



COMO
ACELERAR
O SEU SITE
WORDPRESS

Ultimate
Guide

Brian Jackson × **KINSTA**



Digitalize o código para ver a versão sempre atualizada deste guia!

<https://kinsta.com/speed-up-wordpress>

Publicado por Kinsta

Desenho de livro por Maja Szakadat



Sobre o Autor - Brian Jackson

Brian é o diretor de marketing da Kinsta. Ele se concentra em tudo, desde o desenvolvimento de novas estratégias de crescimento online, criação de conteúdo, SEO e divulgação na comunidade. Ele gosta muito de escrever artigos longos e publicou mais de 2.500 posts no blog.

Uma das maiores paixões de Brian é o WordPress, que ele usa há mais de uma década. Tendo trabalhado tanto no setor de rede de distribuição de conteúdo quanto no setor de hospedagem de alto desempenho, ele é especialista em otimização de desempenho da Web e em formas estratégicas de acelerar sites.

Como Acelerar o Seu Site WordPress

Capítulos

08

Tipos de Sites do
WordPress: Estático ou
Dinâmico

11

Escolha uma Hospedagem
WordPress de Alto
Desempenho

23

Escolha um Servidor
mais Próximo aos Seus
Visitantes

32

DNS Premium é Melhor
que DNS Gratuito

36

O Tema do Seu
WordPress é Importante

45

Fatos Sobre Plugins
do WordPress

50

Ótimas Configurações do
WordPress

56

Por Que o Cache é
Tão Importante

66

Otimização de Imagem é
Uma Obrigação

75

Ajuste o seu
Banco de Dados

93

Use uma Rede de Entrega
de Conteúdo (CDN)

103

Descarregue Mídia
e E-mail Quando
Necessário

108

Como Encontrar Restrições
e Plugins Lentos

127

Recomendações Sobre
Otimização do Back-end

141

Dicas Sobre Otimização
do Front-end

168

Otimização da Versão
Mobile em Mente

Publicamos muitos tutoriais ao longo dos anos com maneiras de otimizar e acelerar o WordPress. Mas às vezes pode ser confuso tentar encontrar tudo o que você precisa em um só lugar. Então, hoje vamos compartilhar com você tudo o que sabemos sobre turbinar o WordPress, mais de 15 anos de experiência e lições difíceis aprendidas, tudo em um guia final. Se você está apenas começando a usar o WordPress ou é um desenvolvedor experiente, prometemos que encontrará algo de útil neste post.

Mais de um terço da web agora é alimentado pelo WordPress. Embora isso seja incrível, também significa que existem milhares de diferentes temas, plugins e tecnologias, todos tendo que coexistir. Para o usuário diário do WordPress, isso pode se transformar rapidamente em um pesadelo quando o site começa a afunilar e eles não sabem por que ou até onde começar a solução de problemas.

Hoje, vamos nos aprofundar nas etapas aplicáveis que você pode realizar agora para ver melhorias em seus próprios sites do WordPress. Também compartilharemos alguns recursos inestimáveis para nós.

CAPÍTULO 01:

Tipos de Sites do WordPress: Estático ou Dinâmico



Antes de nos aprofundarmos nas otimizações, é importante entender primeiro que nem todos os sites do WordPress são iguais. É por isso que muitos usuários têm problemas, pois você não pode lidar com todos os problemas da mesma maneira. Nós sempre damos aos sites do WordPress uma classificação: estática ou dinâmica. Então, primeiro vamos explorar as diferenças entre esses dois tipos de sites.

Principais Sites Estáticos

Sites estáticos normalmente inclui sites como blogs, sites de pequenas empresas, sites de notícias de menor volume, pessoais, fotografia etc. Por estática, queremos dizer que os dados nesses sites do WordPress não estão mudando com muita frequência (talvez algumas vezes por dia). Mesmo a maioria do nosso site Kinsta seria considerado um site estático.

Isso se torna incrivelmente importante, pois muitas das solicitações podem ser atendidas diretamente do cache no servidor em velocidades extremamente altas! Não se preocupe; vamos mergulhar no tópico do cache mais abaixo. Isso significa que eles terão menos chamadas de banco de dados e não serão necessários tantos recursos para atingir o desempenho do Google.

Sites Altamente Dinâmicos

Por outro lado, temos sites altamente dinâmicos. Estes incluem sites como eCommerce (WooCommerce ou Easy Digital Downloads), comunidade, membros, fóruns (bbPress ou BuddyPress) e sistemas de gerenciamento de aprendizado (LMS). Por dinâmico, queremos dizer que os dados nesses sites do WordPress estão mudando frequentemente (as transações do servidor estão ocorrendo a cada poucos minutos ou até mesmo a cada segundo). Isso significa que nem todas as solicitações para o servidor podem ser fornecidas diretamente do cache e requerem recursos adicionais do servidor e consultas de banco de dados.

Esses sites também costumam ter um grande número de visitantes e sessões simultâneas. Em um site WordPress informativo ou corporativo que é basicamente estático, um visitante pode permanecer por cinco ou dez minutos até encontrar o que precisa (e esse é um número alto, geralmente as taxas de rejeição são muito mais altas). Em sites dinâmicos, você tem o oposto acontecendo. Normalmente, os visitantes chegam ao site para se envolver com algo ou alguém. Se eles estão participando de um curso online, não é incomum que fiquem por horas.

Você pode ver onde isso está indo. Os visitantes simultâneos conectados ao seu host do WordPress são adicionados rapidamente. Para piorar, você tem um grande número de visitantes simultâneos em cima de um problema de “conteúdo sem cache”.

Você não pode tratar todos os sites do WordPress da mesma forma quando se trata de desempenho. Sites estáticos e altamente dinâmicos são dois animais muito diferentes.

CAPÍTULO 02:

Escolha uma Hospedagem WordPress de Alto Desempenho



Um host do WordPress é uma empresa que armazena todos os dados do seu website. Você se inscreve em um plano e todas as suas imagens, conteúdo, vídeos, etc., residem em um servidor localizado no data center do host. O host do WordPress oferece uma maneira fácil de acessar os dados, gerenciá-los e direcioná-los aos seus visitantes. Muito simples, certo? Bem, não é bem assim.

Existem três tipos muito diferentes de hosts do WordPress que você encontra na web. Vamos mergulhar nos prós e contras de cada um. É importante escolher o caminho certo desde o início, caso contrário, você simplesmente só causará dores de cabeça e perderá tempo na estrada.

