News: Not in the Script--News of Java's Demise Is Premature

Greg Goth

Java is receiving plenty of negative attention lately. Though barely a decade old, the world's most popular programming language is being called bloated, too complicated for the world of quickly written and deployed Web applications, and ripe for dethronement by scripting languages. Forum flames regarding choice of language or development environment are nothing new among developers and analysts. However, CNET's widely read technology news site ran a story featuring comments from industry visionary Marc Andreessen headlined, "Andreessen: PHP Succeeding Where Java Isn't" (http://news.com.com/Andreessen+PHP+succeeding+where+Java+isn't/2100-1012_3-5903187.html) in October 2005, and Businessweek.com followed in December with a story titled "Java? It's So Nineties." (http://www.businessweek.com/technology/content/dec2005/tc20051213_042973.htm) Both stories implied that Java was rapidly losing ground and buzz factor to the scripting languages, notably PHP, for Web development. Implicit in the stories were sentiments that Java had become too rich and difficult for current Web-based application deployment schedules.

Unless you're familiar with the day-to-day programming requirements of enterprise applications, you might well conclude that Java is headed for the pile of "late, great" languages like IBM's PL/I, Cobol, and Fortran. It might still command dedicated niche adherents, but the momentum, market share, and mind share would be gone.
Coevolution, not revolution

Even the people most intimately involved with PHP say such conclusions are mistaken. "My perspective on all these things—it always makes me unpopular—is that the truth usually lies somewhere in between," says Marco Tabini, publisher of the Toronto-based *PHP Architect Magazine* (http://www.phparch.com). "I don't think Java is dying. I think it's evolving, and PHP is evolving."

Andi Gutmans, vice president of technology and cofounder of Zend (http://www.zend.com), a dominant PHP vendor, says the recent headlines are helping give PHP visibility to become a first-class citizen among programming languages while its strategists position it with stronger tools to convince conservative executives that it's ready for enterprise use. "For years, the industry was highly entrenched into either J2EE or Microsoft," Gutmans says. "Those were the only two solutions that were feasible. But that's not the case anymore."

PHP has been a dominant Web development language for the past two years, but it's open source and used mostly by small companies. "It wasn't getting visibility like Java or .Net," says Gutmans. "Through our work of continuously talking to the press, showcasing companies using it, and gaining endorsements by companies like IBM and Oracle and pioneers like Marc Andreessen, we've shown it's not just a small community project, it actually has significance."

Service-oriented changes

Tabini says the anticipation of a service-oriented network architecture is bringing arguments over development environments and languages to the fore. "We're coming close to the end of another life cycle in many business applications," he explains. "The last time that happened, Java was right there and ready. If it wasn't necessarily the best choice all the time, it was a very solid product backed by a very large corporation."

The situation differs today. Requirements for the next cycle of applications aren't "dictated by a 55-year-old mainframe that is starting to smoke," Tabini says, "but dictated by an evolution in the public's requirement for new technologies. Java has had to play catch-up in a way that some languages like PHP don't necessarily have to. PHP has to play catch-up in building confidence in the commercial community, whereas Java has to catch up in the perception of being suitable for things that are more relevant to the Web."
IBM, which has long been one of the industry's most prominent Java backers, has also become an enterprise champion of PHP. Marie Wieck, vice president of the WebSphere platform at IBM, says the company's partners and customers are still strong Java markets, but they've also begun to explore using PHP. The market shouldn't reflect a direct comparison of the two, she says. Both the IBM partner base and its customers still show strong and continued use of Java in the mainstream.

"We do see PHP being used when you want to do something quick, when you want to drive a project-based approach," Wieck says. She reports one WebSphere customer finding PHP cropping up in departmental uses, but the company basically prohibits leveraging it across their IT infrastructure because of manageability concerns.

**Synthesis in the marketplace**

Next year will likely bring synthesis rather than the rise of the insurgent grassroots-level scripting languages at Java's expense. The engineers behind Java tools will evangelize their efforts to make writing Java code easier while PHP's backers try making it more enterprise-friendly.

