^{n+1} - T_a^n}{\Delta t} = \frac{C_H |\vec{v}_a^n| (T_s - T_a^{n+1})}{z_a}. \quad (111)$$

The superscripts n and $n + 1$ represent the current time step (after the update from the large-scale condensation scheme) and the future time step, respectively. Note, that on the right-hand-side of the equation the only variable taken implicitly is T_a . $|\vec{v}_a^n|$ is evaluated at the current time step and C_H is constant. The equation can now be solved for T_a^{n+1}

$$T_a^{n+1} = \frac{T_a^n + C_H |\vec{v}_a^n| T_s \frac{\Delta t}{z_a}}{1 + C_H |\vec{v}_a^n| \frac{\Delta t}{z_a}}. \quad (112)$$

Similar equations for \vec{v}_a and q_a can be calculated

$$\vec{v}_a^{n+1} = \frac{\vec{v}_a^n}{1 + C_d^n |\vec{v}_a^n| \frac{\Delta t}{z_a}} \quad (113)$$

$$q_a^{n+1} = \frac{q_a^n + C_E |\vec{v}_a^n| q_{sat,s}^n \frac{\Delta t}{z_a}}{1 + C_E |\vec{v}_a^n| \frac{\Delta t}{z_a}}, \quad (114)$$

with the time-level dependent coefficient C_d^n . Notice that the second term in the numerator of (113) is absent in the case of the zonal and meridional wind. This is because the wind is set to zero at the surface.

D. SIMPLIFIED MIXING IN THE PLANETARY BOUNDARY LAYER

The forcing by the planetary boundary layer is described in [9] and is partly reproduced here. To parameterize the surface fluxes that impact the zonal velocity u , the meridional velocity v and moisture q we start with the time rate of change equations

$$\frac{\partial u}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho}{\partial z} \overline{w'u'} \quad (115)$$

$$\frac{\partial v}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho}{\partial z} \overline{w'v'} \quad (116)$$

$$\frac{\partial q}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho}{\partial z} \overline{w'q'}. \quad (117)$$

Potential temperature, as opposed to temperature, is used in the boundary layer parameterization because the vertical profile of the potential temperature is a suitable indicator of static stability. This adds the time rate of change equation

$$\frac{\partial \Theta}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho}{\partial z} \overline{w'\Theta'}. \quad (118)$$

Here u' , v' , w' , Θ' and q' symbolize the deviations of the zonal velocity, meridional velocity, vertical velocity, potential temperature and specific humidity from their averages, respectively. The average is indicated by an overbar. Note, assuming pressure is held constant (which is a common assumption in physical parameterizations), the potential temperature time tendency can be converted back to a

temperature tendency of the following form

$$\frac{\partial T}{\partial t} = -\frac{1}{\rho} \left(\frac{p}{p_0} \right)^\kappa \frac{\partial \rho}{\partial z} \overline{w' \Theta'}. \quad (119)$$

with the reference pressure $p_0 = 1000$ hPa.

The turbulent mixing is characterized by a constant vertical eddy diffusivity to represent Ekman-like profiles of boundary layers

$$\overline{w' u'} = -K_m \frac{\partial u}{\partial z} \quad (120)$$

$$\overline{w' v'} = -K_m \frac{\partial v}{\partial z} \quad (121)$$

$$\overline{w' \Theta'} = -K_E \frac{\partial \Theta}{\partial z} \quad (122)$$

$$\overline{w' q'} = -K_E \frac{\partial q}{\partial z}. \quad (123)$$

Here, K_m is the eddy diffusivity coefficient for momentum and K_E is the eddy diffusivity coefficient for energy and set equal to that for water vapor. In order to calculate the eddy diffusivity coefficients, the eddy diffusivity is matched to that for the surface flux calculated in Appendix C at the lowermost model level. To allow for a smooth transition above the boundary layer ($p_{top} = 850$ hPa) the diffusivity coefficients for momentum taper to zero as

$$\begin{aligned} K_m &= C_d |\vec{v}_a| z_a & \text{for } p > p_{top} \\ K_m &= C_d |\vec{v}_a| z_a \exp \left(- \left[\frac{p_{top} - p}{p_{strato}} \right]^2 \right) & \text{for } p \leq p_{top}. \end{aligned} \quad (124)$$

Here the constant p_{strato} determines the rate of decrease and is set to 100 hPa. K_E is defined by

$$\begin{aligned} K_E &= C_E |\vec{v}_a| z_a & \text{for } p > p_{top} \\ K_E &= C_E |\vec{v}_a| z_a \exp \left(- \left[\frac{p_{top} - p}{p_{strato}} \right]^2 \right) & \text{for } p \leq p_{top}. \end{aligned} \quad (125)$$

We suggest implementing the boundary layer scheme with an implicit temporal discretization to avoid numerical instabilities. The details of this discretization are somewhat complicated, and so we refer to implementation details in Appendix D of [9]. In addition, we supply the DCMIP modeling groups with the complete “simple-physics” package as used in the model CAM which can serve as a template routine.

