

cyber_Folks™

>site_Boosted_

PRZEWODNIK PO
TECHNOLOGIACH
PRZYSPIESZAJĄCYCH
STRONY WWW.

Artur Pajkert



cbr _____

Flks™



Artur

Customer Happiness Architect

>Przedmowa_

Jeśli jesteś świadomym właścicielem lub twórcą stron to doskonale wiesz, że szybkość wczytywania strony to szalenie ważna sprawa. Na forach dyskusyjnych i w mediach społecznościowych często spotykam pytania o doradzenie najlepszego hostingu.

Napisałem tego ebooka, aby podzielić się z Tobą swoimi blisko 20-letnimi doświadczeniami w zakresie hostingu tak, żebyś łatwiej dokonał własnego wyboru. Z lektury tego ebooka dowiedz się, na co warto zwracać uwagę, a czym w zasadzie nie należy się nadmiernie przejmować, jeśli chodzi o wydajność stron www w kontekście technologii dostarczanych przez firmę hostingową. Poznasz też kilka narzędzi, przy pomocy których możesz łatwo sprawdzić, który hosting będzie dla Ciebie najlepszy pod kątem wydajnościowym.

Przyjemnej lektury!

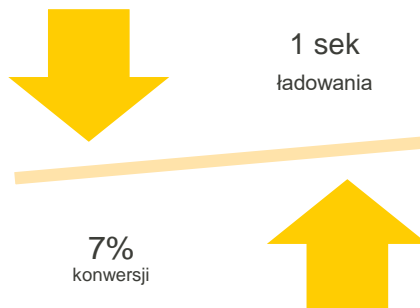
Artur Pajkert

> Dlaczego szybkość jest ważna? _

Kiedy zaczynasz zdawać sobie sprawę...

Początkujący użytkownicy nie podchodzą z uwagą do szybkości działania strony. Zazwyczaj po prostu satysfakcjonuje ich, że strona działa i nie potrzebują „do szczęścia” wiele więcej.

Z czasem jednak zaczynają oczekiwać, że strona będzie działać DOBRZE, że zacznie być widoczna w Google oraz że będzie zarabiać. Wkrótce początkujący właściciel strony poznaje takie określenia, jak ranking Google, doświadczenie użytkownika, konwersja i stopniowo zaczyna rozumieć, że czas ładowania strony to istotny parametr...



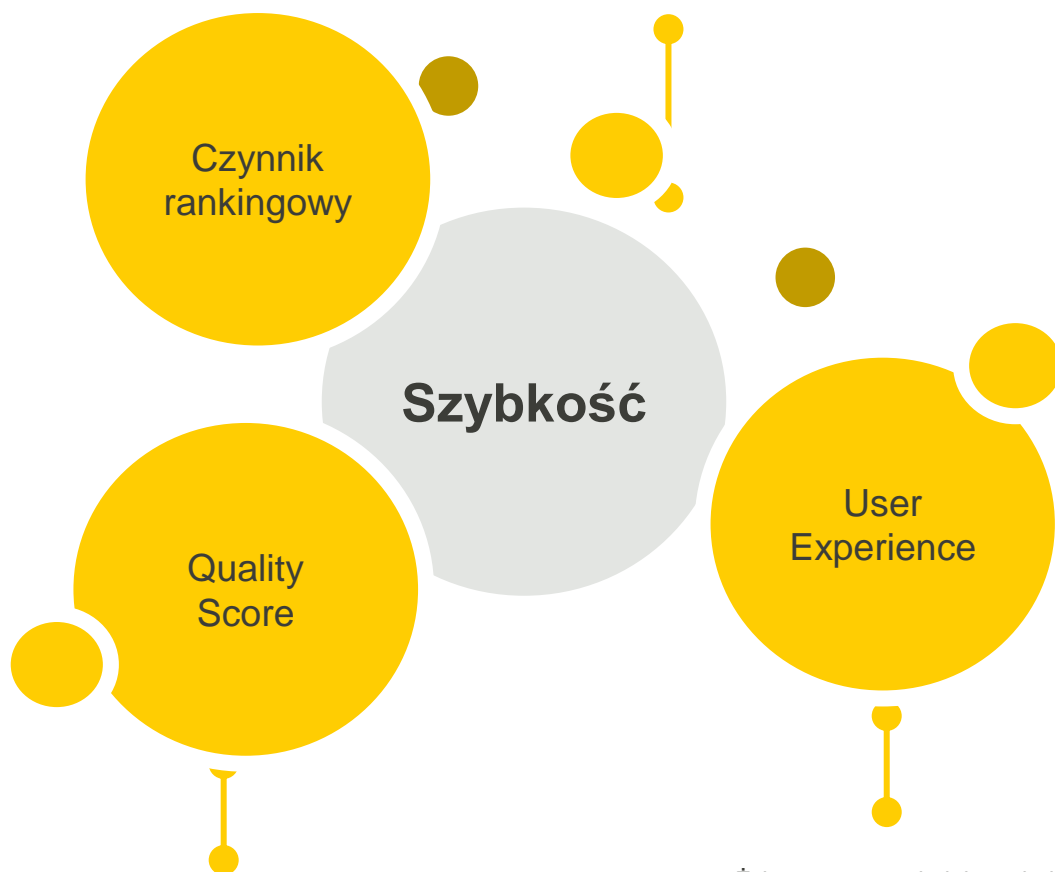
źródło: <https://blog.hubspot.com/marketing/page-load-time-conversion-rates>

Wg badań opublikowanych na hubspot.com – każda sekunda czasu ładowania powoduje obniżenie konwersji aż o 7%. Czy zaczynasz już rozumieć, jak ważne jest, żeby tych sekund było jak najmniej?



>Trzy powody_

Zdaniem większości specjalistów SEO szybkość otwierania stron jest istotnym czynnikiem rankingowym, wpływającym na pozycję Twojej strony w wyszukiwarce Google. Wolniej ładujące się strony mają zdecydowanie mniejszą szansę na osiągnięcie dobrych wyników.



Quality Score – ang. Wynik Jakości – to syntetyczny wskaźnik jakości strony używany w kampaniach Google Ads. Jednym jego składowych jest tzw. doświadczenie strony docelowej – bazujące m.in. na szybkości jej otwierania. Im szybciej otwiera się dana strona, tym lepszy wynik.

Im lepszy Quality Score tym bardziej efektywne są Twoje działania w Google Ads.

Żyjemy w czasach, jak to niedawno usłyszałem od Pawła Tkaczyka „kultury nanosekundy”. Wszyscy pędzimy i oczekujemy jak najszybszego rezultatu. Każda sekunda ładowania strony istotnie obniża poziom cierpliwości użytkownika, a kiedy ten się wyczerpie... cóż. Jak to mawia S. Krug – konkurencja jest na kliknięcie obok...

> Trzy obszary wydajności

Szersze spojrzenie

W tym ebooku omawiam wydajność z punktu widzenia hostingu i możliwości, które on daje. Zanim przejdziemy to tego, co być może interesuje Cię najbardziej, czyli przyspieszenia Twojej strony – zatrzymajmy się chwilę na obszarach wydajności.

Jak się okazuje, istotnym obszarem wydajnościowym w kontekście wyboru firmy hostingowej może być... poczta elektroniczna! Ma to duży związek z samą stroną, kampaniami i konwersjami, co pokażę Ci na studium przypadku.



1

POCZTA

Zdecydowana większość osób przechodzi nad zagadnieniem serwera pocztowego do porządku dziennego.

„Mam przecież pocztę na gmailu” – myślą. Sprawa zaczyna wyglądać inaczej, kiedy prowadzimy aktywne działania reklamowe, o czym przeczytasz już za chwilę w studium przypadku.

2

PHP

Radykalna większość współczesnych stron www to strony dynamiczne, bazujące na języku programowania PHP. WordPress, Joomla!, PrestaShop – to popularne systemy oparte o tę technologię.

Szybkość działania strony będzie, między innymi, zależała od szybkości przetwarzania języka PHP.

