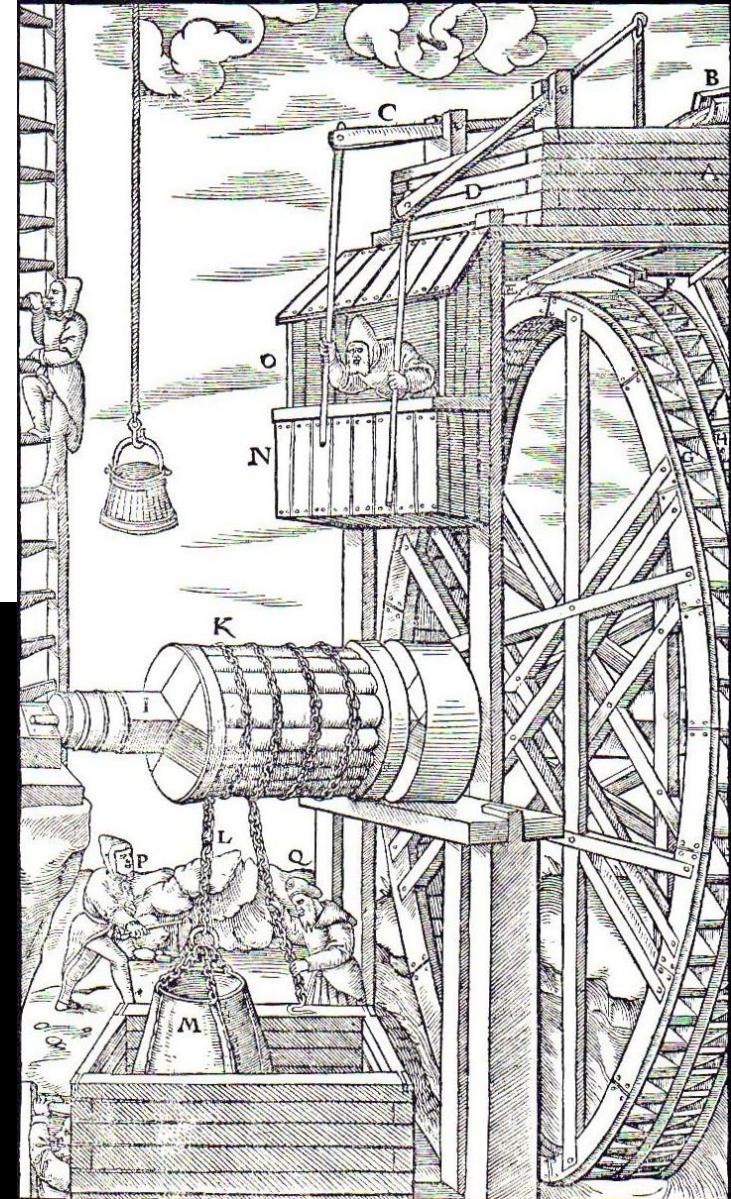


Zarządzanie procesami PHP

mod_php, php-fpm, php-pm



Kehrrad als Teil einer Trommelfördermaschine
in De re metallica (Georgius Agricola, 1566)

