Seven PHP frameworks, four measurements

Cold request, load test, OPcache preload and the Laravel preload curve: the measured cost of seven PHP frameworks on a single page.

4.4×

01 EMPTY ROUTE, REQ/S

gap between the fastest and the slowest

Slim 25,975, Laravel 5,966 req/s. Once the database joins in, the gap shrinks to 3.7×.

02 FOOTPRINT

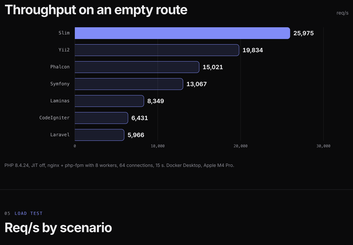
Files loaded per request



Source: PHP framework footprint measurement

03 FOOTPRINT

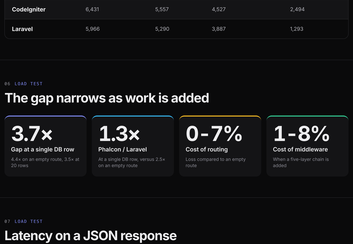
Cold first request



Source: PHP framework footprint measurement

04 LOAD TEST

Throughput on an empty route



PHP 8.4.24, JIT off, nginx + php-fpm with 8 workers, 64 connections, 15 s. Docker Desktop, Apple M4 Pro.

05 LOAD TEST

Req/s by scenario

FRAMEWORK	EMPTY ROUTE	TEMPLATE	DB SINGLE ROW	DB 20 ROWS
Slim	25,975	22,780	14,443	4,551
Yii2	19,834	16,995	9,366	3,245
Phalcon	15,021	10,563	5,222	2,040
Symfony	13,067	9,355	6,972	3,060
Laminas	8,349	7,419	5,421	2,874
CodeIgniter	6,431	5,557	4,527	2,494
Laravel	5,966	5,290	3,887	1,293

06 LOAD TEST

The gap narrows as work is added

3.7×
Gap at a single DB row
4.4× on an empty route, 3.5× at 20 rows

1.3×
Phalcon / Laravel
At a single DB row, versus 2.5× on an empty route

0–7%
Cost of routing
Loss compared to an empty route

1–8%
Cost of middleware
When a five-layer chain is added

07 LOAD TEST

Latency on a JSON response

FRAMEWORK	P50	P95	P99
Slim	0.25 ms	0.43 ms	0.53 ms
Yii2	0.35 ms	0.59 ms	0.73 ms
Phalcon	0.46 ms	0.73 ms	0.94 ms
Symfony	0.58 ms	0.89 ms	0.96 ms
Laminas	0.88 ms	1.52 ms	2.23 ms
Codeigniter	1.16 ms	1.84 ms	1.98 ms
Laravel	0.75 ms	1.97 ms	2.59 ms

−63%

08 LARAVEL

throughput taken by the default web group

Choosing a framework can open a gap of up to 4×, but a single default middleware group can take away 63%.

09 LARAVEL

Web group on and off

ROUTE	REQ/S	P50	P99
No group	5,858	1.32 ms	2.88 ms
Default web group	2,176	1.73 ms	71.89 ms

10 OPCACHE PRELOAD

Cold request: before and after preload

FRAMEWORK	NO PRELOAD	BEST PRELOAD	GAIN	FPM COST
Laravel	76.31 ms	13.98 ms	5.5×	+2,340 ms
Symfony	35.58 ms	2.50 ms	14.2×	+40 ms
Laminas	26.70 ms	3.81 ms	7.0×	-7 ms
Codeigniter	20.75 ms	3.13 ms	6.6×	-12 ms
Yii2	16.98 ms	4.80 ms	3.5×	+13 ms
Slim	7.46 ms	1.86 ms	4.0×	+28 ms
Phalcon	2.68 ms	1.99 ms	1.3×	+43 ms

11 OPCACHE PRELOAD

Preload is not a setting, it is a maintenance job

2/7
Official preload file
Only in Symfony and Codeigniter

14.2×
Symfony's gain
From 35.58 ms to 2.50 ms

2.3 s
Laravel's deploy cost
In return for a 62 ms gain per visitor

1.3×
Phalcon
Negative control: 3 files

12 LARAVEL PRELOAD CURVE

Cold request as more files are preloaded



Median values; the filters are specific to this application.

13 LARAVEL PRELOAD CURVE

The sweet spot is at 2,809 files

12.77 ms
Cold request at the sweet spot
−81% compared to 68.37 ms without preload

+1,132 ms
fpm readiness cost
From 382 ms to 1,514 ms

15.7 ms
Gain from the critical 123 files
Eight times what the 1,912 files deliver

1.8 ms
Gain from the last 1,912 files
654 ms of cost per millisecond gained

14 AS YOU READ

The limits of these numbers

ENVIRONMENT The load test ran in a Docker Desktop VM; absolute values do not represent bare metal, and the fast candidates are, if anything, under-reported.

SCOPE JIT was off, worker mode was not measured. Slim has no data layer or template engine of its own.

COLD Cold means an empty OPcache. A single shot strays from the median: Symfony takes 76.5 ms in a single shot, against a median of 35.58 ms.

CONSISTENCY Zero non-200 responses across 546 load measurements; in 150 samples the worst value is 84% of the best.