The History of Python

A series of articles on the history of the Python programming language and its community.

Tuesday, January 27, 2009

Personal History - part 2, CNRI and beyond

The Python workshop (see previous posting) resulted in a job offer to come work on mobile agents at CNRI (the Corporation for National Research Initiatives). CNRI is a non-profit research lab in Reston, Virginia. I joined in April 1995. CNRI's director, Bob Kahn, was the first to point out to me how much Python has in common with Lisp, despite being completely different at a superficial (syntactic) level. Python work at CNRI was funded indirectly by a DARPA grant for mobile agent research. Although there was DARPA support for projects that used Python, there was not much direct support for language development itself.

At CNRI, I led and helped hire a small team of developers to build a mobile agent system in pure Python. The initial team members were Roger Masse and Barry Warsaw who were bitten by the Python bug at the Python workshop at NIST. In addition, we hired Python community members Ken Manheimer and Fred Drake. Jeremy Hylton, an MIT graduate originally hired to work on text retrieval, also joined the team. The team was initially managed by Ted Strollo and later on by Al Vezza.

This team helped me create and maintain additional Python community infrastructure such as the python.org website, the CVS server, and the mailing lists for various Python Special Interest Groups. Python releases 1.3 through 1.6 came out of CNRI. For many years Python 1.5.2 was the most popular and most stable version.

GNU mailman was also born here: we originally used a Perl tool called Majordomo, but Ken Manheimer found it unmaintainable and looked for a Python solution. He found out about something written in Python by John Viega and took over maintenance. When Ken left CNRI for Digital Creations, Barry Warsaw took over, and convinced the Free Software Foundation to adopt it as its official mailing list tool. Hence Barry licensed it under the GPL (GNU Public License).

The Python workshops continued, at first twice a year, but due to the growth and increased logistical efforts they soon evolved into yearly events. These were first run by whoever wanted to host them, like NIST (the first one), USGS (the second and third one) and LLNL (the fourth one, and the start of the yearly series). Eventually CNRI took over the organization, and later (together with the WWW and IETF conferences) this was spun off as a commercial effort, Fortec. Attendance quickly rose to several hundreds. When Fortec faded away a while after I left CNRI, the International Python Conference was folded into O'Reilly's Open Source Conference (OSCON), but at the same time the Python Software Foundation (see below) started a new series of grassroots conferences named PyCon.

We also created the first (loose) organization around Python at CNRI. In response to efforts by Mike McLay and Paul Everitt to create a "Python Foundation", which ended up in the quicksand of bylaw drafting, Bob Kahn offered to create the "Python Software Activity", which would not be an independent legal entity but simply a group of people working under CNRI's legal (non-profit) umbrella. The PSA was successful in rallying the energy of

About Me

Guido van Rossum

Python's BDFL

View my complete profile

a large group of committed Python users, but its lack of independence limited its effectiveness.

CNRI also used DARPA money to fund the development of JPython (later shortened to Jython), a Python implementation in and for Java. Jim Hugunin initially created JPython while doing graduate work at MIT. He then convinced CNRI to hire him to complete the work (or perhaps CNRI convinced Jim to join -- it happened while I was on vacation). When Jim left CNRI less than two years later to join the AspectJ project at Xerox PARC, Barry Warsaw continued the JPython development. (Much later, Jim would also author IronPython, the Python port to Microsoft's .NET. Jim also wrote the first version of Numeric Python.)

Other projects at CNRI also started to use Python. Several new core Python developers came out of this, in particular Andrew Kuchling, Neil Schemenauer, and Greg Ward, who worked for the MEMS Exchange project. (Andrew had contributed to Python even before joining CNRI; his first major project was the Python Cryptography Toolkit, a third party library that made many fundamental cryptological algorithms available to Python users.)

On the wings of Python's success, CNRI tried to come up with a model to fund Python development more directly than via DARPA research grants. We created the Python Consortium, modeled after the X Consortium, with a minimum entrance fee of $20,000. However, apart from one group at Hewlett-Packard, we didn't get much traction, and eventually the consortium died of anemia. Another attempt to find funding was Computer Programming for Everybody (CP4E), which received some DARPA funding. However, the funding wasn't enough for the whole team, and it turned out that there was a whole old-boys network involved in getting actually most of the money spread over several years. That was not something I enjoyed, and I started looking for other options.

Eventually, in early 2000, the dot-com boom, which hadn’t quite collapsed yet, convinced me and three other members of the CNRI Python team (Barry Warsaw, Jeremy Hylton, and Fred Drake) to join BeOpen.com, a California startup that was recruiting open source developers. Tim Peters, a key Python community member, also joined us.

