

PHP on Windows

The pros and cons of implementing PHP in a Windows infrastructure

By Cal Evans

Abstract

This document was written for Information Technology managers in companies with existing Windows infrastructures that are considering the deployment of PHP as part of their technology mix. It discusses options that are available for installing and managing PHP on a Windows server. It reviews several commercial and open source tools that can make the process easier.

Table of Contents

Executive Summary	3
Introduction	4
Where we came from	5
Why Windows?	5
Infrastructure	5
Support for Windows specific features	6
Why PHP over .NET	6
Increased productivity through off-the-shelf solutions	6
Higher ROI on PHP based projects	6
Community Support	7
Web Server Choices	8
Deploying and Managing PHP	9
PHP.NET	9
XAMPP	9
Zend Server	9
Microsoft Web Platform Installer	9
Beyond the install	10
Conclusion	11

Executive Summary

PHP is the most popular dynamic language in use on the web today. Born on Linux, its performance on Windows has historically been poor and plagued with stability issues. However since 2006, with the help of Microsoft, PHP's performance on Windows has been steadily increasing to where now, in many cases, it is on par with Linux. With Microsoft's support of FastCGI, most, if not all, of the stability issues have been resolved. Thanks to strong support from the community and investments on the part of Microsoft, PHP is spreading out from its traditional internet roots and into corporate networks.

Now that PHP is a first class citizen on Windows, there are several reasons why IT managers are considering deploying it.

- PHP is the most popular dynamic language on the web and used to build more of the Top 10 web sites than any other language.
- PHP's support for Windows only features like COM access and Native APIs

Once you have determined that PHP is an appropriate technology to introduce into your stack, there are four tools to help you install and manage PHP on a Windows platform:

- PHP Binaries and installer from php.net
- XAMPP from ApacheFriends.org
- Zend Server from Zend technologies, Inc.
- Web Platform Installer from Microsoft

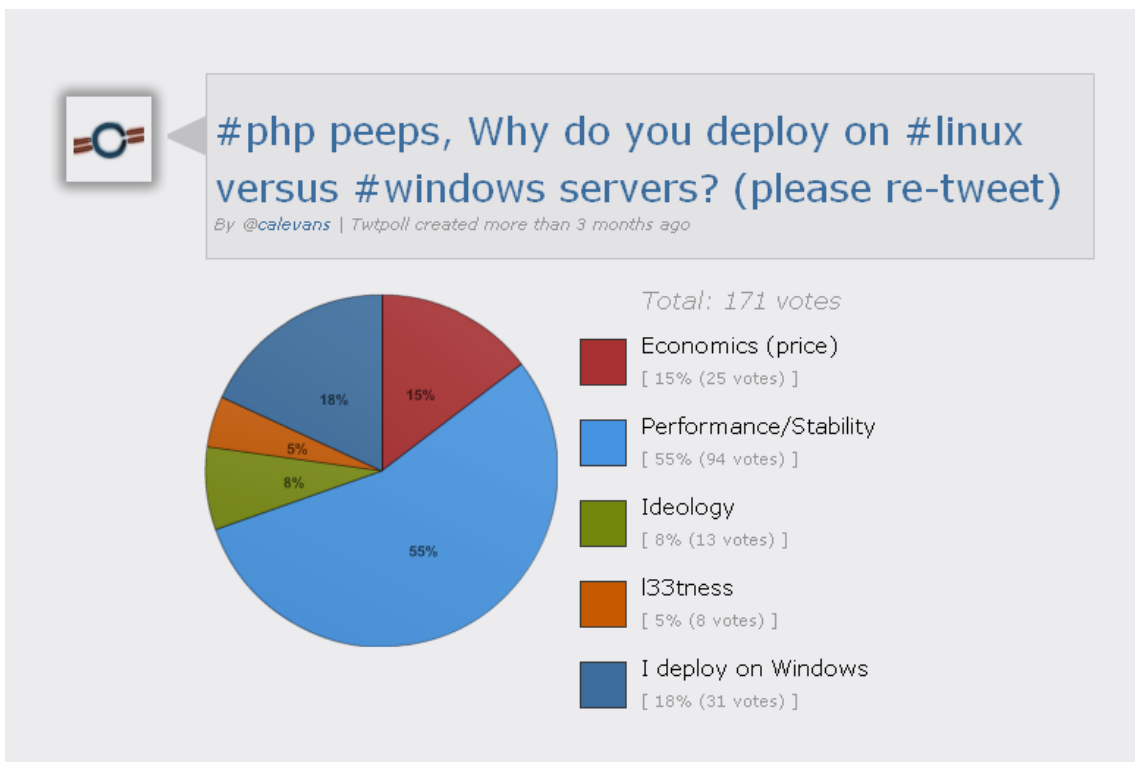
Each of these tools has strengths and weaknesses. The best tool for your situation depends on your needs and level of expertise.