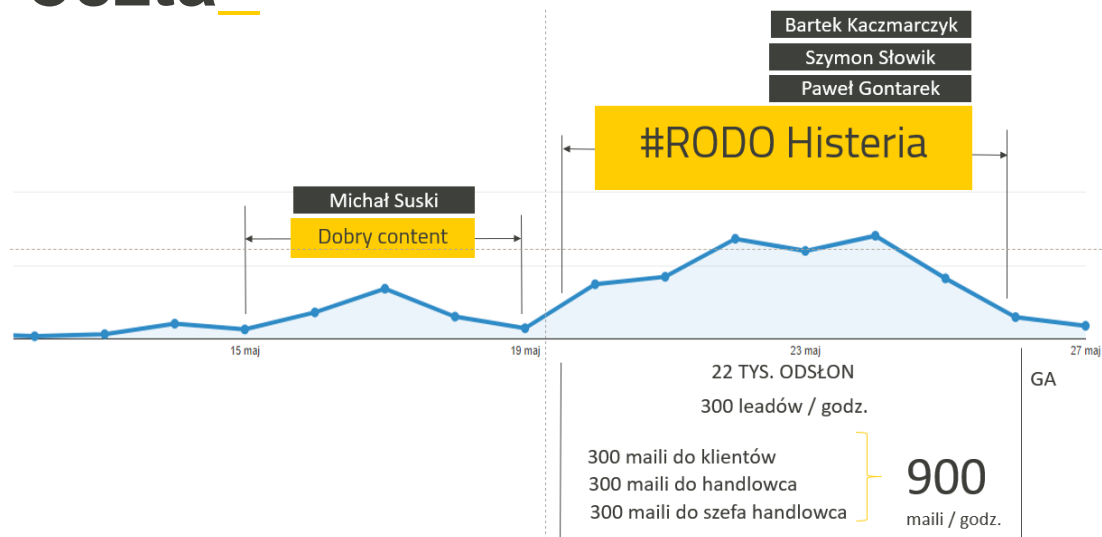
3

MYSQL

Niemal każda współczesna strona www korzysta z bazy danych. Blogi, strony firmowe i sklepy – to wszystko wymaga sprawnej bazy.

Operacje bazodanowe często stanowią wąskie gardło w działaniu strony.

Warstwa łącza i transferu danych – kolejne pokłady wydajności drzemą w samej metodzie przesyłania danych między serwerem i przeglądarką. Bez względu na wynik operacji PHP wykonanych w oparciu o bazy danych – zawsze powstaje coś, co należy przesłać do przeglądarki użytkownika. W tej warstwie także tkwi potencjał do optymalizacji.



CASE STUDY: RODOHISTERIA

Ta historia dotyczy jednej z naszych marek hostingowych, ale równie dobrze mogła wydarzyć się wszędzie. Jak wiele marek, Hekko prowadziło firmowego bloga. Od czasu do czasu na blogu pojawiały się artykuły budzące więcej emocji niż inne.

Jednym z takich wydarzeń okazuje się wpis Michała Suskiego z SurferSEO – dot. doboru fraz w modelu data driven (Michał – jeszcze raz dziękuję Ci za ten świetny wpis!). Jak widzisz powyżej na wykresie, wzbudził on spore zainteresowanie czytelników.

Okazało się jednak, że to jeszcze nic przy tym, jak w momencie wejścia w życie przepisów RODO „rynek” rzucił się na bezpłatny poradnik dla właścicieli stron, który udostępniliśmy wówczas bezpłatnie wszystkim zainteresowanym. Akcję poparły znane osoby ze środowiska „digitalowego” i w ciągu zaledwie kilku dni wygenerowała ona 22 tys. odsłon. W „szczyt” poradnik był pobierany 300 razy na godzinę

W tej akcji konwersją było pobranie poradnika, a konwersja bardzo często oznacza wysłanie maila lub kilku maili (podziękowanie za zakup, faktura, kopia maila do handlowca itp.). Powiedzmy, że są to 3 maile. W wypadku tej akcji oznaczałoby to konieczność wysłanie 900 maili w ciągu godziny. Okazuje się jednak, że firmy hostingowe stosują tzw. parametry bezpieczeństwa. Jest to konieczne, aby balansować obciążenia i – obrazowo mówiąc – uniknąć sytuacji w której jeden pasażer zajmuje wszystkie miejsca w autobusie.



>Poczta_

Mamy zatem do wysłania 900 maili w ciągu godziny. W tym momencie trzeba zwrócić uwagę, że ze względów bezpieczeństwa firmy hostingowe na standardowych rozwiązaniach stosują zabezpieczenia, chroniące przed nadmiernym wykorzystaniem zasobów.

Popatrz na przykładowe ograniczenia kilku operatorów w tym zakresie:

Operator	Ograniczenie
cyber_Folks	5000 / dobę
Ogicom	2000 / godz. (z Polski)
AZ	500 / godz.
Atthost	1000 na dobę

Dane ze stron operatorów. Maj, 2018.

SMTP w chmurze

Jeśli przewidujesz wysyłanie znacznej liczby maili w krótkim czasie – rozważ dedykowaną usługę. Być może Twój operator ma w ofercie hosting poczty o podniesionych parametrach.

Istnieją dedykowani dostawcy rozwiązań, specjalizujący się w obsłudze takiego ruchu pocztowego. Moim zdaniem w Polsce godny uwagi jest zwłaszcza EmailLabs. Od wielu lat korzystam z tego właśnie rozwiązania. To naprawdę proste – zamiast klasycznego SMTP podajesz w konfiguracji strony lub Twojego skryptu wysyłkowy adres SMTP uzyskany od EmailLabs i masz niemal pewność, że nawet duża wolumenowo poczta dotrze do adresata.



Pamiętaj! Zanim zaczniesz przyspieszać Twoją stronę www i prowadzić intensywne kampanie reklamowe, upewnij się, że Twój system pocztowy jest na to odpowiednio przygotowany.

Zmarnowanie
pieniędzy na
kampanię

Obniżenie reputacji
marki

NIEDOSTARCZONA POCZTA

Stracone konwersje

Czas stracony na
szukanie przyczyny

>PHP_

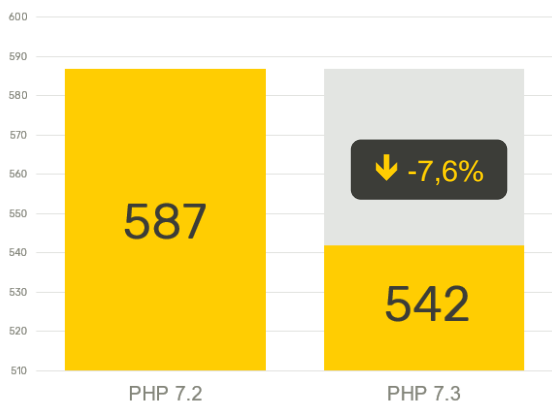
PHP to podstawowy język dynamicznego budowania stron. To w oparciu o PHP działają popularne systemy CMS, jak WordPress, Joomla!, PrestaShop. Kod PHP jest przetwarzany na serwerze, a jego wynik jest przekazywany do przeglądarki użytkownika. Mówiąc zatem w uproszczeniu, aby użytkownik w ogóle mógł zobaczyć stronę, aby w ogóle zaczęła do niego docierać odpowiedź z serwera – kod PHP musi najpierw się na nim wykonać.

Oznacza to, że szybkość wykonywania kodu PHP wpływa na to, jak szybko będzie działać strona.

Kod PHP – co siedzi w środku?

Kod PHP to instrukcje wewnętrzne tego języka oraz polecenia, korzystające z zewnętrznych zasobów, głównie baz danych. Oznacza to, że na czas wykonania skryptu składa się najczęściej czas wykonania poleceń SQL kierowanych do bazy danych, oraz czas niezbędny na wykonanie instrukcji PHP, zazwyczaj wykonujących różne operacje na danych zwróconych z bazy lub skierowanych do zapisania do niej.

Czas wykonania skryptu PHP Benchmark w [ms]



Najważniejsza sprawa, to możliwie świeża, aktualna wersja PHP na serwerze. Okazuje się, że wraz z każdą kolejną wersją rośnie wydajność PHP. To jeden z powodów, dla których warto korzystać z najnowszej dostępnej wersji, choć w praktyce jest ich więcej. Obok znajdziesz porównanie, średnio PHP 7.3 jest szybsze o 7% od swojego poprzednika.

Do najważniejszych, obok **wydajności**, należą **bezpieczeństwo** oraz **funkcjonalność** - nowe instrukcje wprowadzane w kolejnych wersjach.

- Lepsza kompatybilność ze starszymi aplikacjami
- Nic nie musisz zmieniać

PHP
STARSZE

PHP
NOWSZE

- Większa wydajność
- Nowe funkcje
- Poprawione bezpieczeństwo
- Potencjalne problemy z kompatybilnością (rzadko, ale jednak możliwe)

>PHP – jak to zmierzyć?_

Skoro wiemy już, że szybkość interpretera PHP może mieć znaczenie, to poznaj jeden z lepszych sposobów na pomiar czystej wydajności PHP – skrypt PHP Benchmark. Jest to bezpłatny skrypt, który wykonuje w pętli operacje często wykonywane przez aplikacje i w ten syntetyczny sposób mierzy wydajność.

Oznacza to, że szybkość wykonywania kodu PHP wpływa na to, jak szybko będzie działać strona. Ten benchmark mierzy wydajność czterech głównych grup funkcji, a następnie podaje wynik syntetyczny. Dzięki temu, że funkcje te nie korzystają z baz danych ani zewnętrznych aplikacji – zmierzona zostaje szybkość przetwarzania samych instrukcji PHP.

