

The Python Software Foundation: A Primer

David Ascher

Abstract

The Python Software Foundation is the youngest cousin of the open source foundations such as the Apache Software Foundation and the Perl Foundation. In this article, I present the PSF—who its ancestors were, why it was created, what it is today, and what we hope to accomplish with it.

1 Python’s History

Python, like a large number of open source projects created in the early 1990s, has a fairly complex history. While Guido van Rossum has an unchallenged position as “creator and principal developer” for the language, Guido did most of the work on Python while being employed by a series of organizations with very different interests in and approaches to Python. Guido came up with the first version of Python while working for CWI, a research institute in the Netherlands. CWI therefore owned the original copyright to the Python language, name, *etc.*, collectively referred to as the Python Intellectual Property. Python’s original license was a fairly “loose” license—a copy of the MIT X11 license at the time, it granted Python users pretty much every right except the right to claim Python as their own invention. Much later, CWI would grant Guido some rights to the portions of Python developed while he was a CWI employee.

In 1995, Guido moved to Virginia in the United States to work for the Corporation for National Research Initiatives (CNRI), a mostly government-funded non-profit corporation chartered with developing Internet infrastructure. CNRI viewed Python as a key technology that they wanted to use for their research and hired several other key Python developers, providing key financial support for Python’s development. One of the developers hired in 1997 was Jim Hugunin, who had started the JPython port of Python to the Java platform. Thanks to CNRI’s support, JPython went from being a rough prototype to a successful compatible port of Python. After Jim left CNRI for Xerox PARC, Barry Warsaw took over the work on JPython.

The CNRI years coincided with the massive explosion of interest in Python around the world. CNRI hosted the python.org website, owned the CVS source code repository for Python (behind their firewall), and took over the management of the Python mailing list from CWI (although CWI still runs a gateway to Usenet). Part of this explosion of interest led to the desire to form a “user group,” with the goal of both building a more formalized community as well as a means to gather financial support for Python development. Two such user organizations were started—the Python Software Association, and the Python Software Consortium. The PSA was a mostly informal organization (<http://www.python.org/psa/FAQ.html>), consisting of individuals who gave nominal membership fees (\$25/year for students, \$50/year otherwise), and “cheap” corporate memberships (\$500 or more). Some of the early Python conferences also provided some income. The PSA’s funds were used to partially defray the costs of running the website and the Python conferences. While a nice, friendly organization, the PSA did not really aim to accomplish much, and neither could it—it had no legal identity and no bank account. The PSA also did not provide the right forum for funded development of Python. Several large organizations (Hewlett-Packard, Lawrence Livermore National Labs among others) were looking for a mechanism through which they could pay Guido and other staff to add features they needed in Python. Thus the Python Consortium (<http://www.python.org/consortium/>) was created in

1999, modeled on the successful W3C and X Consortia. Python Consortium members were by definition corporations, with membership fees based on the corporation's size. Python's support for Unicode strings and the development of the new regular expression library was partially funded by Hewlett-Packard, through the Python Consortium, but that model did not grow successfully.

In May 2000, Guido, Barry, Jeremy Hylton and Fred Drake left CNRI for a California startup company, BeOpen.com (although they stayed in the Washington, D.C. area). BeOpen's business plans involved financing Python's development through a group called PythonLabs, which combined the abovementioned four ex-CNRI employees and Tim Peters, a veteran of software and hardware companies Cray, Kendall Square Research, and Dragon Systems. BeOpen was an exciting opportunity for Guido and the developers, since the promise was that the dot-com's business would provide the salaries for several full-time developers devoted to nothing but Python development. With the move to BeOpen, much changed in the Python world:

- The key developers got to work on Python full-time.
- Python shifted from version 1.6 to version 2.0, with all of the technical and non-technical changes implied by such a change in an open source project.
- Python development was decentralized, with a public CVS server at SourceForge. CVS access given to a larger number of developers (about 30).
- Public bug and patch databases hosted by Sourceforge also became key parts of the Python development process.
- A formalized "Python Enhancement Proposal" process was defined, through which anyone could suggest changes to Python.
- Much legal wrangling ensued over license details.

