

Python: Batteries Included



This issue's special theme is the computer programming language Python and the increasing role it plays in scientific projects. Free and universally available, Python comes with a vast standard library containing support for nearly every area of com-

puter science. An even more extensive set of third-party tools and modules covers additional tasks, from managing a Web site to doing a fast Fourier transform to distributed or parallel programming. Python's motto, "batteries included," is meant to convey the idea that Python comes with everything you need.

Interpreted Doesn't Mean Slow or only Interactive

Python is an interpreted language, and it can be used interactively. Some might assume that this limits its uses—for example, that an interpreted

1521-9615/07/\$25.00 © 2007 IEEE
Copublished by the IEEE CS and the AIP

PAUL F. DUBOIS
Contributing Editor

Table 1. Basic Python resources.

Description	Tool	URL
Python with standard library	Python	http://python.org
SciPy project (which includes Python, NumPy, and f2py)	SciPy/NumPy	http://scipy.org
Enhanced interactive Python	IPython	http://ipython.scipy.org
Two-dimensional graphics	Matplotlib	http://matplotlib.sf.net
Three-dimensional graphics	MayaVi	http://mayavi.sf.net
Connect Python to C/C++	SWIG	http://swig.org
	Boost/Python	http://boost.org
	PyCXX	http://cxx.sf.net
Connect Python to Fortran	f2py	(in SciPy)
Python Cheese Shop	2,000 more packages	http://cheeseshop.python.org

language can't possibly be fast enough for scientific programming—but as you'll see, this isn't true. Others might assume that an interactive language can't be used in a large code, or in a batch or parallel system, but that's not right either.

Python's numerical extension NumPy adds an array language similar in power to the one in modern Fortran, in which operations are performed in compiled code. Together with modules for numerical mathematics and graphics, Python by itself is a powerful computational tool. Moreover, it's easy to make your own compiled code callable from Python (and able to call Python itself). Various tools help you make this connection quickly and easily, and once you connect to Python, you have access to Python's "batteries."

Steering

My own interest in Python focuses on using it for computational steering: Python serves as the input language to a scientific application, and the actual computations are performed both in Python itself and in compiled extensions. This approach gives users the chance to be creative, serves as a built-in symbolic debugger and interactive graphics capability, and reduces development time. Nobody I know who has experienced it has ever been willing to be without it in the future.

I first wrote a system for producing steered code in 1984 at Lawrence Livermore National Laboratory. This system, called Basis, was such a new idea that it was difficult at the time to explain it to people, but it proved very successful and developers have written at least 200 applications with it. Some of the larger ones are still in use today, and Basis is still an active project (<http://basis.llnl.gov>). The key to its success is that the interpreted language I wrote for the steering was an array language quite similar to what was eventually in Fortran 95. By using the array operations, I could do real work in

the interpreter (besides calling the compiled routines to do the bulk of the work). Most importantly, the language was simple—it was enough like Fortran that users could easily read it and learn to write it.

However, trouble was coming. Basis supported Fortran 77, and I could see that not only was Fortran going to evolve but the object-oriented revolution was upon us. So in the early 1990s, I contemplated my "Act II." I even designed an object-oriented interpreter and implemented a prototype. We held periodic meetings with Basis users to discuss their requirements.

One day I found Python and saw that it had a great similarity to my prototype, was better thought out, and much further along. The one thing it lacked was an array-language extension, but a special-interest group was already looking into that. At the next meeting, I mentioned it favorably, and David Grote, a member of the group, said that he, too, had just discovered it and thought it would do the job. I decided to throw my efforts into helping design the array extension that became Numerical Python. Jim Hugunin volunteered to write the code; he later moved on to create Jython and IronPython, the Java and .NET versions of Python. I took over as the project's coordinator, and five years later, I passed the torch to Perry Greenfield (whose article about telescopes appears in this issue). Now the project is led by Travis Oliphant, who describes it more fully on p.10.

The happy ending here is that we made a good choice, and LLNL now has many Python-based efforts built from scratch or wrapped around legacy codes, and others that evolved from Basis codes: hundreds of thousands of lines of C++, Python, and Fortran 95, all working together just as we hoped, doing compute-intensive calculations on massively parallel computers.

In this Issue

We begin the issue with a basic introduction to Python in general and the SciPy project in particular. SciPy gathers the high-performance array extension together with many modules for doing common mathematical and statistical functions. Our next major article introduces the advanced interactive interpreter IPython and the matplotlib graphics package. IPython is the computational tool of choice for some people, used in much the same way as commercial products such as Matlab but with access to the full Python world and at no cost. Matplotlib is rapidly becoming accepted as the standard two-dimensional graphics utility for Python, and the Scientific Programming department on p. 90 discusses it in even greater detail.

After the larger introductory articles, we have a series of shorter pieces that present specific scientific, engineering, and educational applications. To show you a wide variety, we tried to extract the basic material on the language and its tools into the first two articles, so we suggest you read those first after sneaking a peek at the pretty pictures in the application pieces. An extra article on Python in the classroom appears in the Education department.