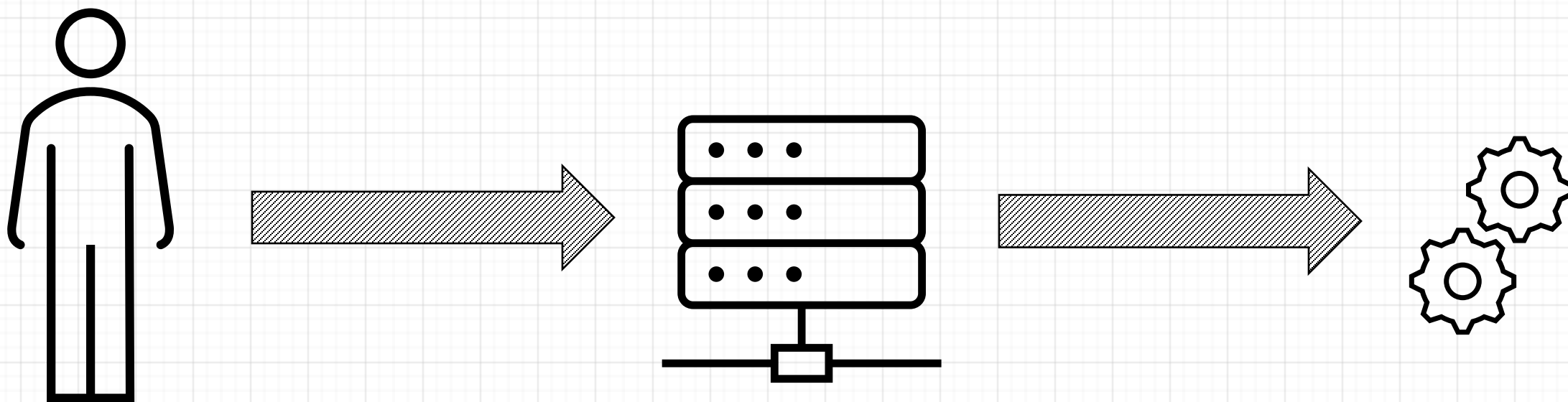
Agenda

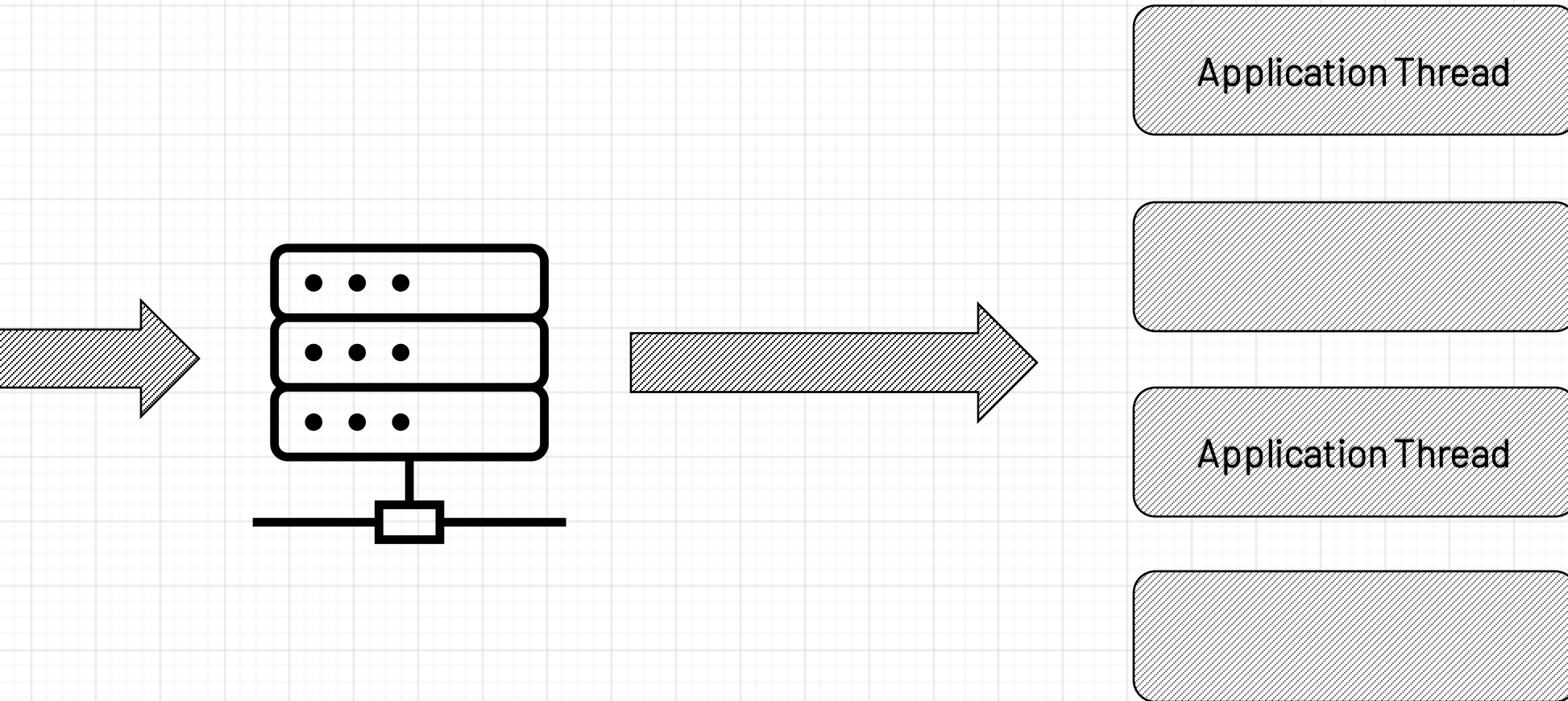
- 1/ Jak aplikacje procesują requesty
- 2/ Wydajność rozwiązań
- 3/ Jak to jest robione w innych językach?
- 4/ Jak procesować ruch jeszcze szybciej?
- 5/ Podsumowanie



Jak aplikacje procesują requesty

01/





> mod_php vs php-fpm

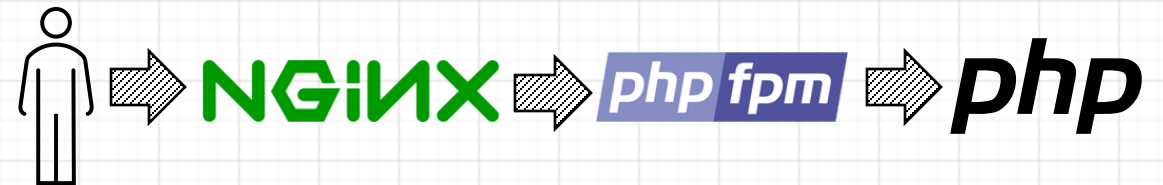
mod_php

- Moduł do serwera WWW Apache2



php-fpm

- Dodatkowa aplikacja którą można podłączyć do Apache2, nginx lub innego serwera WWW