"I think we clearly have not gotten the message out well enough," says Tor Norbye, senior staff engineer at Sun Microsystems. "I think people have the idea that scripting would be something we're against. In fact, we talked a lot about it at last year's JavaOne [conference]."

The Java community is currently finalizing Java Specification Request (JSR) 223 (http://www.jcp.org/en/jsr/detail?id=223), which adds scripting support to the platform. The next version of Java already integrates Javascript support, and its designers are working on support for the BeanShell and Groovy scripting languages.

The community is also working to make Java development tools easier to use. Sun's Norbye points to Java Studio Creator 2 (http://developers.sun.com/prodtech/javatools/jscreator), specifically architected with novice programmers and expert users in mind. It won 2005 product-of-the-year awards from both eWeek Labs and developer.com. "We've tried to simplify Java APIs across the board," he says. "It's all about developer productivity and ease of use now. We've had features for years, and now we're trying to make it accessible."

PHP's backers have taken steps to heighten its credibility with enterprise users like the IBM customer that didn't want to see it move beyond departmental projects. In 2005,
Zend established partnerships with IBM and Oracle. The partnerships promote Zend Core, which features a development platform with integrated PHP and database accessibility. Additionally, Zend announced the Zend Framework, which the company calls "a body of PHP code that provides developers with the essential structure and services for an industrial-grade PHP Web application." The framework is part of a broader community effort called the PHP Collaboration Project (http://www.zend.com/collaboration). Initial contributors include IBM, Ning (Marc Andreessen's latest company), Sugar CRM (the popular open source customer relationship management company), and PHP Architect publisher Tabini among others.

One project goal is to forge a relationship with the Eclipse Foundation around PHP. The open source Eclipse platform has become the de facto standard among Java integrated development environments. The proposed Eclipse PHP IDE project (http://www.eclipse.org/proposals/php-ide) could provide the same swift jump-start for PHP among Java developers who swear by that language's IDE.

Developers' view

The Zend Framework and Eclipse work also might bring some positive cutting-edge buzz to the way developers look at PHP. Among book sales tracked by oreilly.com (http://radar.oreilly.com/archives/2005/12/ruby_book_sales_surpass_python.htm) (a ballpark estimation of any given language's popularity at a specific point in time), Java sales were down 4 percent in late 2005, as were PHP sales. However, Tim O'Reilly wrote in compiling the statistics that Java sales still showed it to be the most popular language, with about twice the number of sales as for C# and 2.5 times the number for PHP.

In contrast to falling book sales, a global survey (http://www.tiobe.com/index.htm?tiobe_index) conducted by the Eindhoven, Netherlands-based TIOBE Software showed that Java had actually gained almost 5 percent in worldwide popularity in 2005. With a 22 percent share, Java surpassed C as the most-used language, while PHP, at 9.4 percent, lost a fraction of a percentage to remain fourth most popular, behind Java, C, and C++.

Global surveys conducted by analyst firm Evans Data have shown relatively minor ebbs and flows in both Java and PHP over recent months. Evans Data President John Andrews says the trends might portend a major shift in application demands and so in development patterns.
"Is there going to be a coexistence for a number of years? Absolutely," Andrews says. "Is it greater than five years? Absolutely. Is there a momentum shift? It's hard to tell right now. As you think about back end stretching out to the front end in service-oriented architectures, as you move farther out in the front end, you see Java's importance diminish and PHP and some of the others starting to take advantage of the simplicity and the ability for them to morph much faster. When we look at some of our empirical data, looking at the innovation inflection point, yeah, we're getting very close to it, within 24 months."

**OO capabilities hold the short-term key**

Enterprise developers hesitate before jumping on any scripting language's bandwagon just yet. Although script-based applications might be much easier to write than Java's object-oriented code, they are also harder to maintain. For any application intended to run for years instead of weeks or months, Java will probably still reign supreme for some time. Chris Justus, director of software development for Ottawa-based consultancy Alcea Technologies, says OO principles such as encapsulation and reuse of objects are clearly properties enterprise developers prefer when writing complex projects. Additionally, OO code is far easier to troubleshoot than script-based applications.

