

The Evolution of Open Source

A new generation of companies offer a combination of open source software, smarts, and service

By Steven Grandchamp

Let's get one thing straight: open source software isn't "on its way" into the enterprise. It isn't creeping in, slowly but steadily. It isn't sneaking in the back door. And it isn't just a cheap alternative to *Windows*. No, open source software is well-established in the enterprise, and adoption goes way beyond Linux.

In a Forrester Research report issued earlier this year, 75 percent of large enterprises surveyed said they're either using or plan to use open source software, up from 60 percent in 2004. Another survey, one conducted by IDC, reported that half of the developers queried claimed that their employers' use of open source was rising. And in general, the consensus is that most enterprises are already making use of open source and plan to expand adoption to include more packages in more places. As penetration increases, the cost savings add up commensurately, and executives at all levels are taking notice.

But the rapid arrogation comes with some growing pains. Enterprise customers expanding beyond the usual open source suspects are wrestling with familiar issues, including sourcing, integration, maintenance, policy management, and support — issues historically and typically associated with proprietary software. For open source to progress and accelerate, open source must evolve to satisfy the concerns of the enterprise. Fortunately, a number of third-party companies now address the unique challenges of large-scale deployment of open source.

The Problem

The adoption of open source can be arduous, especially as enterprises deploy a broader set of open source solutions. An enterprise using the top five or so open source products (say, *Linux*, the *Apache HTTP Server*, *MySQL* and *PostgreSQL*, *JBoss*, and the many scripting languages) may find satisfactory services and support from the community or a vendor

(as is the case with *JBoss* and *MySQL*), but finding equivalent assistance and expertise for the more specialized applications can be a major challenge.

There are significant sourcing and selection issues, too. How do you choose the right open source package from a plethora of options? How reputable is the source of the package? When buying proprietary commercial software, an enterprise typically adheres to well-established procurement processes; however, the same processes have generally not been applied to open source solutions. Instead, open source software comes into enterprises directly, downloaded by developers, bypassing procurement entirely. The result can spawn legal headaches. For example, is that new Web server's license compatible with its intended use?

Support can also be particularly vexing, with enterprise customers juggling support contracts with dozens of different commercial open source companies or trying to tackle problems internally without proper expertise. Either approach is a risky and time-consuming proposition.

Open source software also introduces technical challenges. Since open source is highly componentized, you often need to deploy and integrate a variety of open source products to address a particular problem. Just think of the common "LAMP" stack: its components tend to change frequently and independently, requiring additional effort to re-deploy, re-integrate, and re-validate the entirety of the solution. Creating and maintaining such a complex environment — managing version dependencies, synchronizing updates, dealing with license compatibilities, ensuring compliance with

company policies, and maintaining many combinations, one for each development team, say — can drain resources and quickly turn savings into spending. Moreover, enterprises generally combine open source products with commercial and proprietary software. Hence, open source products don't just need to coexist with each other; the software must also integrate seamlessly with the closed-source solutions an enterprise has already invested in.

So, open source software often comes free, but it doesn't always come cheap. (Or as Richard Stallman famously said, "Free as in free speech, not free as in free beer.") The price tag ticks up as resources are wasted on policy development, education, integration, and oversight. Without a cost-effective solution, such unforeseen expenses could become a barrier to wider success in the enterprise.

The Solution

Fortunately, there's a cost-effective solution emerging in the form of third-party, open source services companies. This new breed of open source company doesn't sponsor any particular package; instead, it focuses on the broader challenges of open source, helping enterprises take advantage of the many benefits of open source, while minimizing the burdens of deployment and management.

Open source service companies can assume many forms: One form is the stack supplier; others include the support provider, systems integrator, consultant, software developer, and a combination of all of the above. Among the newcomers are Virtuas, BitRock, SpikeSource, and OpenLogic [the author's company], but the larger, traditional IT players are also beginning to take notice and build practices in this area.

While the strengths of the service companies differ, all share a common philosophy: each eschews the typical business model for the commercialization of open source. Specifically, rather than latch onto a particular open source product with a big installed base and earn revenue with subscriptions, support contracts, and consulting, each company provides solutions that extend across many open source products. Each supplier traffics in expertise about how to use open source in concert with software downloaded, purchased, and developed internally. By backing broader solutions, offering support that stretches across projects and guiding the development of sound policies, these companies offer concerned enterprises a way to mitigate risk and reduce the costs associated with open source.