1. Hospedagem Compartilhada do WordPress

O primeiro e mais popular tipo de hospedagem WordPress é o que chamamos de “hospedagem compartilhada”. Estes incluem os maiores hosts do setor, como empresas EIG como Bluehost e HostGator, além de provedores como Siteground, GoDaddy e InMotion Hosting. Eles geralmente utilizam o cPanel, e o cliente médio geralmente paga entre US\$ 3 e US\$ 25 por mês.

Qualquer pessoa que use esse tipo de hospedagem terá, em algum momento, lentidão, é apenas uma questão de tempo. Por quê? Como os hosts compartilhados tendem a sobrecarregar seus servidores, o que, por sua vez, pode afetar o desempenho de seu

site. Suspensões de site ou erros freqüentes de 500 são coisas comuns que você vai experimentar, pois eles têm que colocar limites em tudo e consolidar recursos para sobreviver. Ou pior ainda, o tempo de inatividade do site. Mesmo que você não saiba, seu site WordPress provavelmente está no mesmo servidor de mais de 200 pessoas. Qualquer problema que surja com outros sites pode chegar ao seu site.



Shared WordPress hosting

Não importa como você faz as contas, após as despesas, US\$ 3,00 por mês não geram receita para a empresa de hospedagem. Especialmente quando você atribui apoio a isso. Um ticket de suporte e eles já estão no vermelho. A maneira que eles ganham muito dinheiro está em upselling e taxas escondidas. Esses upsells incluem coisas como migrações, registros de domínio, certificados SSL, etc. Outra tática comum é fornecer grandes descontos de inscrição. Mas uma vez que a renovação chega, você recebe a conta real.

Atendimento ao cliente e suporte com a hospedagem compartilhada são quase sempre abaixo do esperado devido ao grande volume de sites versus representantes de suporte. Os hosts compartilhados precisam se espalhar muito pouco para obter lucro, o que geralmente leva a uma experiência desagradável para o cliente.

Quando se trata de hospedagem compartilhada, você geralmente recebe o que paga

2. Hospedagem VPS DIY WordPress


O segundo tipo de hospedagem do WordPress é o VPS DIY, ou “Faça você mesmo em um servidor privado virtual”. Esse público normalmente é composto de startups de bootstrap e usuários com um pouco mais de desenvolvimento, gerenciamento de servidor e experiência no WordPress. Eles são a multidão de DIY. Essas pessoas normalmente ainda estão tentando economizar, mas geralmente também estão preocupadas com o desempenho e percebem sua importância no sucesso de seus negócios. As configurações do Commons podem incluir o uso de um provedor de VPS de terceiros, como Digital Ocean, Linode ou Vultr; junto com uma ferramenta como o ServerPilot para gerenciá-lo mais facilmente.

Um pequeno VPS da DigitalOcean começa em US\$ 5 por mês e o popular plano no ServerPilot começa em US\$ 10 por mês. Então, dependendo da sua configuração, você pode estar olhando para um custo entre US\$ 5 a US\$ 15 ou mais por mês. A abordagem DIY pode reduzir custos, mas também significa que você é responsável se algo quebrar e para otimizar o desempenho do seu servidor.

A abordagem DIY “Faça Você Mesmo” pode ser ótima, mas também pode ser contraindicada se você não for cuidadoso. Não siga este caminho se você não é especialista em tecnologia ou apenas porque quer mexer! Seu tempo vale dinheiro e você deve gastá-lo no crescimento do seu negócio.

3. Hospedagem Gerenciada do WordPress

O terceiro tipo de hospedagem é o que oferecemos aqui na Kinsta, e que é gerenciado pelo WordPress. Esses tipos de hosts lidam com todas as tarefas relacionadas ao servidor de backend, além de fornecer suporte quando você precisar. Eles são normalmente ajustados para trabalhar com o WordPress e geralmente incluem recursos como ambientes de teste de um clique e backups automáticos. Suas equipes de suporte serão mais bem informadas quando se trata de conhecer o caminho do CMS, já que estão focadas em uma plataforma diariamente.



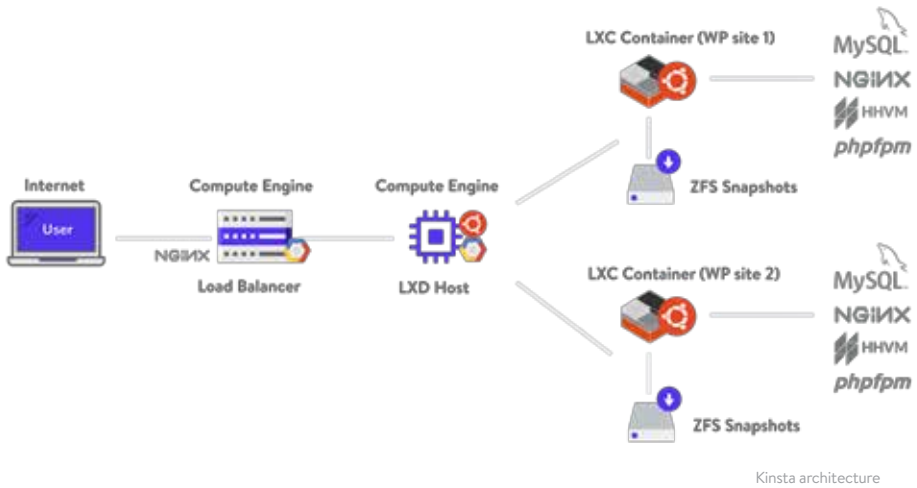
*Se você quiser economizar tempo,
hospedagem gerenciada do
WordPress é o caminho a percorrer!*

Os planos para hospedagem gerenciada do WordPress normalmente variam de US\$ 25 a US\$ 150 por mês ou mais, dependendo do tamanho do site e das necessidades. Grandes empresas como jQuery, Intuit, Plesk, Dyn, NGINX e até mesmo a Casa Branca estão usando o WordPress para hospedar seu site. Alguns hosts WordPress gerenciados populares com os quais você provavelmente está familiarizado, ou que talvez esteja usando no momento, incluem o WP Engine, Flywheel, Pressable, Media Temple, Pressidium e o Pagely.

Kinsta Adota Uma Abordagem Diferente

No entanto, Kinsta leva a hospedagem gerenciada do WordPress para o próximo nível. Nossa plataforma de hospedagem não se enquadra em nenhuma das categorias de hospedagem tradicionais. Toda a nossa infraestrutura é criada no Google Cloud Platform e é diferente da infraestrutura tradicional compartilhada, VPS ou dedicada.

Cada site WordPress em nossa plataforma é executado em um contêiner de software isolado que contém todos os recursos de software necessários para executar o site (Linux, NGINX, PHP, MySQL). Isso significa que o software que executa cada site é totalmente privado e não é compartilhado nem entre seus próprios sites.



