Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 104 (2017) 51 – 56

Procedia
Computer Science

ICTE 2016, December 2016, Riga, Latvia

Analysis and Practical Application of PHP Frameworks in Development of Web Information Systems

Natalya Prokofyeva[a], Victoria Boltunova[a,]*

[a]*Riga Technical University, 1 Kalku Street, Rīga, LV-1658, Latvia*

Abstract

This paper provides various PHP programming framework (CakePHP2, CodeIgniter, Symfony2, Yii and PhalconPHP) popularity overview and comparison using various criteria. Based on the data obtained during the study two frameworks were selected for deeper analysis - Symfony2 and PhalconPHP. This article offers a description of the architecture and main features of selected frameworks (routing, template engine, etc.). During framework comparison a performance test was developed with a goal to determine performance and effectiveness of frameworks during the same task. For performance testing a „Ticket Reserving System" cashier list section was chosen. Tests were performed using ab.exe (Apache Benchmark) tool that comes with the Apache Web server. Based on the comparison results recommendations are made that allow Web developers to choose a framework for creating a real-world Web project. It is planned to expand the developed Web system capabilities using Web services that are based on open standards and protocols (SOAP, XML-RPC, REST, etc.).

Peer-review under responsibility of organizing committee of the scientific committee of the international conference; ICTE 2016

Keywords: PHP frameworks; Web programming; Information system development

1. Introduction

Framework facilitates Web programming and makes it better organized in many ways. First of all, frameworks increase programming productivity because writing a piece of code that usually takes hours and takes up hundreds

* Corresponding author. Tel.: +371-29989413.
E-mail address: lboltunova@apollo.lv

Peer-review under responsibility of organizing committee of the scientific committee of the international conference; ICTE 2016
doi:10.1016/j.procs.2017.01.059

lines of code can be done in minutes with the help of framework built-in functions. Secondly, widely used framework has a major security advantage since its users become long-term testers. If a user finds a security problem, he can notify about it on the framework developers website so that the developer team would be able to fix it. Thirdly, often most popular frameworks are free and since they help the developer to write code faster the final cost will be smaller. Fourthly, a framework usually comes along with a support team, documentation or large support forums where users can quickly get answers. Framework selection is an important step because the speed and quality of the work depends on it. You can find many articles on how to choose an appropriate framework for your task[1,2]. Main aspects to take into consideration are usage context, license, software pattern, hosting requirements, ease of installation, core library, learning curve, DB abstraction and ORM, included JS libraries etc[3].

For a long time PHP programming language was not considered as a sufficiently serious language for large Web application development. It was known that it is popular and perhaps good for small projects but most of the appreciation was reserved for the aristocratic framework elite like Spring, Ruby on Rails or Django. Only recently the situation has significantly changed. It became clear that PHP has not only one but three web frameworks that expand the capabilities of the language: Symfony, CakePHP and Zend Framework. The development pace was fast and stable. Object-oriented source code that was written in PHP5 was elegant and maintainable. More and more new projects began to use PHP frameworks and successful completion of these projects made PHP frameworks even more popular. Popular programming languages for website development are PHP, Ruby, Python and Java. Selecting from these languages PHP was chosen for framework analysis and web site development, because it is an open source scripting language with many advantages: support for different database types, many language structures were taken from C and Perl languages. Since PHP is a popular and widely used language, it has a lot of documentation on the Web. Also as shown in Fig. 1 PHP is used by 81.7% of all websites whose server-side programming languages are known[4].

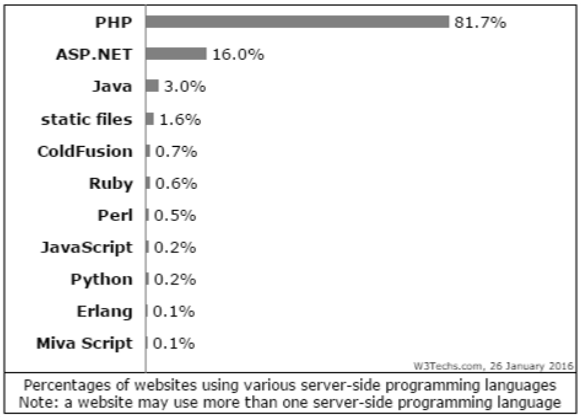

Fig. 1. Usage of server-side programming languages for websites.