E. ALTERNATIVE PLANETARY BOUNDARY LAYER FOR TROPICAL CYCLONE TEST

An alternative approach for boundary layer mixing has been proposed using a K-profile parameterization formulation for the tropical cyclone test described in Section 2. The implementation only impacts the manner in which the eddy diffusivity coefficients are calculated.

In particular the calculation of K_m in (124) is replaced with

$$\begin{aligned} K_m &= \kappa u^* z \left(1 - \frac{z}{h}\right)^2 & \text{for } z \leq h \\ K_m &= 0 & \text{for } z > h, \end{aligned} \quad (126)$$

where $\kappa = 0.4$, $u^* = \sqrt{C_d}|\vec{v}_a|$ and $h = 1$ km. K_E in (125) is then defined as

$$\begin{aligned} K_E &= \kappa e^* z \left(1 - \frac{z}{h}\right)^2 & \text{for } z \leq h \\ K_E &= 0 & \text{for } z > h, \end{aligned} \quad (127)$$

where $e^* = \sqrt{C_E}|\vec{v}_a|$. This implementation will be used in supplemental simulations of the tropical cyclone test.

F. REQUIRED NETCDF OUTPUT FORMAT

As mentioned in section 0.6 a fundamental requirement for the exchange of scientific data is the ability to precisely describe the physical quantities being represented. We require data in the ‘Network Common Data Form’ (netCDF) [1] that adhere to the netCDF Climate and Forecast (CF) metadata convention (if possible to version 1.6 from Dec. 2011 [2]). NetCDF files should have the file name extension “.nc”. In particular, netCDF metadata need to be present. If models cannot adhere to the CF standards, we will work with these modeling groups before the DCMIP workshop and evaluate the application of ‘NCO’ (netCDF operator [19]) tools to help make the output netCDF-CF compliant after the model execution.

F.1. Global attributes

We ask for netCDF “global attributes” that make the output files self-describing and searchable by cyberinfrastructure tools. We ask for the inclusion of the global attributes

- model
- test_case
- horizontal_resolution
- levels
- grid
- equation
- time_frequency
- description

The first six entries need to follow the file naming convention outlined in Tables IV and V with one exception for models with non-latitude-longitude computational grids. The “grid” attribute needs to indicate the computational base grid like “cubed” for a cubed-sphere model regardless of any interpolations to a regular latitude-longitude output grid. The models will then be searchable on the DCMIP webpage according to their computational meshes. The “time_frequency” attribute indicates the output frequency in seconds (s), hours (hr) or days (day), and needs be specified as e.g.

```

time_frequency = "1hr"
time_frequency = "6hr"
time_frequency = "day"
time_frequency = "100s"

```

for the 1-hourly, 6-hourly, daily, or 100-second output. Other global attributes might also be present as shown in the example in section F.3.

F.2. Coordinates, variable names, metadata

The standard netCDF variable names (here denoted as “acronyms”), the requested physical ‘units’ attribute, a suggested ‘long_name’ and the standardized netCDF attribute ‘standard_name’ are listed in Tables IX and X. The entry for the netCDF attribute ‘long_name’ can be freely selected. However, if ‘standard_name’ is present its value must come from the standard netCDF CF-compliant entries listed below. The ‘units’ and ‘long_name’ attributes need to be part of the metadata of the netCDF output file. The ‘standard_name’ might be added as an option. Remember that the case is significant in netCDF names, and that all variable listed in Tables IX and X are written with upper case letters, all others like the dimensions or coordinates are written with lower case letters.

