Dr.Dobb's
THE WORLD OF SOFTWARE DEVELOPMENT

What Developers Think

By Jeffrey Hammond, January 16, 2010

Consider these 7 trends as you craft IT strategy

Jeffrey Hammond is a research analyst focusing on software development for Forrester Research.

When it comes to the technologies that software developers use to do their jobs, IT managers and executives don't always have a clear picture of what's going on at the ground level in their development shops. A series of Forrester Research surveys over the past three years have shown that.

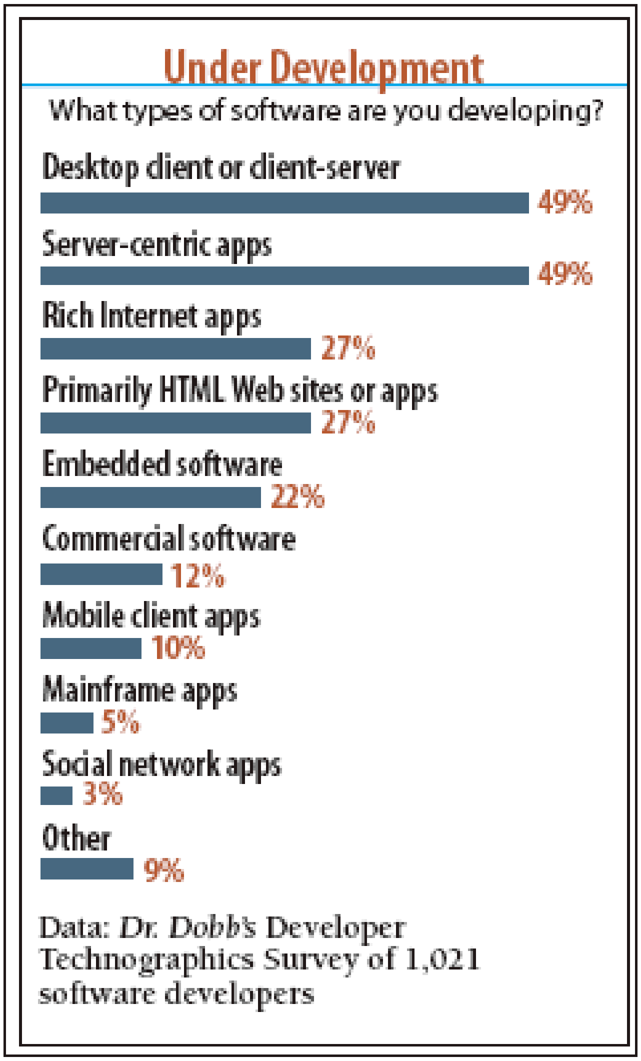
We attribute this knowledge gap, which exists regardless of company size or industry, to "technology populism" -- the growing trend where tech-savvy workers such as developers make their own decisions about what technologies to use. They help themselves to the software, collaborative tools, and information sources that best fit their needs, with minimal or no support from central IT. Much of this activity flies under management radar.

Developers were among the first users to bring consumer technologies -- like instant messaging, mashups, wikis, and mobile devices -- to their jobs. They're often the first IT staffers to evaluate and use new software technologies. So it's valuable to track what they're doing.

To better understand what's going on in development shops, Forrester conducted a survey for Dr. Dobb's of more than 1,000 platform-agnostic, programming-language-independent Dr. Dobb's readers. Collectively, they best represent the software development community. We asked them what types of applications they're writing, how they're writing them, and what they think about the state of application development. Our survey turned up seven trends that could have major implications for your IT strategy.

1. RIAs Are For Real

While client-server and server-based applications are the most common types of software projects that our survey respondents are involved in, there's a third technology to keep an eye on: rich Internet applications (see Figure 1).



Data: *Dr. Dobb's* Developer Technographics Survey of 1,021 software developers

Figure 1: What types of software are you developing?

RIAs are slowly replacing HTML when it comes to Web site development, because they combine the more-interactive user experience of rich client applications with the location-independence and reduced deployment costs of traditional Web applications. Industry stalwarts such as Microsoft, Oracle, and Sun are doing well with this transition, but other vendors, including Adobe and Google, are gaining mind-share as developers opt for their tools and frameworks, and push to get these vendors added to their companies' stables of platform suppliers.