> mod_php vs php-fpm

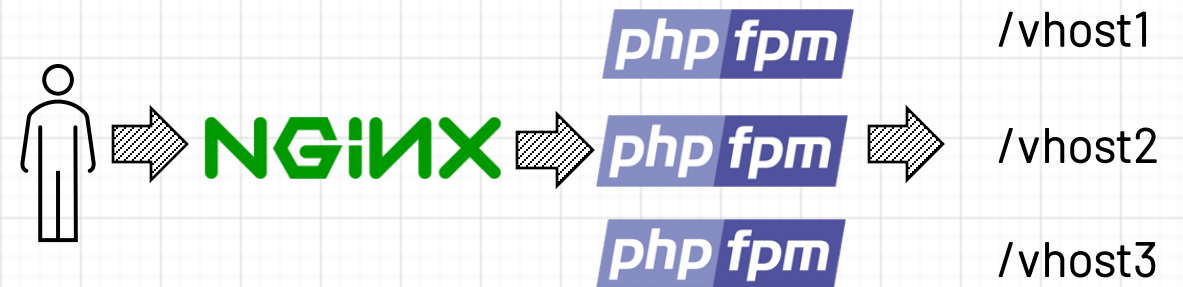
mod_php

- Popularny do prostych i dedykowanych zastosowań



php-fpm

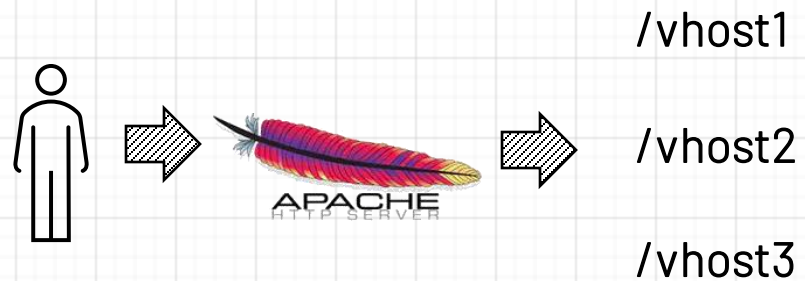
- Popularny na hostingach współdzielonych



> mod_php vs php-fpm

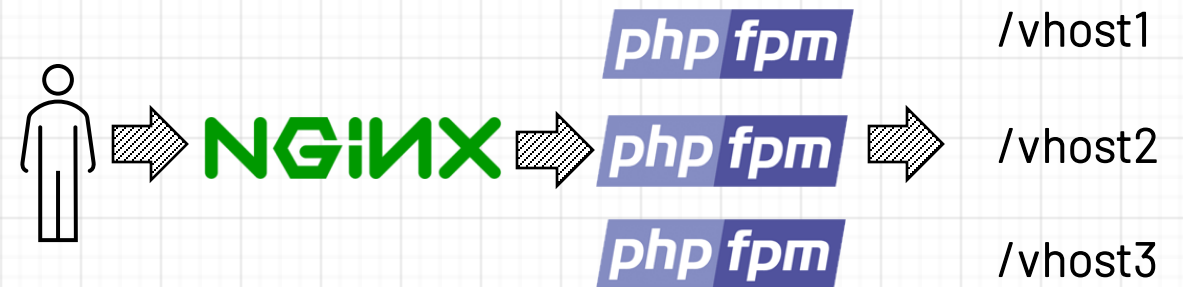
mod_php

- Brak domyślnej separacji vhostów



php-fpm

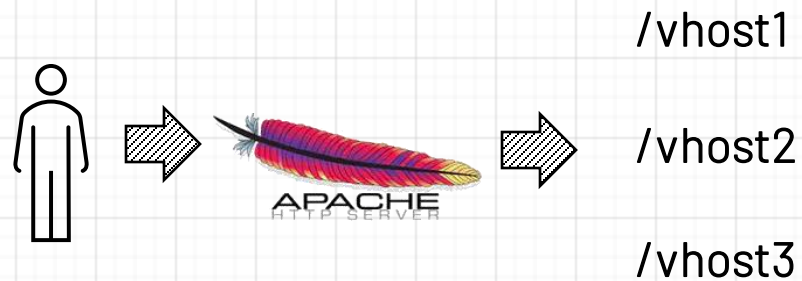
- Wbudowana obsługa pooli zapewniających separację



> mod_php vs php-fpm

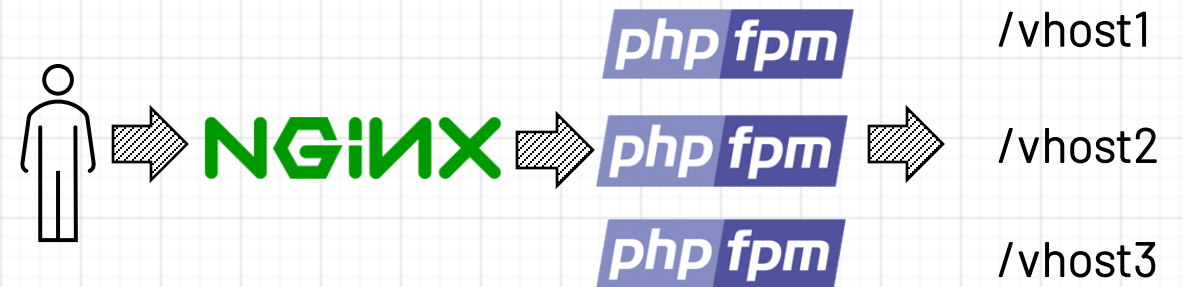
mod_php

- Brak domyślnej separacji vhostów
- Można skorzystać z modułu mpm_itk, ale nie działa on idealnie



php-fpm

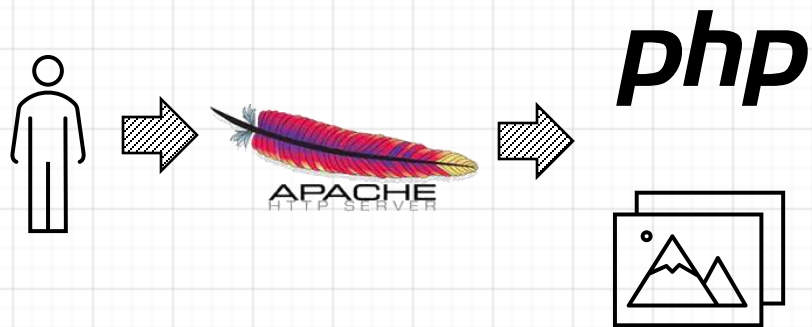
- Wbudowana obsługa pooli zapewniających separację