The Legitimization of Open Source

It's important to understand that from the very beginning, enterprise customers set out to minimize the high risk open source seemed to pose. The cost savings and flexibility of

open source were surely always tempting, but when the open source movement was just getting started, the community wasn't prepared or equipped to provide commercial-grade support. Additionally, the cultural gap between corporate executives (the "suits") and open source developers (the "geeks") was severe. To make enterprises comfortable with open source, a market had to develop to instill the suits with the confidence needed to buy in. In other words, one of the scruffy faithful needed to shave, don a suit, and wake up before noon on behalf of open source.

One of the first companies to turn open source into income was Cygnus Solutions, founded in 1989 by three GNU hackers with a good business idea and fortuitous timing. Back then, nearly every computer vendor offered its own, proprietary C compiler, linker, and loader, but the GNU tools were also very popular. Chip makers, seeking to provide the GNU tools on each new processor, would pay a big premium to Cygnus to port the GNU software to each new architecture. Cygnus was really the first company to demonstrate that you could make money off of "free" software without adding any intellectual property. According to the Cygnus model, all you needed was the right service for a popular set of open source solutions.

Open source products don't just need to coexist with each other; the software must also integrate seamlessly with closed-source solutions an enterprise has invested in

Sleepycat Software, the makers of *Berkeley DB*, was the first company to demonstrate that you could make money selling code even as you gave it away for free. Sleepycat's novel "dual-license" model has been applied by a few other players, including Trolltech and MySQL, with great success. However, such success typically depends on the company owning a significant amount of intellectual property — a very rare thing in a world where contributions are made by engineers all across the globe.

But perhaps the biggest step in the maturation of open source has been the evolution of commercial Linux distributions. Linux was the first open source project to penetrate into enterprise environments, but it wasn't until companies such as Red Hat and Novell packaged it and backed it with service contracts that Linux really took off. To achieve success, it was essential that Linux take on some of the characteristics traditionally seen in commercial software. Enterprises needed code that was stable and a reliable vendor to turn to for support. By becoming a trusted source and certifying its code, Red Hat lessened the risk for enterprise customers.

However, even with advancements in enterprise-grade

open source in specific categories like operating systems and databases, open source continued to be hurt by a lack of more complete solutions. Even when leaders accepted the value of niche open source options, they remained skeptical that a group of packages could work together seamlessly in the same way a suite from Oracle or IBM could.

Enter the LAMP stack. In 1998, German journalist Michael Kunze described how a user could save money on software by combining Linux, Apache, MySQL, and *PHP* to form a high-performance, open source Web platform. Since LAMP's inception, it's become a major influence in the software industry, and has dispelled a lot of myths about open source. The LAMP stack was exactly what the open source community needed to really prove its worth.

Where Are We Now?

Over the last few years, commercial open source has come of age, with more enterprises using it in more ways than ever before. Each open source success adds credibility, making it easier for IT managers to choose open source the next time around. Five years ago, for instance, no one expected to see Linux in mission-critical deployments. Today, Linux is a standard platform. And the trend continues, as open source database and application servers become de rigueur.

Five years ago, no one expected to see Linux in mission-critical deployments

Oddly, or perhaps predictably, the titans of the software industry still talk as if open source is stuck on the fringe. That's dead wrong: companies already rely on open source solutions for all kinds of fundamental needs. The castle walls have been breached, and the biggest wave of the invasion is still to come.

This open source onslaught heralds a burgeoning market opportunity for open source IT services companies. Perhaps the most common business model among these companies is *aggregation*. A vendor gathers as much open source software as possible, ties a nice bow around it, and sells support contracts.

However, most customers don't want yet another vendor to provide core infrastructure software; an established vendor, such as Red Hat or Novell, can easily provide such software. Hence, the real opportunity for growth is in demystifying the use of open source. Those third-party, open source firms that focus on helping enterprises develop policies, pick projects, and manage deployments are the ones most likely to succeed and excel.

Indeed, there are companies already out there aiming to fill this need — giving enterprises the right tools and assur-

ances to sustain the broad, long-term use of open source. But this market is just getting started, and we're going to see big changes in the next year or two. The changes start with stack suppliers.

Where Are We Headed?

Right now a lot of companies are in the business of packaging and certifying open source software stacks, trying to emulate the success of stacks like LAMP and "SASH" — the open source *Java* cocktail of *Apache Struts*, *Apache Axis*, *Spring Framework*, and *Hibernate*. Certainly, a market for stacks will continue as products, but such products will likely settle into a niche, serving the tight budgets of the SMB space and the very specific needs of the production data center.

Ultimately, the business of selling fixed configuration stacks is going to get squeezed out from both sides. As customers increasingly expect infrastructure software to come along with an operating system, operating system vendors like Red Hat, MandrakeSoft, and Novell are going to do the bundling of key components with each distribution. Meanwhile, application vendors are going to work with one another to provide joint solutions. Moreover, large enterprises, where the real money is, don't just need a seal of approval — they need customization and extensibility.