The last issue is one which alone could take dozens of pages to recount in detail, but we will save you the pain those details represent. It can be summarized thus:

Due to Python's development history, the current Python license is a layered license, with CWI's license at the bottom, CNRI's license in the middle, BeOpen.com's license on top of that (and as we'll see, the Python Software Foundation's license on top of all the others). The current Python license is deemed compliant with the OSI definition of open source licenses, as well as compatible with the GNU General Public License (which means that one can use Python to build GPL systems, not that the terms of the GPL apply to Python). Those interested in the details of the Python license are welcome to read it at <http://www.python.org/2.2/license.html>.

In October of 2000, the PythonLabs team left BeOpen.com (as you will remember, that was not a good time for dot-coms), and was hired by Digital Creations (now Zope Corporation). Part of the move to Digital Creations involved the realization in many people's minds that the intellectual property state of Python could not depend on who Guido happened to be working for. Each company has a different legal staff, different business priorities, and different views on open source in general and Python in particular. The Python community at large needed a stable, long-term, solid intellectual property basis on which to continue developing the Python software toolset. Thus, as part of the PythonLabs team's employment contract with Digital Creations, it was agreed that all of the intellectual property rights from PythonLabs' Python work would go to a third party, the Python Software Foundation (which was yet to be created at the time of the contract signing).

2 The Python Software Foundation

The Python Software Foundation was officially started in February 2001, with initial support from the PythonLabs team, Digital Creations, and ActiveState—another company with a strong desire to see Python’s intellectual property issues clarified. The PSF bylaws were drafted, modeled on the Apache Software Foundation, which is a widely recognized, successful foundation responsible for the Apache HTTP server as well as a variety of other open source projects.

2.1 Bylaws

The bylaws of the foundation are rather boring reading material, so only the truly dedicated will take the effort to read them—but if you are one of those, feel free: <http://www.python.org/psf/bylaws.html>.

2.2 Membership

The membership currently consists of thirty-five Nominated Members, individuals identified by the founding members as being significant members of the Python community. Most have CVS access, or have otherwise made significant contributions to Python (important non-core modules, books, advocacy, etc.). There are also currently six corporate Sponsor Members. These are corporations who were invited by the membership and contribute a \$2000 yearly fee. Sponsor Members and Nominated Members each get a single vote on any matter requiring membership votes. Members vote to select new members, and may serve on PSF committees. Once the PSF gets going, it is the committees who will have the real day-to-day power in the PSF (because they will do the real day-to-day work).

2.3 Board of Directors, Officers

The Board of Directors represents the membership and is in charge of day-to-day governance of the organization. Each board member is elected by the membership periodically. The board currently consists of myself, Jeremy Hylton, Marc-André Lemburg, Martin von Löwis, Tim Peters, Guido van Rossum and Thomas Wouters. The board appoints officers, who are charged with various administrative duties. Jeremy Hylton is Secretary and Treasurer, and Guido van Rossum is President and Chairman.

2.4 Mission Statement of the PSF

The Python Software Foundation (PSF) is a non-profit membership organization devoted to advancing open source technology related to the Python programming language. It intends to qualify under the US Internal Revenue Code as a tax-exempt 501(c)(3) scientific and educational public charity, and will conduct its business according to the rules for such organizations.

By “open source” we mean freely available technology licensed under terms compatible with Version 1.9 (or later) of the Open Source Definition established by the Open Source Initiative (<http://www.opensource.org/>).

