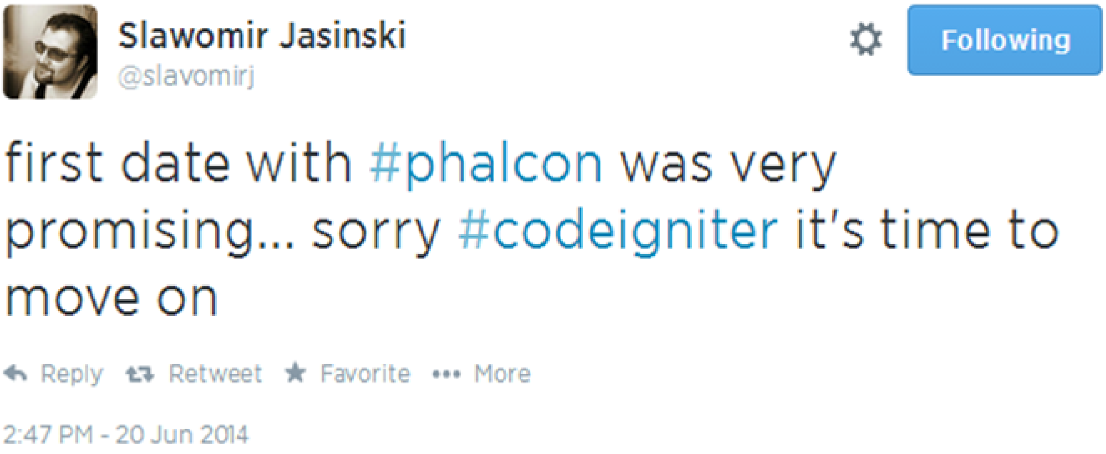
WHY PHALCON IS THE FRAMEWORK TO DIE FOR



Slawomir Jasinski
@slavomirj

first date with #phalcon was very promising... sorry #codeigniter it's time to move on

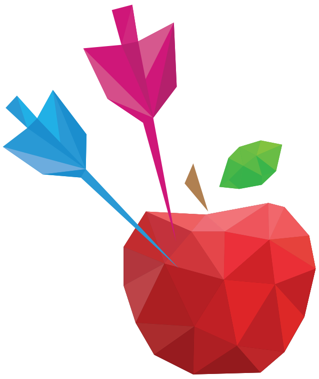
2:47 PM - 20 Jun 2014

Introduction

A PHP framework is in essence a set of classes and functions that a developer adheres to when developing a site. It offers a foundation or structure for developers to arrange their code. It also facilitates online sharing of documentation, tips and advice, solutions to problems, and more, in true open source fashion. The absence of a framework would result in different coders employing their own different techniques and methodologies in arranging code and designing architecture, creating all-round complexities that would subvert the open-source environment.

There are over 20 different open source PHP frameworks, ranging from full-stack frameworks to micro frameworks. And each framework claims uniqueness based on their niche USP, which could be speed, documentation, beautiful syntax, or anything else.

Phalcon is the new kid on the block, but it has soared in popularity ever since its launch on 14 November 2012, and now ranks among the most popular frameworks for PHP5. Written and implemented as a C language extension, it offers the full development stack from user interface to data store.



Performance

While PHP ranks as the most popular open-source web programming language, it still faces a big drawback in performance. The program has to read all files from the hard disk, convert them into bytecode, and execute, every single time the user makes a request. This causes PHP to lag in performance, compared to alternative languages such as Ruby on Rails and Python (Django, Flask). Using the Phalcon framework mitigates this drawback.

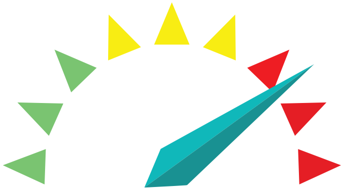
With Phalcon, the whole framework is already in RAM, and as such, it is not necessary to process the entire set of framework files. This boosts performance by no small means.

Phalcon is distributed as a shared library, as a DLL (Dynamic-Link Library) in Windows, or as a shared object in most Unix flavors. This DLL/Shared Object structure comprises of a Variable Data Section, a Read Only Data Section, and a Code Section. The Code Section and the Read Only Data Section are shared across requests/processes, and the compilation produces a very small binary of about 1.5 MB. Thus, if the server runs 1,000 requests, the same 1.5 MB is shared by all the requests, thereby reducing the memory usage dramatically.

Memory consumption is reduced further by multiple processes loading the same DLL/ Shared Object library at the same or different base address, while still sharing the same physical copy of the DLL/ Shared Object in memory. This also facilitates disc swapping.

All requests share every constant string and number in the extension as well, further saving huge amounts of memory, while also allowing extensions to grow much larger and more robust without impacting performance or load time.

Turnaround Time



Phalcon takes the least amount of time to handle requests, compared to other PHP frameworks. While Phalcon serves 822.96 requests per second (req/sec), the next fastest framework is Slim, which serves just 399.83 req/sec. To put things in perspective, the same statistic for other frameworks is, Kohana - 217.34, CodeIgniter - 187.78, Silex - 179.01, Laravel - 135.9, YY - 123.5, Feul PHP - 116.34, Hazaar MVC - 103.53, Zend 1 - 103.02, Cake PHP - 54.97, Nette - 53.48, Symfony2 - 39.22, and Zend2 - 36.1.

Phalcon's extraordinary speed is due to the optimization of instructions, owing to the compilation process.

The operating system vendors have optimized and matured the DLL/Shared Objects structure that Phalcon uses, over the last decades. When a function is called from the source code of the web application, it already comes interpreted in the binary language, and is executed without wasting time or resources for interpretation, compilation, or any further preparations.