2. Wide Use Of Open Source

Nearly four out of five developers use open source software for development or application deployment (see Figure 2). That's much higher than the 40% of IT managers and executives who say their companies use open source software.

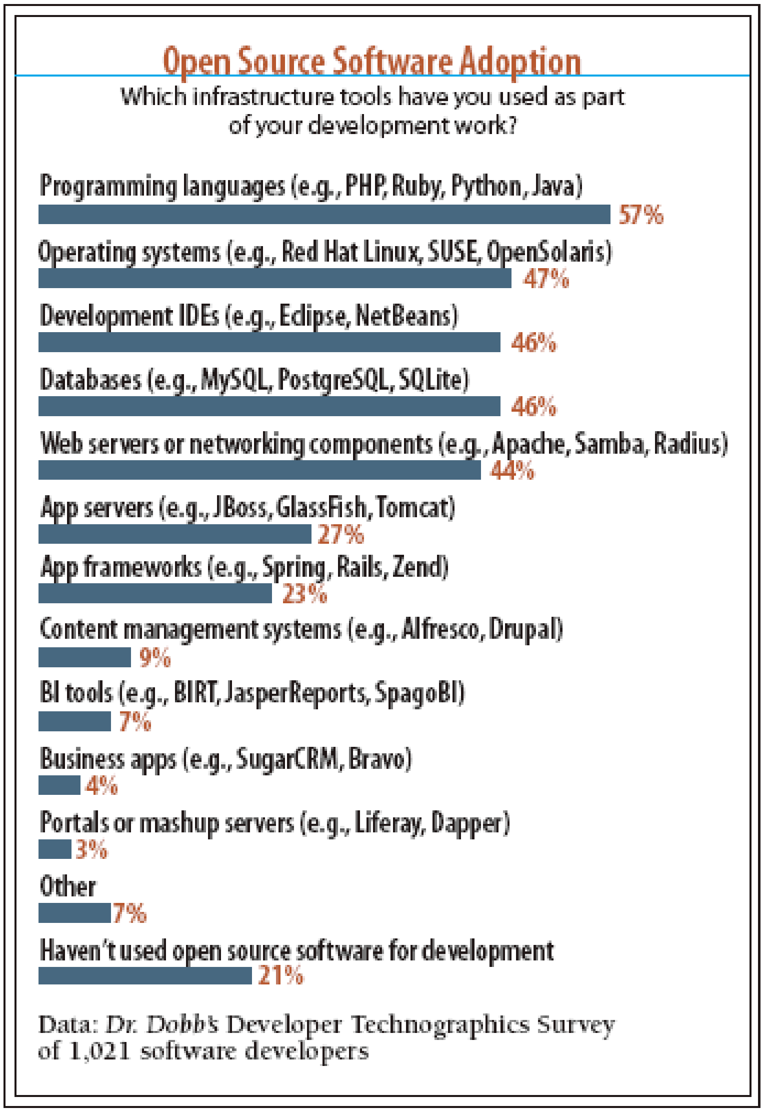


Figure 2: Which infrastructure tools have you used as part of your development work?

Developers are incorporating open source software into their application platform infrastructure in various ways, including:

- More Java developers use Apache Tomcat and Red Hat JBoss as their primary application server than either IBM WebSphere or Oracle WebLogic.
- MySOL and PostgreSQL rank third and fourth, respectively, behind Microsoft SQL Server and Oracle as the primary databases used for deployed apps.
- More than one-third of developers use Subversion for source code management; that's almost triple the share of the next most used SCM tool, Microsoft SourceSafe.

The majority of developers still write code on Windows PCs, but when they deploy apps, almost one in four targets Linux as their primary deployment operating system. And while 5% of developers have moved to a Mac as their primary development machine, most of them are targeting Linux or Windows as a primary deployment operating system, not OS X.

Linux's popularity has grown as more developers build Web applications and shift their focus to the server components, especially in the Java/Web space. The economies of scale when deploying to a LAMP stack, or Linux/Apache/Tomcat or Linux/Apache/JBoss combinations, make Linux on the server an attractive proposition. This is especially true in environments making heavy use of virtualization, since developers don't have to stop and ask, "Do I have a license for that image I just created?"

The implication of all this is clear: If you don't have an open source strategy in place, you need one -- now!

3. Virtualization, Cloud Use Evolves

Developers appear to understand the benefits of virtualization and are getting rid of the testing server under the desk that used to be a development machine. More than a third (36%) of respondents develop applications on a virtual image that IT manages, and almost as many (28%) deploy to virtual server instances.