PHP Benchmark

Kod PHP to instrukcje wewnętrzne tego języka oraz polecenia, korzystające z zewnętrznych zasobów, głównie baz danych. Oznacza to, że na czas wykonania skryptu składa się najczęściej czas wykonania poleceń SQL kierowanych do bazy danych, oraz czas niezbędny na wykonanie instrukcji PHP, zazwyczaj wykonujących różne operacje na danych zwróconych z bazy lub skierowanych do zapisania do niej.



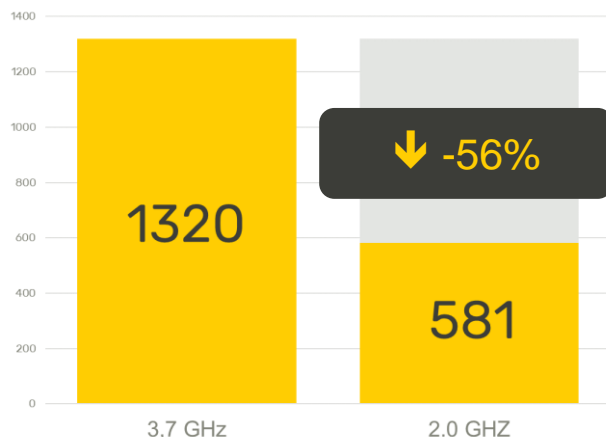
Oto fragment skryptu. Jak widzisz, wykonuje on 130 tysięcy iteracji na funkcjach przetwarzających ciągi tekstowe, jak np. dodanie ukośników czy policzenie hashu sha1 z danego ciągu tekstowego. Następnie (nie jest to pokazane na tym fragmencie kodu) mierzy czas wykonania. Analogicznie dla pozostałych bloków „tematycznych”.

```
private static function test_StringManipulation($count = 130000) {
    $time_start = microtime(true);
    $stringFunctions = array("addslashes",
        "chunk_split", "metaphone", "strip_tags", "md5", "sha1",
        "strtoupper", "strtolower", "strrev", "strlen", "soundex", "ord");
    foreach ($stringFunctions as $key => $function) {
        if (!function_exists($function))
            unset($stringFunctions[$key]);
    }
    $string = "the quick brown fox jumps over the lazy dog";
    ...
}
```

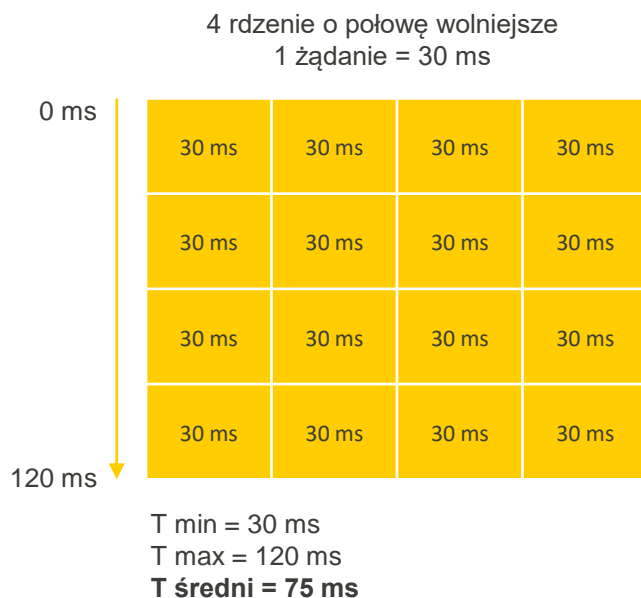
Skrypt możesz uruchomić praktycznie na dowolnym hostingu w trakcie trwania okresu testowego i samodzielnie sprawdzić wydajność serwera.

>PHP vs procesory w serwerze_

Szybkość przetwarzania kodu PHP na serwerze jest wyraźnie skorelowana z częstotliwością taktowania procesora. Na wykresie obok przedstawiam czas wykonania skryptu PHP Benchmark, w zależności od wykorzystywanego przez serwer procesora. Jak widzisz, procesor taktowany częstotliwością 3.7 GHz jest w tym zadaniu ponad dwukrotnie szybszy, niż procesor o taktowaniu 2.0 GHz. Można zatem odnieść wrażenie, że im wyżej taktowany procesor, tym lepiej dla Twojej strony... i może to być prawda – ale tylko częściowo!



Chodzi o to, że proces PHP zajmuje zazwyczaj po prostu jeden wątek procesora – nie skaluje się automatycznie na więcej rdzeni albo wątków. Od początku do końca idzie jednym wątkiem. Dla uproszczenia wyobraź sobie dwa serwery. Jeden ma procesor o wysokim taktowaniu, ale dwurdzeniowy. Drugi ma cztery rdzenie, ale wolniejsze. Wykonanie tego samego zadania na szybszym zegarze procesora trwa 20 ms, a na wolniejszym – 30 ms. Oto uproszczony schemat graficzny przetwarzania dla 16 identycznych skryptów:



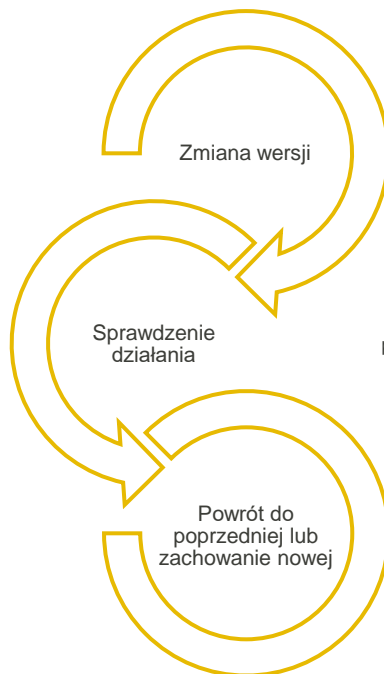
Lepsze czasy wykonania uzyskaliśmy tu dla serwera o większej liczbie rdzeni, mimo, że pojedyncze rdzenie są taktowane po prostu wolniej. Przy większej liczbie uruchomionych skryptów (np. więcej odwiedzin na stronach) lepiej więc sprawdzą się procesory z dużą liczbą rdzeni.

>PHP – co jeszcze?_

Jak zatem samodzielnie zmienić wersję PHP na serwerze?

Zacznijmy od tego, żebyś upewnił się, czy w ogóle Twój hosting umożliwia samodzielny wybór PHP. Większość operatorów tak robi, ale oczywiście mogą być wyjątki. Czasami firmy hostingowe możliwość zmiany wersji PHP nazywają **multiphp**.

Poprzez odpowiednie wpisy wykonane samodzielnie w pliku .htaccess, sterującym pracą serwera www, możesz nakazać serwerowi przetwarzanie plików PHP z użyciem odpowiedniej wersji interpretera. Najczęściej operatorzy udostępniają też wygodny panel, w którym jednym kliknięciem dla danej domeny możesz ustawić wybraną wersję interpretera PHP.



Rozważ dopasowanie skryptów w razie problemów. To lepsze, niż powrót do starszej wersji PHP, ale wymaga większych nakładów.

PHP 7.3 - Ciekawostka

PHP 7.3 pojawiło się stosunkowo niedawno. Jako jedni z pierwszych wykonaliśmy benchmarki i okazało się, że jest szybsze o ok. 7% od swojego poprzednika... Jest jednak świetnym przykładem możliwości niekompatybilności i dość interesującej.

Funkcja `mb_strtoupper()` zamienia alfabetyczne litery ciągu (Unicode) na wielkie. Sama funkcja nie jest nowa, jednak wynik jej działania różni się między PHP 7.3 i poprzednimi wersjami. Chodzi o obsługę niektórych znaków, w szczególności w języku niemieckim. Dla niektórych może to mieć duże znaczenie w aplikacjach, a także w SEM/SEO, bo fraza „Straße” po przekształceniu w wielkie litery nie brzmi teraz „STRAßE” ale „STRASSE” i niektóre systemy mogą ją rozpoznać jako całkiem inne słowo!



Co to jest cache?

Kiedy ktoś spyta Cię, ile to jest $1141 * 34$ potrzebujesz chwili na wykonanie takiego działania, np. na policzenie w pamięci lub wpisanie tych liczb do kalkulatora. Jeśli jednak kilka sekund później ponownie usłyszysz to pytanie – możesz natychmiast podać odpowiedź, którą wciąż widzisz na wyświetlaczu.

Cacheowanie polega na przechowywaniu zasobów, do których często się odwołujemy, a do których dostęp jest „powolny”, w taki sposób, żeby w kolejnych żądaniach można było je serwować szybciej.