To sum up, all of the information mentioned above confirms that framework usage for web site development is quite a hot topic. The paper is structured as follows. It starts with a comparison of PHP frameworks and a brief overview of Model-view-controller (MVC) architecture style. Then a description of the architecture and main features of selected frameworks (routing, template engine, etc.) are offered. Also a performance test is created and performed in order to examine framework demands to resources and ability to finish certain tasks within a reasonable time frame. We conclude the paper with a summary where an attempt to make advices is made.

2. Framework comparison

Several PHP programming environment frameworks were inspected (see Table 1).

Selected frameworks were compared using following criteria: documentation availability, framework usage requirements, tools which accelerate the development progress (ORM, code generation, pattern system, CRUD, etc.).

Table 1. Framework comparison[5].

	CakePHP2	CodeIgniter	Symfony2	Yii	Phalcon
Newest version	3.3.6	3.1.0	3.1.4	2.0.9	3.0.0
Release date	2016 Oct 14	2016 Jul 26	2016 Sep 3	2016 Jul 11	2016 Jul 29
License	MIT	BSD-style	MIT	BSD	New BSD License
Requirements	>= PHP 5.2.8	>= PHP 5.1.6	>= PHP 5.5.9	>= PHP 5.4	>= PHP 5
ORM	Custom	ActiveRecord	Doctrine 2, Propel	Database Access Objects (DAO), Active Record	Phalcon
Code generation	CLI	Combustor	CLI	Yii CLI, Gii (Web based)	Phalcon Developer Tools
Template engine	Custom, bet Smarty/Twig	PHP, Simple template parser "{var_name}"	PHP, Twig	Razor, Smarty, Twig etc.	Volt, PHP
CRUD generation	Scaffolding	Grocery CRUD	SensioGenerator Bundle	Gii	Scaffolding
Web-site	cakephp.org	codeigniter.com	symfony.com	yiiframework.com	phalconphp.com

The Table 1 results show that all described frameworks are recently updated and require at least version 5 of PHP language. They all provide different ORM implementations however template engine Twig is popular amongst other frameworks, not only Symfony2. All described frameworks except Symfony2 are based on the MVC architecture style[6] MVC concept was described by Norwegian computer scientist Trygve Reenskaug in 1979. In the article "Applications Programming in Smalltalk-80: How to use Model-View-Controller» two varieties of MVC were described: passive and active model[7]. The active model is considered a classic MVC concept realization. The Model notifies Views about its changes, but the Views are interested in obtaining notifications about changes in the Model by subscribing to them. This allows maintaining Model independence from the Controller and the View.

Two frameworks were selected for deeper analysis - Symfony2 and PhalconPHP. According to sitepoint.com these two frameworks are two of the most promising frameworks in 2014[8].

2.1. PhalconPHP

Phalcon framework was chosen because it is an open source PHP framework. The fact that Phalcon is written as a PHP extension using C programming language distinguishes it from other frameworks. It is based on the MVC principles and is developed by Phalcon Team. Phalcon is distributed under the BSD license taking the “New BSD License” into account. Phalcon framework has the following characteristics: all components are written in C programming language; there are different versions for popular operating systems: Linux, Windows and Mac; high performance and low server resource expenses. According to the tests Phalcon is one of the fastest PHP frameworks; interaction with the databases is implemented in C language using ORM technologies.

When looking at routing capabilities it is clear that Phalcon has two routing modes: MVC mode and a “match only” mode. Phalcon built-in router in “match only” mode automatically tries to find the controller and its method based on the input URL. MVC mode allows the programmer to manually configure routes and redirect requests to the appropriate controller and method. The router can be configured with PHP code or using annotations (metadata that is attached to the source and which can be read during program execution) which are written directly in the controller. Using regular expressions it is possible to define laws which will be used to validate input parameters. Regular expressions are a widely used pattern describing method that allows searching for text and check its compliance with the pattern. Phalcon framework has its own template engine "Volt". It is written using C language and compiled together with Phalcon as a PHP extension. Volt transforms all templates to PHP code so it is possible to use raw PHP code inside a template. To reuse already written code Volt offers a variety of mechanisms. The code can be split using the {%block%} operator. This functionality can be used, for example, to create a single master file and then simply insert other templates. When extending a template it is necessary to define blocks that will be replaced. Templates that extend another template can contain only blocks otherwise an error will occur[9].