> mod_php vs php-fpm

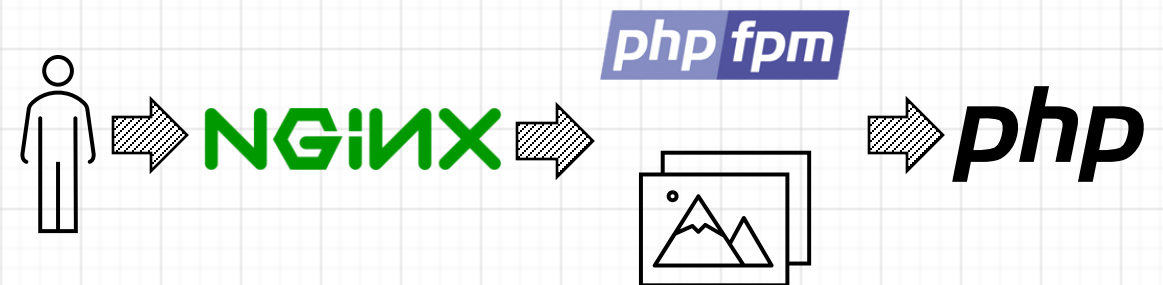
mod_php

- Aktywowany dla każdego requestu



php-fpm

- Może obsługiwać tylko requesty do PHP



> mod_php vs php-fpm

mod_php

- Moduł do serwera WWW Apache2
- Popularny do prostych i dedykowanych zastosowań
- Brak domyślnej separacji vhostów
- Aktywowany dla każdego requestu

php-fpm

- Dodatkowa aplikacja którą można podłączyć do Apache2, nginx lub innego serwera WWW
- Popularny na hostingach współdzielonych
- Wbudowana obsługa pooli zapewniających separację
- Może obsługiwać tylko requesty do PHP

> mod_php vs php-fpm

mod_php

- Moduł do serwera WWW Apache2
- Popularny do prostych i dedykowanych zastosowań
- Brak domyślnej separacji vhostów
- Aktywowany dla każdego requestu

php-fpm

- Dodatkowa aplikacja którą można podłączyć do Apache2, nginx lub innego serwera WWW
- Popularny na hostingach współdzielonych
- Wbudowana obsługa pooli zapewniających separację
- Może obsługiwać tylko requesty do PHP

Które rozwiązanie jest lepsze?



Wydajność rozwiązań

02/

> Benchmark: środowisko testowe

- Dwa środowiska zamknięte w docker-compose
 - apache2 + mod_php
 - nginx + php-fpm
- Aplikacja testowa z phpbenchmarks.com
- Całość w repozytorium git z wersjonowanymi zmianami pod finetuning, podsumowaniem każdego kroku
- <https://github.com/lpiob/php-serving-benchmarks>

> Benchmark: środowisko testowe

- Dwa środowiska zamknięte w docker-compose
 - apache2 + mod_php
 - nginx + php-fpm
- Aplikacja testowa z phpbenchmarks.com
- Całość w repozytorium git z wersjonowanymi zmianami pod finetuning, podsumowaniem każdego kroku
- <https://github.com/lpiob/php-serving-benchmarks>

> Benchmark: przebieg i wyniki pośrednich testów

- Omówienie na podstawie <https://github.com/lpiob/php-serving-benchmarks/tree/master/test-results>
- Pliki 001-007

Key takeaways:

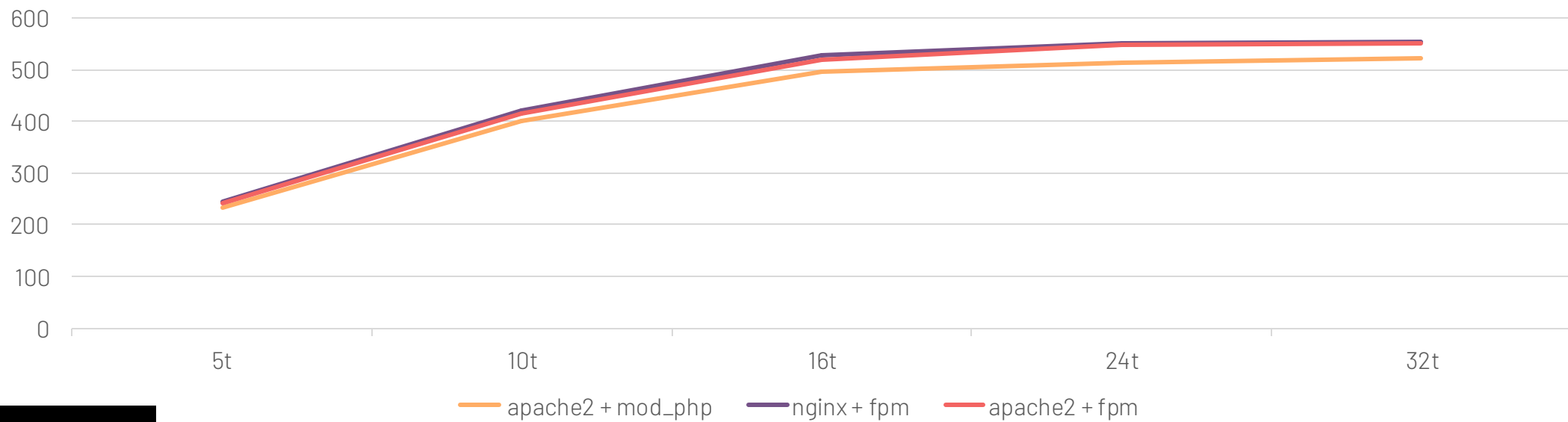
- Zmiana połączenia TCP na socket nie miała zmierzalnego wpływu na wydajność
- Największy zysk wydajnościowy po zmianie domyślnego process managera w FPM.