Np. jeśli cachujemy lokalnie obrazek, to oznacza, że przeglądarka pobierze go i zapisze sobie w komputerze lokalnym i następnym razem gdy wejdiesz na stronę – nie będzie już pobierać go z serwera (byłoby to wolniejsze niż korzystanie z zapisanego już na Twoim lokalnym dysku zasobu).

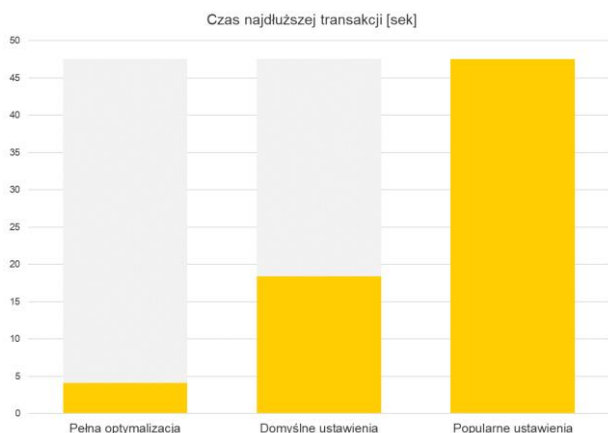
Cache_	Na czym to polega_
LSCache	Niektóre firmy hostingowe oferują ten rodzaj cache, jeśli ich serwerem www jest LiteSpeed. Obejmuje on cache'owanie całego kodu HTML strony internetowej tak, żeby była ona „gotowa” do zaserwowania przy kolejnym żądaniu. Mechanizm zapewnia bardzo wydajne serwowanie statycznych zasobów. Współpracuje z pluginem LS Cache do WordPress.
OP Cache	To system przyspieszający działanie PHP w ten sposób, że serwer przechowuje gotowe wyniki działania skryptu PHP. Skrypt nie musi być uruchamiany za każdym razem od nowa, skoro można przechować tylko wynik jego działania i od razu podać ten wynik. OP Cache działa niskopoziomowo, nie wymaga ze strony właściciela strony dodatkowych zabiegów, musi jednak zostać włączony na serwerze (naturalnie firma hostingowa musi go udostępniać.)
Object Cache	Klasyczne cache'owanie przechowuje w pamięci lub podręcznych plikach np. obrazki lub całe strony internetowe, a cache obiektowe – wyniki zapytań do bazy danych, przechowywane w formacie par klucz=>wartość.
CDN	Content Delivery Network – czyli sieć dostarczania treści. To sieć serwerów, które przechowują określone zasoby, niezbędne do wyświetlenia strony, np. JavaScript. Chodzi o to, żeby taki JavaScript załadował się z serwera możliwie bliskiego użytkownikowi. Nie ma to większego znaczenia, kiedy użytkownicy Twojej strony są blisko Twojego serwera (np. serwer w Polsce, użytkownik w Polsce), ale im większa ta odległość tym większe korzyści możesz odnieść z sieci CDN, bo użytkownik będzie pobierał dane z serwera CDN blisko siebie, a nie z Twojej „macierzystej” maszyny. Ten rodzaj cachowania w niewielkim stopniu wiąże się z samym dostawcą hostingu.
Cache przeglądarki	Ten rodzaj cacheowania w żadnym stopniu nie łączy się z operatorem hostingu, ale wpływa na komfort otwierania stron. Po prostu przeglądarka użytkownika cachuje (zapisuje lokalnie) pewne zasoby, jak np. obrazki – po to, żeby przy kolejnym żądaniu nie pobierać ich ponownie z serwera, tylko podawać lokalnie z komputera użytkownika, co jest znacznie szybsze.

>Wydajność vs szybkość

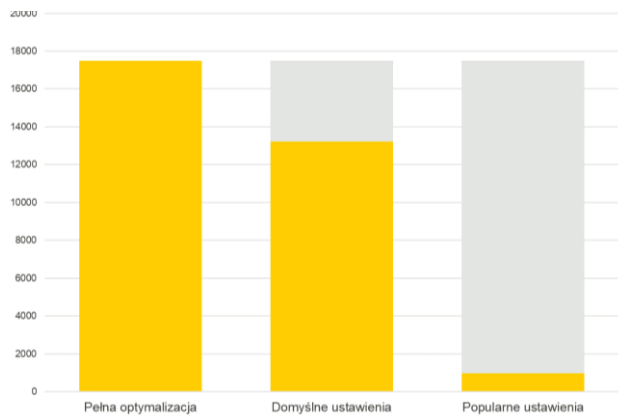
Szybkość

Kiedy rozważamy zagadnienie wydajności i szybkości ważne, żeby rozumieć, że wcale nie oznaczają one tego samego. Szybkość otwierania strony to czas, w jakim dany użytkownik otworzy stronę internetową. Zależy ona od wielu czynników, w tym od tych leżących po stronie firmy hostingowej. Ważne jednak, aby rozumieć, że chodzi tu o doświadczenie JEDNEGO użytkownika.

Czas najdłuższej transakcji [sek]



Obsłużone transakcje 1/min



Wydajność

O wydajności hostingu i strony mówimy natomiast raczej w kontekście zdolności do obsłużenia wielu żądań w krótkim czasie.

To tak, jak w sklepie. Może być bardzo szybka kasjerka, ale jeśli jest tylko jedna, to tylko ten klient, który jest przy kasie, będzie zadowolony, pozostali będą musieli czekać. Wydajny serwer to taki, który ma „dużo kas” – jest w stanie jednocześnie obsługiwać wielu użytkowników strony www, najlepiej – z dość przewidywalną szybkością.

Siege - Obłężenie

Siege to aplikacja działająca z poziomu linii komend SSH. Umożliwia ona symulację wielu jednoczesnych wejść na stronę www. Mierzy liczbę żądań obsłużonych w ciągu minuty oraz podaje czas obsługi najdłuższego żądania. Stanowi zatem interesujące narzędzie oceny tego, ile będzie obsługiwany ten „najmniej zadowolony” klient.

Jeśli strona ma spory ruch, to poza miarami szybkości ważne są też miary wydajnościowe.

Wykresy pokazują parametry dla tej samej strony (sklep PrestaShop), zależnie od ustawień po stronie hostingu i optymalizacji samego sklepu.

Jak widać, albo jesteś w stanie obsłużyć tysiąc, albo siedemnaście tysięcy użytkowników na minutę. Słupki po lewej jasno pokazują, że w wariancie czerwonym najdłuższa transakcja jest też wielokrotnie krótsza.

Ten wariant zakłada m. in. korzystanie LS Cache oraz optymalizację ustawień, baz i obrazków.

>Mysql_

Wydajność baz danych możesz mierzyć podobnie, jak wydajność interpretera PHP. W tym wypadku jednak proponuję Ci inny skrypt, który wykonuje przede wszystkim operacje na bazie danych, a minimalizuje wpływ samych poleceń MySQL na wydajność.



Simple DB Speed Test ver. 1.2

Your speed report:

2019-04-29 17:02:53
host: localhost [C:] WINNT, using PHP version: 7.2.3, using MySQL, host: localhost
engine type: InnoDB mode: Transactions

Operation type	Speed	Rating
Inserting records E.g. Inserting new article, placing a new order, registering as new customer.	12045/sek	Good
Finding record by primary key E.g. Finding product by ID in e-commerce application, or article by ID in CMS.	9381/sek	Good
Finding unindexed integer records with exact match E.g. Finding all shirts with size 40 in e-commerce application.	625/sek	Good
Text search. Searching for 3-characters string inside records E.g. Searching for all customers' names containing 'et'.	217/sek	Good
Finding unindexed string records with exact match E.g. Searching for all items of specific colour.	459/sek	Average
Select records where integer field value * 1.23 > random number E.g. Example: Searching for all products with size bigger than...	13587/sek	Good
Updating records E.g. Updating 'color' field - changing color in all item, where color is green for new color blue.	39/sek	Average

Benchmark analizuje wydajność wstawiania rekordów do bazy (np. złożenie nowego zamówienia w sklepie), odnajdowanie rekordów po kluczu głównym (np. aby wyświetlić dany artykuł), wyszukiwania liczb, wyszukiwanie pełnotekstowe (np. kiedy użytkownik usiłuje odnaleźć w sklepie towar zawierający określone słowa w opisie) i inne operacje. Każda operacja jest wykonywana w wielokrotnej pętli, a aplikacja liczy liczbę operacji w jednostce czasu, podając wydajność dla danego rodzaju operacji bazodanowych,

Simple Sql Benchmark Script

To stworzony przeze mnie skrypt, który wykonuje w pętli polecenia SQL, stosując różne mechanizmy składowania oraz tryby:

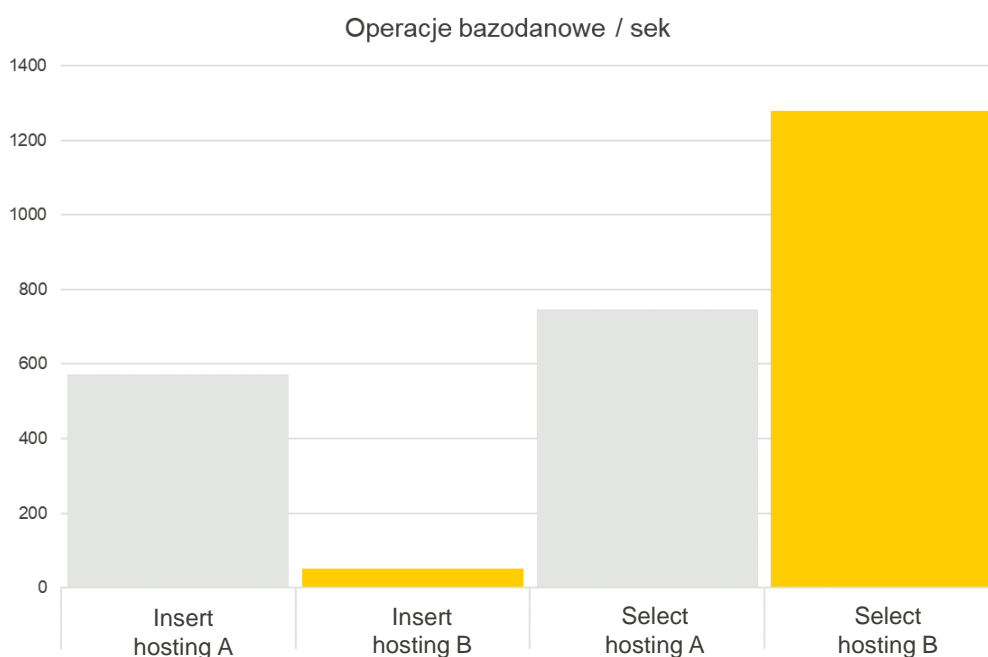
- jedno połączenie z bazą na każde polecenie sql
- jedno połączenie z bazą i kilka poleceń
- kilka poleceń zebranych w transakcje



>Mysql_

Okazuje się, że uzyskane różnice wydajnościowe są znaczne. Warto przeanalizować, jakie operacje dokładnie wykonuje Twoja aplikacja i wybierając hosting zwrócić uwagę na ten rodzaj operacji, który może stanowić potencjalne „wąskie gardło”. Przykładowo, generalnie rzecz biorąc, odczyty i zapisy do bazy mogą uzyskiwać istotnie różne poziomy wydajności.

Porównałem dla Ciebie wydajność dwóch przykładowych serwerów, dla operacji INSERT, czyli dopisywania do bazy nowych rekordów oraz operacji SELECT, czyli odczytywania wyszukanych rekordów z bazy. Badania były wykonane na w środowiskach hostingowych 2 różnych marek.



Okazuje się, że w wypadku dodawania rekordów – INSERT – Hosting B uzyskał w tych samych warunkach zaledwie kilkadziesiąt rekordów na sek. Podczas gdy hosting B blisko 600 rekordów na sekundę. Jest to więc miażdżąca wręcz przewaga hostingu A.

Sytuacja zmienia się jednak diametralnie w chwili, kiedy mówimy o odczycie rekordów. W teście znajdowania rekordów po kluczu głównym (np. po ID produktu w bazie) na serwerze w hostingu B uzyskujemy ok. 1300 rekordów na sekundę, podczas gdy w A nieco ponad 700 rekordów.

Okazuje się zatem, że z punktu widzenia wydajności MySQL kiedy w naszej aplikacji wąskim gardłem są zapisy – hosting A będzie lepszym wyborem, a kiedy Odczyty – hosting B.

>Redis/Memcached

Za każdym razem, kiedy w kodzie Twojej strony pojawia się wykonanie polecenia SQL, serwer bazodanowy musi je wykonać... ale przecież bardzo często są to dokładnie takie same polecenia, dające takie same wyniki! Po co więc za każdym razem wykonywać takie polecenie. Znacznie lepiej byłoby przechowywać w szybciej pamięci wynik takiego polecenia i serwować gotową odpowiedź.

Dwie technologie hostingowe, które wspierają cachowania wyników zapytań do bazy to Memcached i Redis. Trudno wskazać, która jest bardziej wydajna, natomiast Redis jest nowszy i ma większą funkcjonalność, zapewniając przykładowo autoryzację użytkowników. Dobrze, kiedy Twój hosting wspiera co najmniej jedną z tych technologii.

Cache'owanie obiektowe

Klasyczne cache'owanie przechowuje w pamięci lub podręcznych plikach np. obrazki lub całe strony internetowe, a cache obiektowe – wyniki zapytań do bazy danych, przechowywane w formacie par klucz=>wartość.

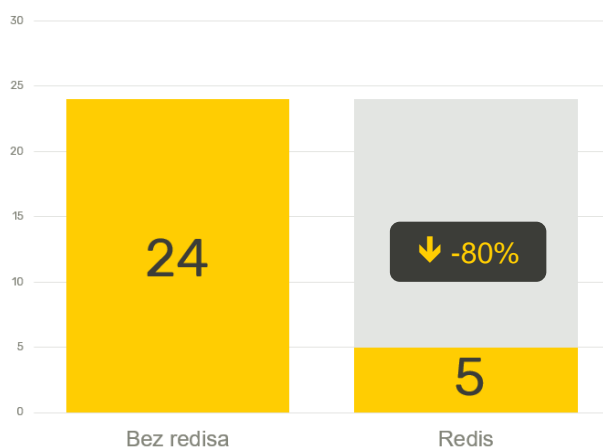
Sam możesz przekonać się, ile tych zapytań jest w Twoim wypadku, wpisując np. w stopkę strony kod:
<p>Liczba zapytań do bazy: <?php echo get_num_queries();?></p>

Przykładowo w WordPress jest przygotowany na takie rozwiązanie. Object cache w WordPressie zaimplementowany jest za pomocą klasy WP Object Cache i polega na cache'owaniu wyników zapytań SQL do bazy danych. Eliminuje to konieczność częstego łączenia się z bazą i pozwala na szybszy dostęp do danych. Domyślnie informacje te nie są jednak trwale i przeładowują się przy każdym wejściu użytkownika na stronę. Stosowanie technologii cacheowania obiektowego Redis/Memcached rozwiązuje ten problem.

Ile zapytań do bazy wykonuje Twój WordPress kiedy wchodzisz na stronę?

Okazuje się, że może to być 20-30 zapytań. Kiedy zamiast każdorazowo odpytywać serwer SQL korzystasz z metod cacheowania obiektowego może się okazać, że liczba realnie wykonywanych zapytań spadnie nawet 5-krotnie!

Liczba wykonanych komend SQL



Domyślna instalacja WordPressa, w motywie potomnym Twenty Seventeen przy każdym wejściu na stronę.

>HTTP/2_

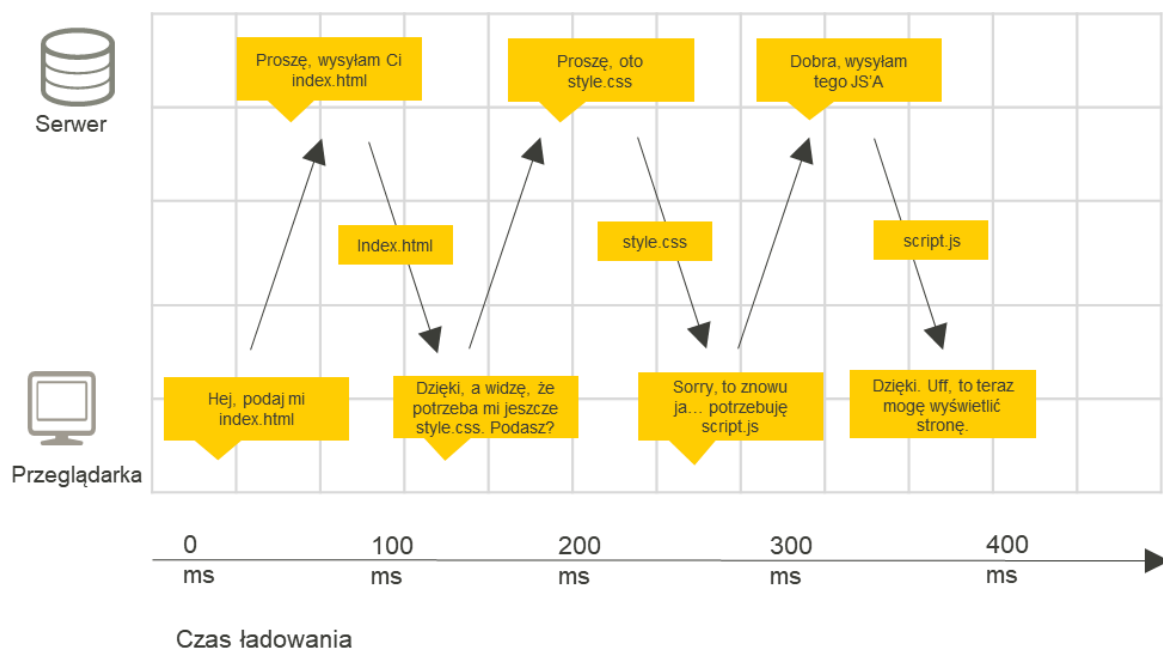
Mówiąc w uproszczeniu: tradycyjny protokół HTTP 1.x działa w taki sposób, że kiedy użytkownik wpisuje w przeglądarce adres Twojej strony – prosi ona serwer o główny plik strony, zazwyczaj index.html

Serwer znajduje ten plik i odsyła do przeglądarki użytkownika. Przeglądarka czyta go i widzi, że jest potrzebny arkusz stylów css, odpowiedzialny za wygląd. Prosi zatem serwer o ten arkusz.

