

Shining a Light on LAMP

By Peter Yared
October 31, 2005

Shining a Light on LAMP

Welcome to LAMP 101. Two years ago, the general IT population still didn't know exactly what LAMP was; it was a well-kept "hobby" in the dark corners of enterprise development. Although it still hasn't reached the height of cocktail conversation, LAMP has come a long way, building momentum among developers and reaching the ears of C-level executives and boardrooms of some of the world's leading organizations as they consider the bottom line: Can LAMP deliver a more flexible, low-cost environment?

LAMP has arrived. Whether you're already curious or still in the dark, the enterprise, maybe even *your* CIO, is starting to take a serious look at LAMP's benefits, and here are the basics that every developer needs to know.

What Is LAMP?

LAMP is an acronym used to describe an increasingly popular software stack, composed of widely-accepted open source projects:

- L = Linux operating system
- A = Apache web server
- M = MySQL database
- P = PHP/Python/Perl scripting languages

Although the components of the LAMP stack were never designed to work together, they have been used for years by an increasing number of developers to create numerous Web sites. The LAMP components are widely distributed and adopted, included at virtually every ISP and bundled with most Linux distributions, including Red Hat and SuSe.

According to Wikipedia, the term LAMP was first used in the late nineties, by Michael Kunze in an article for a German computing magazine. In Kunze's coining of the term, the "P" in LAMP stood for PHP. In practice, however, this letter has come to stand for any of the three scripting languages: PHP, Perl, or Python. Scripting languages are ideally suited for Web site development because they are easy to learn and highly oriented towards processing text. PHP, Perl, and Python have evolved into full-fledged languages over the past few years.

It's not just the "P" in LAMP that has expanded to include multiple meanings. In fact, LAMP has become a somewhat amorphous term, because various components are often arbitrarily swapped out, such as BSD instead of Linux (BAMP), Postgres instead of MySQL (LAPP), and so on. Today, "LAMP" is sometimes used as an umbrella term, representing open source software stacks that are pieced together, as distinct from homogeneous vendor stacks like J2EE or .NET.

Who Is Using LAMP?

In the past, LAMP, whether as a whole or as mix-and-match components, was primarily used by large, massively scalable Web sites such as Amazon, Google, Yahoo, and Friendster. Now, LAMP is seeing increasing adoption from enterprise customers, particularly for Wiki and content management solutions. Furthermore, large transactional LAMP applications are already deployed at major companies including Lufthansa, Sabre (Travelocity), Boeing, and Disney.

LAMP components are used more and more widely on their own, as well as in LAMP installations. In just the past year, IBM and Oracle endorsed PHP, Sun announced integration of PHP and Java, and Microsoft included Python as a CLR language. Linux is perhaps the biggest success story among the LAMP components, but the usage of PHP has also grown dramatically over the past five years, as shown in the following chart.

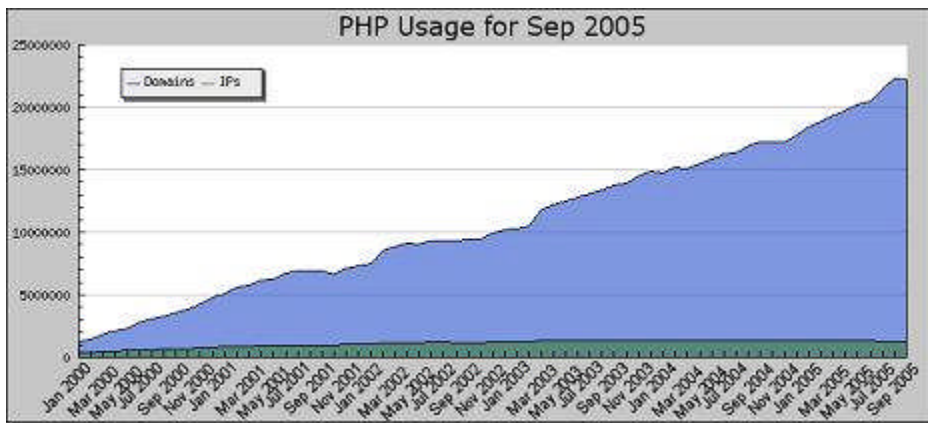


Figure 1: PHP Usage on the Rise 2000-2005

This broad vendor support, growing attention from the news media, and the increasing number of major enterprises running LAMP, are all clear indicators that LAMP has emerged as a lasting platform that is fast becoming a peer to J2EE and .NET.

