SunWorld
December 1999

Search

Go

Navigate

Home
Mail this Article
Printer-Friendly Version

Navigate

Subscribe, It's Free
Topical Index
Backissues
SunWHERE
Letters to the Editor
Events Calendar
TechDispatch
Newsletters

Technical FAQs

Solaris Security
Secure Programming
Performance Q&A
SE Toolkit

SunWorld Partners

Software Store
Career Central
Sun Microsystems
ITcareers

About SunWorld

SunWorld FAQ
Advertising Info
SunWorld Editors
Masthead
Editorial Calendar
Writers Guidelines
Privacy Policy

Python reaches for stardom

Object-oriented scripting languages shows signs of celebrity

Summary
Python news, and techniques for mixing and matching. *(1,300 words)*

REGULAR EXPRESSIONS

By Cameron Laird and Kathryn Soraiz

The Eighth International Python Conference will be held in Arlington, Va., during the week of January 24, and if you're not planning to attend, you might want to reconsider. It will be a great chance to immerse yourself in the culture of scripting, as well as its technology. A few high points:

- Most of the leading lights of Pythonia will be there, some traveling from as far afield as Scandinavia and Australia.

- The technical content promises to maintain its usual high standard. The presenters really *care* about how many percentage points of performance improvement can be achieved with different schemes of type inference, and whether all attested Mesoamerican calendrical systems can be fit into a single Palm Pilot.

- It's always worthwhile to encourage the Left Coast crowd to get out a bit. In this case, the lure is Arlington, just across the river from Georgetown and the District of Columbia.

- This conference, with a projected participation of 150 working developers, might be the last Python gathering more or less free of suits.

- Finally, this might be your last opportunity to accelerate your Python education before the language *really* takes off in the next few months.

Speculations and accomplishments

We're not out to advertise the conference, and if all you want to do is learn Python, we repeat a recommendation we've made before: download and install a copy of the interpreter, read the online tutorial, and pick up *Learning Python* or one of several other good introductory books (see the Resources section

Link To Us!
Copyright

or one of several other good introductory books (see the Resources section below for more details). That's the surest investment you can make.

Beyond education, however, the mid-January conference has the potential to be a watershed in Python's social, if not technical, history. It's a handy peg on which to hang the bagful of Python news that has come out recently.

Granted, much of that news is commercial. Since its origins as a species of educational experiment, Python activity has been dominated by academic motivations. Pythoneers delight in constructing metaobjects, modeling arcane physics or human behaviors, and refining their language toward ideals of purity and rigor. However, along with the usual density of technical achievements -- including stackless Python, Python inside Java, three-dimensional Python, tuning Python for type safety, and more -- Python displays a number of symptoms of growing business respectability:

- The Python job board is bustling.
- At the end of October, the Python Consortium officially began its mission "to ensure Python's continued support and development."
- Python inventor Guido van Rossum's "Computer Programming for Everybody" project continues to receive new encouragement and grant money from the Department of Defense's Advanced Research and Projects Agency.
- New Python books seem to be hitting the store shelves every month, with David Beazley's *Python Essential Reference* being the most recent offering.
- Digital Creations has been a public-relations powerhouse, churning out weekly press releases on its open source, Python-based Web application server.
- Hewlett-Packard's ecommerce integration platform, called e-speak, seems poised to promote Python in a big way.

The last two of these deserve a bit more attention. Digital Creations has positioned itself as the model open source company, giving away Zope, its principal product, for free. The acceptance of Zope has led to strategic alliances with Netscape, Fast Engines, Microsoft, Sybase, and UserLand.com. Zope's successes certainly bring attention to Python's fitness for Web applications.

'The needs of the new digital service economy'

But while observers comment on Zope as a killer app for Python, HP is positioning e-speak to be a killer app for entire *industries*. E-speak is another technology with a Python story to tell.

HP seems very serious about the business implications of e-speak. It has redesigned the e-speak Website at least a couple of times already and, with a banner ad that recommends "new e-intelligence alliances and portals [that] turn your data into profits at Internet speed," clearly is targeting the crowd that watches the stock market. Other prominent slogans at the site define e-speak as "architecture, protocols, specification, and implementation" that addresses "the needs of the new digital service economy," and promise that "enabled by e-speak, Internet Chapter 2 is about changing the Internet experience from 'do it yourself' to 'do it for me.'"

No doubt. But there is interesting engineering behind the hype, which we'll examine more carefully in the future. Here are the essentials:

- Despite the claims in some early stories on e-speak, it is *not* helpful to think of e-speak as a language analogous to other computer languages. It's more like a platform, something best compared to CORBA or JavaSpaces, an implemented and specified architecture and protocol with conventions for use. Mike Rank, marketing manager for e-speak operations at HP, labels it a *portfolio of software;* it includes a processing core, APIs, components, service brokers, and tools.

- E-speak is a massively scalable approach to distributing processing. It has been implemented so that commercial realities of buying, selling, and communicating fit comfortably into its model of services and requests. In that sense, it's considerably more focused than a general-purpose technology such as JavaSpaces.

- The e-speak core is open source. HP released the beta 2.2 source of its development implementation on December 13.