The PSF:

- Produces the core Python distribution, made available to the public free of charge. This includes the Python language itself, its standard libraries and documentation, installers, source code, educational materials, and assorted tools and applications.
- Establishes PSF licenses, ensuring the rights of the public to freely obtain, use, redistribute, and modify intellectual property held by the PSF.
- Works with the Open Source Initiative to ensure that PSF licenses conform to the Open Source Definition.
- Holds Python’s intellectual property rights for releases 2.1 and following.
- Seeks to obtain the intellectual property rights for Python releases prior to 2.1, for relicensing under the PSF Python license, to relieve the legal burden on Python’s users. The PSF may also seek rights to other Python-related software for relicensing under a PSF license.
- Protects the Python name, and the names, service marks and trademarks associated with all other intellectual property held by the PSF.
- Solicits and manages contributions to the Python codebase, and may perform these services on behalf of other open source Python-related codebases.
- Raises funds to support PSF programs and services. The regulations for public charity funding are complex. Some consequences are that the vast bulk of funding must come from private contributions (including sponsoring memberships) and public grants, must come from a broad base, and that no single private donor can supply a substantial percentage of the PSF’s total funding. Additional revenue may be pursued in ways consistent with then-current rules for public charities and with Python’s standing as an open source project. For example, the PSF may offer to sell conference proceedings, special Python distributions, or merchandise with distinctive insignia.
- Publicizes, promotes the adoption of, and facilitates the ongoing development of Python-related technology and educational resources. This includes, but is not limited to, maintaining a public web site, planning Python conferences, and offering grants to Python-related open source projects.
- Encourages and facilitates Python-related research in the public interest.

3 Frequently Asked Questions about the PSF

Why did the PSF board pick that specific mission statement, and why does it matter?

If (or, thinking optimistically, when) the PSF gets charity status, we will be limited in our actions to the scope defined by the mission statement and bylaws. Thus, the mission statement is as inclusive as possible while remaining “on focus” and appropriate to a non-profit charitable organization dedicated to Python’s future.

Why yet another software foundation? Why not just be part of the ASF or YAS?

Representatives of both the Apache Software Foundation and Yet Another Society (the organization behind YAPC and the Perl Foundation) have generously offered to host Python within their organizations. For better or worse, we, the PSF founders, decided against it. One key reason for going it alone is that we

wanted to have sponsor members. The core purpose of the PSF is to be an organization with a mission focused on Python, not on anything else. While we are likely to share a great deal with the goals of the ASF and YAS, the overlap is not 100%, and as such, it was deemed worth the extra headache of reinventing this particular wheel. It is not unlike the decisions for doing Unicode handling differently in Python than in any other comparable language—it had to be appropriate to the rest of Python, even if it was more work than “borrowing” someone else’s code. We are in touch with folks at those foundations, and do exchange information on a regular basis.

Why does the PSF have sponsor members unlike ASF and YAS?

Corporations have always had a seat at the table when it came to shaping important decisions about Python. The PSA had corporate members, the Python Consortium was an organization made up only of corporate members. Furthermore, sponsor members were seen at the foundation of the PSF as an important mechanism by which regular, predictable income for the foundation could be established. The yearly membership fee from self-selected corporate sponsors with an explicitly stated interest in Python’s well-being and future seemed a good match. The model of sponsor members, however, has led to some accounting challenges. As the board has learned (after painful dredging through accounting books on non-profit accounting rules), one of the key requirements for getting charity status from the IRS is that the PSF must show evidence of broad public support. The precise rules are beyond the interest of most sane readers, but the short version of it is that financial support must come from a large number of parties and be relatively evenly distributed. Having just the sponsor members as the source of financing becomes a problem.

Given that it is so much work, why bother getting non-profit status?

There are a variety of reasons to seek non-profit status that apply to all similar organizations (donations are tax-deductible, tax laws are kinder, people are more willing to donate to a “designated good cause”). In addition, the PSF has a very specific motivation, due to the fact that public charities are prohibited by law from transferring their assets to anything but a like charity. This means that CNRI, which holds the intellectual property to several versions of Python, cannot even consider transferring those rights to the PSF until we are granted charity status.