Although I asked the authors to state briefly why they find Python helpful, I also asked them not to extensively argue for it over some other technology choice. As in *Field of Dreams*, we think we've built it and that you will come once you see it.

I hope you enjoy our special issue and will try the Python approach to scientific computing. Table 1 should get you started, with a list of basic Python resources that are open source and available without charge. There are many, many more; start your hunt at the Python Cheese Shop (<http://cheeseshop.python.org>). ☺

Paul F. Dubois is retired and lives in Pleasanton, California, where he contributes to open source projects and writes for CiSE. His column "Cafe Dubois" will return next issue. Contact him at paul@pfdubois.com.

Coming up in our next issue

July/August: New Directions

The articles in this special issue cover a wide range of topics, from problem-solving environments to algorithm design to physics curricular material.

PURPOSE: The IEEE Computer Society is the world's largest association of computing professionals and is the leading provider of technical information in the field.

MEMBERSHIP: Members receive the monthly magazine *Computer*, discounts, and opportunities to serve (all activities are led by volunteer members). Membership is open to all IEEE members, affiliate society members, and others interested in the computer field.

COMPUTER SOCIETY WEB SITE: www.computer.org

OMBUDSMAN: Email help@computer.org.

Next Board Meeting: 18 May 2007, Los Angeles

EXECUTIVE COMMITTEE

President: Michael R. Williams*

President-Elect: Rangachar Kasturi; * **Past President:** Deborah M. Cooper; * **VP, Conferences and Tutorials:** Susan K. (Kathy) Land (1ST VP); * **VP, Electronic Products and Services:** Sorel Reisman (2ND VP); * **VP, Chapters Activities:** Antonio Doria; * **VP, Educational Activities:** Stephen B. Seidman; † **VP, Publications:** Jon G. Rokne; † **VP, Standards Activities:** John Walz; † **VP, Technical Activities:** Stephanie M. White; * **Secretary:** Christina M. Schober; * **Treasurer:** Michel Israel; † **2006–2007 IEEE Division V Director:** Oscar N. Garcia; † **2007–2008 IEEE Division VIII Director:** Thomas W. Williams; † **2007 IEEE Division V Director-Elect:** Deborah M. Cooper; * **Computer Editor in Chief:** Carl K. Chang †

* voting member of the Board of Governors † nonvoting member of the Board of Governors

BOARD OF GOVERNORS

Term Expiring 2007: Jean M. Bacon, George V. Cybenko, Antonio Doria, Richard A. Kemmerer, Itaru Mimura, Brian M. O'Connell, Christina M. Schober

Term Expiring 2008: Richard H. Eckhouse, James D. Isaak, James W. Moore, Gary McGraw, Robert H. Sloan, Makoto Takizawa, Stephanie M. White

Term Expiring 2009: Van L. Eden, Robert Dupuis, Frank E. Ferrante, Roger U. Fujii, Anne Quiroz Gates, Juan E. Gilbert, Don F. Shafer

EXECUTIVE STAFF

Associate Executive Director: Anne Marie Kelly; **Publisher:** Angela R. Burgess;

Associate Publisher: Dick J. Price; **Director, Administration:** Violet S.

Doan; **Director, Finance and Accounting:** John Miller

COMPUTER SOCIETY OFFICES

Washington Office. 1730 Massachusetts Ave. NW, Washington, DC 20036-1992

Phone: +1 202 371 0101 • Fax: +1 202 728 9614

Email: hq.ofc@computer.org

Los Alamitos Office. 10662 Los Vaqueros Circle, Los Alamitos, CA 90720-1314

Phone: +1 714 821 8380 • Email: help@computer.org

Membership and Publication Orders:

Phone: +1 800 272 6657 • Fax: +1 714 821 4641

Email: help@computer.org

Asia/Pacific Office. Watanabe Building, 1-4-2 Minami-Aoyama, Minato-ku, Tokyo 107-0062, Japan

Phone: +81 3 3408 3118 • Fax: +81 3 3408 3553

Email: tokyo.ofc@computer.org

IEEE OFFICERS

President: Leah H. Jamieson; **President-Elect:** Lewis Terman; **Past President:** Michael R. Lightner; **Executive Director & COO:** Jeffrey W. Raynes; **Secretary:** Celia Desmond; **Treasurer:** David Green; **VP, Educational Activities:** Moshe Kam; **VP, Publication Services and Products:** John Baillieul; **VP, Regional Activities:** Pedro Ray; **President, Standards Association:** George W. Arnold; **VP, Technical Activities:** Peter Staecker; **IEEE Division V Director:** Oscar N. Garcia; **IEEE Division VIII Director:** Thomas W. Williams; **President, IEEE-USA:** John W. Meredith, P.E.

revised 23 Mar. 2007