Po otrzymaniu arkusza przeglądarka wraca do HTML i okazuje się, że już po chwili natrafia na linię, nakazującą pobranie javascriptu – pliku .js. Ponownie zwraca się więc do serwera... i tak dalej.

Prędkość światła w próżni

Odległość z Los Angeles do Warszawy to ok. 10 tys. km. Światło potrzebuje ok. 32 ms, aby pokonać tę odległość. Nie zdążysz mrugnąć w tak krótkim czasie, ale jeśli pomyślisz, że takie opóźnienie może pojawić się dla KAŻDEGO zasobu ładowanego na Twojej stronie (obrazy, pliki js, css etc.) – to zrozumiesz, dlaczego HTTP/2 jest takie ważne. Suma liczby zasobów, wymnożona przez 20-30 ms daje już odczuwalne części sekundy, a nawet – więcej niż sekundę!



Powyższy diagram pokazuje, jak wygląda wymiana informacji między przeglądarką a serwerem w uproszczonym ujęciu. „Gołym okiem” widać, że następuje „pingpong” kolejnych wymian informacji między serwerem i przeglądarką. Na każdym takim połączeniu tracimy czas na zestawienie połączenia.

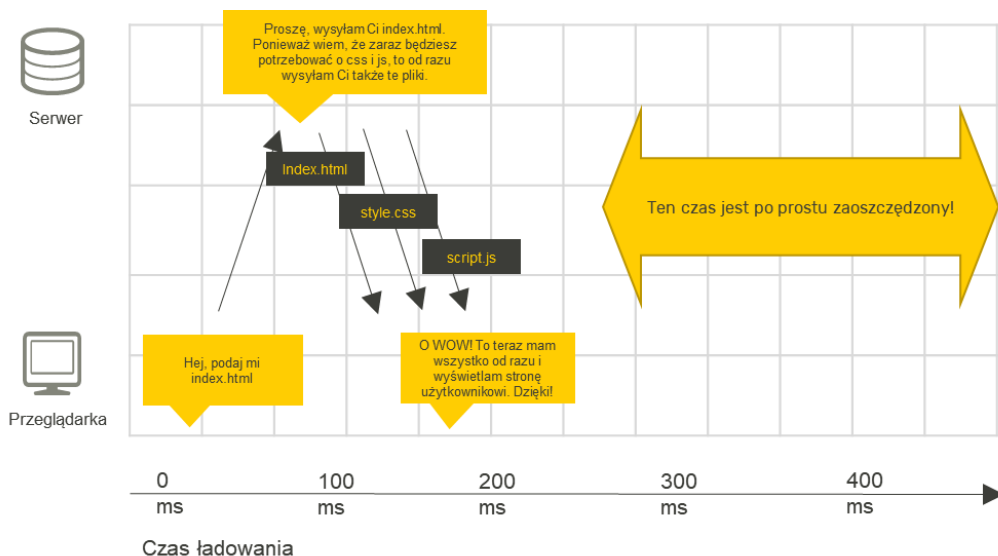
W praktyce tak działający protokół był stopniowo zastępowany kolejnymi wersjami i współczesne przeglądarki umieją łączyć się z serwerem poprzez kilka równoległych połączeń jest to jednak działanie ograniczone, zazwyczaj do maksymalnie 6-8 zasobów.

>HTTP/2_



Zalety HTTP/2

- Pracuje na danych binarnych, a nie tekstowych (szybciej, mniejsza podatność na błędy)
- Obsługuje wiele zasobów w jednym połączeniu
- Pozwala na równoległą transmisję – bez czekania na zakończenie pobierania poprzedniego zasobu
- Kompresuje nagłówki
- Pozwala wypychać zasoby ZANIM przeglądarka o nie poprosi



W eksperymencie, na stronie opartej o WordPress z WooCommerce, po włączeniu pluginu LS Cache wraz z funkcją PUSH dla zasobów css i js uzyskaliśmy blisko 3-krotne skrócenie czasu ładowania strony.

Nawet jeśli nie chcesz albo nie możesz korzystać z pluginu, to z funkcji PUSH możesz śmiało korzystać także „ręcznie” poprzez odpowiednie wpisy w nagłówku strony w PHP lub przez .htaccess.

-63%

>DNS Anycast

Właściciele i twórcy stron i sklepów deklarują, że kluczową cechą hostingu jest wydajne otwieranie stron. Jednocześnie ochrona przed atakami jest wskazywana jako kluczowa cecha dla 80,7% respondentów – mówią badania przeprowadzone przez H88 w marcu 2019 roku. Właśnie dlatego uruchomiliśmy system DNS Anycast.

Kiedy użytkownik wpisuje adres Twojej strony do przeglądarki, to w pierwszej kolejności przeglądarka musi ustalić adres IP serwera, na którym jest strona – do tego właśnie służą serwery DNS, to one, na podstawie wpisanej nazwy domenowej, dostarczają jej wiedzy o tym, skąd stronę można pobrać. Działanie serwerów DNS wpływa na szybkość i bezpieczeństwo. Dlatego są one atrakcyjnym celem do atakowania – wyłączenie lub sparaliżowanie takiego serwera oznacza bowiem poważne problemy.

91%

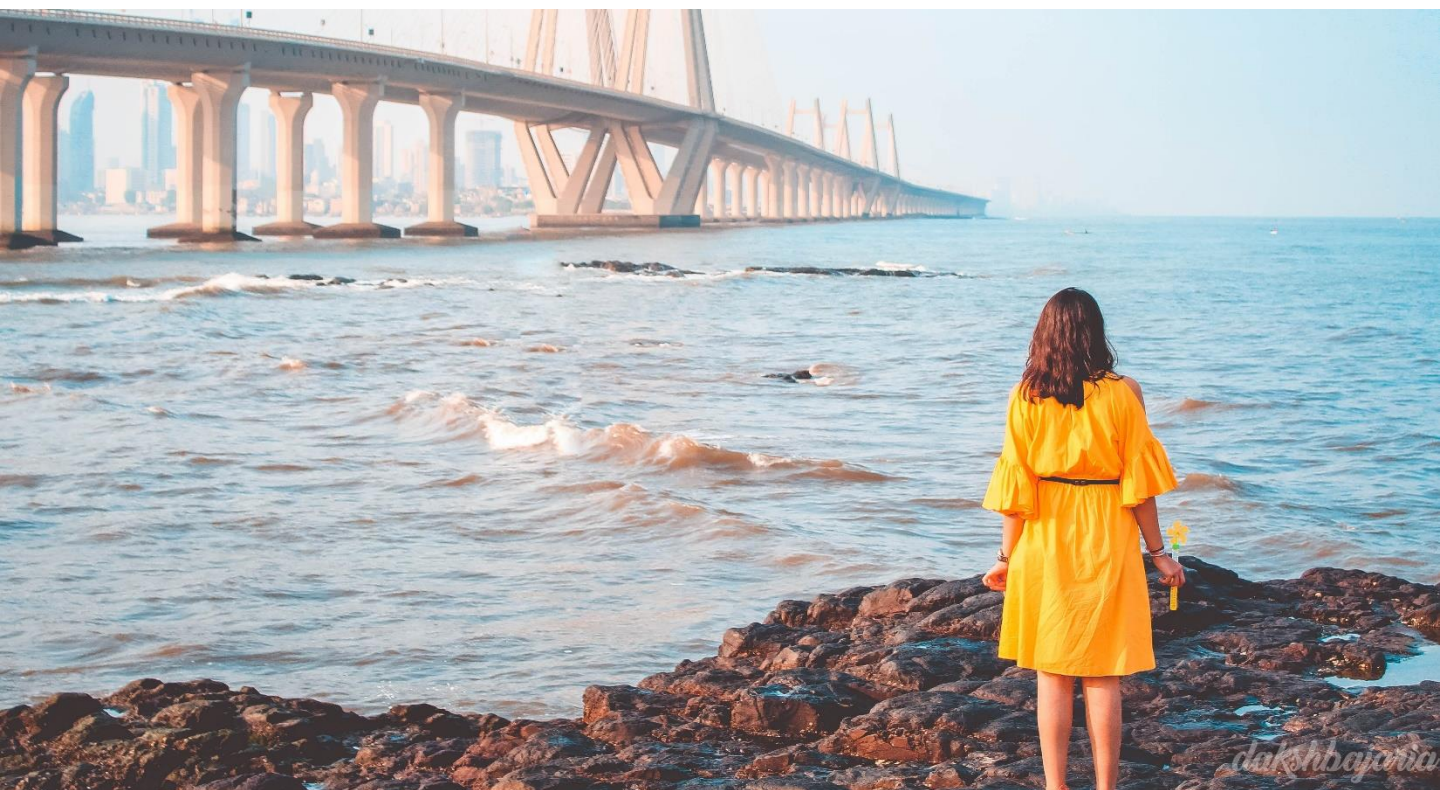
Właściciele i twórców sklepów wskazuje na kluczową rolę szybkiego otwierania stron.