The future of this practice lay in finding a way to serve up tested, certified, and stable stacks that are easily tailored to each customer's specifications. Much like Dell revolutionized the customization of PCs and made it the standard, she future of open source stacks is in made-to-order flexibility, not a menu with a handful of options.

Management Tools Mature

Anyone who's dealt with procuring open source on a large scale knows it isn't an easy task. Let's consider a partial list of options for a business unit looking to save money on licenses with an open source mail server: You can choose one of *Sendmail*, *Qmail*, *Postfix*, the new *Zimbra*, *Courier*, *SquirrelMail*, *JBoss Mail Server*, *Apache James*, and *Exim*.

These are all "open source" on some level, but narrowly conflating so many different projects into one "open source" bucket is dangerous. There's a lot more to the procurement decision than that. What are the tradeoffs of each package? Which ones work with an existing platform? Where should they be obtained? Which ones have the right licenses for a company's scheme?

In a similar vein, enterprises need ways to keep some control over how open source is used. CIOs in large companies don't want to squash the use of open source, but they do need to ensure that they're in compliance with regulations, open source licenses, and corporate policies. The CTO needs visibility

into how and where open source is used across all of the different areas of his or her organization.

Lastly, they need easy ways to manage the uptake of rapidly changing open source software. This means automated tools and processes for configuring and deploying new software. It also means ways to manage complex upgrades, so that customers can ensure that the open source software they use stays compatible while still taking advantage of new features.

This is an area where companies focused on easing the use of open source have a lot of value to add. Their core competency is expertise that stretches across a wide range of projects, and such companies can put the right packages in the hands of an enterprise's developers and give them the tools to maintain them at a much lower price point than an enterprise could do internally.

Support Consolidates and Separates

For enterprises eyeing open source, the name of the game remains the mitigation of risk, and that means support contracts. In an August 2006 report from The 451 Group, about half of all respondents identified support as the most important service they want from third-party stack providers; certification and testing was a distant second with only 16% of the vote. Right now, many enterprises are either trying to juggle a dozen or more different support contracts from providers of point solutions, or they aren't getting support at all (whether by choice or because the package they're using has no company behind it). Neither of these scenarios is sustainable if open source is to build market share in the enterprise. In the first case, customers are pouring resources into interfacing between different sources of support, and struggling to solve integration issues spread across multiple packages. In the second case, they're missing out on the tremendous potential of open source solutions that don't come from a big, monolithic vendor.

There's no cut-and-dry solution to this dilemma. Enterprises already have one or more big, incumbent IT services providers on the payroll, usually the professional services arm of someone like HP, IBM, or Sun. These practices aren't going away, not by a long shot, and large consulting customers will continue to use outside services for deeper-level support when a proprietary framework is involved. It's also not practical for an open source support company to specialize in too many projects. You run into a "long tail" problem where there isn't enough demand to justify developing expertise around every individual project — only the most popular ones call for it.

What customers need is not one shop with its own experts in hundreds of different open source packages — it's just not feasible. But they still want "one throat to choke" to feel comfortable with using a diverse group of packages. With open source, expertise is distributed far and wide, so the best

way to offer that assurance is by aggregating different sources into some form of "federated support." By developing relationships with the right developers and serving as an intermediary to secure their services, third-party open source companies can offer support for a hundred or even a thousand projects with a single contract.

Good Habits Come With Good Policies

Simplifying support and offering more flexibility in certified stacks will help enterprises take advantage of open source cost savings, but the best way to cut the cost of use in the long run is to establish clear policies early. There's a big need for consulting and training services geared towards developing policies in the enterprise dealing with the concerns unique to open source.

Policies managing procurement, deployment, version control, and code contribution must be defined. What projects are allowed? What versions? Where do they need to come from? What's the procedure to introduce them? Is the legal department involved? Are developers allowed to interact with the project community on their own?

By retaining experts to help them answer these questions ahead of time, customers can avoid lapses in maintenance and the introduction of unreliable software down the road.

When it comes to open source policies, the granddaddy of all concerns has always been licensing. The *SCO v. IBM* case in 2003, with a billion-dollar figure attached, led to a lot of paranoia about legal exposure from the poorly supervised introduction of open source software. Most of that initial fear has ebbed, but it brought the issue of open source licensing front and center as a business concern and renewed some of the earlier preconceptions of open source as a high-risk gambit.

The fact that companies like Black Duck Software and Palamida are appearing to address this issue almost exclusively should tell you just how sensitive people are to the provenance and restrictions on open source software these days. But managing open source licenses and preventing code contamination doesn't have to be hard — a well-articulated policy can go a long way in eliminating problems before they happen. Expertise in license compatibility and policies is already in high demand, and the need for it is going to explode as open source components find their way into more and more code.