Cada contêiner de site é executado em máquinas virtuais em um dos vários datacenters do GC. Cada máquina tem até 96 CPUs e centenas de GB de RAM. Os recursos de hardware (RAM/CPU) são alocados para cada contêiner de site automaticamente por nossas máquinas virtuais conforme a necessidade (um recurso interessante que chamamos de dimensionamento automático).

Todos os anos, a Review Signal lança seus benchmarks de desempenho de hospedagem WordPress, e estamos orgulhosos de que, por cinco anos consecutivos, a Kinsta provou ser a melhor companhia em todos os níveis! E não apenas em um ou dois dos nossos planos, mas em todos os planos, desde o Starter até o Enterprise.

A Kinsta tinha testes LoadStorm e Blitz essencialmente perfeitos. Eles também não tiveram falhas em nenhum outro teste. Não tenho palavras para elogiar o desempenho deles.



Kevin Ohashi
Fundador e consultor WP, ReviewSignal

Também não temos representantes de suporte de nível 1 ou nível 2. Toda a nossa equipe de suporte é formada por desenvolvedores do WordPress e engenheiros de hospedagem Linux, muitos dos quais gerenciaram seus próprios servidores, criam temas e plugins e

contribuem de volta ao núcleo. Isso garante que você receberá conselhos de especialistas de alguém que usa e desenvolve ativamente com o WordPress.

Você pode conversar com os mesmos membros da equipe de suporte que apóiam nossos clientes da Fortune 500 e clientes Enterprise. Somos tão exigentes com a qualidade de nossa equipe de suporte que contratamos apenas menos de 1% dos candidatos que se candidatam. Você não encontrará melhor suporte em nenhum outro lugar.

Para saber mais sobre por que você deve escolher Kinsta para hospedagem gerenciada do WordPress, acesse nosso site em [kinsta.com](https://www.kinsta.com). Mas, independentemente de quem você escolher como seu provedor de hospedagem, você deve sempre procurar os seguintes recursos de servidor para garantir que seu site seja executado o mais rápido possível.

Com o WP Engine, os problemas básicos geralmente são resolvidos rapidamente. No entanto, para quaisquer problemas que sejam complexos, uma resolução levará algum tempo e haverá muitas idas e vindas. Este é um problema quando você está executando um site WordPress high-end, e há um problema urgente que precisa ser tratado rapidamente. Se você está pedindo minha única recomendação entre os dois, na minha opinião, Kinsta é melhor. Eles oferecem muito mais do que prometem. Você nunca precisa se preocupar com a lentidão do site, o tempo de inatividade, a obtenção de suporte de qualidade ou qualquer outro problema relacionado à hospedagem.



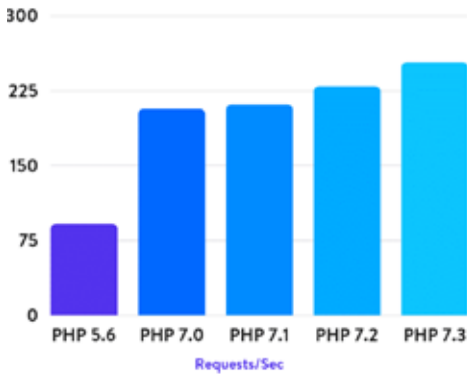
Harsh Agrawal
Blogger premiado, ShoutMeLoud

PHP 7 ou Superior Para o Melhor Desempenho

O PHP é uma linguagem de programação e script do lado do servidor, de código aberto, usada principalmente para o desenvolvimento da web. A maior parte do núcleo do software WordPress é escrita em PHP, juntamente com seus plugins e temas, o que torna o PHP uma linguagem muito importante para a comunidade WordPress. Você deve garantir que seu host WordPress ofereça pelo menos PHP 7 ou superior.

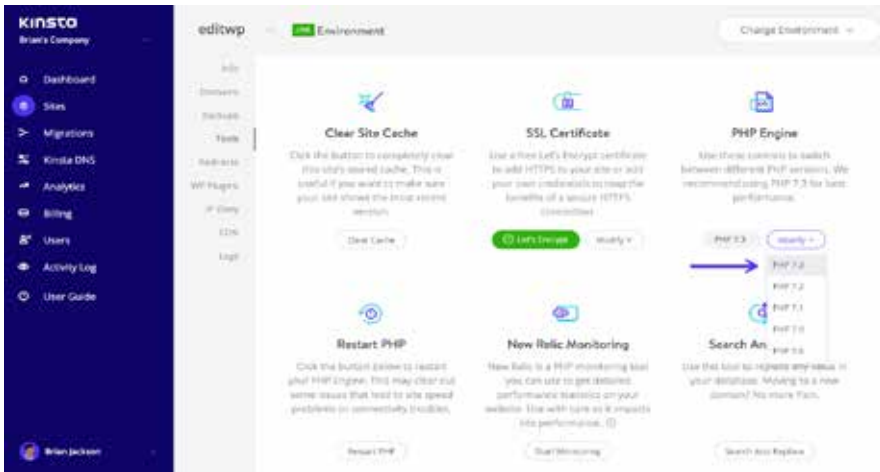
Existem diferentes versões do PHP que seu host irá fornecer a você em seu servidor, com o novo PHP 7.3 oferecendo grandes melhorias de desempenho.

WordPress 5.0



De fato, em nossos recentes testes de benchmarks do PHP, se você comparar o PHP 7.3 ao PHP 5.6, ele pode lidar com 3x mais pedidos (transações) por segundo! O PHP 7.3 também é, em média, 9% mais rápido que o PHP 7.2. Isso também pode afetar a capacidade de resposta do painel de administração do WordPress.

Velocidades mais rápidas e segurança aprimorada, é por isso que Kinsta sempre oferece as versões mais recentes do PHP. Você também pode alterar as versões do PHP com um único clique.



É desconfie de qualquer host do WordPress que ofereça o HHVM como uma alternativa ao PHP. O HHVM não é mais uma solução adequada para o WordPress, já que ele não suporta mais o PHP.

Escolha um Host que Use o NGINX

Nos bastidores, cada host do WordPress usa um servidor da web para alimentar seus sites do WordPress. As escolhas mais comuns são o NGINX e o Apache.

É altamente recomendável ir com um host que use o NGINX devido às suas raízes na otimização de desempenho em escala. O NGINX geralmente supera outros servidores populares da Web em testes de benchmark, especialmente em situações com conteúdo estático ou solicitações simultâneas altas, e é por isso que Kinsta usa NGINX.