After reading this white paper you will be able to determine where PHP is an appropriate technology for your Microsoft technology stack and identify the best deployment and management tools for your needs.

Introduction

Few can dispute that PHP is the most popular dynamic language on the web. More of the Top 10 web sites use PHP than any other language. The majority of PHP developers, however, ignore Windows, the most popular server operating system, as production platform for corporate networks.

Anecdotal evidence suggests that the overwhelming reasons that developers do not consider Windows as a production platform are stability and performance. Given the choice, most PHP programmers still develop and test their applications on Windows computers. When asked why developers do not deploy on Windows, here are the answers they gave.



This dichotomy has not been lost on Microsoft. In recent years, Redmond's attitude towards open source in general, and PHP in particular, has changed from an adversarial stance to one of open cooperation. This is true with regards to many other open source projects, but their efforts and impact on PHP and its surrounding community seem to be the most notable example. Not only is Microsoft working with prominent PHP projects to ensure they perform well on Windows and with other Microsoft technologies, they are investing in the core of PHP itself to bring PHP into "First Class Citizen" status on Windows. This attention, and the gains that are coming from it, are starting to turn heads not only in the PHP community but in IT management as they begin to consider PHP as a tool in their toolbox.

Where we came from

Until 2006, the problems with PHP on Windows were largely ignored. The only two solutions offered were the “thread-safe” binaries that would run as ISAPI modules but were missing important extensions, or the “non-thread safe” binaries that, while stable and complete, were not fast enough for most applications.

In 2006, Microsoft and Zend announced a partnership to improve PHP’s performance on Windows. Most developers consider this the turning point for PHP on Windows. While improvements to the core language were not immediately forthcoming, labors of this partnership soon began to bear fruit.

The early results of the collaboration between Microsoft, Zend and the WinPHP community were improvements in the speed of PHP running under Microsoft’s version of FastCGI on Windows IIS. FastCGI allows the PHP process to be recycled instead of killed off after each request. This dramatically improved the speed of PHP while not sacrificing the stability.

Microsoft’s involvement with PHP began with its partnership with Zend but it certainly did not end there. Using that partnership as a springboard they have begun making progress in other areas that will be discussed in coming sections. However, with all of their investment and posturing, Microsoft is not abandoning their core technologies like .NET. Instead, they position PHP as another tool you can deploy, not their preferred tool for dynamic scripting tasks. So, if Microsoft won’t discuss why IT departments should develop using PHP instead of .NET, we will tackle that here.

Why Windows?

The first question that comes to mind when discussing PHP on Windows is naturally, why? PHP was developed on Linux and runs fine on it. If companies want PHP, why not install Linux and run it from Apache where it’s a known entity? The answer to this is two-fold.