> Benchmark: wyniki

Sposób serwowania	Requests per second
apache2 + mod_php	400.10
apache2 + php-fpm	413.06
nginx + php-fpm	420.84

> Benchmark: wyniki

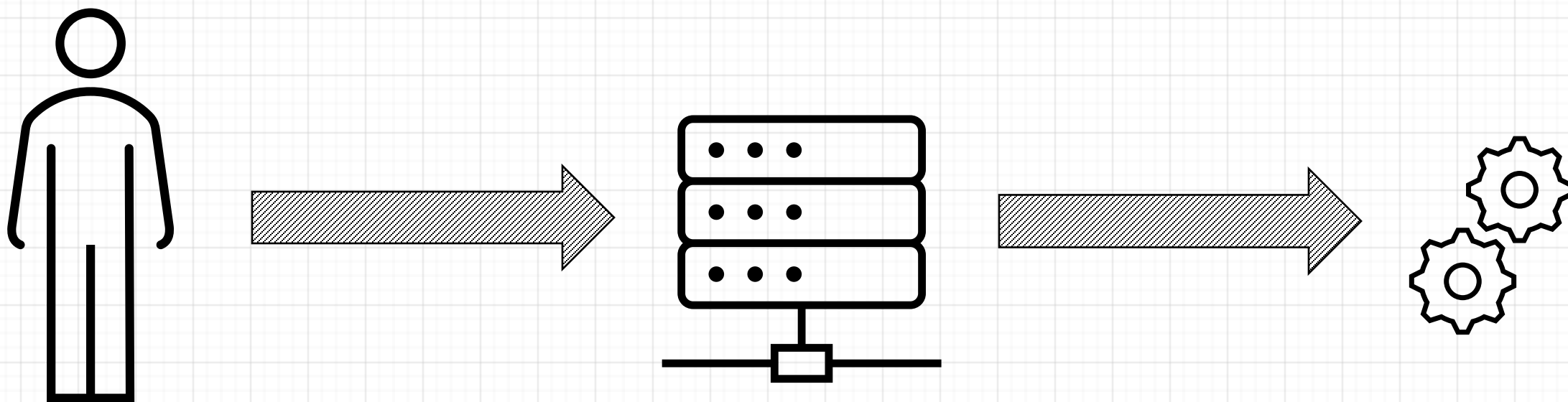
Sposób serwowania	Requests per second
apache2 + mod_php	400.10
apache2 + php-fpm	413.06
nginx + php-fpm	420.84