Algumas empresas de alto perfil que usam NGINX incluem a Autodesk, Atlassian, Intuit, T-Mobile, GitLab, Microsoft, IBM, Google, Adobe, Salesforce, VMWare, Xerox, LinkedIn, Cisco, Facebook, Target, Sistemas Citrix, Twitter, Apple, Intel e muitos mais. (fonte: Sifter.com)

De acordo com a W3Techs, o Apache alimenta 44,0% de todos os sites, o que o torna a opção mais amplamente utilizada. Mas se você olhar para o servidor web mais popular entre os sites de alto tráfego (top 10.000), NGINX potencia 41,9% deles, enquanto Apache só potencia 18,1%. É usado por alguns dos sites mais intensivos em recursos existentes, incluindo Netflix, NASA, e até mesmo WordPress.com.

A Rede da sua Hospedagem é Importante

Ao escolher um host do WordPress, você pode nem pensar em perguntar ou pesquisar em qual rede eles estão usando, mas você deveria. A rede pode ter um enorme impacto no desempenho do seu site e até mesmo no snappiness do seu painel do WordPress. Muitos hosts deixarão isso de lado, pois optarão pela rede mais barata para reduzir custos.

Aqui estão algumas perguntas que você deve fazer:

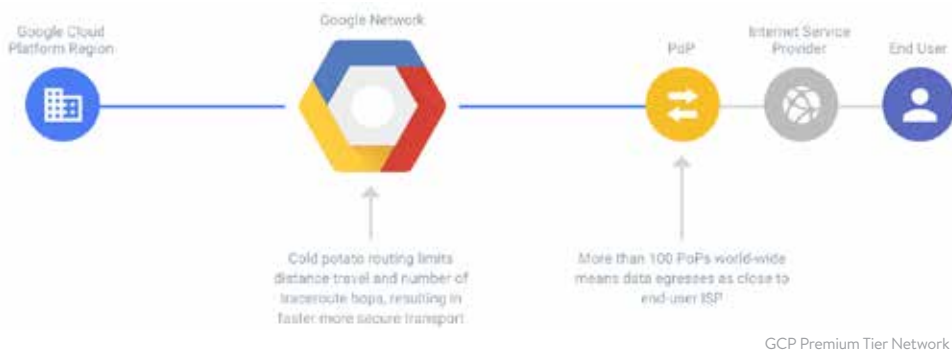
- **Em quais redes você está transmitindo dados?** É a maioria sobre redes públicas de ISP ou infraestruturas privadas, como Google ou Microsoft? Esses grandes provedores possuem redes que são construídas e otimizadas para baixa latência e velocidade. Eles ainda têm seus próprios cabos de internet sob o oceano!

- **As redes que você está usando são redundantes?** O que acontece se um cabo for cortado acidentalmente? Isso acontece com mais frequência do que você pensa.

Em 2017, o Google anunciou sua rede de nível padrão, que é uma rede mais lenta, mas com um custo mais barato. Na Kinsta, utilizamos sua rede de **nível premium** para todos os nossos planos de hospedagem. Embora isso seja um custo extra para nós, isso garante que você obtenha velocidades extremamente rápidas.

Kinsta usa a rede de nível premium do Google Cloud Platform. Porque seus sites merecem o melhor

Segundo o Google, a rede de nível premium consegue melhorar o desempenho da rede reduzindo a duração das viagens na Internet pública; os pacotes entram (e saem) da rede do Google o mais próximo possível do usuário e, em seguida, viajam no backbone do Google antes de chegar à VM. A camada padrão fornece tráfego de saída do GCP para a Internet em redes de transporte público (ISP) em vez da rede do Google.



Para colocar de outra forma que pode ser mais fácil de entender:

- **Os pacotes de nível premium passam mais tempo na rede do Google**, com menos tráfego, e portanto, apresentam melhor desempenho (mas custam mais).
- Os pacotes de nível padrão gastam menos tempo na rede do Google e mais

tempo lidando com problemas em redes públicas, e portanto, apresentam desempenho pior (mas custam menos).

Quanto impacto isso tem? Bem, para dados que viajam por todos os continentes, sua rede de nível premium é em média, **41% mais rápida** que a rede de nível padrão. Para dados que viajam para uma região próxima (mesmo continente), o nível premium é cerca de **8% mais rápido**. Embora a rede apenas represente uma fração do seu tempo total de carregamento da página, cada milésimo de segundo é adicionado!

Para os dados que viajam pelos continentes, a rede premium do Google Cloud é em média 41% mais rápida!

A redundância também é fundamental, e é por isso que o Google usa pelo menos três caminhos independentes (redundância N+2) entre dois locais da rede do Google, ajudando a garantir que o tráfego continue fluindo entre os locais, mesmo em caso de interrupção.

Como você provavelmente pode dizer agora, muita coisa acontece nos bastidores quando se trata de redes. Certifique-se de que seu host do WordPress esteja usando uma reputação e não esteja optando por reduzir os custos para as camadas inferiores.

É essencial ter HTTP/2

O HTTP/2 é um protocolo da web lançado em 2015 que foi desenvolvido para acelerar a forma como os sites são entregues. Por causa do suporte ao navegador, ele requer HTTPS (SSL). Se seu host do WordPress não for compatível com HTTP/2, você deve começar a procurar um novo provedor. Com o movimento de toda a web para HTTPS, isso não é mais apenas um bom recurso para ter; é uma necessidade.

A melhoria no desempenho com o HTTP/2 deve-se a vários motivos, como suporte a multiplexação melhor, paralelismo, compressão HPACK com codificação Huffman, a

extensão ALPN e push de servidor. Costumava haver um pouco de sobrecarga de TLS quando se tratava de executar HTTPS, mas isso agora é muito menos graças ao HTTP/2 e ao TLS 1.3.

Outra grande vitória com o HTTP/2 é que, na maioria dos sites WordPress, você não precisa mais se preocupar com a concatenação (combinação de arquivos) ou com o sharding de domínio. Estas são agora otimizações obsoletas. E o HTTP/3 é o próximo!

CAPÍTULO 03:

Escolha um Servidor Mais Próximo aos Seus Visitantes



Uma das primeiras coisas que você deve fazer ao hospedar seu site WordPress é determinar de onde vem a maioria de seus visitantes ou clientes. Por que isso é importante? Porque o local em que você hospeda seu website desempenha um fator significativo na determinação da latência geral da rede e do TTFB. Ele também afeta suas velocidades de SFTP e a capacidade de resposta do painel administrativo do WordPress.

Latência de rede: Refere-se ao tempo e/ou atraso envolvido na transmissão de dados em uma rede. Em outras palavras, quanto tempo leva para um pacote de dados ir de um ponto a outro. Hoje em dia isso é tipicamente medido em milissegundos; no entanto, pode ser segundos dependendo da rede. Quanto mais próximo de zero melhor.