2.2. Symfony2

The reasons for choosing Symfony2 framework are based on the fact that it offers a fast web application development and management. This easily enables Web programmers to perform daily tasks. Symfony2 has support for multiple databases (MySQL, PostgreSQL, SQLite, etc.). Information about the relational database in the project must be linked to the object model. This can be accomplished by using an ORM tool. By default Symfony2 is equipped with Doctrine2 ORM. Symfony2 router has a lot of options to configure routes - annotations, YAML, XML and PHP. Annotations and YAML (data format, which focuses on working with complex data structures) are the most popular configuration file formats used with Symfony2. Twig is a Symfony2 template engine with open source code which is written in PHP programming language. Twig and Volt syntax comes from Jinja and Django template engines so they are very similar. Block including and extending is similar to the Volt engine but with one difference. Twig unlike the Volt supports multiple block levels. This allows creating a single master template with included CSS and JavaScript files and using it repeatedly[10].

To compare Symfony2 and Phalcon frameworks a performance test is offered in the next section.

3. Performance test

The authors have developed a performance test. The goal of this test is to determine performance and efficiency of both frameworks while performing the same task. For performance testing the “Theatre ticket reservation” systems cashier list output section was developed and chosen. Output data and template structure in both versions of the website are almost the identical. Tests were executed using ab.exe (ab - Apache Benchmark) tool that comes with the Apache Web server. Test method: both frameworks were loaded with 2,000 requests with 10 simultaneous connections. The number of loaded files was found using the PHP function get_included_files[11]. Memory consumption was found by using the PHP function memory_get_usage[12].

The Symfony2 framework handled 49.33 requests per second but Phalcon framework handled 231.99 requests which is almost 6 times better than Symfony2 as shown in Fig. 2.

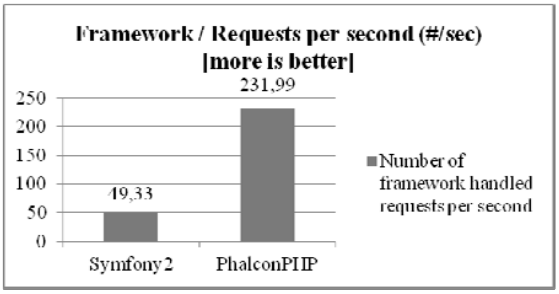

Fig. 2. Number of framework handled requests per second.

Symfony2 framework request time is 20,273 ms per request but Phalcon framework request time is 3.106 ms per request which is more than 6 times better than Symfony2 as shown in Fig. 3.

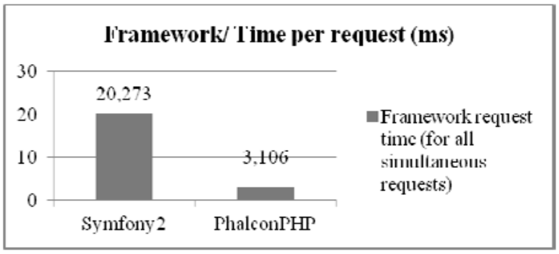

Fig. 3. Framework request time.

The Symfony2 framework loads 301 file for each request while Phalcon framework needed only 10 files which is roughly 30 times better than Symfony2 as shown in Fig. 4.

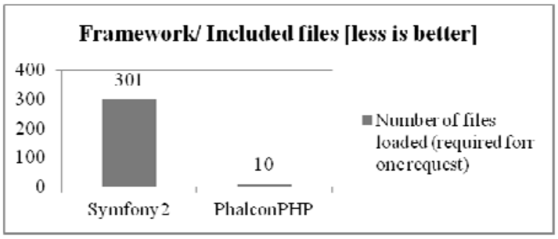

Fig. 4. Number of files loaded.

Symfony2 framework consumes 15.86 MB per request while Phalcon needs only 0.53 MB per each request which is almost 30 times better than Symfony2 as shown in Fig. 5.

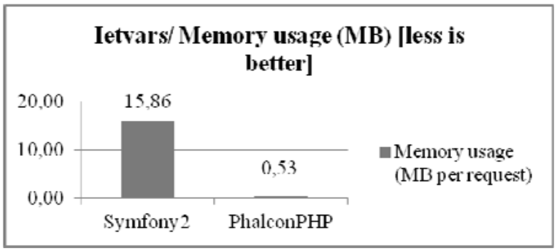

Fig. 5. Memory consumption of framework.

Phalcon unique structure provides excellent performance and exceeds Symfony2 results in this test context. Thus the two frameworks were compared and based on the received results conclusions were made which are shown in the next section.

