Useful Links and Support

CDAT: Getting Support

The CDAT Home Page:

http://cdat.sf.net

On-line tutorials:

- From the Tutorials section of the download facility at **http://cdat.sf.net**, download the data files tarball (**cdat_tutorial_data.tar.gz**) and unpack it.

CDAT Mailing list:

http://sourceforge.net/mailarchive/forum.php?forum_id=6820

CDAT Links (1)

CDAT documentation:

http://esg.llnl.gov/cdat/documentation.html

CDMS:

http://esg.llnl.gov/cdat/cdms.pdf

VCS:

http://esg.llnl.gov/cdat/vcs_html/vcs.html

VCS Template library:

http://esg.llnl.gov/cdat/templates/templates.html

CDAT Links (2)

cdtime docs:

http://esg.llnl.gov/cdat/cdms_html/cdms-3.htm

cdutil on-line docs:

http://esg.llnl.gov/cdat/cdat_utilities/cdat_utilities-1.htm#pgfId-998201

genutil on-line docs:

http://esg.llnl.gov/cdat/cdat_utilities/cdat_utilities-2.htm#pgfId-998200

Numeric:

http://www.pfdubois.com/numpy/html2/numpy-9.html#pgfId-36422

VCDAT and “doc strings”

Using VCDAT

The VCDAT graphical user interface has a “**help**” button that provides access to all the documentation that a user may require at the click of a button. The interface also has helpful information in pop-up balloons that instruct the user when the cursor is moved over the buttons.

Using “docstrings”

Python also has a powerful feature: most objects in Python, including the modules, classes, and functions, have documentation strings (“docstrings”) attached to them. The special attribute name “__doc__” (which has two underscores on each end) is used to access this string supplied by the module author. If you have an object x, try the command:

>>> print x.__doc__ OR print help(x)

Pydoc – for documentation (1)

Using pydoc to generate documentation

You can also use the **pydoc** utility. This is a standard facility for extracting documentation from Python installations.

You can generate Python documentation in HTML or text for interactive use.

- **Interactive use**

  In the Python interpreter, do "**from pydoc import help**" to provide online help. Calling help(thing) on a Python object documents the object. Also, typing help() will put the user in a help environment indicated by the prompt changing to:

  help>

Pydoc – for documentation (2)

- **From the shell**

  At the shell command line outside of Python: Run "pydoc <name>“ to show documentation on something. <name> may be the name of a function, module, package, or a dotted reference to a class or function within a module or module in a package. If the argument contains a path segment delimiter (e.g. slash on Unix, backslash onWindows) it is treated as the path to a Python source file. Run "pydoc –k <keyword>" to search for a keyword in the synopsis lines of all available modules.

- **Starting a browser / server**

  pydoc -g starts an HTTP server and also pops up a little window for controlling it.

Pydoc – for documentation (3)

- **Writing out HTML**

  Run "pydoc -w <name>" to write out the HTML documentation for a module to a file named "<name>.html".

- **Using happydoc.**

  A utility named “happydoc” is able to generate documentation for modules written in Python.

Reference guides

- **Quick Reference**

  The documents ***cdms_quick_start.pdf*** and ***vcs_quick_start.pdf*** provide a quick reference of useful commands in the core CDAT modules; **cdms** and **vcs** respectively on a single page that is useful even to the experienced user.

- **Programming reference guides**

  The core CDAT modules **cdms** and **vcs** are extensively documented in separate documents that can be downloaded from **http://cdat.sf.net**. The most up-to-date versions of the following guides can be found there.

  - **cdat.pdf**: This document
  - **cdms.pdf**: *Climate Data Management System*
  - **vcs.pdf**: *Visualization and Control System: Python Command Line and Application Programming Interface*
  - **cdat_utilities.pdf**: *CDAT Utilities Reference Guide*
  - **numpy.pdf**: *Numerical Python*

Johnny Lin CDAT help

IDL to Numeric/numarray mapping:

http://www.johnny-lin.com/cdat_tips/tips_array/idl2num.html

Endless other useful tips from Johnny Lin:

http://www.johnny-lin.com/cdat_tips/

Python Links

Python:

http://www.python.org/doc/

Beginners' Guide to Python:

http://www.python.org/moin/BeginnersGuide

Python built-in functions:

http://www.python.org/doc/lib/built-in-funcs.html

Numeric:

http://www.pfdubois.com/numpy/html2/numpy.html

Binding Python to other languages

Pyfort:

http://pyfortran.sourceforge.net

F2PY:

http://cens.ioc.ee/projects/f2py2e

Comparison of Pyfort and F2PY:

http://www.prism.enes.org/WPs/WP4a/Slides/pyfort/pyfort.html

SWIG:

http://www.swig.org

SWIG Python Presentation:

http://www.swig.org/papers/PyTutorial98/PyTutorial98.pdf

Python FAQT website

- FAQTs.com is a growing collection of high quality information. You can browse the knowledge bases or perform specialised searches. Anyone can contribute and everyone is encouraged to do so:

http://www.faqts.com/knowledge_base/index.phtml/fid/199/

NetCDF and CF Convention

CF Metadata Convention:
http://www.cgd.ucar.edu/cms/eaton/cf-metadata/

CF Mailing list:
http://www.cgd.ucar.edu/mailman/listinfo/cf-metadata

NetCDF Users' Mailing list:
http://my.unidata.ucar.edu/mailsubscribeform.php?list_name=netcdfgroup&list_label=netCDF+users

Best practice for writing NetCDF:
http://my.unidata.ucar.edu/content/software/netcdf/BestPractices.html

NetCDF CF-convention checker:
http://titania.badc.rl.ac.uk/cgi-bin/cf-checker.pl

NetCDF4 and HDF5 convergence:
http://my.unidata.ucar.edu/content/software/netcdf/netcdf-4/index.html).

NetCDF Manipulation Packages

xconv/convsh:

http://badc.nerc.ac.uk/help/software/xconv/

NCO:

http://nco.sourceforge.net/

NCBROWSE:

http://www.epic.noaa.gov/java/ncBrowse/

NCVIEW:

http://meteora.ucsd.edu/~pierce/ncview_home_page.html

Reading GRIB in CDAT

GrADS (including ***gribmap*** utility):

http://grads.iges.org/grads/

grib2ctl.pl utility:

http://www.cpc.ncep.noaa.gov/products/wesley/grib2ctl.html