How does LAMP Work?

LAMP is singularly focused towards Web applications. The architecture is very straightforward, as illustrated in Figure 1. Linux forwards HTTP connections to Apache, which serves static content directly from the Linux kernel. Dynamic pages are forwarded by Apache to PHP, which runs the PHP code to design the page. Database queries are sent to MySQL through PHP. Administration is commonly handled through phpMyAdmin, and every major enterprise management system can manage Apache and Linux.

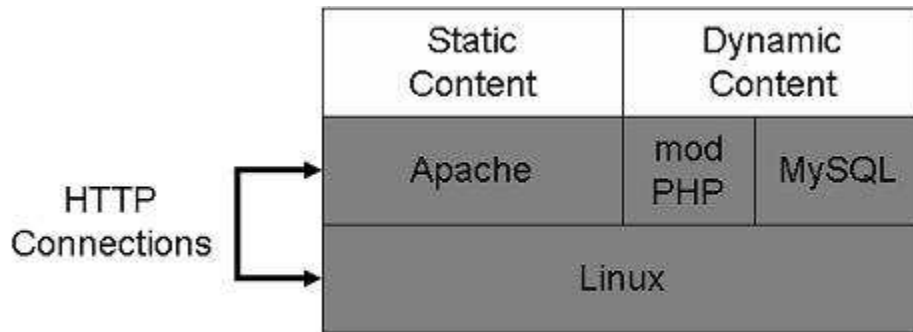


Figure 2: LAMP Architecture

There are numerous resources on the Web explaining how to quickly build a LAMP application. Most developers are productive within hours of installing the LAMP stack. Linux isn't a necessary standard because development often occurs on a Windows platform running Apache, MySQL, and either PHP, Python, or Perl (this configuration is called "WAMP").

Where Is LAMP?

LAMP is composed of several separate open source projects, each with its own owners, packages, and distributors. Each of the core components of LAMP has a major commercial or nonprofit foundation behind it:

- Linux: Red Hat, Novell
- Apache: Covalent
- MySQL: MySQL AB
- PHP/Python/Perl: Zend, Python Software Foundation, the Perl Foundation

Because it can be onerous to piece together a LAMP stack, a few stack vendors have emerged, including: (Disclosure: this includes ActiveGrid, which provides the LAMP stack underneath its application server) ActiveGrid, BitRock, SpikeSource, and SourceLabs

Although the projects are each under distinct open source licenses, the code that a developer writes for his or her applications does not need to be released as open source. However, if the software contains modifications to some of the projects (including the Linux kernel and the MySQL database), those modifications need to be published under the GPL license.

Why Lamp?

For Web applications, LAMP has been proven faster, cheaper, more flexible, and easier than any alternative. There is a strong push to LAMP by vendors ranging from IBM to Oracle to numerous startups—and these vendors are adding enterprise-grade capabilities and management to LAMP. There is no question that LAMP is not a passing trend, but now entering the mainstream as a serious contender to J2EE and .NET. Now that you know what it is, who uses it and how it works, it's time to start building a LAMP application yourself! Who knows, you just might love it.

About the Author

Peter Yared is the founder, CEO, and CTO of ActiveGrid, the Enterprise LAMP company. Most recently CTO of Sun Microsystems' Liberty Network Identity initiative, Peter was also CTO of Sun Microsystems' Application Server Division. Before its acquisition by Sun, Peter served as CTO of NetDynamics, which pioneered the then-leading J2EE application server. Earlier, he was founder and CEO of JRad Technologies, an enterprise Java company acquired by NetDynamics. Peter is a developer at heart, still actively involved in architecting the ActiveGrid Application Builder and LAMP Application Server, designed to simplify the development and deployment of rich Web 2.0 applications. Contact the author at pyared@activegrid.com.