In anticipation of the transition to BeOpen.com, a difficult question was the future ownership of Python. CNRI insisted on changing the license and requested that we release Python 1.6 with this new license. The old license used while I was still at CWI had been a version of the MIT license. The releases previously made at CNRI used a slightly modified version of that license, with basically one sentence added where CNRI disclaimed most responsibilities. The 1.6 license however was a long wordy piece of lawyerese crafted by CNRI's lawyers.

We had several long wrestling discussions with Richard Stallman and Eben Moglen of the Free Software Foundation about some parts of this new license. They feared it would be incompatible with the GPL, and hence threaten the viability of GNU mailman, which had by now become an essential tool for the FSF. With the help of Eric Raymond, changes to the CNRI Python license were made that satisfied both the FSF and CNRI, but the resulting language is not easy to understand. The only good thing I can say about it is that (again thanks to Eric Raymond's help) it has the seal of approval of the Open Source Initiative as a genuine open source license. Only slight modifications were made to the text of the license to reflect the two successive changes of ownership, first BeOpen.com and then the Python Software Foundation, but in essence the handiwork of CNRI's lawyers still stands.

Like so many startups at the time, the BeOpen.com business plan failed rather spectacularly. It left behind a large debt, some serious doubts about the role played by some of the company's officers, and some very disillusioned developers besides my own team.

Luckily year my team, by now known as PythonLabs, was pretty hot, and we were hired as a unit by Digital Creations, one of the first companies to use Python. (Ken Manheimer had preceded us there a few years before.) Digital Creations soon renamed itself Zope Corporation after its main open source product, the web content management system Zope. Zope's founders Paul Everitt and Rob Page had attended the very first Python workshop at NIST in 1994, as did its CTO, Jim Fulton.

History could easily have gone very differently: besides Digital Creations, we were also considering offers from VA Linux and ActiveState. VA Linux was then a rising star on the stock market, but eventually its stock price (which had made Eric Raymond a multi-millionaire on paper) collapsed rather dramatically. Looking back I think ActiveState would not have been a bad choice, despite the controversial personality of its founder Dick Hardt, if it hadn't been located in Canada.

In 2001 we created the Python Software Foundation, a non-profit organization, whose initial members were the main contributing Python developers at that time. Eric Raymond was one of the founding members. I'll have to write more about this period another time.

Posted by Guido van Rossum at 2:01 PM

---

6 comments:

Uzah January 27, 2009 at 8:38 PM

Can you explain the following (quoted from python built-in doc strings):

>>> print dict.setdefault.__doc__
D.setdefault(k[,d]) -> D.get(k,d), also set D[k]=d if k not in D

Isn't that the worst (and most confusing) explanation for a function ever ? Notice how the "set" part is mentioned after the "get" part and almost seems like an afterthought ("..., also set....")

Reply

Aleksandr Motsjonov January 29, 2009 at 12:08 AM

"I'll have to write move about this period another time." Is my English so bad, or here is actually should be "more" instead of "move"? =)

Thanks for article.
Aleksandr, Estonia.

Reply

verte January 29, 2009 at 5:40 PM

@Uzah:

How? it does exactly the same thing as d.get(k, d), but it also sets D[k] = d if k is not in D.

Maybe you should submit a patch.

@Guido van Rossum:

One thing I'm curious about is how languages like ML influenced your design of Python. The basic types, including the unwritten rule that lists are homogenous and tuples are heterogenous, and that iterating over strings yields more strings, suggests that you had a mild taste for ML. What other languages had a marked effect on your design? Seeing as you've bought up lisp, how do you feel about that language in context?

Reply

Gareth McCaughan January 30, 2009 at 4:54 AM

Uzah has "contributed" ill-informed whingeing comments about Python to several of the posts here. I have no idea why s/he thinks this is worth while, or why s/he thinks this is an appropriate place to complain about the design of Python.

I would suggest not feeding the troll.

Reply

Guido van Rossum January 30, 2009 at 9:07 PM

@verte: I was not aware of ML at all when I designed Python. The examples you mention were all explicit in ABC, which was Python's greatest influence and itself born around the same time as ML.

Seeing tuples as records and lists as arrays in other languages also provides a clear model for that preference.

I have plenty of material about the technical influences on Python which I will post in the coming weeks.

Reply

Unknown June 19, 2015 at 9:00 AM

So, rather than over-analyzing the naming problem, I decided to under-analyze it. I picked the first thing that came to mind, which happened to be Monty Python's Flying Circus, one of my favorite comedy troupes.

That's 100% describing me!

Reply

Enter your comment...

Comment as: xandhy3369@g ▼ Sign out

Publish Preview Notify me

Note: Only a member of this blog may post a comment.

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)