4. Conclusion

Phalcon and Symfony2 have very similar routing capabilities - annotation and regular expression usage, but Symfony2 provides a greater choice of configuration file formats - XML and YAML. Phalcon „match only" routing mode is better suited for small or medium-sized projects because it allows developers to pay more attention to the implementation of functional components for the project instead of spending time planning routes. Symfony2 template engine „Twig" allows for better template structure organization as it supports nested template blocks. Very large choice of filters also helps to maintain the code clean and maintainable.

Phalcon template engine „Volt” is friendlier to programmers who are just starting to work with template engines because it allows the use of PHP code blocks in case it is hard to implement the desired functionality using the offered tools. Both engines cache the generated template code after the first run though Volt does it much quicker since it is written in C language. Despite the fact that both engines generate template cache to accelerate further loading Twig templates will load slower because its cache is divided into too many small files. After the tests, it is clear that Phalcon provides outstanding performance and exceeds Symfony2 results. This article may be useful for choosing a framework for Web information system.

To continue the research, it is necessary to expand the developed Web system capabilities using Web services that are based on open standards and protocols (SOAP, XML-RPC, REST, etc.). Web service is a way of how applications can interact with each other in the Web. Applications can use different platforms and be written in different programming languages. Thanks to the fact that Web services are developed to ensure machine collaboration in the Web (Machine-to-Machine approach) each service interface can be described using a machine-readable format.

References

1. Samra J. Comparing Performance of Plain PHP and Four of Its Popular Frameworks. Thesis Project. Linnaeus University, Sweden. Available: https://www.diva-portal.org/smash/get/diva2:846121/FULLTEXT01.pdf.
2. Nylén. H. PHP Framework Performance for Web Development Between Codeigniter and CakePHP. Bachelor Thesis. School of Computing Blekinge Institute of Technology, Sweden. Available: http://www.diva-portal.org/smash/get/diva2:831188/FULLTEXT01.pdf.
3. Siddharth S. 15 Most Important Considerations when Choosing a Web Development Framework; 2009. Available: http://code.tutsplus.com/tutorials/15-most-important-considerations-when-choosing-a-web-development-framework-net-8035.
4. W3Techs. Usage of server-side programming languages for websites. Available: http://w3techs.com/technologies/overview/programming_language/all.
5. SocialCompare. PHP frameworks comparison; 2016. Available: http://socialcompare.com/en/comparison/php-frameworks-comparison.
6. Basher K. MVC Patterns (Active and Passive Model) and its implementation using ASP.NET Web forms. CodeProject; 2013. Available: http://www.codeproject.com/Articles/674959/MVC-Patterns-Active-and-Passive-Model-and-its.
7. Burbeck S. Applications Programming in Smalltalk-80(TM): How to use Model-View-Controller (MVC). Available: http://www.dgp.toronto.edu/~dwigdor/teaching/csc2524/2012_F/papers/mvc.pdf.
8. Skvorc B. Best PHP Frameworks for 2014. Available: https://www.sitepoint.com/best-php-frameworks-2014/.
9. Phalcon Team. Phalcon 3.0.1 Documentation. (in Russian); 2016. p. 272-333. Available: https://media.readthedocs.org/pdf/phalcon-docs-ru/latest/phalcon-docs-ru.pdf.
10. SensioLabs. The Symfony Book v. 3.1; 2016. p. 4-68. Available: http://symfony.com/pdf/Symfony_book_3.1.pdf.
11. PHP Documentation. PHP’s memory_get_included_files. Available: http://php.net/manual/en/function.get-included-files.php.
12. PHP Documentation. PHP’s memory_get_usage function. Available: http://php.net/manual/en/function.memory-get-usage.php.

Natalya Prokofyeva, Dr. sc. ing. (2007). Position: Riga Technical University, Department of Computer Science and Information Technology, Chair of Software Engineering, an Associate Professor. The main fields of research activity: E-learning systems (model, methods, technologies), Modern Internet Technologies. Scientific activity: 2014 Joint International Conference on Engineering Education & International Conference on Information Technology (ICEE/ICIT-2014 Riga). Contact her at natalija.prokofjeva@rtu.lv.

Victoria Boltunova, Bachelor of engineering science in computer control and computer science (2015). Education: 2015 – Riga Technical University, Faculty of Computer Science and Information Technology, Institute of Applied Computer Systems, Master Student. Scientific activity: 4th International Workshop on Intelligent Educational Systems, Technology-enhanced Learning and Technology Transfer Models (INTEL-EDU 2016). Contact her at lboltunova@apollo.lv.