Table IX. List of symbols and corresponding netCDF attributes

Symbol	Acronym	‘units’	Suggested ‘long_name’	NetCDF ‘standard_name’
λ	lon	degrees.east	longitude	longitude
φ	lat	degrees.north	latitude	latitude
p_s	PS	Pa	Surface pressure	surface_pressure
Φ_s	PHIS	m2/m2	Surface geopotential	surface_geopotential
u	U	m/s	Zonal wind	eastward_wind
v	V	m/s	Meridional wind	northward_wind
w	W	m/s	Vertical velocity	upward_air_velocity
ω	OMEGA	Pa/s	Vertical pressure velocity	lagrangian_tendency_of_air_pressure
p	P	Pa	Pressure	air_pressure
T	T	K	Temperature	air_temperature
q	Q	kg/kg	Specific humidity	specific_humidity
qc	Qc	kg/kg	Cloud water mixing ratio	
qr	Qr	kg/kg	Rain water mixing ratio	
P_{ls}	PRECL	m/s	Large-scale precipitation rate	rainfall_rate
q_{Cl}	Q1	kg/kg	Singlet chlorine mixing ratio	
q_{Cl2}	Q2	kg/kg	Chlorine gas mixing ratio	

F.3. Example: Selected entries of a netCDF file with latitude-longitude grid

An example of selected entries of an NCAR CAM-FV output file ‘cam-fv.42.medium.L30’ is shown below. The simulation was run at the medium resolution on a regular 181×360 latitude-longitude grid with grid spacing $1^\circ \times 1^\circ$ (including the poles) and 30 hybrid η -levels. Note that this output data set also lists the approximate (reference) pressure positions of the 30 full levels (lev) and 31 model interface levels (ilev) as well as the hybrid coefficients for the full (hyam, hybm) and half levels (hyai, hybi). The latter can be used in combination with the surface pressure to reconstruct

Table X. Optional model variables: List of symbols and corresponding netCDF attributes

Symbol	Acronym	'units'	Suggested 'long_name'
SST	SST	K	Sea surface temperature
u_{850}	U850	m/s	Zonal wind at 850 hPa
v_{850}	V850	m/s	Meridional wind at 850 hPa
w_{500}	W500	m/s	Vertical velocity at 500 hPa
w_{850}	W850	m/s	Vertical velocity at 850 hPa
ω_{500}	OMEGA500	Pa/s	Vertical pressure velocity at 500 hPa
ω_{850}	OMEGA850	Pa/s	Vertical pressure velocity at 850 hPa
T_{500}	T500	K	Temperature at 500 hPa
T_{850}	T850	K	Temperature at 850 hPa

the actual pressure at each grid point. The surface geopotential PHIS is provided as a 3D data set despite its time-independency. The time-dependent data sets PS, U, V, T and OMEGA (actual data not listed) contain 61 instantaneous 6-hourly snapshots between day 0 and 15. In addition, the NetCDF header also lists variables on the 850 hPa pressure surface.

Desirable output quantities are the time step 'mdt' used for the simulation (here it represents the physics time step 1800 s), and the 'gw' field. The latter contains the latitudinal area-based ("Gaussian") weights that need to be used for area-averages on the latitude-longitude grid. The sum of these 'gw' weights is 2.

Example of a NetCDF file (header and selected entries and data sets):

```
netcdf cam-fv.161.r100.L30.latlon.hydro.4th_order_div_damping.nc {
dimensions:
    lat = 181 ;
    lon = 360 ;
    lev = 30 ;
    ilev = 31 ;
    time = UNLIMITED ; // (61 currently)
variables:
    double P0 ;
        P0:long_name = "reference pressure" ;
        P0:units = "Pa" ;
    double lat(lat) ;
        lat:long_name = "latitude" ;
        lat:units = "degrees_north" ;
    double lon(lon) ;
        lon:long_name = "longitude" ;
        lon:units = "degrees_east" ;
    double lev(lev) ;
        lev:long_name = "hybrid level at midpoints(1000*(A+B))" ;
        lev:units = "level" ;
        lev:positive = "down" ;
        lev:standard_name = "atmosphere_hybrid_sigma_pressure_coordinate" ;
```

```

    lev:formula_terms = "a: hyam b: hybm p0: P0 ps: PS" ;
double ilev(ilev) ;
    ilev:long_name = "hybrid level at interfaces (1000*(A+B))" ;
    ilev:units = "level" ;
    ilev:positive = "down" ;
    ilev:standard_name = "atmosphere_hybrid_sigma_pressure_coordinate" ;
    ilev:formula_terms = "a: hyai b: hybi p0: P0 ps: PS" ;
double time(time) ;
    time:long_name = "time" ;
    time:units = "days since 2000-01-01 00:00:00" ;
    time:calendar = "none" ;
double mdt ;
    mdt:long_name = "timestep" ;
    mdt:units = "s" ;
double hyai(ilev) ;
    hyai:long_name = "hybrid A coefficient at layer interfaces" ;
double hybi(ilev) ;
    hybi:long_name = "hybrid B coefficient at layer interfaces" ;
double hyam(lev) ;
    hyam:long_name = "hybrid A coefficient at layer midpoints" ;
double hybm(lev) ;
    hybm:long_name = "hybrid B coefficient at layer midpoints" ;
double gw(lat) ;
    gw:long_name = "gauss weights" ;
float PHIS(lat, lon) ;
    PHIS:units = "m2/s2" ;
    PHIS:long_name = "Surface geopotential" ;
float PS(time, lat, lon) ;
    PS:units = "Pa" ;
    PS:long_name = "Surface pressure" ;
float PRECL(time, lat, lon) ;
    PRECL:units = "m/s" ;
    PRECL:long_name = "Large-scale precipitation rate" ;
float Q(time, lev, lat, lon) ;
    Q:units = "kg/kg" ;
    Q:long_name = "Specific humidity" ;
float T(time, lev, lat, lon) ;
    T:units = "K" ;
    T:long_name = "Temperature" ;
float U(time, lev, lat, lon) ;
    U:units = "m/s" ;
    U:long_name = "Zonal wind" ;
float V(time, lev, lat, lon) ;