How are you going to resolve the problem of showing broad public support given the issue with sponsor members?

Getting financial support from a few corporate members does not in itself pose a problem. Only if that is the only source of income and if there are relatively few of these donors is that a problem. The solution is therefore to establish a mechanism by which more people can donate more money. Luckily, we know that we can, provided we can figure out the accounting. The PSA received a continuous flow of membership checks, even though the PSA had no legal status at all, and the only thing members got in exchange was a listing on the website.

Sounds good! How can I help?

If you are a member, read the PSF-members mailing list and participate. The board wants all the input from the membership it can get in order to represent it as best as it can. When committees are formed, such as the “Public Support Committee”, participate!

If you want to help financially, get in touch with the board (psf-board@python.org), and we will let you know when and how we are ready to accept funds.

If you are an accountant, a lawyer, or know something about running a non-profit and want to provide free advice, get in touch with the board (psf-board@python.org) and let us know.

How do I become a member?

Work hard on things Pythonic. Take care of bugs, patches, releases, newbies. Write library modules, test code, documentation, docstrings. Establish credibility with existing members. Learn about non-profit accounting rules or go to law school (kidding!).

Where can I learn more?

The PSF has its own section on the python.org website (<http://www.python.org/psf>). This includes the bylaws, mission statement, records of meetings, listings of members, directors and officers, and any other information we felt appropriate to put up there. A version of this article will find its way soon after this journal goes live.

4 Current Status and Future Plans

The PSF is incorporated, and is catching up on the paperwork that is involved in both being a corporation and filing for 501(c)(3) status with the IRS. We have gotten some of the sponsor fees and are expecting more in the coming weeks, which will go towards the administrative fees accrued and pending, as well as towards getting legal and accounting advice. The legal advice is particularly important with respect to the PSF contributor forms, which will go a long way towards establishing a strong claim of ownership over the Python intellectual property. The accounting advice will be particularly important to ensure that our accounting practices will help, not hinder, our non-profit status.

The emphasis of the PSF board at this point is squarely on making sure that we can secure the intellectual property which makes up Python—and that starts with getting 501(c)(3) status, since CNRI, which owns a good chunk of Python IP, will not be legally able to transfer that IP to the PSF until that happens. On a day-to-day basis, much of what is involved in that medium-term goal makes a great deal of sense both in the short term and in the long term. Getting our financial books in order, establishing a routine for the board and the membership, reaching out and establishing broad public support—these are all goals that we have anyway—but the IRS is helping us prioritize them.

The challenging issues the board is facing right now involve the mismatch between our skills and experience (mostly doing things in the mostly logical software world) and the immediate goals for the corporation—accounting, legal, and administrative details. We are learning more about Rules of Order and accounting than we had expected. Luckily, we have not had to worry too much about purely legal issues to date—although that will come, I am sure. Setting up a non-profit foundation is turning out to be a lot more work than we ever expected, but then again, that is not altogether different from any other software project, now is it?

Some of the readers probably remember “Computer Programming for Everybody” (CP4E)—a research and educational effort aimed at making Python the ideal programming language for teaching programming. The PSF could very well solicit government grants to pursue goals not unlike CP4E—and those funds would in fact help the accounting issues, since government grants are by definition evidence of public support. Other long-term goals include more educational/promotional work, possibly funding research work, etc.

5 Acknowledgements

Thanks to Greg Stein, Tim Peters, and Guido van Rossum for their feedback on a draft of this document.

6 About the Author

David Ascher is a Director of the Python Software Foundation. He is currently employed by ActiveState and works on development tools.

7 Resources

The Python Homepage – <http://www.python.org>

The Python Software Foundation – <http://www.python.org/psf>

Computer Programming for Everybody (CP4E) – <http://www.python.org/cp4e>

The Apache Software Foundation – <http://www.apache.org/foundation>

Yet Another Society – <http://www.yetanother.org/>

The Perl Foundation – <http://www.perl-foundation.org/>