Em outras palavras, quanto maior a distância, maior a latência, o que equivale a um atraso maior. Você não pode superar a velocidade da luz quando se trata de dados transmitidos por cabos de fibra óptica. Por exemplo, abaixo está um mapa da rede do Google Cloud Platform. Eles têm milhares de quilômetros de cabos indo por baixo do oceano que abrangem todo o globo. Esta é uma das razões pelas quais é uma das redes mais rápidas do mundo. E mesmo com essa extensa rede de ponta, ainda haverá atrasos de latência.

Sua empresa se beneficiaria ao colocar seu site em um servidor na Austrália, em vez de hospedá-lo na Europa ou nos Estados Unidos.

TTFB: Isso significa tempo para o primeiro byte. Para simplificar, esta é uma medida de quanto tempo o navegador deve aguardar antes de receber seu primeiro byte de dados do servidor. Quanto mais tempo demora para obter esses dados, mais tempo leva para exibir sua página. Mais uma vez, quanto mais próximo de zero melhor.

Um equívoco comum é que isso é calculado após os tempos de pesquisa de DNS, no entanto, o cálculo original do TTFB na rede sempre inclui a latência da rede. Isso envolve um processo de 3 etapas e atrasos e latência podem ocorrer em qualquer ponto intermediário, somando-se ao TTFB total.



1. Pedido Para o Servidor: Quando alguém visita seu site, a primeira coisa que acontece é que uma solicitação HTTP é enviada do cliente (navegador) para o servidor. Nesta etapa, há vários fatores que podem introduzir atrasos. Os tempos de pesquisa de DNS lentos podem contribuir para aumentar o tempo da solicitação. Se o servidor estiver localizado geograficamente longe, isso pode introduzir latência na distância que os dados precisam percorrer. Além disso, se você tiver regras de firewall complexas, isso poderá aumentar o tempo de roteamento. E não esqueça a velocidade da internet do cliente.

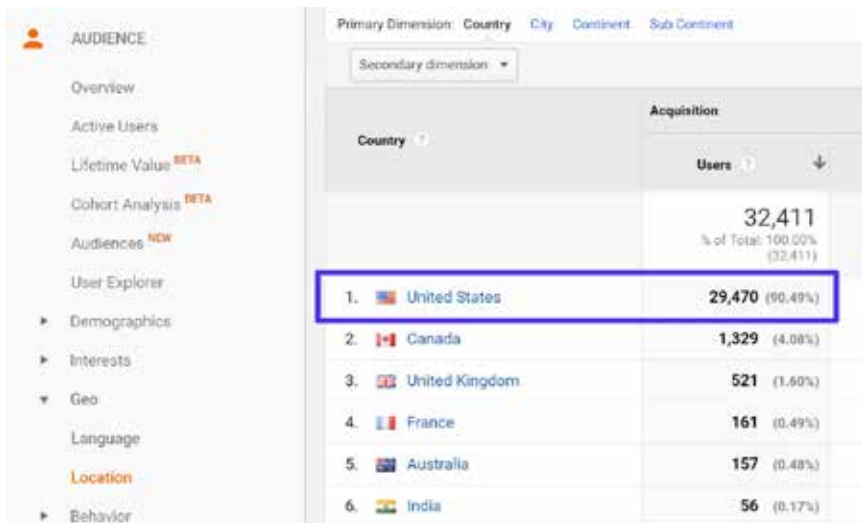
2. Processamento de Servidores: Depois que a solicitação foi enviada, o servidor agora precisa processá-la e gerar uma resposta. Isso poderia introduzir vários atrasos diferentes, como chamadas de banco de dados lentas, muitos scripts de terceiros, não armazenar em cache sua primeira resposta, código mal otimizado ou tema WordPress e recursos de servidor ineficientes, como E/S de disco ou memória.

3. Resposta ao Cliente: Depois que o servidor processa a solicitação, ele deve enviá-lo de volta ao cliente (ou, em vez disso, enviar de volta o primeiro byte). Isso é muito afetado pela velocidade da rede do servidor e do cliente. Se o cliente tiver internet lenta a partir de um hotspot Wi-Fi, ele será refletido no TTFB.

Geralmente, **qualquer coisa abaixo de 100 ms é ótima e boa TTFB.** O Google PageSpeed Insights recomenda menos de 200 ms para o tempo de resposta do servidor. Se você estiver na faixa de 300 a 500 ms, isso é bastante normal. E se você tiver mais de 600 ms, poderá ter algo configurado incorretamente em seu servidor ou talvez seja hora de atualizar para uma pilha da web melhor. Ou siga nossas recomendações abaixo sobre como reduzir seu TTFB. E lembre-se de que a negociação SSL / TLS também pode ser um fator.

Não vamos aborrecer você com todos os detalhes técnicos. Tudo o que **você precisa saber é que deseja que a latência de sua rede e o TTFB sejam os mais baixos possíveis.** Uma das maneiras mais fáceis de conseguir isso é escolher um servidor mais próximo de seus visitantes. Você pode determinar a melhor localização seguindo as dicas abaixo.

Dica 1 – Verificar a Geolocalização dos Seus Visitantes no Google Analytics



Uma das primeiras coisas que você pode fazer é olhar para a geolocalização de seus visitantes no Google Analytics. Você pode encontrar isso em “Público-alvo → Localização geográfica”.

Neste exemplo abaixo, você pode ver que mais de 90% do tráfego é proveniente dos Estados Unidos. Então, na maioria dos casos, você gostaria de colocar seu site WordPress em um servidor nos Estados Unidos. Você também pode filtrar os dados ainda mais para as cidades. Isso é especialmente importante se você for uma empresa local. Mas normalmente nós recomendamos uma localização central como Iowa, EUA.

Dica 2 – Verificar Dados de Comércio Eletrônico

Se você administra uma loja de comércio eletrônico, verifique também de onde seus clientes estão vindo. É claro que você gera receita, então esses são seus visitantes mais importantes. Isso deve coincidir com seu tráfego acima; no entanto, isso nem sempre é o caso. Se você tiver uma configuração ou metas de dados de comércio eletrônico no Google Analytics, poderá facilmente sobrepor essas informações aos dados de geolocalização para tomar uma decisão mais fundamentada. Ou verifique as informações de localização armazenadas no banco de dados da sua plataforma de comércio eletrônico.

Dica 3 – Faça um Teste de Latência Rápida

Há muitas ferramentas gratuitas úteis para medir a latência de sua localização atual para diferentes provedores de nuvem. Isso pode ajudar você a avaliar rapidamente qual região pode ser a melhor opção para seu site.