Infrastructure

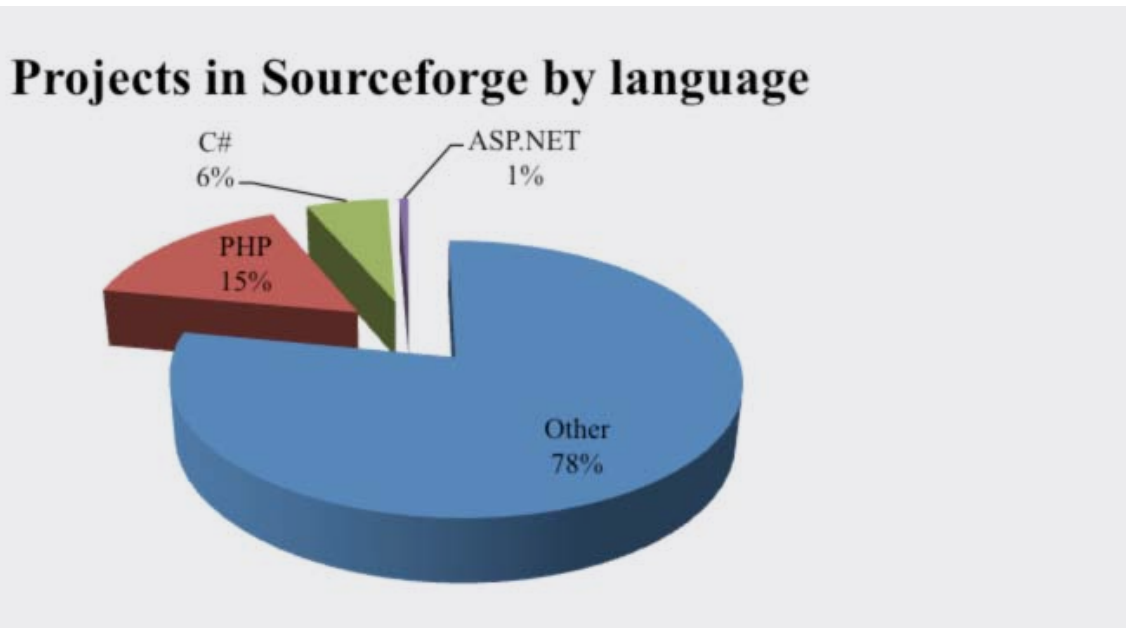
First, the existing infrastructure must be taken into account. If a company has an investment in a Windows infrastructure, they may not want to extend that infrastructure to include Linux based machines. While the tools exist to allow these two co-exist, the skills necessary to manage one infrastructure are not necessarily the skills necessary to manage the other. The investment in personnel and training may be more than a company wants to make. PHP on Windows allows companies with an existing Microsoft infrastructure to deploy the web’s most popular dynamic language.

Support for Windows specific features

On Windows, PHP has features that can't be duplicated on Linux. In addition to access to Windows native API calls, PHP on Windows has native access to COM objects like the Microsoft Office tool suite. It is true that some of this functionality can be duplicated via third-party classes; access to the original COM objects, however, can only be had on a Windows platform. Therefore, if your application requires access to Windows COM objects or access to the Windows API to function, you will need to run PHP on Windows.

Why PHP over .NET?

Once you have determined that Windows will be your platform to build on, the next question you must answer is why choose PHP over .NET. There are several reasons why a company would want to deploy PHP and PHP web based applications over .NET. First and foremost is the long list of open source solutions ready to deploy on both the intranet and Internet that will get you up and running quickly. We are not talking about frameworks for development but pre-packaged solutions, ready to deploy. Everything from forum systems to CRM systems to shopping carts can all be downloaded and quickly deployed on systems running PHP. There are, of course, open source .NET software projects available but not to the extent of PHP.



Increased productivity through off-the-shelf solutions

With IT budgets being stretched further each year, managers are now looking to off-the-shelf OS solutions to allow them to deploy projects quickly and spend their scarce resources on implementing the key business features needed. A quick scan of sourceforge.net, the largest repository of open source projects on the Internet, shows that PHP has more than twice the number of projects available than C# and ASP.NET combined.

Higher ROI on PHP based projects

The second reason to consider deploying PHP inside a Windows infrastructure is the ROI of PHP. While companies that have an existing .NET development team may not find that PHP is a suitable choice for new development, many companies are finding that the low barrier of entry into PHP development, the deep labor pool of PHP developers, and the quality of current generation frameworks and class libraries mean that PHP allows them to bring products to market quicker and cheaper than other competing technologies. Even for internal projects, speed of delivery is an important metric to be measured. Finding an open source product that has an acceptable license and is 50% of what you need to deliver can significantly impact your delivery time.