```

```

V:units = "m/s" ;
V:long_name = "Meridional wind" ;
float OMEGA(time, lev, lat, lon) ;
  OMEGA:units = Pa/s" ;
  OMEGA:long_name = "Vertical pressure velocity" ;
float OMEGA850(time, lat, lon) ;
  OMEGA850:units = Pa/s" ;
  OMEGA850:long_name = "Vertical pressure velocity at 850 hPa" ;
float T850(time, lat, lon) ;
  T850:units = "K" ;
  T850:long_name = "Temperature at 850 hPa" ;
float U850(time, lat, lon) ;
  U850:units = "m/s" ;
  U850:long_name = "Zonal wind at 850 hPa" ;
float V850(time, lat, lon) ;
  V850:units = "m/s" ;
  V850:long_name = "Meridional wind at 850 hPa" ;

// global attributes:
      :Conventions = "CF-1.0" ;
      :model = "cam-fv" ;
      :test_case = "161" ;
      :horizontal_resolution= "r100" ;
      :levels = "L30" ;
      :grid = "latlon" ;
      :equation = "hydro";
      :time_frequency = "6hr";
      :description = "4th-order divergence damping" ;

data:

P0 = 100000 ;

lat = -90.0, -89.0, -88.0, -87.0, -86.0, -85.0, -84.0, -83.0,
      -82.0, -81.0, -80.0, -79.0, -78.0, -77.0, -76.0, -75.0,
      -74.0, -73.0, -72.0, -71.0, -70.0, -69.0, -68.0, -67.0,
      -66.0, -65.0, -64.0, -63.0, -62.0, -61.0, -60.0, -59.0,
      -58.0, -57.0, -56.0, -55.0, -54.0, -53.0, -52.0, -51.0,
      -50.0, -49.0, -48.0, -47.0, -46.0, -45.0, -44.0, -43.0,
      -42.0, -41.0, -40.0, -39.0, -38.0, -37.0, -36.0, -35.0,
      -34.0, -33.0, -32.0, -31.0, -30.0, -29.0, -28.0, -27.0,
      -26.0, -25.0, -24.0, -23.0, -22.0, -21.0, -20.0, -19.0,
      -18.0, -17.0, -16.0, -15.0, -14.0, -13.0, -12.0, -11.0,

```

```

-10.0, -9.0, -8.0, -7.0, -6.0, -5.0, -4.0, -3.0,
-2.0, -1.0, 0.0, 1.0, 2.0, 3.0, 4.0, 5.0,
6.0, 7.0, 8.0, 9.0, 10.0, 11.0, 12.0, 13.0,
14.0, 15.0, 16.0, 17.0, 18.0, 19.0, 20.0, 21.0,
22.0, 23.0, 24.0, 25.0, 26.0, 27.0, 28.0, 29.0,
30.0, 31.0, 32.0, 33.0, 34.0, 35.0, 36.0, 37.0,
38.0, 39.0, 40.0, 41.0, 42.0, 43.0, 44.0, 45.0,
46.0, 47.0, 48.0, 49.0, 50.0, 51.0, 52.0, 53.0,
54.0, 55.0, 56.0, 57.0, 58.0, 59.0, 60.0, 61.0,
62.0, 63.0, 64.0, 65.0, 66.0, 67.0, 68.0, 69.0,
70.0, 71.0, 72.0, 73.0, 74.0, 75.0, 76.0, 77.0,
78.0, 79.0, 80.0, 81.0, 82.0, 83.0, 84.0, 85.0,
86.0, 87.0, 88.0, 89.0, 90.0;