Phalcon's bootstrap file is called on every request. Requests are sent to the bootstrap through directives stored in a .htaccess file. It is possible to modify the directory layout to access everything which exists as public/index.php, via the bootstrap file.

Responses in Phalcon usually generate in about 10-35ms (milliseconds), even with high server loads. Pages are generated and delivered to users faster even with network latency in the region of 200-225 ms. Additional planning and tolls such as CDN could reduce network latency significantly.

Usability

Phalcon offers all the features inherent in any state-of-the-art PHP framework, including routing, controllers, view template, Object-Relational Mapping (ORM), and caching. The easy to use ORM technique facilitates data conversion between incompatible type systems. While there is not much differentiation compared to other frameworks on this front, Phalcon still offers some distinct superiority.

Phalcon offers the Model–view–controller (MVC) architecture for implementing user interfaces. The MVC architecture divides the software application into three interconnected portions, separating the way information is represented internally, and how the same info is shown to or received from the user. The central component, called the model, features the application data, as well as the business logic, rules, and functions. This not only makes possible a clean and user friendly interface, it also makes possible multiple views of the same information, such as, say, a bar chart for management and a tabular view for accountants, using the same data.

Phalcon's own query language - Phalcon Query Language (PHQL) makes database interaction more expressive and clean. PHQL can be integrated with models to easily define and use relationships between tables. This is especially useful to handle complicated queries with ease.

Extending the Phalcon MVC Model class offers immediate access to common handy methods such as find(), save(), and validate(), very easily.

Phalcon offers a well-rounded package, including ORM, template engine, a request object library, PHQL, caching, pagination, and much more, all in a single place, without having to rely on third party add-ons. The advantage is that everything stays in-memory, C-based and super fast.

Deployment

Users download and install Phalcon as a PHP module. This is when most other frameworks have to be downloaded and extracted to a directory. Phalcon being installed as an extension offers many advantages:

- The controllers and models come autoloaded, making it possible to simply create files and use them from anywhere in the project.
- It weeds out the hobbyists from the serious developers, providing integrity to the framework.
- The install process takes just a few minutes. The documentation offers installation instructions.

Phalcon as a framework is coupled loosely. This allows developers to utilize the complete framework, or just a few desired objects as the glue components.

Learning Curve

Phalcon is open-source, making it possible to modify the code and recompile it as and when required. Phalcon’s learning curve is very easy.

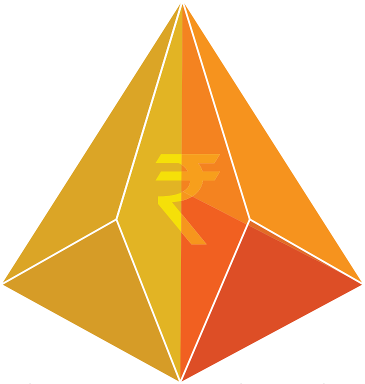
The API is intuitive, neat and clean. The robust code is written on potent design patterns, making navigating through the framework a breeze. Native namespaces help to distinguish from similar class names or even controllers, and also develop modules or bundles easily.

Phalcon is different from other full-stack frameworks in that it comes as a PHP module written in C. While debugging the code will require knowledge in C, using C is overall an advantage for Phalcon, as C-compilers are very mature. It reveals uninitialized variables, wrong data types, and other bugs that generate errors. It also detects unnecessary code, thereby optimizing the code.

But even knowledge of C may not become essential going forward.

Phalcon 2, a full rewrite of the original Phalcon Framework, offers a new Zephir language. Zephir is a mixed-type derivative of PHP supporting both static and dynamic types, and allows developers to write “almost-PHP” code and compile it down to C based extensions for PHP.

Phalcon 2 is built almost entirely in Zephir, and this makes it easier than ever to maintain, update and accept contributions from the community.



Support & Costs

Like all open source programs, the robustness of Phalcon's support depends on its community. Phalcon has a tight-knit and hardcore community, filled with enthusiasts genuinely interested in it, enough to not get bogged down by the basics.

Phalcon scores over other PHP frameworks on the cost front as well. Its blazing-fast framework pre-loaded in memory, not only improves performance, but also reduces cloud bills substantially.

Moreover, the framework costs next-to-nothing in overheads.

> “Phalcon is the ultimate framework. Its compiled nature delivers unparalleled performance, outdoing other frameworks like **Slim, Silex, Yii, Symfony, Zend, Framework, Kohana, FuelPHP, Laravel and CodeIgniter,** measured for benchmarks like requests per second, time across all concurrent requests, number of included PHP files on a single request, and memory usage per request.”

Suyati
dream. dare. do

Suyati Technologies

At the core of Suyati's corporate culture, is the spirit of collective good, much like the Open Source movement itself. Our **expertise in FOSS technologies** covers application development, web application development, and mobile app development.

To know more about how we can help you, get in touch with us:
services@suyati.com

www.suyati.com

Reference:
1. http://www.sitepoint.com/phalconphp-yet-another-php-framework/
2. http://www.sitepoint.com/best-php-frameworks-2014/
3. http://systemsarchitect.net/performance-benchmark-of-popular-php-frameworks/
4. http://www.sitepoint.com/phalcon-2-0-alpha-landing/
5. http://www.tisindia.com/blog/7-best-php-frameworks-2014/
6. http://phalconphp.com/en/
7. http://www.developed.be/2013/07/16/php-frameworks-which-to-choose/
8. http://www.creativebloq.com/design/choose-right-php-framework-12122774
9. http://systemsarchitect.net/performance-benchmark-of-popular-php-frameworks/