The Danger of Open Source Exceptionalism and the Era of the Mixed-Source Enterprise

As open source service companies mature, enterprises eager to better leverage open source are going to take advantage, and the result will be a lot more winners in the enterprise

open source market. Smaller, more specialized open source companies will find a stronger market for their solutions when enterprises have the right support system to fit them into a larger framework.

The prospects for commercial open source in the enterprise are excellent, but let's dispel any delusions about the "open source enterprise" of the future. There are open source advocates out there who are quick to forecast the eventual doom of proprietary software, but this idea of open source exceptionalism is myopic. Open source software carries with it unique benefits and challenges, but in business terms, it is not so unlike the rest of the software industry — it's still just code. It is productized and bundled in similar ways, and there are enough failed open source businesses to prove that going open source is not a panacea for mediocre software or a weak business model. People don't just throw money at you because you throw software at them.

The more dangerous consequence of this attitude is when people fail to recognize that even as it grows like gang-busters, open source is going to remain but one piece of a much larger puzzle. Each large enterprise uses its own unique mix of open source packages, closed source software from big vendors, and proprietary code developed in-house. Many of the challenges for the future aren't just about quilting together different open source packages, but about getting open source to play well with code from the other side.

By and large, people are well past the knee-jerk, open source purist sentiments of ten years ago, but some still get carried away and talk as if open source is some unstoppable force that will subsume closed-source software for good. It's abundantly clear that corporations will continue to depend on a mix of open source, commercial, and proprietary software for the foreseeable future.

When it comes to certifying and supporting open source software for enterprise use, customers don't care about whether it works in a vacuum. What matters is that it works when it's rolled up with other code they use, whether it's purchased from Red Hat or Redmond.

Open source software in business has long since come of age, but so far, commercial open source companies have generally been narrowly focused, oriented towards providing a license, support, and other services for a specific open source solution or set of solutions. That's why the co-evolution of this second market is key.

You can call them by a lot of different names — IT services companies, stack providers, support consolidators, policy consultants, but they're all project-agnostic shops devoted to easing the wider use of open source in the enterprise. The firms cast a wide net in a way that individual project maintainers can't, providing services, support, and expertise on hundreds, if not thousands, of open source projects across the board. They're going to be the flag bearers in the next

THE OPENLOGIC STORY

OpenLogic is an enterprise open source management company and a leading provider of software, stacks, and support that enable enterprises to easily deploy and manage customized open source environments.

Originally founded in 1998, OpenLogic has been called an "elder statesman" of the open source certification space and has expanded its offerings as enterprise requirements for open source usage have evolved.

In late 2005, OpenLogic was the first company to provide an extensible product that allowed enterprises to build stacks made up of proprietary, commercial, and open source components. In 2006, OpenLogic launched a number of products and services to further help their enterprise customers manage and control open source usage.

OpenLogic now provides enterprise support for more than 160 certified open source products, offering a single point of contact for enterprise open source issues. OpenLogic handles tier 1 and tier 2 support, while tapping the open source development community through the *OpenLogic Expert Community* for help in resolving more complex issues. OpenLogic shepherds enterprise issues through the entire process from introduction to resolution, providing enterprises with commercial-grade service levels for a wide variety of products.

This year, OpenLogic also launched *OpenLogic Enterprise 4.0* (formerly known as *BlueGlue*), which uses a new distributed enterprise architecture to allow enterprises control over the use of open source across the enterprise. OpenLogic Enterprise 4.0 gives enterprises a central repository of approved, certified open source products within the corporate firewall; enables companies to automatically install, configure and integrate this software on remote servers and desktops (using existing software deployment tools if they choose); and provides an audit trail of open source software deployment.

OpenLogic's *JumpStart* program is a package of services and software to help organizations streamline the adoption of open source software in the enterprise. Under the JumpStart program, OpenLogic's open source experts guide customers in establishing a policy for the selection, sourcing, licensing, approval, usage, maintenance, and support of open source software, including short-term licenses for desired software and intensive on-site consultation.

Large enterprises continue to come to OpenLogic to help manage and control open source usage and provide consolidated support across a large library of open source projects. OpenLogic has more than 700 customers in 25 countries.

phase of open source growth, and as their practices mature, they will empower enterprises to venture farther down the open source path than ever before.

Steven L. Grandchamp is the CEO of OpenLogic, Inc. Steven has over twenty-five years of experience in the software industry. He was formerly President and CEO of Information Management Research, where he led the company's move into Enterprise Content Management, and he has also held various senior management positions with Microsoft, including the application development segment of Microsoft Consulting Services.