

## Editorial

Gael Varoquaux (gael.varoquaux@normalesup.org) – *Neurospin, CEA Saclay, Bât 145, 91191 Gif-sur-Yvette FRANCE*

Travis Vaught (tvaught@enthought.com) – *Enthought, Austin TX USA*

Jarrod Millman (millman@berkeley.edu) – *University of California Berkeley, Berkeley CA USA*

---

The Annual Scipy Conference began in 2002 with a gathering of some extremely bright (and admittedly odd and passionate) folks. They gathered at the Caltech campus in Pasadena, California to discover and share ideas about a compelling approach to scientific computing. These pioneers had been using Python, a dynamic language, to perform and drive their modeling, data exploration and scientific workflows. At that time, Python already had the advantage of a very fast numerics library as well as the flexibility of being interpreted, procedural (if you wanted it to be), and, in a true academic spirit, permissively open source. That gathering 7 years ago yielded many interesting ideas, but more importantly, it crystalized a community and established relationships that persist today. From those inauspicious beginnings, the conference has now grown to a fully international meeting attracting a variety of interests—yet still inspired by the proposition of a humane language interface to fast computation, data analysis and visualization.

This year marked the 7th edition of the conference; however, it is the first edition for which proceedings are to be published. We are thrilled by this new development. These proceedings are the sign of a maturing community not only of developers, but also of scientific users. Python's use as a tool for producing scientific results is, by now, well established. Its status as the subject of research and academic publication is now being recognized and we hope that the proceedings of the SciPy Conference will help communicate how scientists and engineers are using and building upon various Python tools to solve their problems. Communication is indeed paramount to both the software and the scientific community. A scientist needs access to the methods of others; he also needs to get academic credit for his work, which is often measured in publications and citations. A developer needs to be aware of the ongoing software efforts to reduce duplication of effort; he also needs to advertise his work to potential users.

The variety of subjects covered in the talks, and in the following proceedings, is striking because it shows the extent of the use made of these tools in heterogeneous communities. This is the power of open source projects and a key factor to their success. Different actors are able to contribute to the quality of the tools in ways unique to their field and interests, resulting in a whole stack that is much greater than the sum of its parts.

Among the scientific results presented this year, there is a strong presence of the astronomy community—a community that has a long history in software engineering and is an early adopter of Python. Also

of note were the inroads that Python has gained in many other contexts, ranging from ecology and neuroimaging, to mechanical engineering and computer vision. As presented at the conference, scientists are attracted to Python by the quality and ease of use of the language itself, and the richness of the existing scientific packages. Another major presence at the conference this year was the more abstract-oriented research fields of mathematics or graph theory. The SAGE project has been very successful at building upon Python to give a consistent package for number theory, or computational mathematics in general. More component-oriented mathematical projects such as NetworkX, for graph theory, or pyinterval, for interval arithmetic, contribute to the stack from a very different perspective than the usual array-manipulating numerical packages. Finally, a sizable fraction of the talks were focused on the current effort to improve the available tools. On the numerical side, the focus is on speed, with some research in compiling part or all of the Python code to fast machine code. We also note a strong effort on the documentation of the tools, where there has been a tremendous amount of work toward filling a historical gap in user docs.

The SciPy Conference has been supported since its creation by the Center for Advanced Computing Research, at CalTech, and Enthought Inc. In addition, this year we were delighted to receive funding from the Python Software Foundation which allowed us to pay for travel, conference fees and accommodation for 10 students. We are very grateful to Leah Jones, of Enthought, and Julie Ponce, of the CACR (Caltech), for their invaluable help in organizing the conference. The conference, like much of the software presented, was truly a global effort and we are hopeful that the impact will continue to grow.

---

## Organizers

### Organizing Committee

### Conference Chairs

- Jarrod Millman, UC Berkeley (USA)
- Travis Vaught, Enthought (USA)

### Tutorial Coordinators

- Fernando Pérez, UC Berkeley (USA)
- Travis Oliphant, Enthought (USA)

---

**Web Masters**

- Gaël Varoquaux, Neurospin, CEA - Saclay (France)
- Stéfan van der Walt, University of Stellenbosch (South Africa)

**Program Committee****Program Chair**

- Gaël Varoquaux, Neurospin, CEA - Saclay (France)

**Program Members**

- Anne Archibald, McGill University (Canada)
- Matthew Brett, MRC Cognition and Brain Sciences Unit (UK)
- Perry Greenfield, Space Telescope Science Institute (USA)

- Charles Harris, Space Dynamics Laboratory (USA)
- Ryan Krauss, Southern Illinois University Edwardsville (USA)
- Stéfan van der Walt, University of Stellenbosch (South Africa)

**Proceedings Reviewers**

- Matthew Brett, MRC Cognition and Brain Sciences Unit (UK)
- Ryan Krauss, Southern Illinois University Edwardsville (USA)
- Jarrow Millman, UC Berkeley (USA)
- Gaël Varoquaux, Neurospin, CEA - Saclay (France)
- Stéfan van der Walt, University of Stellenbosch (South Africa)