From PHP4
to PHP5

Author: Ivo Jansch / Ibuildings, The Netherlands
September 2007

Zend
The *php* Company

Table of Content

Preface

A little over three years ago, the PHP community released version 5 of its popular programming language. PHP5 has now become the version of choice for many new PHP projects. PHP4 however remains popular and is still widely used. This whitepaper is aimed at developers who are still using PHP4 and considering switching to PHP5. It is also intended to provide IT managers responsible for PHP applications additional information for their decision on whether to stick with their current version or migrate.

This paper seeks to answer the questions, "What are the benefits of migrating to PHP5? And, equally importantly, what are the risks?" The information below will put you in a better position to make an informed decision between PHP4 and PHP5.

"If it ain't broke, don't fix it"

Programming is a delicate business. Different software applications are frequently so closely intertwined that a change in one part of a system can have undesired side effects elsewhere. If an application has been live for some time and is running smoothly, it's better not to meddle with it if there is no pressing need to change anything. That's a fairly persuasive argument. But what happens if something does go wrong? Will fixing bugs depend on whether or not the application is written in PHP5? Not necessarily, but it might. And that represents a risk.

The PHP community has announced that December 2007 will be the end of life for PHP4. While this may sound ominous, it doesn't mean that PHP4 will simply vanish. But it does mean that the PHP community will stop maintaining PHP4. So if the bug in your application is a result of erratic behaviour in PHP4, finding a fix may be difficult. Starting in December, only critical security errors will be addressed. In August of 2008, this support will also cease.

This means if someone discovers a security flaw in PHP4 after August 2008 and it affects your application, you may not be able to solve the issue immediately. At that point you could upgrade to PHP5, but as we'll see later in this paper, the migration does require some effort. During a crisis caused by software issues may not be the most opportune time for your company to carry out the upgrade.

We don't mean to be alarmist, but end-of-life announcements should be given serious consideration. If you migrate now, you can do so on your own schedule, on your own terms, and with enough time to properly test the application in PHP5. There's still plenty of time before end-of-life. But the sooner you start, the better.

The Benefits

There are plenty of other reasons to switch to PHP5 that don't involve doom and gloom. There are enormous advantages to migrating to the latest version. The first item to mention requires no effort whatsoever and simply comes along with the package:

Performance

PHP5 is faster than PHP4. The performance increase depends on a lot of factors, including the way an application is built, but in most cases, you will notice significant gains.

A key factor in the performance jump is the Zend Engine 2, the heart of the PHP language. Object handling has been almost completely rewritten to transform it from what was basically a wrapper for associative arrays in PHP4 to a true object model. Object-oriented code in particular will perform faster in PHP5 as a result.

Sebastian Bergmann, author of the PHPUnit test framework, benchmarked several PHP versions. The graph below summarizes his results.
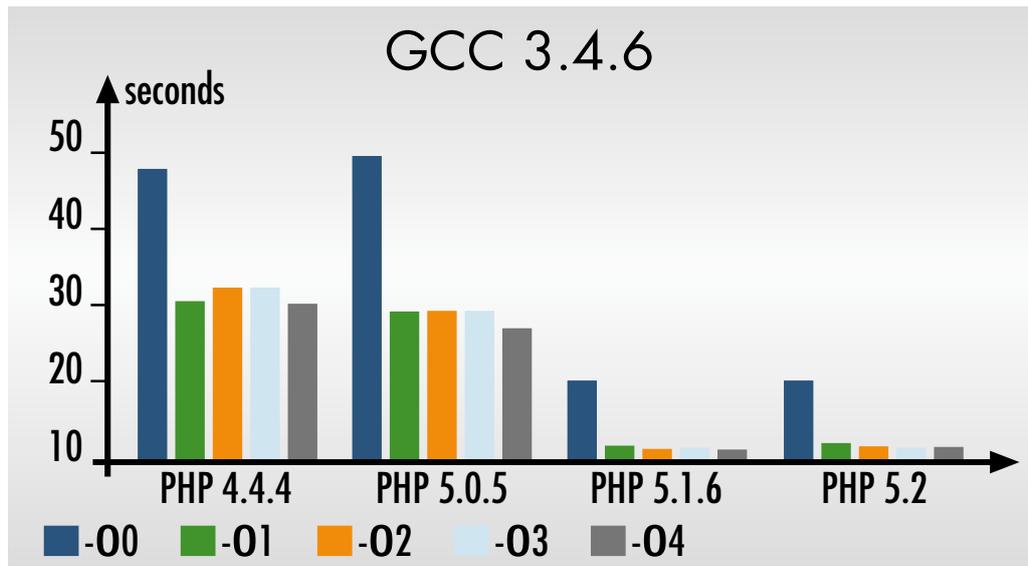
O0 to Os are the different scripts that Sebastian used to test the PHP versions listed on the horizontal axis. In early PHP5 versions, the performance increase was noticeable but not spectacular. In PHP 5.1, after the Zend Engine 2 was refined and tuned, the benefits became clear – PHP 5.1 is about 4 times faster than PHP 4. Increased performance means better response times for your visitors and less hardware needed to run your website.