- **gcping.com:** (medida de latência para as regiões do Google Cloud Platform, incluindo os servidores Kinsta)
- **cloudping.info:** (medida de latência para regiões da Amazon Web Services)
- **azurespeed.com:** (medida de latência para regiões do Azure)

Neste exemplo abaixo, podemos ver que o Los Angeles, EUA (us-west2) é o mais rápido de onde estamos localizados. No entanto, se você estiver atendendo a clientes em todos os Estados Unidos, talvez seja melhor escolher Iowa, EUA (us-central1) para garantir baixa latência para visitantes da costa oeste e leste.

Meça a sua latência para as regiões GCP

| REGION | MEDIAN LATENCY |
|-------------------------------------|----------------|
| Los Angeles, USA us-west2 | 28ms |
| Global HTTP Load Balancer global | 31ms |
| Oregon, USA us-west1 | 53ms |
| Northern Virginia, USA us-east4 | 82ms |
| Iowa, USA us-central1 | 83ms |
| South Carolina, USA us-east1 | 88ms |

Aqui na Kinsta, oferecemos 23 data centers diferentes em todo o mundo. Você pode facilmente escolher um site que tenha baixa latência e baixo TTFB! Isso também ajuda a reduzir o salto de rede.



Formas Adicionais para Reduzir a Latência e o TTFB

Além de escolher um local de servidor próximo, aqui estão algumas outras maneiras de reduzir a latência.

- Implemente o cache no seu site WordPress. Em nossos testes, o cache reduziu nosso TTFB em impressionantes 90%!
- Utilize uma rede de entrega de conteúdo (CDN) para veicular ativos em cache de POPs em todo o mundo. Isso ajuda a negar a latência da rede para visitantes que podem não estar próximos de seu servidor host.
- Aproveite o protocolo HTTP/2 para minimizar o número de viagens de ida e volta, graças à paralelização. O HTTP/2 está ativado em todos os servidores Kinsta.
- Reduza o número de solicitações HTTP externas. Cada um deles pode ter sua própria latência adicional com base na localização de seu servidor.
- DNS desempenha um papel no TTFB, portanto, você deve usar um provedor de DNS premium com tempos de pesquisa rápidos.
- Utilize pré-busca e pré-inicialização para executar tarefas nos bastidores enquanto a página é carregada.

Não se preocupe; abordaremos todas as recomendações mencionadas acima mais abaixo nesta postagem.

Velocidades SFTP e Painel de Controle do WordPress

Seus visitantes e clientes devem ser sempre sua prioridade. Mas outro aspecto do desempenho sobre o qual muitos não falam é como **algumas dessas decisões afetam seu trabalho diário**. A localização do data center escolhida afeta a velocidade de download e a velocidade de upload do SFTP (transferência de arquivos com um cliente FTP), bem como a capacidade de resposta do painel de controle do WordPress.

Então, enquanto você quer ter certeza e escolher um local que seja melhor para seus visitantes, lembre-se de que isso pode afetar o gerenciamento do site. Tarefas como o upload de arquivos para a biblioteca de mídia do WordPress serão mais rápidas quando o site estiver hospedado em um data center mais próximo de você.

Ouvimos constantemente de clientes na Kinsta que eles ficam surpresos com a rapidez com que o painel de administração deles está mais rápido conosco. Há uma infinidade de fatores que influenciam isso, mas ter 20 centros de dados diferentes é grande! Escolha um local que funcione tanto para seus visitantes quanto para você! Afinal, você provavelmente gastará milhares de horas trabalhando em seu site.

CAPÍTULO 04:

DNS Premium é melhor que DNS Gratuito



DNS, abreviação de Domain Name System, é um dos componentes mais comuns, mas incompreendidos, da paisagem da web. Para simplificar, o DNS ajuda a direcionar o tráfego na Internet, conectando nomes de domínio a servidores da Web reais. Essencialmente, é necessário um pedido amigável para o ser humano - um nome de domínio como kinsta.com - e traduz em um endereço IP de servidor amigável ao computador - como 216.58.217.206.



Como funciona o DNS

Você pode encontrar DNS gratuito e DNS premium. Todos os clientes da Kinsta têm acesso ao DNS premium via Amazon Route 53. E, em geral, acreditamos que o DNS premium é uma necessidade no mundo atual.

Um grande motivo para escolher o DNS premium é a **velocidade e a confiabilidade**. Pesquisar registros de DNS e direcionar o tráfego leva tempo, mesmo que seja apenas uma questão de milissegundos.

Normalmente, o DNS gratuito que você recebe do registrador de nomes de domínio é relativamente lento, enquanto o DNS premium geralmente oferece melhor desempenho. Por exemplo, em nossos testes, descobrimos que o **DNS gratuito do NameCheap é 33% mais lento** que o DNS premium do Amazon Route 53. Além disso, o DNS premium pode oferecer melhor segurança e disponibilidade, especialmente quando você está sob um ataque de DDoS.

Você pode usar uma ferramenta de teste de velocidade SolveDNS para verificar seus tempos de pesquisa de DNS. O DNSPerf também fornece excelentes dados de desempenho em todos os principais provedores de DNS.

Para um bom meio-termo entre o DNS gratuito fornecido pelo registrador de domínios e o DNS premium, o Cloudflare DNS é um serviço gratuito que ainda oferece muitos dos benefícios do DNS premium. E eles estão brilhando rapidamente com menos de 20 ms de tempo médio de resposta em todo o mundo (como visto abaixo).

| Name Server | Result | Average (milliseconds) | Standard Deviation | 95% Confidence Interval |
|---------------------------|---------------------------|------------------------|--------------------|-------------------------|
| algin.ns.cloudflare.com | ✓ Los Angeles: 5.65 ms | 5.48 ms | 2.69 | [3.49, 7.47] |
| | ✓ Dallas: 6.04 ms | | | |
| | ✓ New York: 3.47 ms | | | |
| | ✓ Singapore: 3.20 ms | | | |
| | ✓ London: 3.81 ms | | | |
| | ✓ Amsterdam: 4.04 ms | | | |
| | ✓ San Francisco: 12.4 ms | | | |
| | ✓ Sydney: 6.57 ms | | | |
| | ✓ Tokyo: 4.02 ms | | | |
| heather.ns.cloudflare.com | ✓ Los Angeles: 1.55 ms | 3.44 ms | 3.12 | [1.43, 5.75] |
| | ✓ Dallas: 1.98 ms | | | |
| | ✓ New York: 2.66 ms | | | |
| | ✓ Singapore: 2.11 ms | | | |
| | ✓ London: 2.95 ms | | | |
| | ✓ Amsterdam: 2.17 ms | | | |
| | ✓ San Francisco: 12.04 ms | | | |
| | ✓ Sydney: 1.52 ms | | | |
| | ✓ Tokyo: 3.62 ms | | | |

No entanto, uma ressalva com o Cloudflare é que ele também tem mais tempo de inatividade do que muitos outros provedores. Se você veicular principalmente visitantes nos Estados Unidos, o DNS Made Easy é outro ótimo provedor de DNS premium que você pode querer conferir. Eles têm a reputação de fornecer o melhor tempo de atividade do DNS na última década.

