

Anatomy of a Modern PHP Application

Overview

- PHP: Why so bad?
- Ye Olden Days
- PHP 5 & PHP 7
- Community Influences
- Modern Frameworks
- Tooling

PHP: Why so bad?

PHP: Why so bad?

- Interpreted: Easy to Host
- Low barrier to entry
- Easy to do badly
- Can be dangerous
- Lots of bad quality tutorials online

Ye Olden Days

Ye Olden Days: Procedural PHP, Monoliths & FTP

- 5000 line functions.php
- One file per page
- Mixed code and HTML
- Global scope pollution

PHP 5: A Turning Point

PHP 5: A Turning Point

- 5.0: Proper Objects! (13 July 2004)
- 5.1: PDO Database Access (24 November 2005)
- 5.2: Native JSON Support (2 November 2006)
- 5.3: Namespaces, Closures, Late Static Binding (30 June 2009)

PHP 5: A Turning Point (part 2)

- 5.4: Traits, Short Array Syntax (1 March 2012)
- 5.5: Generators, Password Hashing API (20 June 2013)
- 5.6: Variadics, Argument Unpacking (28 August 2014)

PHP 7

PHP 7: Getting Somewhere!

- Released 3 December 2015
- +100% performance increase on 5.6
- Scalar Type Hinting
- Return Type Declarations
- Spaceships!

PHP 7: Type Hinting & Return Declarations

```
declare(strict_types = 1);

function square(int $value) : int
{
    return $value * $value;
}

$a = square(5); // int 25
$b = square("5"); // Fatal error: Uncaught TypeError
```

PHP 7: Spaceship Operator

`$a <=> $b`

- **-1** if `$a` is *less than* `$b`
- **0** if `$a` is *equal to* `$b`
- **1** if `$a` is *more than* `$b`

In PHP 5 you could use:

`(($a < $b) ? -1 : ((($a > $b) ? 1 : 0))`

PHP 7: Null Coalesce Operator

PHP 5:

```
$a = isset($b) ? $b : "default";
```

PHP 7:

```
$a = $b ?? "default";
```

Community Influences

PHP Framework Interop Group (PHP-FIG)

- Govern the PHP Standards Recommendations (PSRs)
- 6 Accepted (Active) PSRs
- 11 Draft PSRs
- 1 Deprecated PSR

Active PSRs

- PSR-1: Basic Coding Standard
- PSR-2: Coding Style Guide
- PSR-3: Logger Interface
- PSR-4: Autoloading Standard
- PSR-6: Caching Interface
- PSR-7: HTTP Message Interface

Composer & Packagist

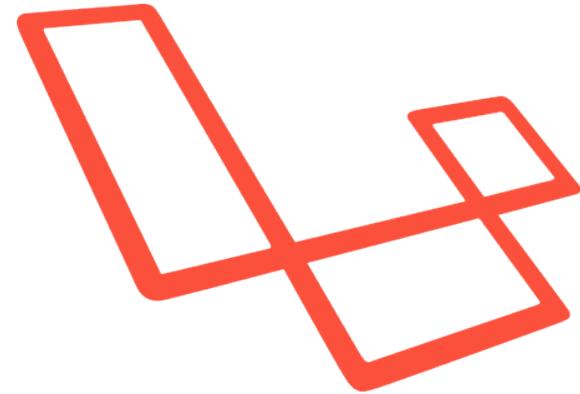


- Composer¹: Dependency Manager for PHP
- Packagist²: PHP Package Repository

¹ <https://getcomposer.org/>

² <https://packagist.org/>

Modern PHP Frameworks



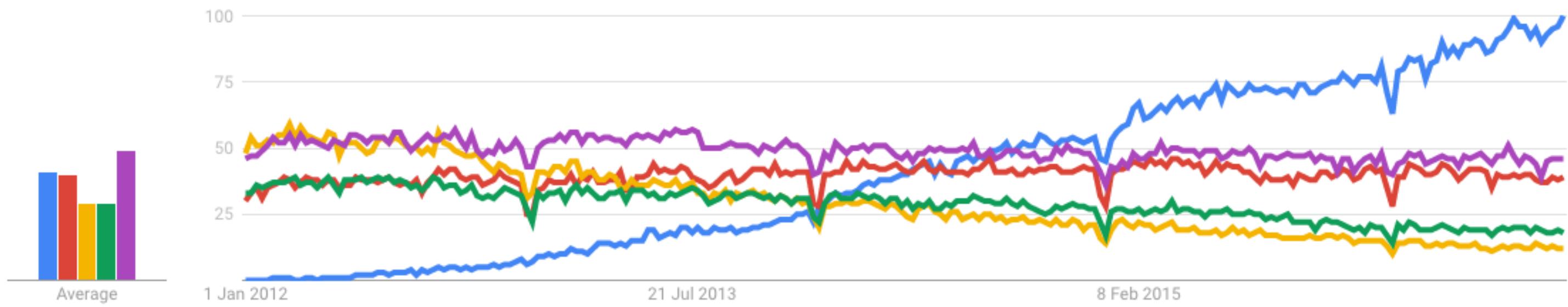
Heavy Hitter Frameworks

- Most features available out of the box
- Good for building full applications

Examples:

- Laravel
- Symfony
- Zend Framework

Framework Popularity (2012 - present)



Laravel - Symfony - Zend Framework - CakePHP - CodeIgniter

Micro Frameworks

- Slimline & Lightweight
- Optimised for API Builds

Examples:

- Lumen (Laravel)
- Silex (Symfony)
- Slim PHP

Tooling



Version Control

- Git (GitHub)
- Github Flow
 - Branch per Feature
 - Pull Requests

Github Flow

- Production always runs master
- New feature, new branch
- Pull Request
- CI & Code Review
- Deploy the branch
- Merge to master

Continuous Integration

- SemaphoreCI
- Automatically runs based on Git hooks
 - Automated Test Suites
 - Code Linting
 - Code Quality Tools

Continuous Delivery

(Automated Deployments)

Envoy SSH Task Runner

- git clone into new directory
- composer install to build
- Copy Config Files
- Create Symlinks
- Web Server Reload

Questions?

Twitter: @johnblackmore

Web: johnblackmore.com

Email: john@johnblackmore.com