Features

The qtcm package (Lin 2009) is a Python wrapping of the Neelin-Zeng (2000) Quasi-equilibrium Tropical Circulation Model, a primitive equations-based intermediate-level atmospheric model written in Fortran. By using Python as a wrapper, we create an integrated modeling and visualization environment with the following features:

Interactive model runs as Python objects: Visualization and analysis is integrated in with model execution.

Execution control using run lists: Enables runtime control of subroutine execution order and content.

Doing science more easily: In this interactive modeling environment, the traditional sequence of “hypothesis → modeling → visualization and analysis” is made nonlinear and flexible, enabling more science questions to be easily addressed.

Interactive modeling with qtcm

Fig. 1 shows a screenshot of an interactive Python session running an instance of the qtcm tropical atmosphere model (model). During a model run, you have access to and can change all model variables. The visualization is done interactively at run time.

Execution control using run lists

qtcm uses “run lists,” lists of string names and dictionaries, to describe what subroutines are executed and in what order. For instance:

```python
>>> model = Qtcm(**inputs)
>>> print model.runlists['qtcminit']
[{'__qtcm': 'qtcm.wrapcall.qtcminit', '__qtcm': 'qtcm.wrapcall.wbndinit', '__qtcm': 'qtcm.wrapcall.wparinit', '__qtcm': 'qtcm.wrapcall.wtimemanager'}: [1], '__lsm': 'parts'}
```

Because lists can be manipulated at run time, subroutine execution order and content is completely changeable at run time.

Doing science more easily

Because lists can be manipulated at run time, subroutine execution order and content is completely changeable at run time.

References and acknowledgments


For more information

Email: jwblin@uw.edu
Personal home: http://www.johnny-lin.com
qtcm home: http://www.johnny-lin.com/py_pkgs/qtcm

Fig. 1. Screenshot of an interactive integration of a qtcm model instance.

The upper-left window shows the code for initializing the model instance and running 180 days of simulation.

The lower-right window shows the run session. The first two lines in the window called the qtcm method to generate the two plots. The third line shows variable substitution for a prognostic variable (doubling the existing value), and the fourth line will run the model for another 30 days when executed.

The 180 day model run took a little over a minute of wall-clock time on a 1.83 GHz Intel Core Duo with 1 GB DDR2 SDRAM running Mac OS X version 10.4.11. The horizontal grid for the model is 5.625 x 3.75 degrees longitude and latitude.