```

```

lon = 0., 1., 2., 3., 4., 5., 6., 7., 8., 9.,
10., 11., 12., 13., 14., 15., 16., 17., 18., 19.,
20., 21., 22., 23., 24., 25., 26., 27., 28., 29.,
30., 31., 32., 33., 34., 35., 36., 37., 38., 39.,
40., 41., 42., 43., 44., 45., 46., 47., 48., 49.,
50., 51., 52., 53., 54., 55., 56., 57., 58., 59.,
60., 61., 62., 63., 64., 65., 66., 67., 68., 69.,
70., 71., 72., 73., 74., 75., 76., 77., 78., 79.,
80., 81., 82., 83., 84., 85., 86., 87., 88., 89.,
90., 91., 92., 93., 94., 95., 96., 97., 98., 99.,
100., 101., 102., 103., 104., 105., 106., 107., 108., 109.,
110., 111., 112., 113., 114., 115., 116., 117., 118., 119.,
120., 121., 122., 123., 124., 125., 126., 127., 128., 129.,
130., 131., 132., 133., 134., 135., 136., 137., 138., 139.,
140., 141., 142., 143., 144., 145., 146., 147., 148., 149.,
150., 151., 152., 153., 154., 155., 156., 157., 158., 159.,
160., 161., 162., 163., 164., 165., 166., 167., 168., 169.,
170., 171., 172., 173., 174., 175., 176., 177., 178., 179.,
180., 181., 182., 183., 184., 185., 186., 187., 188., 189.,
190., 191., 192., 193., 194., 195., 196., 197., 198., 199.,
200., 201., 202., 203., 204., 205., 206., 207., 208., 209.,
210., 211., 212., 213., 214., 215., 216., 217., 218., 219.,
220., 221., 222., 223., 224., 225., 226., 227., 228., 229.,
230., 231., 232., 233., 234., 235., 236., 237., 238., 239.,
240., 241., 242., 243., 244., 245., 246., 247., 248., 249.,
250., 251., 252., 253., 254., 255., 256., 257., 258., 259.,
260., 261., 262., 263., 264., 265., 266., 267., 268., 269.,
270., 271., 272., 273., 274., 275., 276., 277., 278., 279.,
280., 281., 282., 283., 284., 285., 286., 287., 288., 289.,

```


290., 291., 292., 293., 294., 295., 296., 297., 298., 299.,
300., 301., 302., 303., 304., 305., 306., 307., 308., 309.,
310., 311., 312., 313., 314., 315., 316., 317., 318., 319.,
320., 321., 322., 323., 324., 325., 326., 327., 328., 329.,
330., 331., 332., 333., 334., 335., 336., 337., 338., 339.,
340., 341., 342., 343., 344., 345., 346., 347., 348., 349.,
350., 351., 352., 353., 354., 355., 356., 357., 358., 359.;

gw = 0.3807694E-04, 0.3045981E-03, 0.6091034E-03, 0.9134232E-03,
0.1217465E-02, 0.1521135E-02, 0.1824343E-02, 0.2126994E-02,
0.2428998E-02, 0.2730262E-02, 0.3030694E-02, 0.3330203E-02,
0.3628698E-02, 0.3926087E-02, 0.4222280E-02, 0.4517187E-02,
0.4810718E-02, 0.5102784E-02, 0.5393296E-02, 0.5682164E-02,
0.5969302E-02, 0.6254621E-02, 0.6538035E-02, 0.6819458E-02,
0.7098804E-02, 0.7375987E-02, 0.7650923E-02, 0.7923528E-02,
0.8193721E-02, 0.8461417E-02, 0.8726535E-02, 0.8988996E-02,
0.9248719E-02, 0.9505624E-02, 0.9759633E-02, 0.1001067E-01,
0.1025866E-01, 0.1050352E-01, 0.1074518E-01, 0.1098357E-01,
0.1121862E-01, 0.1145024E-01, 0.1167838E-01, 0.1190297E-01,
0.1212392E-01, 0.1234118E-01, 0.1255469E-01, 0.1276437E-01,
0.1297016E-01, 0.1317200E-01, 0.1336983E-01, 0.1356358E-01,
0.1375321E-01, 0.1393864E-01, 0.1411983E-01, 0.1429672E-01,
0.1446925E-01, 0.1463738E-01, 0.1480104E-01, 0.1496020E-01,
0.1511480E-01, 0.1526480E-01, 0.1541015E-01, 0.1555080E-01,
0.1568672E-01, 0.1581785E-01, 0.1594417E-01, 0.1606564E-01,
0.1618221E-01, 0.1629385E-01, 0.1640052E-01, 0.1650220E-01,
0.1659886E-01, 0.1669045E-01, 0.1677697E-01, 0.1685837E-01,
0.1693464E-01, 0.1700575E-01, 0.1707168E-01, 0.1713241E-01,
0.1718792E-01, 0.1723819E-01, 0.1728322E-01, 0.1732298E-01,
0.1735746E-01, 0.1738666E-01, 0.1741056E-01, 0.1742915E-01,
0.1744244E-01, 0.1745041E-01, 0.1745307E-01, 0.1745041E-01,
0.1744244E-01, 0.1742915E-01, 0.1741056E-01, 0.1738666E-01,
0.1735746E-01, 0.1732298E-01, 0.1728322E-01, 0.1723819E-01,
0.1718792E-01, 0.1713241E-01, 0.1707168E-01, 0.1700575E-01,
0.1693464E-01, 0.1685837E-01, 0.1677697E-01, 0.1669045E-01,
0.1659886E-01, 0.1650220E-01, 0.1640052E-01, 0.1629385E-01,
0.1618221E-01, 0.1606564E-01, 0.1594417E-01, 0.1581785E-01,
0.1568672E-01, 0.1555080E-01, 0.1541015E-01, 0.1526480E-01,
0.1511480E-01, 0.1496020E-01, 0.1480104E-01, 0.1463738E-01,
0.1446925E-01, 0.1429672E-01, 0.1411983E-01, 0.1393864E-01,
0.1375321E-01, 0.1356358E-01, 0.1336983E-01, 0.1317200E-01,
0.1297016E-01, 0.1276437E-01, 0.1255469E-01, 0.1234118E-01,
0.1212392E-01, 0.1190297E-01, 0.1167838E-01, 0.1145024E-01,

0.1121862E-01, 0.1098357E-01, 0.1074518E-01, 0.1050352E-01,
0.1025866E-01, 0.1001067E-01, 0.9759633E-02, 0.9505624E-02,
0.9248719E-02, 0.8988996E-02, 0.8726535E-02, 0.8461417E-02,
0.8193721E-02, 0.7923528E-02, 0.7650923E-02, 0.7375987E-02,
0.7098804E-02, 0.6819458E-02, 0.6538035E-02, 0.6254621E-02,
0.5969302E-02, 0.5682164E-02, 0.5393296E-02, 0.5102784E-02,
0.4810718E-02, 0.4517187E-02, 0.4222280E-02, 0.3926087E-02,
0.3628698E-02, 0.3330203E-02, 0.3030694E-02, 0.2730262E-02,
0.2428998E-02, 0.2126994E-02, 0.1824343E-02, 0.1521135E-02,
0.1217465E-02, 0.9134232E-03, 0.6091034E-03, 0.3045981E-03,
0.3807694E-04;

lev = 3.64346569404006, 7.59481964632869, 14.3566322512925,
24.6122200042009, 38.2682997733355, 54.5954797416925, 72.0124505460262,
87.8212302923203, 103.317126631737, 121.547240763903, 142.994038760662,
168.225079774857, 197.908086702227, 232.828618958592, 273.910816758871,
322.241902351379, 379.100903868675, 445.992574095726, 524.687174707651,
609.778694808483, 691.389430314303, 763.404481112957, 820.858368650079,
859.53476652503, 887.020248919726, 912.644546944648, 936.198398470879,
957.485479535535, 976.325407391414, 992.556095123291 ;

ilev = 2.25523952394724, 5.03169186413288, 10.1579474285245,
18.5553170740604, 30.6691229343414, 45.8674766123295, 63.3234828710556,
80.7014182209969, 94.9410423636436, 111.69321089983, 131.401270627975,
154.586806893349, 181.863352656364, 213.952820748091, 251.704417169094,
296.117216348648, 348.366588354111, 409.83521938324, 482.149928808212,
567.22442060709, 652.332969009877, 730.445891618728, 796.363070607185,
845.353666692973, 873.715866357088, 900.324631482363, 924.964462406933,
947.432334534824, 967.538624536246, 985.112190246582, 1000 ;

hyai = 0.00225523952394724, 0.00503169186413288, 0.0101579474285245,
0.0185553170740604, 0.0306691229343414, 0.0458674766123295,
0.0633234828710556, 0.0807014182209969, 0.0949410423636436,
0.11169321089983, 0.131401270627975, 0.154586806893349,
0.181863352656364, 0.17459799349308, 0.166050657629967,
0.155995160341263, 0.14416541159153, 0.130248308181763,
0.113875567913055, 0.0946138575673103, 0.0753444507718086,
0.0576589405536652, 0.0427346378564835, 0.0316426791250706,
0.0252212174236774, 0.0191967375576496, 0.0136180268600583,
0.00853108894079924, 0.00397881818935275, 0, 0 ;

hybi = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.0393548272550106,
0.0856537595391273, 0.140122056007385, 0.204201176762581,

```

0.279586911201477, 0.368274360895157, 0.47261056303978,
0.576988518238068, 0.672786951065063, 0.753628432750702,
0.813710987567902, 0.848494648933411, 0.881127893924713,
0.911346435546875, 0.938901245594025, 0.963559806346893,
0.985112190246582, 1 ;

hyam = 0.00364346569404006, 0.00759481964632869, 0.0143566322512925,
0.0246122200042009, 0.0382682997733355, 0.0545954797416925,
0.0720124505460262, 0.0878212302923203, 0.103317126631737,
0.121547240763903, 0.142994038760662, 0.168225079774857,
0.178230673074722, 0.170324325561523, 0.161022908985615,
0.150080285966396, 0.137206859886646, 0.122061938047409,
0.104244712740183, 0.0849791541695595, 0.0665016956627369,
0.0501967892050743, 0.037188658490777, 0.028431948274374,
0.0222089774906635, 0.016407382208854, 0.0110745579004288,
0.00625495356507599, 0.00198940909467638, 0 ;

hybm = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.0196774136275053,
0.062504293397069, 0.112887907773256, 0.172161616384983,
0.241894043982029, 0.323930636048317, 0.420442461967468,
0.524799540638924, 0.624887734651566, 0.713207691907883,
0.783669710159302, 0.831102818250656, 0.864811271429062,
0.896237164735794, 0.92512384057045, 0.951230525970459,
0.974335998296738, 0.992556095123291 ;

time = 0, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2, 2.25,
2.5, 2.75, 3, 3.25, 3.5, 3.75, 4, 4.25, 4.5, 4.75,
5, 5.25, 5.5, 5.75, 6, 6.25, 6.5, 6.75, 7, 7.25,
7.5, 7.75, 8, 8.25, 8.5, 8.75, 9, 9.25, 9.5, 9.75,
10, 10.25, 10.5, 10.75, 11, 11.25, 11.5, 11.75, 12, 12.25,
12.5, 12.75, 13, 13.25, 13.5, 13.75, 14, 14.25, 14.5, 14.75,
15 ;

mdt = 1800 ;

}

```

Note that the notation of the variables uses the NetCDF-specific order of the dimensions (time,lev,lat,lon). We ask for output data (like U, V, T, PS, etc.) with single precision (float) accuracy using 4 bytes per datum. This uses the storage space more economically than double-precision (double) data.

F.4. Dimensions & metadata & time stamp

The NetCDF file must contain information about the grid and time dimensions. For a regular latitude-longitude grid (either as the native computational grid or as the result of interpolations) the dimensions need to utilize the keywords (see also the in section F.3)

```
dimensions:
  lat = 181 ;
  lon = 360 ;
  lev = 30 ;
  ilev = 31 ;
  time = UNLIMITED ; // (61 currently)
```

where “lat” and “lon” denote the number of latitudes and longitudes, “lev” and “ilev” symbolize the number of full model levels and interface levels, and ‘time’ denotes the number of time snapshots. These keywords for the dimensions are also used to define the size of the coordinate arrays like

```
double lat(lat) ;
  lat:long_name = "latitude" ;
  lat:units = "degrees_north" ;
double lon(lon) ;
  lon:long_name = "longitude" ;
  lon:units = "degrees_east" ;
```

and to define the dimensions of the model output variables like

```
float U(time, lev, lat, lon) ;
  U:units = "m/s" ;
  U:long_name = "Zonal wind" ;
```

As in the example above, the coordinates and model output variables need to be accompanied by metadata that describe, at a minimum, the “units” and “long_name” of the entry, except “units” are not necessary for non-dimensional quantities. The “standard_name” as listed in Table IX might also be included.