Szybkość

Kiedy Twoja domena jest utrzymywana na DNS w systemie Anycast, to spośród wielu serwerów DNS o tym samym adresie IP (mamy ich obecnie 13 na 6 kontynentach), odpowie Ci ten, który jest w stanie odpowiedzieć najszybciej.

Ochrona

Rozproszony system DNS Anycast jest znacznie bardziej odporny na awarie i ataki. Napastnik musiałby nie tylko skutecznie zaatakować serwer DNS, ale też wszystkie pozostałe serwery DNS pod tym samym adresem IP, co czyni atak wielokrotnie trudniejszym.



>Kompresja gzip

Na czym polega gzip

Kompresja gzip polega na tym, że na serwerze dane są kompresowane zanim zostaną przesłane do przeglądarki użytkownika. Oznacza to, że objętość danych do przesłania jest mniejsza.

- Strona wczytuje się krócej.
- Łącze jest obciążone w mniejszym stopniu.

Trzeba jednak pamiętać, że skompresowanie danych wymaga wykonania określonej pracy przez procesor serwera. Nowoczesne procesory są bardzo wydajne i świetnie radzą sobie z kompresowaniem. Wg niemal wszystkich specjalistów na świecie – korzyści przeważają nad „ceną” związaną z wykorzystaniem niektórych cykli procesora na wykonanie kompresji.

Jeśli jednak Twoje procesory są obciążone w bardzo wysokim stopniu, to dodanie im dodatkowego zadania w postaci kompresji, może wpłynąć na ogólne spowolnienie systemu. Jest to dość ekstremalna sytuacja, w realnym życiu niemal zawsze korzyści są większe.

Typy plików do kompresji

Największe korzyści odniesiesz z kompresji plików tekstowych. Tego rodzaju pliki mogą zmniejszyć swoją objętość nawet o **70-80%**, a czas przesyłania danych może się skrócić o **15-20%**. Nie polecam natomiast kompresji plików z mediami, ponieważ są one w większości kompresowane wewnątrz algorytmami specjalistycznymi (np. jpeg, mp3 itp.)

Uwaga

Używanie kompresji gzip w połączeniu z HTTPS/SPDY/HTTP/2 może rodzić zagrożenia w postaci ataków BREACH . Atak ten pozwala na ujawnienie poufnych informacji w razie gzipowania. Można się przed nim ochronić blokując tzw. żądania cross-site. Po więcej informacji zajrzyj tutaj:

- <https://en.wikipedia.org/wiki/BREACH>
- <https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>



Przykład sterowania kompresją w pliku .htaccess - jeśli istnieje moduł kompresji i plik spełnia określone kryteria typu – kompresja zostaje włączona. Serwer może mieć domyślnie włączoną kompresję dla różnych plików - dowiedz się w firmie hostingowej, jak to wygląda w Twoim wypadku. W taki sposób możesz ręcznie włączyć kompresję:

```
<IfModule mod_deflate.c>
<filesMatch "\.(js|css|html|txt)$">
SetOutputFilter DEFLATE
</filesMatch>
</IfModule>
```

> Technologie dyskowe

Stosowane technologie dyskowe mogą mieć wpływ na szybkość otwierania stron, jednak nie jest to wpływ bezpośredni. Część osób, wybierając hosting, zwraca uwagę, czy jest on na dyskach SSD, albo wręcz NVMe (choć to też SSD, tylko podłączone bezpośrednio do magistrali PCIe, a nie poprzez starszą technologię SATA3.

W praktyce, moim zdaniem, nie warto nadmiernie ekscytować się technologią dyskową. Może mieć ona znaczenie tylko dla kilku procent największych stron internetowych, albo – jeśli masz szczególne potrzeby w zakresie objętości składowanych danych.

	HDD SATA	HDD SAS	SSD SATA	SSD NVMe
Postać fizyczna	Metalowy prostopadłościan, zawierający dysk o średnicy 3.5 cala lub 2.5 cala.	Metalowy prostopadłościan, zawierający dysk o średnicy 3.5 cala lub 2.5 cala.	Metalowy prostopadłościan prawie zawsze odpowiadający dyskom 2.5 cala, ale bez ruchomych części wewnątrz.	Mała karta, zawierająca zazwyczaj kontroler i układy pamięci, nieco przypominająca moduł RAM.
Ma części ruchome	Tak, dyski wirujące z prędkościami 5400 obr/min do 10 000 obr/min	Tak, dyski wirujące z prędkością obrotową do 15.000 obr/min	Nie	Nie
Pojemność	1-12 TB	0,3-12 TB	0,5-4 TB	0,5-2 TB
Cena za GB	niska	średnia / wysoka	średnia	bardzo wysoka
Transfer danych	Ok. 100 mbps	Ok. 150 mbps	Ok. 500 mbps	1500-5000 mbps
IOPS (operacje na sekundę)	80	150	50.000-100.000	100.000-300.000

W profesjonalnych serwerach bardzo często są stosowane wydajne macierze i kontrolery cache'ujące często używane dane w szybkiej pamięci podręcznej. W zaawansowanych rozwiązaniach są one w stanie cache'ować zarówno zapisy, jak i odczyty danych (pamięć cache jest nieulotna lub podtrzymywana bateryjnie).

Oznacza to, że nawet na pozornie mniej wydajnej technologii strona pod wysokim obciążeniem wciąż może otwierać się bardzo szybko.

Zatem...

W radykalnej większości zastosowań rodzaje dysków są mniej ważne, ważniejszy jest efekt ich pracy. Dyski mogą być wąskim gardłem przede wszystkim podczas obsługi baz danych, zamiast zaprzętać sobie głowę, czy „pod spodem” jest dysk SATA czy NVMe – po prostu wykonaj testy wydajnościowe baz danych.

> Technologie macierzowe

Dyski twarde, na których są składowane Twoje dane, są przechowywane w macierzach. Innymi słowy, dane nie są zapisane na dysku, ale raczej na zespole dysków, pracujących ze sobą w określony sposób, zwany często poziomem RAID. Ten sposób może mieć znaczenie dla osiągniętej wydajności.

Mimo kuszącej efektywności w zakresie wykorzystania przestrzeni w trybach z liczeniem sum kontrolnych, jak RAID 5 czy RAID 6 – w środowiskach produkcyjnych firm hostingowych, które miałem okazję obserwować „od środka” dominował RAID 10, jako mechanizm łączący wydajność, bezpieczeństwo i względnie krótki czas replikacji danych w razie awarii dysku.