Maintainability

There are several features in PHP 5.1 that improve the maintainability of applications. These include the new object model mentioned above which supports features such as abstract classes, interfaces, and encapsulation (private and protected methods and properties). This paper will not dive into the technical details of these features since other sources provide excellent examples of the new functionality. The important thing to note is that these features make it easier to apply software architecture to a PHP application. In other words, it becomes easier to write properly structured, high-quality code. The code base will be cleaner, easier to read, and consequently simpler to maintain.

---

1) The official information can be found at http://sebastian-bergmann.de/archives/634-PHP-GCC-ICC-Benchmark.html.
2) The official PHP manual has an appendix that discusses the changes between PHP4 and PHP5. It can be found at http://www.php.net/manual/en/migration5.php. A Google search for ‚php5 oo' will yield several other useful sites.

Robustness

A new mechanism for error handling was added in PHP5. In PHP4, there was no dedicated mechanism for error handling at all. Programmers had to use regular language constructs such as if-statements to check for errors and use special return values to tell a calling method that something went wrong during the execution of a function.

PHP5 also supports exception handling. Exception handling makes it possible to create what is known as a try block to execute a piece of code that might contain an error and implement a catch that is called whenever something actually does go wrong. This makes it easier to separate the 'what should happen in the case of an error' code from the regular application business logic. This approach leads to cleaner applications, which in turn improves developer productivity and application robustness.

Future-proof

The internet no longer consists solely of stand-alone applications. It is becoming more common for applications to interoperate, share data, integrate, and combine into 'mash-ups' to provide a richer internet experience. These kinds of applications allow you to read CNN's headlines on your personalized Google homepage, display Flickr images of locations in other websites, and track packages in real time on a map of your home town.

There are several technologies involved in these types of applications. Web services play a key role. A web service is basically an interface from one site to the other. When a site displays a UPS package in transit to a destination, it uses UPS's web service to determine the coordinates and Google's or Yahoo's map service to display it on the map. Popular protocols for web services include SOAP and REST, two standards which make sure that websites can communicate with each other.

XML is another very important web app technology. XML is a data format for transferring data from one service to the other and consists of plain text messages that can be read by both computers and human beings, making it an ideal standard for interchanging data.

The proliferation of these technologies is another reason to switch to PHP5. In PHP5, functionality was added to make it extremely easy to work with web services. If someone needs to connect to your application through a web service interface, you can create a web service for your application with just a few short lines of code in PHP5. And the reverse, calling web services of other applications, can often be accomplished with as few as 2 lines of code. This paper doesn't go into the technical details, but an internet search for "PHP5 SOAP" should provide numerous examples. In PHP4, external libraries were necessary to properly implement web services. In PHP5, however, it's a feature of the language itself.

Generating and parsing XML messages is also extremely simple in PHP5. In PHP4, working with XML was very cumbersome using the DOM or SAX parser. In PHP5, a new 'SimpleXML' extension has been added which makes PHP4 XML code look extremely complex by comparison. Using SimpleXML, an XML message can be read and parsed with a single line of code.

Product support

There are more and more products available that take advantage of PHP5's new features and add additional value to PHP5.

Zend Framework

The Zend Framework leverages PHP5's new features to provide a rich set of classes that PHP developers can use to build applications. Examples include interfaces to many popular web services. The framework uses PHP5's web services functionality to communicate seamlessly with web services and provide an easy-to-use interface for developers. This means that with PHP5 and the Zend Framework, you can talk to Amazon's or Flickr's web services without even knowing what a web service is.

Zend Core

Zend Core can be considered a PHP5 distribution. It includes Apache, MySQL and PHP itself, eliminating the hassle of having to compile or install these components manually. It also has an updater that ensures that you'll always be using the latest stable, tested version of PHP and its extensions.