In the end, of course, the decision on whether PHP fits into a company's technology mix is a decision that can only be made by IT management. By weighing the obvious spin-up costs of any new technology with the cost savings of being able to leverage the large pool of developer talent and the open source software on the market, IT management teams are beginning to see that PHP can have a positive impact on their budget.

Community Support

One of PHP's strengths is its large and diverse community. In addition to a multitude of chat channels spread across many services that give developers access to not only help but many of the core developers, the PHP community share what they know openly on blogs and tutorial sites. Web sites like Zend's DevZone and Ibuildings' techPortal give PHP developers worldwide access to high quality tutorials covering just about every topic.

Several times a year, the community gathers at conferences to share information in real time. These conferences, such as the *Dutch PHP Conference* hosted by Ibuildings, are an excellent opportunity for developers to learn the latest techniques and best practices while gaining direct access to the core developers that work on the language itself.

Web Server Choices

Once you have decided to deploy PHP on Windows for your web based applications, then the next choice that has to be made is what web server platform you want to run it on. There are two competing platforms that developers can use to run a PHP based web application under Windows.

APACHE - The dominant web server on Unix based platforms has a Windows version that is stable and a good solution. PHP is safe to run as an Apache module on the Windows platform, just as it is on Unix.

INTERNET INFORMATION SERVER (IIS) - For running PHP under IIS, Microsoft recommends using FastCGI and IIS, both of which can be installed by the Web Platform Installer discussed below. This powerful combination will give you the best possible performance and stability. If you for legacy reasons have to install PHP as an ISAPI module under IIS, you will need the “thread-safe” binaries and will need to research the limitations of this version. Specifically, it will be missing some important extensions that are usually available.

As we discussed above, both platforms have positive and negative aspects to them. One point to consider when making the decision is what your current infrastructure already supports. If you already have IIS servers in your web infrastructure then adding Apache into the mix without good reason is probably a bad idea. However, if your current infrastructure does not have web servers in it, or your other servers are non-Windows and already running Apache, then Apache is a solid choice. Either way, there are packages and installers that you can use to get your web server and PHP up and running quickly.

PHP's stability and speed on Windows has improved dramatically in the past few years thanks to improvements in FastCGI

PHP's stability and speed on Windows has improved dramatically in the past few years thanks to Microsoft's improvements to FastCGI. Microsoft has recognized that PHP on Windows is of strategic importance to them.

Since roughly 2006, Microsoft has invested in making sure that PHP runs well on Windows and IIS. The visible manifestations of their investment are:

- Their partnership with Zend Technologies, Inc.
- Their support for the Windows PHP team
- Their increased visibility inside the PHP community

As a result of their investment, all of the installers discussed hereafter are faster and more stable than they were 3 years ago.

Deploying and Managing PHP

The ways to get PHP installed and running on a Windows server are too numerous to list here. Listed below are the best of breed options for deploying and managing PHP on Windows. Each of them has both positive and negative attributes.

PHP.NET

The obvious starting point when discussing installing PHP on Windows is PHP.NET – the official PHP binaries are located there. If you are an experienced server administrator, this may be your best bet. The current version is stable and does a good job of putting PHP on the server. Starting with PHP 5.3, they will also have a version compiled with MSVC 9.0. To remain compatible with Apache for Windows, dual binaries will be released for PHP 5.3.

XAMPP

If you have decided on Apache for your web server platform then XAMPP is a good choice. It can install Apache, PHP and MySQL and its own small control panel that will allow user to start and stop services. We caution against installing it in production however as it installs many other pieces like phpMyAdmin which are not appropriate on a production server. If XAMPP is to be used in production then the custom install option is strongly recommended so that unnecessary modules can be removed.

