SIVO-PyD or SSSO-PyD or spd
A Python Distribution for
Scientific Data Analysis

Jules Kouatchou
SSAI - Code 610.3
Jules.Kouatchou-1@nasa.gov
I learned it last night! Everything is so simple!
Hello world is just print "Hello, world!"

I dunno...
Dynamic typing?
Whitespace?

Come join us!
Programming is fun again!
It's a whole new world up here!
But how are you flying?

I just typed
import antigravity
That's it?
... I also sampled everything in the medicine cabinet for comparison.

But I think this is the Python.
Background (1)

- Python is used in industry and academia for a wide variety of purposes
- There are more than 15,000 Python related packages
- Few attempts to re-group some of them in a distribution for scientific computing
- Available products are mainly commercial
Background (2)

• GMAO adopted Python for model development and data analysis
• Other NCCS users rely on Python
• In 2009, we started the process of creating a Python distribution
Background (3)

- Parts of the distribution were installed by different staff
- The distribution was static (not possible to change the version of Python for instance)
- Users’ requests were not handled in a consistent way

New Approach: SIVO-PyD
What is SIVO-PyD?

• Collection of Python packages for scientific computing and visualization
• All the packages are accessible within the Python framework
• Self-contained distribution that mimics the commercial Enthought one.
Design Philosophy

• Only use open-source packages
• Follows baselibs concepts
• Automatically installed on *discover* and on Mac
• Does not alter source codes
• New packages or new versions of existing ones are added with minimal effort
• Maintainable in a version control repository
SIVO-PyD Main Packages

Python
Numpy  SciPy
ipython
pygrads
pyhdf  h5py  netCDF4
matplotlib  basemap

More than 33 packages are part of the distribution
How to Use the Distribution

The distribution is available to any NCCS user on *discover/dali*. You only need to load the modules:

- `other/comp/gcc-4.5`
- `lib/mkl-10.1.2.024`
- `other/SIVO-PyD/spd_1.4.0`
- `other/SIVO-PyD/spd_1.0.0`
- `other/SIVO-PyD/spd_1.1.2`
- `other/SIVO-PyD/spd_1.1.2_un`
New Addition

• We recently built the distribution using ATLAS (instead of Intel MKL)
• We added more functionalities in SciPy
• With ATLAS, the distribution becomes more portable

other/SIVO-PyD/spdAtlas
A Good Alternative

- Start with the free distribution (Python, Numpy, SciPy, Matplotlib) of Enthought
- Added more packages (using eggs file) as needed
- Installation is very quick
- Currently causing system problems
Under the Hood

The distribution consists on three main components:

1. The source files
2. A GNUmakefile file
3. A set of Python scripts

All of them are kept in a Git repository.
Source Files

• Directly obtained from the organizations that wrote them.
• Do not perform permanent alteration of any of them.
• Kept in their original form in the repository.
GNUmakefile File

- Sets up build options for individual packages
- Determines dependencies among packages
- Sets up necessary environment variables
- Provides the structure of the installation directory
Python Scripts

• Written to perform specific operations on front end source scripts
• Mainly edit files that need additional configuration options.
• Executed in the GNUmakefile file.
Git Repository

- Easy to manipulate
- Unfortunately became “too large”
- May need to find ways to avoid keeping some of the packages in the repository
- Does not know yet how to use Git to overwrite an existing package with a new version.
User Community

• Growing community of users: Codes 610.1, 610.3, 663.0, 587.0.
• Distribution is flexible enough to address specific needs.
• Might be asked to diverge from the initial requirements: what to do?
• How to meet individual user’s requirements and still maintain SIVO-PyD integrity?
Things to DO

• Install all the Matplotlib backend dependencies
• Install VTK, Mayavi
• Complete the installation procedures for Mac platform and the Cloud
• Avoid keeping source codes in the repository.
Extension

Use the same strategy to install packages that have many dependencies:

• NCL
• R
• etc.
Any Suggestion?

Is there anything you want us to add/remove in the distribution?
More Information

https://modelingguru.nasa.gov/clearspace/docs/DOC-2109