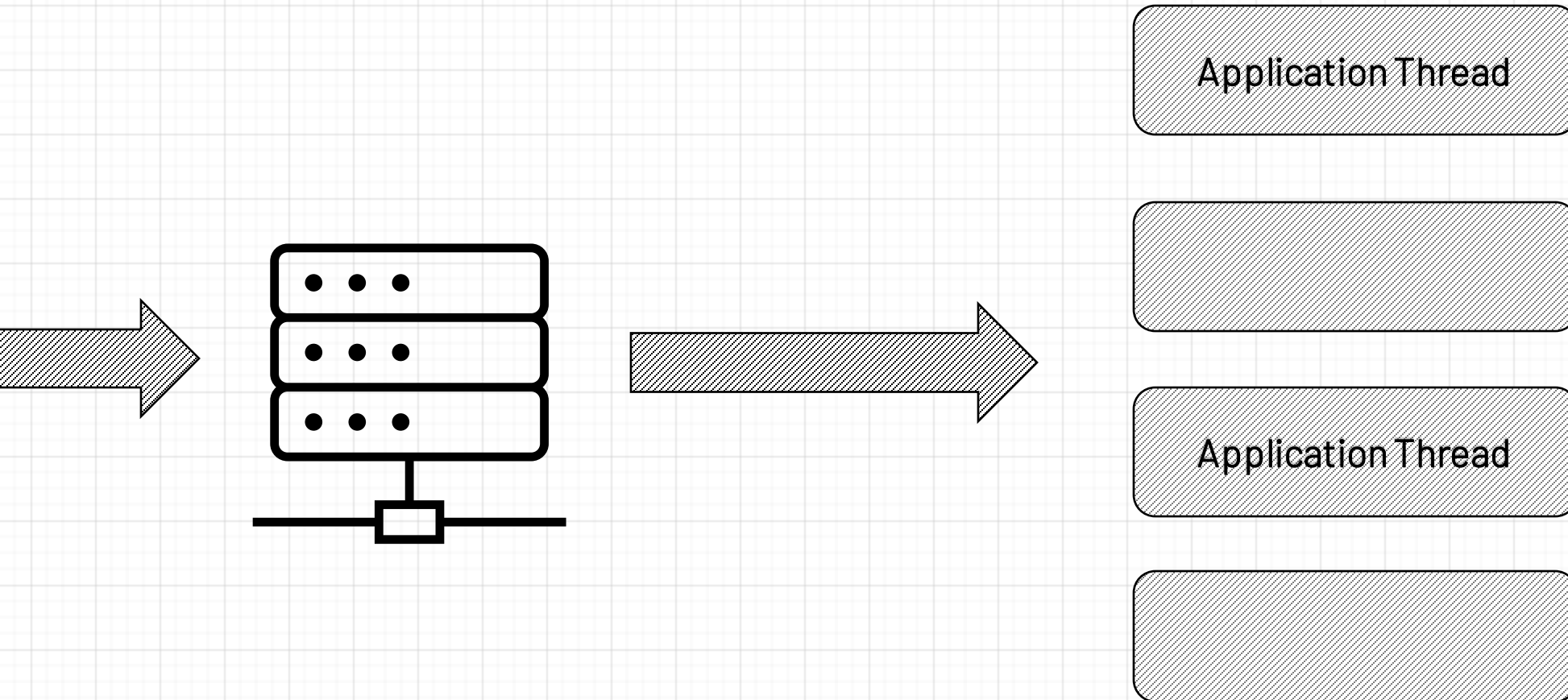


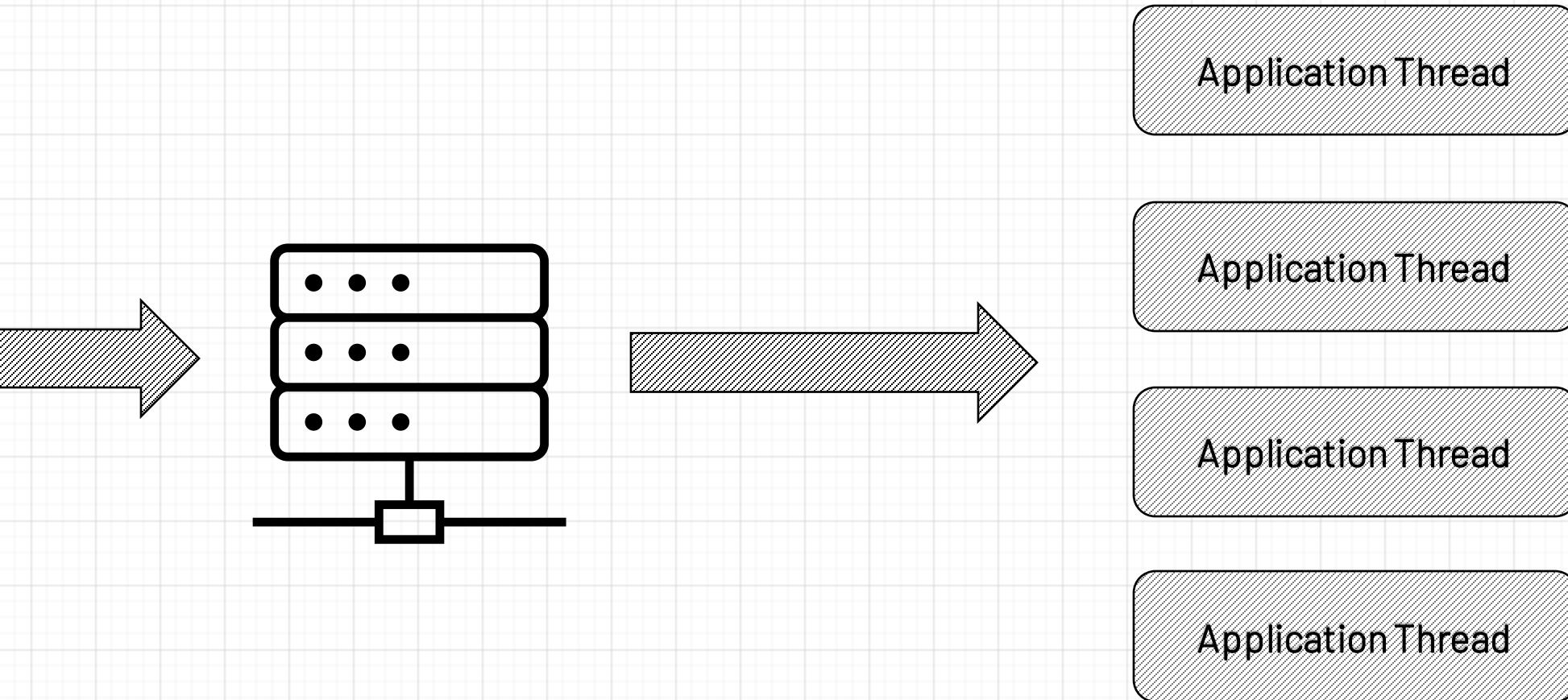


Jak to jest robione w innych
językach?