Zend Server

Zend Server is the evolution of the Zend Core/Zend Platform. It is a subscription-based service that installs and configures PHP and optionally MySQL for a variety of operating systems including Windows. Zend Server's control panel allows administrators to manage PHP options stored in the php.ini, activate or deactivate extensions and monitor the log files. When used in conjunction with Zend Studio, Zend Server allows developers to drill down on an error from the log files and call up the offending code.

While Zend Server lacks the clustering and monitoring features of its enterprise counterpart, Zend Platform, it is still an option for managing single instance servers like small sites or intranet servers.

Microsoft Web Platform Installer

Microsoft has recently unveiled a new tool to make the installation of PHP, IIS extensions, and web applications much easier on Windows servers. Microsoft Web Platform Installer is a universal installer that, among other things, will install PHP and a small but growing selection of PHP applications that can be installed.

If your infrastructure is built on Microsoft technologies, Web PI is your best choice for getting PHP up and running.

While it lacks the managing and monitoring features of Zend Server and the additional options of XAMPP, overall, it does the best job of installing PHP properly and allows you to install most of the other technologies necessary for PHP applications to run.

Beyond the install

As discussed above, in recent years Microsoft has shown its commitment to PHP on Windows with both monetary investments and community involvement. An additional positive sign is their involvement in open source projects in the PHP eco-sphere. For the first time in their history, Microsoft has contributed code to an open source project, ADOdb, an LGPLed database abstraction layer for PHP. This investment in time, plus their open advocacy of PHP projects through their WebPI shows that they are seriously supporting both open source in general and PHP in specific.

Microsoft, through its investments in and evangelism of PHP, is signaling to its customers that PHP is an acceptable solution to their dynamic scripting problems.

Microsoft continues to support the PHP community and its surrounding eco-system. Beyond simple support though, they are now active participants. Their presence can be felt in every corner of the PHP community, from the core development team to sponsoring and attending conferences. Once a rarity at PHP gatherings, these days Microsoft employees are not only present but engaged in the conversation.

All of these factors taken together signal to the PHP community and more importantly to Microsoft's customers, that PHP is well on its way to becoming a first class citizen sitting alongside its own offerings like ASP.NET to solve web based problems.

Conclusion

We have discussed some of the factors that need to be considered when deciding whether or not PHP is right for your particular technology mix. Many of the factors that make PHP the best choice for building web applications, makes it an excellent choice for building web based Intranet applications. Some of these factors include:

- PHP's low TCO compared to other dynamic languages
- PHP's large off-the-shelf codebase
- PHP's broad base of community support
- Microsoft's recent and ongoing support of PHP as a viable technology on Windows

All of these factors are working together to boost PHP's profile in companies with Microsoft based infrastructures. PHP is now an excellent choice for companies looking to enjoy the rapid development and ROI that PHP offers for internal applications. As Microsoft continues to work with the PHP community, it won't be long before Windows is on equal footing with other server platforms and companies can confidently deploy both internal and external applications on it.

About the author

Cal Evans is the Director of the PHP Center of Expertise at Ibuildings. He has been involved in IT development and management for the past 26 years and for the past 9 has developed primarily in PHP and MySQL. Cal is the author of a book and many articles on PHP development and can regularly be seen speaking at PHP conferences around the world. Cal blogs regularly at <http://www.ibuildings.com/blog/authors/Cal-Evans>.

About Ibuildings

Within the Netherlands and UK, Ibuildings is the leading authority in the field of PHP and the representative of Zend Technologies. We supply Internet solutions to organizations that deploy mission-critical web applications. Our strength lies in the design and construction of substantial Internet solutions, based on full-scale outsourcing or on the supplying of specific services (project management, training, consultancy, development).

Netherlands

Ibuildings

www.ibuildings.nl

088 00 24 000

- Goeman Borgesiuslaan 77
3515 ET Utrecht
- Gildeweg 39A
4383 NJ Vlissingen

United Kingdom

Ibuildings

www.ibuildings.co.uk

+44 208 359 1214

Stirling House
9 Burroughs Gardens
London NW4 4AU