	RAID 0	RAID 1	RAID 10	RAID 5
Pokróctce...	Dane są zapisywane naprzemiennie na poszczególne nośniki. To maksymalizuje wydajność, ale w razie awarii choćby jednego dysku - dane są niemożliwe do odczytania.	Dane są jednocześnie zapisywane na nośniki tworzące macierz. Czasem ten tryb nazywa się „mirror” – bo operacje wykonywane na jednym nośniku są w tym samym momencie odwzorowane na drugim. System jest odporny na awarię dysku.	Budowany na min. 4 dyskach, łączy najlepsze cechy RAID 0 i RAID 10. Dyski są łączone w pary i wewnątrz tych par mirrorowane (odporność na uszkodzenie dysku), a pary są łączone na zasadzie RAID 0, co daje znaczne przyspieszenie operacji.	Wymaga min 3 dysków. 1/n (n-liczba dysków) przestrzeni każdego dysku zajmują tzw. sumy kontrolne. Dane są dystrybuowane na wszystkich nośnikach, ale zawsze towarzyszy temu zapisanie sum kontrolnych, służących odbudowaniu danych w razie awarii nośnika.
Wydajność	bardzo wysoka	średnia	wysoka	Średnia
Efektywność wykorzystania przestrzeni	bardzo wysoka	Niska	Średnia	wysoka
Czas odzyskiwania danych	brak możliwości odzyskania	krótki	krótki	Długi (konieczność)



Ciekawostka: BBWC

Battery Backed-up Write Cache, to cache oparty o szybką, ale ulotną pamięć. Dane trafiają najpierw do kontrolera, a później, w sposób możliwie najbardziej wydajny, są zapisywane na dysku. Co stałoby się jednak, gdyby między momentem przesłania danych do pamięci cache, a momentem zapisu na nieulotnym nośniku doszło do awarii zasilania? Żeby uniknąć utraty danych profesjonalne kontrolery często mają wewnętrzne zasilanie bateryjne, chroniące pamięć cache w takich sytuacjach.

>cyber_Boost_

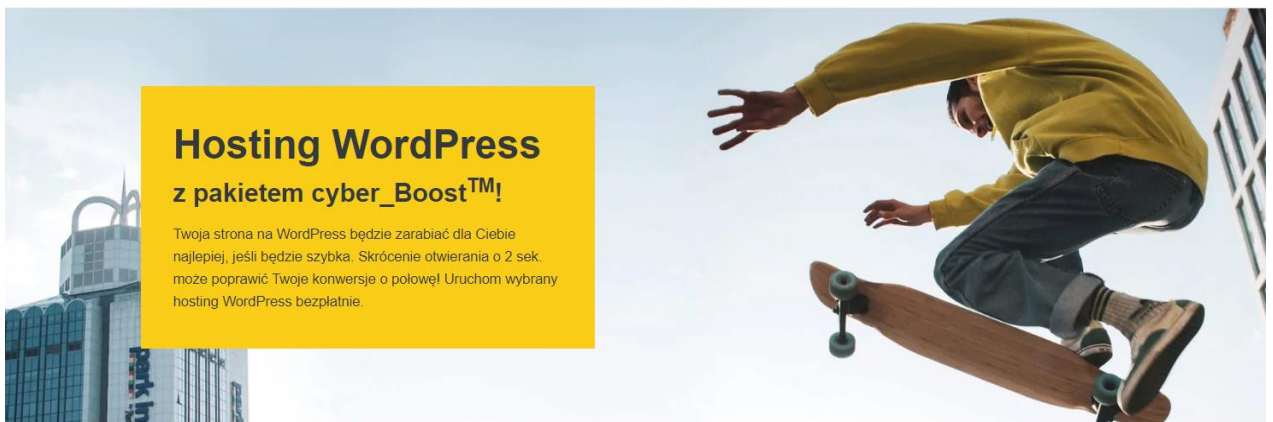
Wykorzystując spostrzeżenia z tego eBooka, wspólnie z blisko 300 deweloperami, opracowaliśmy pakiet technologiczny cyber_Boost.

Pakiet ten oznacza po prostu zbiór technologii przyspieszających, które znasz już z tego eBooka. Ten pakiet technologiczny jest oferowany w wypadku zarówno hostingu dedykowanego dla WordPress, jak i dla PrestaShop

W skład technologii wchodzi:

- Szybkie dyski NVMe
- Serwer LiteSpeed z pluginem LS Cache
- Wsparcie dla HTTP/2
- DNS Anycast z szybkimi odpowiedziami DNS
- Wtyczka optymalizująca obrazy, ustawienia i bazę w wypadku PrestaShop.

cyber_Folks™ DOMENY ▾ HOSTING ▾ E-MARKETING ▾ INSPIRACJE ▾ ZALOGUJ SIĘ ▾



Hosting WordPress z pakietem cyber_Boost™!

Twoja strona na WordPress będzie zarabiać dla Ciebie najlepiej, jeśli będzie szybka. Skrócenie otwierania o 2 sek może poprawić Twoje konwersje o połowę! Uruchom wybrany hosting WordPress bezpłatnie.

14 szybkich dni z hostingiem WordPress

Aktywuj bezpłatnie i używaj nawet dwa tygodnie.
Opłacisz proformę tylko, kiedy poczujesz pełną satysfakcję z dokonanego wyboru.

WYBÓR KLIENTA			
>wp_INI! Do nauki dla uczniów / studentów	>wp_UP! Dla mikrofirmy / blogera	>wp_GO! Dla firmy / sklepu	>wp_FLY! Dla większej firmy / webdevelopera
2 GB	5 GB	20 GB	50 GB
WITRYNY 1	WITRYNY 5	WITRYNY 10	WITRYNY 25
TRANSFER 250 GB	TRANSFER 500 GB	TRANSFER bez limitu	TRANSFER bez limitu
KONTA EMAIL 2	KONTA EMAIL 15	KONTA EMAIL bez limitu	KONTA EMAIL bez limitu

ZOBACZ SZYBKI HOSTING WORDPRESS

>Checklista

Na koniec proponuję Ci checklistę, która ułatwi Ci proces oceny firmy hostingowej pod kątem wydajnościowym. Nie wszystkie kryteria będą równie istotne dla każdego, pamiętaj też, że wydajność to tylko JEDNO z kryteriów wyboru operatora. Na całość zadowolenia z decyzji będą wpływać też inne elementy, jak obsługa klienta, niezawodność,, funkcjonalność etc.



<input type="checkbox"/>	Firma hostingowa publikuje informacje o limitach bezpieczeństwa lub gwarantuje, że takowe nigdy nie wystąpią.
<input type="checkbox"/>	Liczba dopuszczanych wysyłek maili na godzinę jest sprawdzona i znana i wystarczająca dla Twoich zastosowań.
<input type="checkbox"/>	Operator udostępnia kilka wersji PHP.
<input type="checkbox"/>	Wykonano skrypt PHP Benchmark na kilku hostingach i porównano wyniki.
<input type="checkbox"/>	Wydajność PHP mierzona o różnych porach doby jest stabilna dla wybranego wariantu.
<input type="checkbox"/>	Testy PHP wskazują, że wydajność jest na poziomie średnim lub wysokim w porównaniu z innymi hostingami dla tej samej wersji PHP.
<input type="checkbox"/>	Zmiana między wersjami PHP w panelu jest możliwa w prosty sposób.
<input type="checkbox"/>	Operator udostępnia mechanizm OPcache
<input type="checkbox"/>	Mechanizm OPcache można łatwo włączać i wyłączać w panelu.
<input type="checkbox"/>	Operator udostępnia serwer z pluginem LS Cache (dodatkowy atut)
<input type="checkbox"/>	Wykonano testy wydajności MySQL dla rozważanych hostingów.
<input type="checkbox"/>	Testy wydajności MySQL wskazują na satysfakcjonujący poziom wydajności przy kluczowych rodzajach spodziewanych obciążeń.
<input type="checkbox"/>	Firma hostingowa udostępnia serwer Memcached/Redis
<input type="checkbox"/>	Wybrany hosting wspiera HTTP/2
<input type="checkbox"/>	Twoja domena jest/będzie utrzymywana na DNS chronionych systemem DNS Anycast.
<input type="checkbox"/>	Operator udostępnia dodatkowe, autorskie mechanizmy przyspieszające działanie niektórych aplikacji (np. WordPress, PrestaShop)
<input type="checkbox"/>	Przeprowadzono testy wydajnościowe (np. Siege albo Apache Bench) i wyniki są satysfakcjonujące.
<input type="checkbox"/>	Serwer udostępnia kompresję gzip (tzw. mod_deflate)
<input type="checkbox"/>	Prawidłowo skonfigurowano plik .htaccess włączając kompresję dla właściwych typów plików lub świadomie tego zaniechano.
<input type="checkbox"/>	Technologia dyskowa jest znana i uznana za satysfakcjonującą.
<input type="checkbox"/>	Macierze RAID są zoptymalizowane tak, żeby w razie awarii dysku ograniczać czas niezbędny na odbudowę danych.



Chcesz więcej?

Czytasz właśnie ostatnią stronę. Mam więc nadzieję, że Ci się podobało. Jeśli znalazłeś tu choć jedną inspirującą wskazówkę – powiedz o tym innym, może te techniki przydadzą się także Twoim znajomym.

Po więcej dobrych treści dla twórców i właścicieli stron internetowych zapraszam Cię na naszego bloga:

<https://cyberfolks.pl/blog/>



Zapraszam Cię także do osobistego kontaktu na LinkedIn. Każda opinia będzie dla mnie potężnym zastrzykiem motywacyjnym do kolejnych publikacji.

<https://www.linkedin.com/in/arturpajkert/>