03/









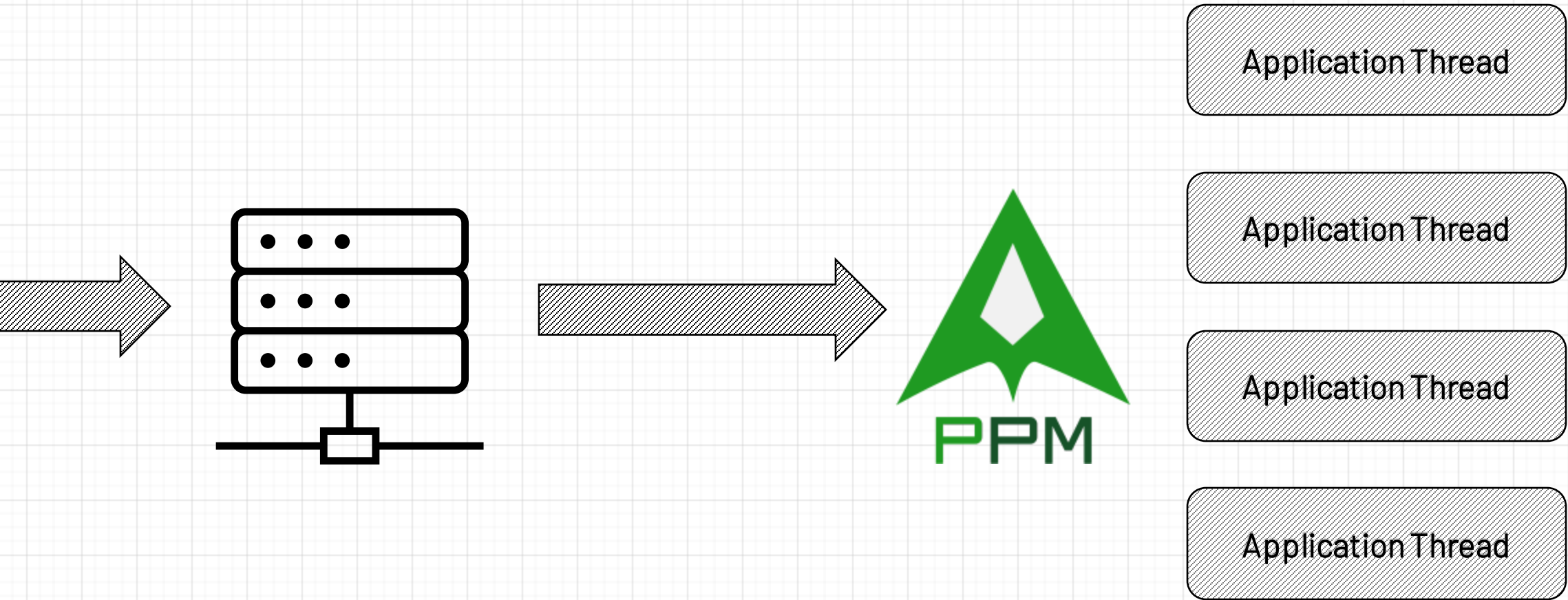
Jak procesować ruch jeszcze
szybciej?

04/

> php-pm

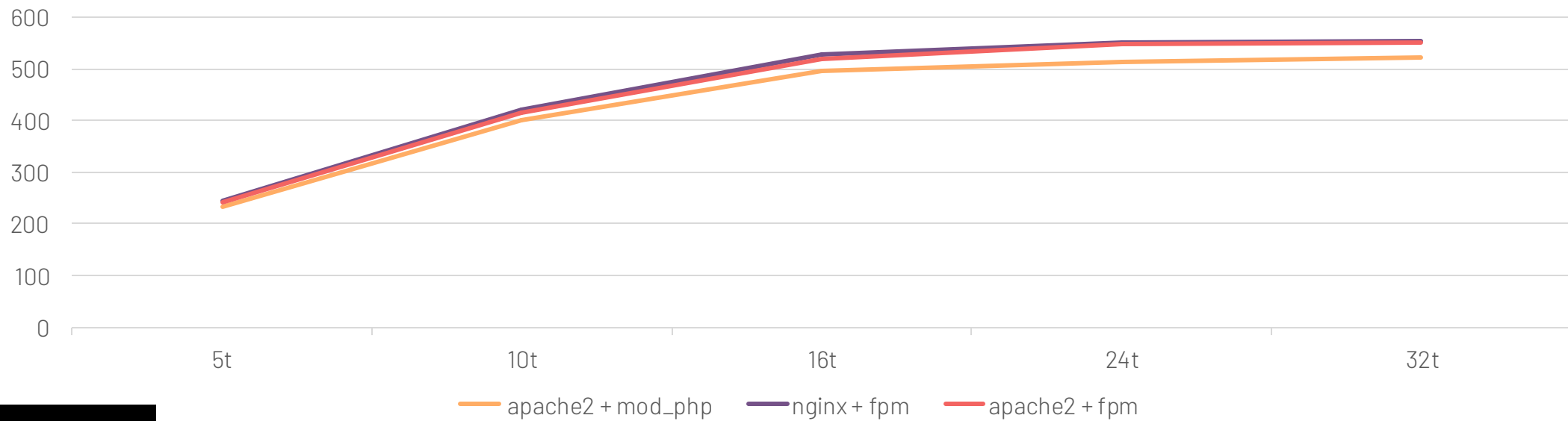
- PPM - PHP Process Manager
<https://github.com/php-pm/php-pm>
- Process manager, supercharger and load balancer for PHP applications
- Performance boost up to 15x (compared to PHP-FPM, Symfony applications).
- Integrated load balancer.
- Hot-Code reload (when PHP files change).
- Static file serving for easy development procedures.
- Support for HttpKernel (Symfony/Laravel), Drupal (experimental), Zend (experimental).