But cloud computing is just gaining traction among developers. About 4% of respondents are deploying applications to the cloud. Of those, Amazon's EC2 is the preferred target. That means if you're just getting started on a cloud strategy, relax -- you still have time to develop one that builds on the idea of a virtual private cloud, combining the best of your own data center resources with those found in the public cloud.

4. Multilingual Developers Emerge

Developers used to identify themselves by the languages that they used -- "I'm a COBOL programmer," "she's a Java developer." But that's changing -- less than 15% of the developers we surveyed spend all their time writing in a single language.

The Web has brought about a new type of developer, one who identifies with platforms and runtimes that are tuned to building particular types of applications such as CRUD-style ("Create, Read, Update, Delete") Web applications that use tools such as Ruby on Rails and media-centric applications that use Adobe Flash. Multilingual developers create applications for these fit-to-purpose platforms, using no one language more than half the time. They're comfortable working in environments where mixing servers written in Java, .NET, and PHP with clients written in JavaScript and Adobe Flex is practically second nature. One implication of this is that when hiring developers, it's just as important to assess how quickly they can learn a new language as it is to look at the languages they already know.

5. Young Developers Drawn To Dynamic Languages

When we looked at age demographics, we weren't surprised to see that COBOL developers tended to be the oldest respondents -- after all, the language turned 50 last year. It's also no surprise to see that dynamic languages such as Ruby, Python, PHP, and JavaScript are proving most popular with developers in the 45-and-under cohort. Dynamic languages are useful when it comes to assembling components into composite Web applications, especially if runtime composition is important.

The implications? As the development staff at a shop turns over, the new generation will push to adopt these dynamic languages. IT managers must ensure that processes and application life-cycle management tools can handle the changes that these new languages bring to the development shop.

6. Agile Processes Resonate

Agile processes are growing ever more popular with developers. Of the 900 developers we surveyed who say they're using a formal methodology, 45% are using an Agile process, with Scrum being the most popular.

Twenty percent of developers using Agile say it's a key part of the success of their projects, compared with only 12% of those using iterative methodologies and 8% of those doing waterfall development. And only 2% of Agile developers feel that their methodology creates significant busywork, compared with 27% of developers doing waterfall development.

Embracing Agile development isn't easy; developers need to change their thinking, processes, and tools to implement practices such as continuous integration and test-driven development. But a fair number of development shops are well along in their transition, and that means IT managers will need to start thinking about how to apply Agile more broadly and measure the results relative to the other development processes they use.

7. Developers As An Untapped Source Of Innovation

When we asked participants to indicate if several statements described them personally, what jumped out was that 60% of respondents don't view their craft as a 9-to-5 job -- something that they leave at behind at the end of the workday. These developers continue to apply their development skills outside their jobs on side projects and in other businesses and organizations they work with.

Whether it's writing the next great iPhone application, helping out a church or local charity, or working on a startup, these developers are extending the use of their creative powers beyond the confines of their companies. In particular, we're surprised to see that one in five contributes to at least one open source project. This augers well for the future evolution of open source.

Wise development managers should recognize the creative potential of these entrepreneurial developers and think about how it can be redirected to the benefit of their organizations.

What It All Means

The picture that emerges from our survey is one of developers in transition. Software developers are adopting new technologies or techniques including RIAs, virtualization, and Agile development. They're using and contributing to open source projects, and challenging the conventions that underpin the way enterprise software and tools are built and sold. This transition will accelerate as developer tech populism takes hold and drives the adoption of new development approaches related to cloud computing, scale-out architectures that can accommodate change, and mobile Web applications.

From drdobbs.com/architecture-and-design/222301141?queryText=%26quot%3Bdevelopers+think%26quot%3B 22 September 2011

Here's the ultimate call to action for people who manage developers: Spend more time understanding what your developers are doing both at work and outside of it, and solicit their ideas about how these technologies could speed up development and cut costs for the organization. Above all, understand how your organization is changing, and that you can’t hold back tech populism among your developers -- they're going to use of all the technology that’s available, whether or not it's approved for use inside the firewall. It's best to embrace this phenomenon and channel it in a positive direction.

From drdobbs.com/architecture-and- 22 September 2011
design/222301141?queryText=%26quot%3Bdevelopers+think%26quot%3B