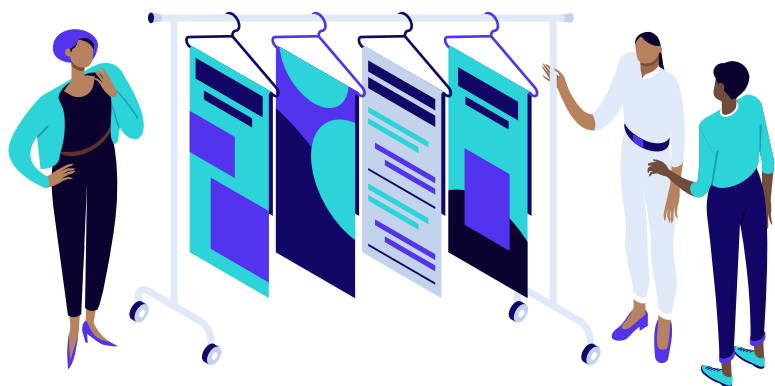
Nos últimos 30 dias, o DNSPerf mostra o seguinte tempo de atividade desses provedores:

- DNS Made Easy: 99,99%, o que equivale a 4m 23.0s de inatividade mensal.
- Amazon Route 53: 99,88%, o que equivale a 52m 35.7s de tempo de inatividade mensal.
- Cloudflare: 99,85%, o que equivale a 1h 5m 44.6s de tempo de inatividade mensal.

O tempo de inatividade é muito importante para os provedores de DNS? A resposta para isso é realmente sim e não. O DNS é normalmente armazenado em cache com ISPs usando o TTL (time to live value) no registro DNS. Portanto, se um provedor de DNS ficar inativo por 10 minutos, você provavelmente não notará nada. O tempo de inatividade importa, no entanto, se o provedor tiver interrupções mais longas e freqüentes, ou se os registros do provedor e do DNS estiverem usando valores TTL realmente baixos.

CAPÍTULO 05:

O Tema do Seu WordPress é Importante



Todo mundo adora um novo tema WordPress, mas tenha cuidado antes de sair e pegando todos que tem novos recursos brilhantes. Existem muitas diferenças quando se trata de temas gratuitos vs pagos. Em relação ao desempenho, todos os elementos que você vê em um tema têm algum impacto na velocidade geral do seu site. E infelizmente, com milhares de temas na natureza, existem bons e maus.

Seu tema WordPress é importante para o desempenho. Escolha o caminho certo desde o início.

Então, como você deve saber qual escolher? Recomendamos seguir uma das duas opções a seguir:

- Um tema WordPress rápido e leve que é construído com apenas os recursos que você precisa, nada mais.
- Um tema WordPress mais rico em recursos, mas você pode desativar recursos que não estão em uso.

Coisas como o Google Fonts, os ícones do Font Awesome, os controles deslizantes, as galerias, os scripts de vídeo e paralaxe, etc. Essas são apenas algumas das muitas coisas que você deve poder desativar se não as estiver usando. Você não quer tentar ajustá-las manualmente depois do fato. E não vamos mostrar a você 50 maneiras diferentes de eliminar essas coisas. Em vez disso, você deve iniciar ou alternar para um tema do WordPress que seja leve desde o início ou forneça essas opções.

Abaixo estão alguns temas do WordPress que recomendamos e que você não pode errar! Confie em nós, você nos agradecerá mais tarde.

Todos os temas mencionados abaixo são compatíveis com WooCommerce e Easy Digital Downloads, WPML, BuddyPress e bbPress. Nós executamos alguns testes de velocidade com cada tema usando a seguinte configuração:

- Hospedado na Kinsta, executando o WordPress 5.0
- PHP 7.3 and SSL (HTTPS)
- Kinsta CDN
- O Imagify foi usado para compactar imagens automaticamente.

GeneratePress

GeneratePress é um tema de WordPress rápido, leve (menos de 1 MB compactado), móvel, construído com velocidade, SEO e usabilidade em mente. Construído por Tom Osborne, um desenvolvedor do Canadá. É ativamente atualizado e bem suportado. Até mesmo alguns membros da equipe da Kinsta usam o GeneratePress em seus projetos.

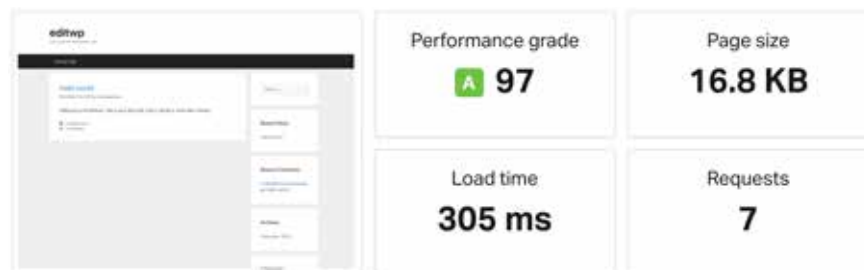
Existe uma versão gratuita e premium disponível. Se você der uma olhada no WordPress.org, a versão gratuita atualmente tem mais de 100.000 instalações ativas, mais de 1.6 milhões de downloads e uma impressionante classificação de 5 de 5 estrelas (mais de 700 pessoas deram a ele 5 estrelas).

Uma das melhores coisas do GeneratePress é que todas as opções usam o WordPress Customizer nativo, o que significa que você pode ver todas as alterações feitas

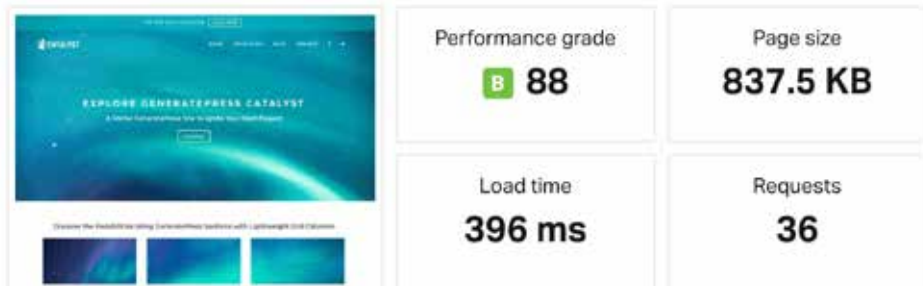
instantaneamente antes de pressionar o botão Publicar. Isso também significa que você não precisa aprender um novo painel de controle de tema.