"I only get called in to disasters for the most part," Justus says, "and from someone who is taking a look at code from the outside for first time, it's easier when I'm looking at a Java project." Because it's object-oriented, Java lets him use a reverse-engineering tool to generate UML and get a high-level view of the software, he says.

Although Gutmans agrees that encapsulation and reuse are sound principles, he also says that expanding PHP's OO capabilities might alienate the very developers who want a language that's easy to learn and quick to deploy. "Don't go crazy with it, don't force it on people," he says. "We have a very simple but powerful object model, and I would want to keep it that way. Where there's a lot of talk about SOAs, oftentimes you want to use the object model because you want to reflect services as objects, and PHP is at a level where it does an excellent job. It's very dynamic and can give you nicer interfaces than Java."

Gutmans' take-it-easy approach isn't universally applauded among PHP developers. PHP programmer and blogger Jeff Moore believes the PHP brain trust should step up its OO efforts to avoid losing ground to Ruby (see the sidebar). Moore writes in his blog (http://www.procata.com/blog):
Support for object-oriented constructs in PHP is relatively new, rough, and uneven. PHP competitors Ruby, Python, Java, and .Net all have OO class libraries, yet PHP does not. The classes and interfaces that PHP does have are not necessarily what I would hold up as paragons of modern OO design.

I'd like to see Zend hire a reputable object guru to help PHP turn around its object reputation. This should be someone who is known outside the PHP community, someone who has been going to OOPSLA for longer than PHP has existed. Someone who has a book on object oriented programming, articles, a blog, and who is an expert at teaching object oriented programming.

Jerry Cuomo, chief technology officer for WebSphere, says industry leaders will figure out a way to meld the quick-to-deploy, quick-to-tear-down advantages of scripting languages with the utility of OO applications in setting the direction for next-generation architectures.

"Certainly PHP can benefit from some of the OO capabilities, but if we add complexity to the point where we set that entry bar too high, we're going to lose what made it successful," Cuomo says. He sees PHP as complementary to Java. "I love the fact we have both," he says.

"We all in IBM, and the Java mecca of WebSphere, would like to see the things come more together—not in PHP becoming Java, but as a bridge. So if you have a set of enterprise Java assets, whether they're Web services, Enterprise Java Beans, or SCA [service component architecture] components, you can connect to these services with these situational applications."

Conclusion

Redmonk analyst Stephen O'Grady says those who've focused on the differences between discrete technologies are missing the larger discussion: how to expand the flexibility of development environments for audiences beyond trained developers.

"What's important about the transition is not the technological distinction as much as it is really giving lots of people the ability to program who never would have been able to do that before," O'Grady says. "That is the really interesting part. How we get at that,
whether it's better tools for some of the more complex languages like Java or C#, or simpler approaches and languages like a PHP or a Ruby, is ultimately academic.

"It's long sort of been the promise of some of the tools people that you can take business analysts and give them the ability to code. We're getting close. We're not there yet. You can't simply hand over PHP to a line of business guy and say 'Knock yourself out,' but we're getting pretty close."

Related Links


"Some Experiments with the Performance of LAMP Architecture" (http://doi.ieeeecomputersociety.org/10.1109/CIT.2005.169)


News: Ruby Buzzes on Rails

In the past several months, another scripting language, Ruby, and its framework— together called Ruby on Rails (http://www.rubyonrails.org) — has been a hot topic among developers. Stephen O'Grady, senior analyst
for Redmonk, an information technology analyst firm, says the savvy management at Zend knows how sound development tools can provide big buzz and bigger market share.

"One needs only to look at Ruby to get a feel for what a successful and constructive and helpful framework can do for a language," O'Grady says. "Ruby's been around 10 years, and even though it's very elegant, it didn't really go anywhere, even though it's dead simple. You could teach Ruby to a monkey. Nonetheless, it was not particularly popular. Along comes Ruby on Rails, and it goes through the roof. It's all developers want to talk about right now, and a lot of that is the framework."

Ruby on Rails has seen phenomenal growth in recent months. However, despite the red-hot buzz surrounding it, Ruby ranked a lowly 22nd place in the TIOBE Software global survey (http://www.tiobe.com/index.htm?tiobe_index) of software languages, with just a three-tenths of a percent market share worldwide.