- While e-speak's new online tutorial illustrates a Java binding various labeled J-ESI and JESI (both abbreviations stand for "Java-based e-speak service interface," and are pronounced *jee-zee*), Python has an increasingly important role in e-speak. HP wants e-speak to be language neutral, and already distributes language bindings that have been contributed for C, Perl, and Python. Moreover, the customization language for HP's current broker-services product is Python.

Accessorizing your (tool)kit

Every month, people write in to tell us, "I love Tk, but hate Tcl," or vice versa, or both, or with various substitutions of other languages. Often, however, there is no need to feel stuck with a particular language. Even if you choose to hold onto a particular antipathy, you have a good chance of finding like-minded fellow developers who have created tools just for you.

Giandomenico De Sanctis, for example, wrote that his daily work involves "traditional" languages, such as COBOL, Visual Basic, Pascal and its relatives, and Rexx. Tk's expressive power attracts him, but Tcl itself simply feels too foreign, starting with its "set a 5" assignment syntax. How can he do good GUI work with languages that lack GUI libraries, such as Unix-based Pascal and Rexx?

Mr. De Sanctis isn't the first with these needs, of course, and it's for him and others that XRexx, an X-savvy Rexx, exists. Although there has been chatter in the past about connecting Tk to Rexx and Pascal, we know of no one who has completed such a project. It certainly looks feasible.

Mr. De Sanctis pushes the envelope by asking also for a Tcl interpreter that preprocesses each line before parsing, as well as a way to have the interpreter "recognize some syntactical forms specific to other languages -- 'Pascal-like,' 'BASIC-like,' or whatever." In other words, it would recognize `a := 10` or `set a 10` or `a = 10`, concepts common to all languages, but exposed in different forms," as the same statement.

Again, these ideas have appeared before. **Regular Expressions** advisor Alexandre Ferrieux, for example, has sketched a slick, built-in preprocessor for Tcl (see Resources), and several developers use home-grown preprocessing sequences for special purposes. More dramatically, efforts are underway to unify different languages. We devoted an entire installment of **Regular Expressions** to this topic in July. What we didn't mention then was the quiet work going on behind the scenes to create a *monoidal processor*. This is, roughly speaking, a single interpreter that understands the syntaxes of several different languages.

We're looking forward to this and other advances, because scripting languages are good not only for swift creation of useful applications -- they're also great testbeds for new *kinds* of computer work. And how will Python fare as a newly popular language? It looks as though we'll learn that in the year 2000. ■

> **About the author**
> Cameron Laird and Kathryn Soraiz manage their own software consultancy, Phaseit, Inc., from just outside Houston, Texas.

Home | Mail this Story | Printer-Friendly Version | Comment on this Story |
Resources and Related Links

Also this month in SunWorld ▼ Go

Resources and Related Links

- The Eighth International Python Conference:
  http://www.python.org/workshops/2000-01/
- Download Python software:
  http://www.python.org/download/
- Introductory material on Python:
  http://www.python.org/doc/Intros.html
- Learning Python:
  http://www.ora.com/catalog/lpython/noframes.html
- Books about Python:
  http://www.python.org/psa/bookstore/
- Python job board:
  http://www.python.org/Jobs.html
- Press release -- Python Consortium created:
  http://www.python.org/consortium/press1.html
- The "Computer Programming for Everybody" project:
  http://www.python.org/doc/essays/cp4e.html
- The Department of Defense's Advanced Research Projects Agency:
  http://www.darpa.mil
- Python essential reference:
  http://islab.cs.uchicago.edu/python/
- Zope:
  http://starbase.neosoft.com/~claird/comp.lang.python/web_python.html#Zope
- Digital Creations:
  http://www.digicool.com/
- E-speak home:
  http://www.e-speak.net/
- Evolution of e-speak:
  http://www.e-speak.net/community/community_spot.html
- E-speak tutorial:
  http://www.e-speak.net/library/pdfs/tutorial.pdf
- Cameron Laird's personal notes on Rexx:
  http://starbase.neosoft.com/~claird/comp.lang.rexx/rexx.html
- XRexx:
  http://www.metronet.com/~ocon/xrexx/
- `tcl_preprocess`:
  http://www.deja.com/=dnc/getdoc.xp?AN=461112541
- "Serious programming doesn't have to be difficult," Cameron Laird and Katherine Soraiz (*SunWorld*, July 1999) -- putting different languages together:
  http://www.sunworld.com/sunworldonline/swol-07-1999/swol-07-regex-2.html
- Full listing of past **Regular Expressions** columns:
  http://www.sunworld.com/sunworldonline/common/swol-backissues-columns.html#regex

Tell Us What You Thought of This Story

○ -Very worth reading
○ -Worth reading
○ -Not worth reading

○ -Too long
○ -Just right
○ -Too short

○ -Too technical
○ -Just right
○ -Not technical enough

Comments:

Name:

Email:

Company Name:

Send data

If you have technical problems with this magazine, contact webmaster@sunworld.com

URL: http://www.sunworld.com/swol-12-1999/swol-12-regex_2.html
Last modified: Friday, February 11, 2000