Open Source products

Many Open Source PHP projects are already taking advantage of PHP5's features. Increasingly, these projects are recognizing the added value PHP5 provides and are considering dropping support for PHP4 altogether. Several projects have joined forces and launched the website http://gophp5.org, which features a list of applications that will fully adopt PHP5 by February, 2008.

The Migration Process

Earlier in this paper, I promised to also discuss the risks of moving from PHP4 to PHP5. Every software upgrade involves certain risks, and it's important to account for these risks and ensure that they are minimized.

One risk everyone faces is that their applications may not work properly under PHP5. Because of all the changes between PHP4 and PHP5, in particular in PHP's object model, you may need to modify your code before it will run properly on PHP5. Mitigating this risk is a matter of solid preparation and testing.

The first step is to familiarize yourself with the changes between PHP4 and PHP5. The PHP manual contains an appendix that lists the most important changes:

http://www.php.net/manual/en/migration5.php

The next step is to assess the impact of these changes on your code base, for example by comparing the list of new functions in PHP5 to your own functions to avoid naming conflicts. Some changes might be obvious, others might only be found after thorough testing. Companies who practice test driven development for automated testing of the code base will have an advantage here.

The most important thing to remember is that the upgrade is not a trivial task. It is not a matter of just updating PHP — the application also needs to be checked and tested. If you have large applications you need to migrate, migration support services provided by companies such as Zend can help you carry out the migration.

Conclusion

This paper lists several benefits of PHP5 over PHP4, and also explains how the upgrade process is not to be taken lightly. It also stresses the importance of migrating to PHP5 before PHP4 has reached end-of-life.

While upgrading will require some effort, there are enormous benefits to be gained from the migration. These benefits include:

• Better performance
• Improved maintenance thanks to more structured object-oriented code
• Increased productivity
• Better support for popular technologies such as Web Services and XML

Finally, the paper has provided a migration path that can be followed in order to efficiently migrate existing environments from PHP4 to PHP5.

About Ivo Jansch and Ibuildings

Ivo Jansch is the CTO of 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. We can run complete projects or take part in existing development projects supplying specific services (project management, training, consultancy, development) and software products and toolkits.

About Zend Technologies
Zend Technologies, Inc., the PHP company, is the leading provider of products and services for developing, deploying and managing business-critical PHP applications. PHP is used by more than twenty-two million Web sites and has quickly become the most popular language for building dynamic web applications. Deployed at more than 15,000 companies worldwide, the Zend family of products is a comprehensive platform for supporting the entire lifecycle of PHP applications. Zend is headquartered in Cupertino, California. For more information, please visit www.zend.com

ZEND: The holistic approach to PHP
• Application management and availability with Zend Platform™
• Development of PHP applications with Zend Studio™, the leading development environment for PHP
• Certified and officially supported PHP installations with Zend Core™
• Access to expertise of leading PHP experts with Zend Professional Services™
• Improved PHP knowledge through Zend Training™ offers
• Protection of intellectual property and source code and administration of licensing models with Zend Guard™
• The certified and manufacturer-supported collection of PHP components and PHP libraries – Zend Framework™
• First class 24/7 support via the Zend Network™

Corporate Headquarters:
Zend Technologies, Inc.
19200 Stevens Creek Blvd.,
Suite 100
Cupertino, CA 95014
Tel.:   1-888-PHP-ZEND
         1-888-747-9363
Fax: 1-408-253-8801

International:
Zend Technologies, Ltd.
12 Abba Hillel Street
Ramat Gan
Israel 52506
Tel.:   972-3-753-9500
Fax: 972-3-613-9501

Italy:
Zend Technologies
Largo Richini 6
20122 Milano
Italy
Tel.:   +39 02 5821 5832
Fax: +39 02 5821 5400

Central Europe:
(Germany, Austria, Switzerland)
Zend Technologies GmbH
Bayerstraße 83
80335 Munich
Germany
Tel.:   +49-89-516199-0
Fax: +49-89-516199-20

UK:
Zend Technologies
Stirling House
9 Burroughs Gardens
London, NW4 4AU
United Kingdom
Tel.:   +44 20 8458 8550
Fax: +44 20 8458 8550

France :
Zend Technologies SARL
5 Rue de Rome
ZAC Les Algorthmes
93110 Rosny sous Bois
France
Tel .:   +33 1 4855 0200
Fax :   +33 1 4812 3132

Your local partner: