from sympy import (TimeDifferencingWrapper, NetCDFMonitor)
import climt
from datetime import timedelta

# Define model timestep in minutes
model_timestep = timedelta(minutes=1)

# Create Components
radiation = climt.RRTMGLongwave()
convection = climt.EmanuelConvection()
boundary_layer = TimeDifferencingWrapper(climt.SimplePhysics())
time_stepper = GFSDynamicalCore([radiation, convection, boundary_layer])

# Create model grid
model_grid = climt.get_grid(nx=64, ny=64, nz=28)

# Create model state
model_state = climt.get_default_state([time_stepper], grid_state=model_grid)

# Create monitor
monitor = NetCDFMonitor('moist_agcm.nc')

# step model forward
for step in range(10):
    diagnostics, new_state = time_stepper(model_state, model_timestep)
    model_state.update(diagnostics)
    monitor.store(model_state)
    model_state.update(new_state)
    model_state['time'] += model_timestep