InformationWeek :: reports

Dr.Dobb's REPORT

Mobile Drives Language Use

Scripting languages fall out of favor with developers

By Andrew Binstock

Last year was a year of many changes in programming: Mobile devices emerged as a major programming platform, and, at the other end of the spectrum, clouds became an established platform for data and applications.

In between, desktops and laptops got substantially more RAM and some - what more processor cores. Predictably, some of these changes trickled down to the developers' choice of languages.

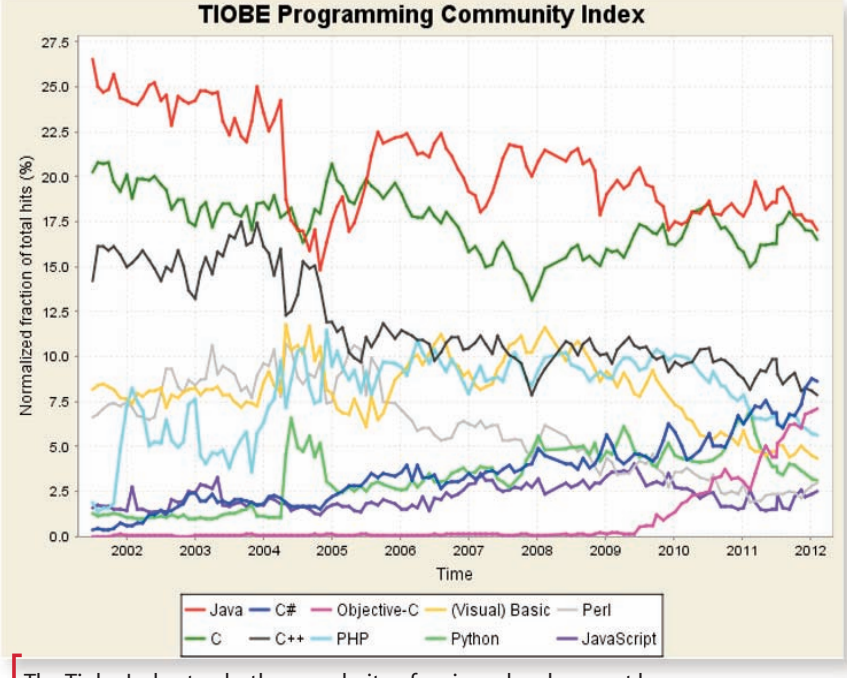
The well-known Tiobe Index, which culls frequency of online mentions of languages and language products, and translates them into a percentage of overall mentions, found the greatest language growth last year to be in Objective-C (see chart, at right). I believe few readers would be surprised by this. Between the iPhone, iPod, and iPad (and to a lesser extent Macs), the demand for Objective-C skills has clearly grown.

Mobile development also appears to be having an effect on Java, which over the last 10 years of Tiobe data has been in a steady decline. Two years ago, it began something of a comeback—driven by Android development, I believe—and this year, Java stayed essentially even with last year. My belief is that Android developers are filling the gap caused by defections to JVM languages, such as Scala, Groovy, and JRuby, which are drawing Java developers away from the language on desktop and server platforms.

As mobile programming takes off, it brings developers back to a lower level of programming that's closer to the hardware. Typically, because applications on a mobile device tend to have small code bases and require specific languages to exploit every new hardware feature, scripting languages have gained little traction in this area. (Apple's tight controls on languages and tools also have contributed to this.) As a result, for the first time in years, and possibly ever, all the primary scripting languages—Perl, Python, Ruby, and PHP—declined last year. Of these, Python's and Ruby's showings in the Tiobe Index are the most interesting.

Ruby's results are seconded by its numbers on Ohloh.net, which tracks the number of contributions to open source projects by programming language. Lines of code of Ruby changed or added last year were at their lowest level since 2006—a fifth of what they were in 2008. Part of the reason for this, I expect, is that the Ruby on Rails jubilation has finally subsided, not because of any inherent defects in the framework, but because it's only one solution to a larger problem. In addition, other frameworks have begun adopting some of the original innovations that Ruby on Rails brought to the fore.

Python's fall in Tiobe isn't reflected in



The Tiobe Index tracks the popularity of various development languages

the Ohloh numbers, suggesting that the decline is likely due to the overall market expanding faster than Python's own ranks, thereby giving it a smaller share.

PHP's decline is tied, in my view, to JavaScript's emergence. JavaScript grew modestly this year on Tiobe but significantly in open source software projects. To the extent that PHP and JavaScript functionality overlap, JavaScript will increasingly rule the day. In a new survey (kindly shown to me prerelease by Zend, the company behind PHP), 82% of PHP developers use JavaScript as a second language. (The nearest competitor, Java, came in at a piddling 24%.)

It's hard to say whether JavaScript's importance will continue to grow. I increasingly believe that it will become a universal intermediate language, with other languages, such as CoffeeScript or Dart, serving as the front-end languages. I'm hoping that the browser vendors might agree to a compiled binary representation of JavaScript as a way to further accelerate its performance in the browser. However, this development could leave open possibilities for instruction extensions by individual browser that might ultimately prove to be enough of a portability hindrance to offset any performance benefit.

MORE DR. DOBB'S ONLINE

How To Work With Dropbox
Dropbox client applets allow only basic functionality for interacting with the file storage service
informationweek.com/1325/ddj/dropbox

Using Bluetooth
Bluetooth has well-documented specs and straightforward programming APIs
informationweek.com/1325/ddj/bluetooth

Keep A Healthy Tech Diet
It's important to stay current with the mobile operating systems your customers use
informationweek.com/1325/ddj/diet

The trend away from scripting to more native languages was also evident in the .NET world. C# saw a huge rise in adoption (second only to Objective-C's jump). The surge was sufficient to move C# ahead of C++ for third place in the Tiobe Index. A significant portion of this rise, I believe, came from developers moving away from Visual Basic. This trend, curiously, undercuts the core .NET proposition, namely that developers would use multiple interoperable languages on the platform. As the number of languages on .NET consolidates around C#, that benefit is becoming less valuable. (The opposite is happening on the JVM, where there is a proliferation of alternative, interoperable languages.)

One year does not a trend make, so changes must be looked as indicative, but not conclusive. However, the move to nonscripting languages, because it's occurring uniformly across so many idioms, might well augur the end of the approach that held that developers' time was worth the sacrifice of performance and closeness to the execution platform. We'll see.

Share an online version of this story and read others by Andrew Binstock at informationweek.com/andrewbinstock. Write to him at alb@drdobbs.com.