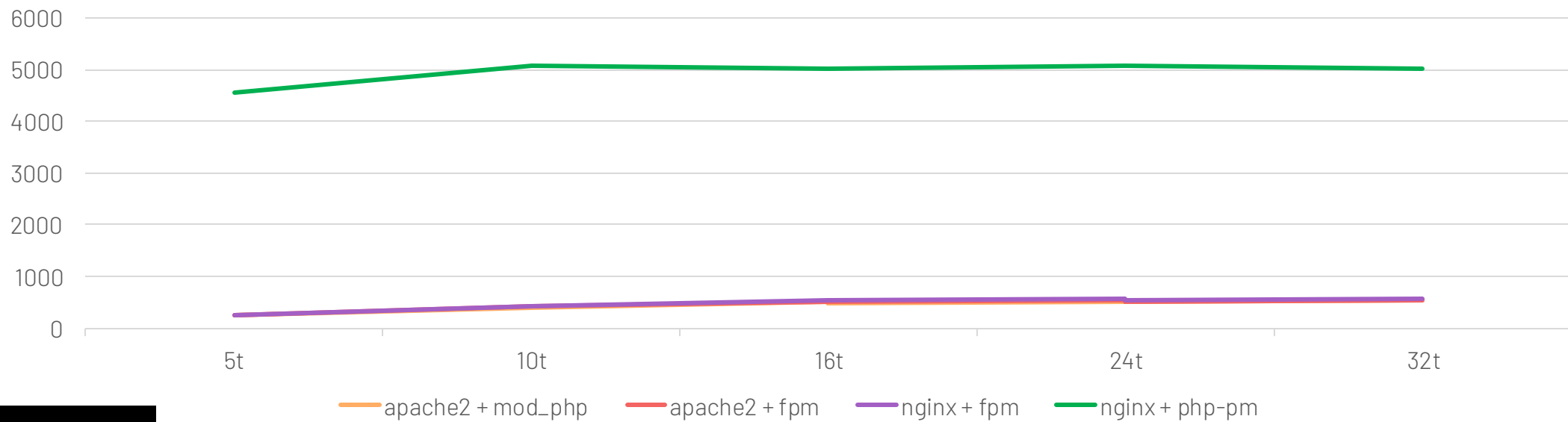
> Benchmark: wyniki

Sposób serwowania	Requests per second
apache2 + mod_php	400.10
apache2 + php-fpm	413.06
nginx + php-fpm	420.84



> Benchmark: wyniki

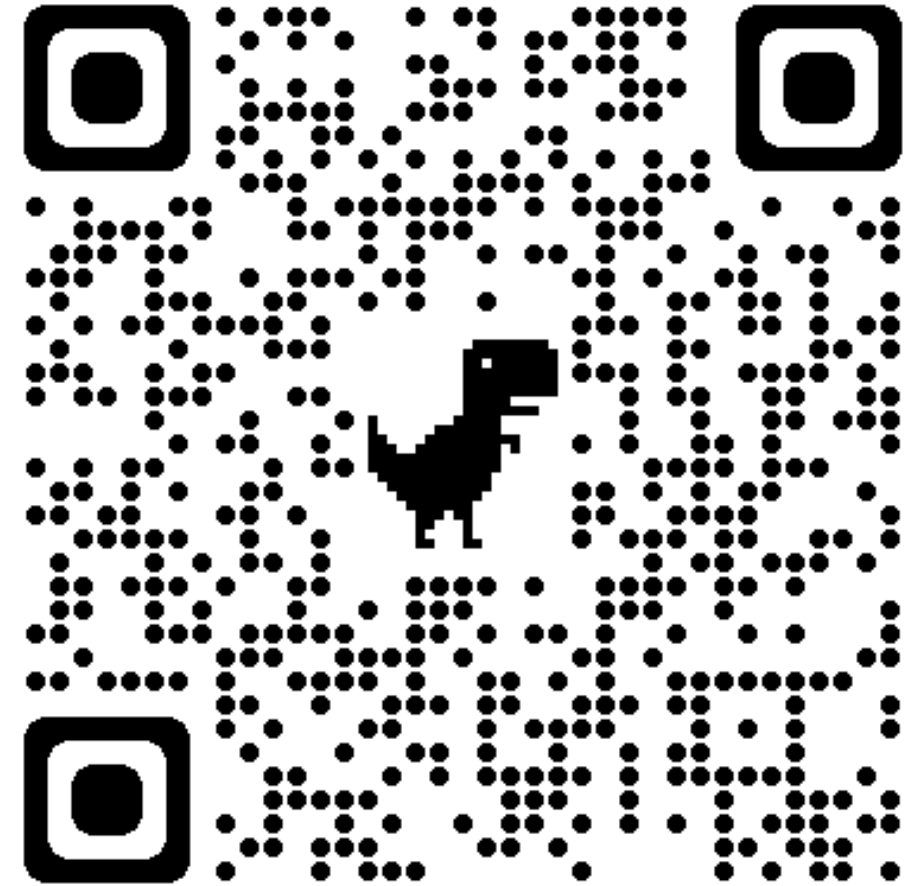
Sposób serwowania	Requests per second
apache2 + mod_php	400.10
apache2 + php-fpm	413.06
nginx + php-fpm	420.84
nginx + php-pm	5061





Podsumowanie

05/



<https://github.com/lpiob/php-serving-benchmarks>





Thank You

unitygroup.com

