Python and Scientific Computing
A very quick overview

Fernando Pérez

<Fernando.Perez@berkeley.edu>
http://fperez.org

Helen Wills Neuroscience Institute, U.C. Berkeley

Py4Science, UC Berkeley
Nov 4, 2009
Interactive, high-level tools are *very* popular in science

- Matlab
- IDL
- Mathematica
- Maple

‘Stack’

- **Numpy**: fast arrays.
- **Scipy**: algorithms for many fields.
- **IPython**: full-time interactive use.
- **Matplotlib**: 2d plotting (basic 3d).
- **Mayavi**: 3d visualization.
- **Sympy**: symbolic manipulations.
- **Cython**: incremental optimization (‘pyrex+’)
- ... lots more I won’t have time to mention.
Numpy: numerical array core

- Fast, flexible and very sophisticated array object.
- Random number generation
- Basic linear algebra (no Fortran dependency)
- FFT
- f2py: support to wrap Fortran codes (Scipy needs this)

Basic stats:

Totals grouped by language (dominant language first):
ansic: 82493 (55.19%)
python: 65694 (43.95%)
cpp: 828 (0.55%)
fortran: 298 (0.20%)
f90: 97 (0.06%)
sh: 61 (0.04%)

Total Physical Source Lines of Code (SLOC) = 149,471
Development Effort Estimate, Person-Years = 38.40
Scipy: lots of libraries

- Fortran, C, C++, Cython, SWIG, Python...
- Linear algebra (full LAPACK), statistics, differential equations, optimization, numerical integration, clustering, special functions, signal processing, n-dimensional image processing, much more...
- **SciKits**: toolkits for audio, image processing, statistical modeling, time-series analysis, ...
- **Basic stats (for scipy library only):**

  Totals grouped by language (dominant language first):
  
  cpp: 217083 (49.70%)
  fortran: 81717 (18.71%)
  python: 77720 (17.79%)
  ansic: 60225 (13.79%)
  sh: 61 (0.01%)

  Total Physical Source Lines of Code = 436,806
  Development Effort Estimate, Person-Years = 118.39
IPython: Matlab/IDL-like interactive use

In [1]: from enthought.tvtk.tools import mlab
In [2]: from scipy import *
In [3]: def f(x, y):
   ...:     return \sin(x+y) + \sin(2*x-y) + \cos(3*x+4*y)
   ...:
In [4]: x = linspace(-5.0, 5.0, 200)
In [5]: y = linspace(-5.0, 5.0, 200)
In [6]: fig = mlab.figure()
In [7]: surf = mlab.SurfRegular(x, y, f)
In [8]: fig.add(surf)
In [9]:
Matplotlib: 3d plotting
MayaVi: 3d visualization (VTK)
MayaVi: customizable application (CFD)
Andrew Straw (Caltech)

Virtual reality based real-time control of fly flight

http://www.archive.org/details/scipy09_day1_05-Andrew_Straw
Perry Greenfield & Michael Droetboom
Hubble Space Telescope Science Institute
Enthought
Prabhu Ramachandran (IIT Bombay)
FOSSE India
Workshops and conferences

- Scipy Conference: 2001-today.
- Scipy India: December 12-19 2009!
- SIAM all-day workshops (2008, 2009)
- Sage days: 20 workshops and counting...
- Supercomputing’09: Python sessions.
Education

- Fossee India
- SECANT: Science Education in Computational Thinking (NSF funded)
- Sage workshops: lots of students.
- MIT 6.0X series: now in Python.

Computing in Science and Engineering (IEEE/AIP)

- Special issue in 2007: one of the most popular ever
- Special issue in 2011.
Growth

**US Federal Labs**
- LBL, LLNL, Los Alamos, Sandia, Oak Ridge, ...
- NASA (JPL, Hubble, ...)
- NIST
- NCAR
- NOAA

**Industry**
- Enthought (Austin, TX). Numpy, Scipy, Mayavi, scipy conference.
- The Python Academy: Germany, Euroscipy
- Visual Numerics: PyIMSL Studio (IMSL+Python+ipython/numpy/scipy/matplotlib)