Idealized test cases are not associated with a particular date. However, a date must be chosen as a reference date, e.g.

```
double time(time) ;
  time:long_name = "time" ;
  time:units = "days since 2000-01-01 00:00:00" ;
  time:calendar = "none" ;
```

so that the data entries for the “time” variable can read

```
time = 0, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2, 2.25,
      2.5, 2.75, 3, 3.25, 3.5, 3.75, 4, 4.25, 4.5, 4.75,
      5, 5.25, 5.5, 5.75, 6, 6.25, 6.5, 6.75, 7, 7.25,
      7.5, 7.75, 8, 8.25, 8.5, 8.75, 9, 9.25, 9.5, 9.75,
```

```

10, 10.25, 10.5, 10.75, 11, 11.25, 11.5, 11.75, 12, 12.25,
12.5, 12.75, 13, 13.25, 13.5, 13.75, 14, 14.25, 14.5, 14.75,
15 ;

```

for 6-hourly snapshots. We recommend picking the reference date “2000-01-01 00:00:00”. This date is arbitrary and irrelevant for practical purposes, since the model simulations are idealized. However, a sensible date such as this must be picked to avoid potential problems in the DCMIP visualization tools. Daily time snapshots (e.g. 13 snapshots for 12 days including the initial state) need to be listed as

```

time = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12;

```

F.5. Computational grid

The netCDF file must list the longitudinal and latitudinal positions of the horizontal grid (in degrees) and must contain sufficient information about the vertical grid. In orography-following hybrid $\sigma - p$ (η) coordinates, as shown in the example in section F.3, the necessary information for the reconstruction of the pressure values along the full model and interface levels is the surface pressure PS, the surface geopotential PHIS and the list of the ‘a’ and ‘b’ hybrid coefficients, here called HYAM, HYBM, HYAI, HYBI. The key point is that the vertical positions of the grid, either in pressure coordinates (for most hydrostatic model) or height coordinates (for most non-hydrostatic models), need to be reconstructable based on the information in the netCDF output file. Details on the vertical grids and the equations for the reconstruction of the pressure or height values need to be provided, possibly in form of a model documentation or a journal paper. In addition, the netCDF file needs to define whether the order of the vertical levels is positive downwards (“down”), as in the example above, or positive upwards (“up”). The metadata entry ‘positive =’ as shown in

```

double lev(lev) ;
lev:long_name = "hybrid level at midpoints(1000*(A+B))" ;
lev:units = "level" ;
lev:positive = "down" ;
lev:standard_name = "atmosphere_hybrid_sigma_pressure_coordinate" ;
lev:formula_terms = "a: hyam b: hybm p0: P0 ps: PS" ;

```

defines this order, and needs to be present. If the vertical arrangement can be freely chosen, the ‘down’ numbering system is recommended (level counter increases downwards).

F.6. Example: NetCDF data structure for a non-latitude-longitude grid

An example netCDF data structure for a non-latitude-longitude (e.g. spherical geodesic) grid is given below.

```

dimensions:
  cell = 2562 ; // number of grid cells
  lev = 30 ;
  ilev = 31 ;
  time = UNLIMITED ; // (11 currently)

```

```

variables:
  double lon(cell) ;
    lon:long_name = "longitude" ;
    lon:units = "degrees_east" ;
    lon:bounds="lon_vertices" ;
  double lat(cell) ;
    lat:long_name = "latitude" ;
    lat:units = "degrees_north" ;
    lat:bounds="lat_vertices" ;
  double time(time) ;
    time:long_name = "time" ;
    time:units = "days since 2000-01-01 00:00:00" ;
  double cell_area(cell) ;
    cell_area:long_name = "area of grid cell" ;
    cell_area:standard_name="area";
    cell_area:units = "m2"
  float PS(time,cell) ;
    PS:units = "Pa" ;
    PS:coordinates = "lon lat" ;
    PS:cell_measures = "area: cell_area" ;
    PS:long_name = "Surface pressure" ;
  float T(time, lev, cell) ;
    T:units = "K" ;
    T:coordinates = "lon lat" ;
    T:cell_measures = "area: cell_area" ;
    T:long_name = "Temperature" ;

```

On non-latitude-longitude grids the area weight “cell_area” of cell should be included in the netCDF output file. This allows for the computation of global error norms based on the original grid point locations. Here, the data structure for the non-orthogonal grid utilizes a single index “cell” for all positions of the horizontal grid. The longitudinal and latitudinal positions are listed according this single array index. Other data structures are also possible.