Quão rápido é isso? Fizemos uma nova instalação do GeneratePress, fizemos cinco testes de velocidade no Pingdom e pegamos a média. O tempo total de carregamento foi de **305 ms** com um **tamanho de página total de apenas 16,8 KB**. É sempre bom ter um teste de linha de base para ver do que o tema é capaz em termos de desempenho bruto.



Em seguida, executamos outro conjunto de testes com um dos temas pré-compilados da biblioteca do site GeneratePress. Contém imagens, planos de fundo, novas seções, etc. Uma vantagem do GeneratePress é que ele possui vários temas pré-criados que não exigem um plugin do construtor de páginas. Você pode ver que ainda tem clock de **400 ms**.



Agora, é claro, em um ambiente real, você pode ter outras coisas funcionando como o Google Analytics, o pixel de remarketing do Facebook, o Hotjar etc. Mas você deve ser capaz de apontar para a marca de 1 segundo facilmente.

Mostraremos mais maneiras de otimizar e acelerar o WordPress abaixo.

OceanWP

OceanWP. Demos Extensions Blog Support Download Sign In / Sign Up

Create Beautiful Websites

With 899,712 downloads, OceanWP is the fastest growing theme for WordPress. It's the favorite choice of thousands of developers and hobby-users.

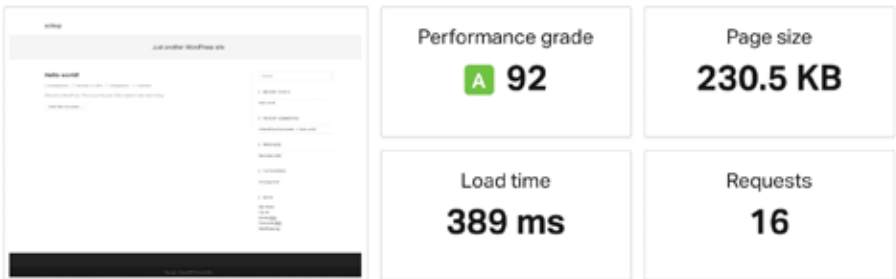
FREE DOWNLOAD

DEVELOPER **HOBBYIST**

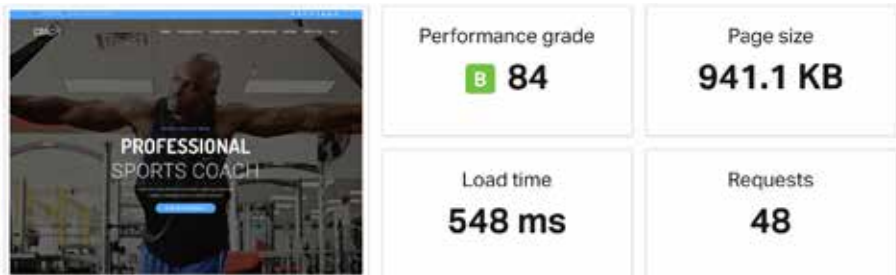
O tema OceanWP é leve e altamente extensível, e permite que você crie praticamente qualquer tipo de site, como um blog, portfólio, site de negócios e loja WooCommerce com um design bonito e profissional. Construído por Nicolas Lecocq, também é atualizado ativamente e bem suportado.

Assim como GeneratePress, existe uma versão gratuita e premium disponível. Se você der uma olhada no WordPress.org, a versão gratuita tem atualmente mais de 200.000 instalações ativas e outra impressionante classificação de 5 de 5 estrelas (mais de 1.200 pessoas deram a ele 5 estrelas).

Quão rápido é isso? Fizemos uma nova instalação do OceanWP, fizemos cinco testes de velocidade em Pingdom e atingimos a média. O tempo total de carregamento foi de **389 ms** com um **tamanho de página total de apenas 230,8 KB**. Os scripts no OceanWP são um pouco maiores, mas nada que se possa escrever.



Em seguida, executamos outro conjunto de testes com um dos temas de demonstração da biblioteca do site OceanWP. Contém imagens, planos de fundo, novas seções e requer o plugin do construtor de páginas Elementor. Você pode ver que ainda tem clock de **600 ms**.



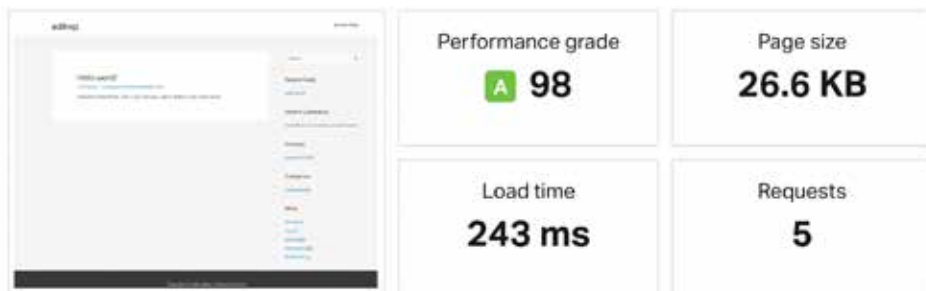
Astra

O Astra é um tema rápido, totalmente personalizável e bonito, adequado para blogs, portfólios pessoais, sites comerciais e vitrines da WooCommerce. É muito leve (menos de 50 KB no frontend) e oferece velocidade inigualável. Criada pela equipe da Brainstorm Force, ela é ativamente atualizada e bem suportada. Você pode reconhecê-los como criadores do popular plugin de Rich Snippets do All In One, que existe há muitos anos.

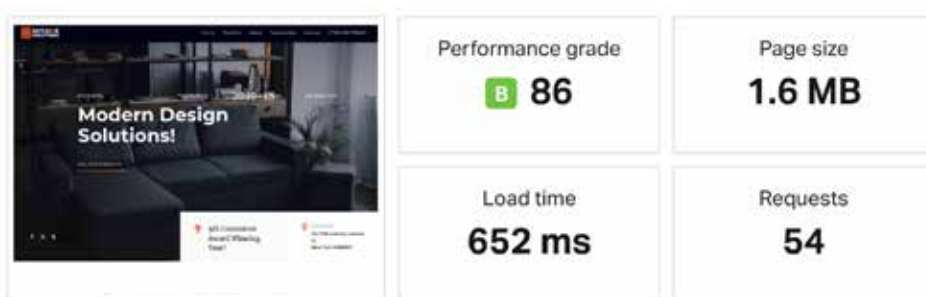
Assim como com o GeneratePress e o OceanWP, há uma versão gratuita e premium disponível. Se você der uma olhada no WordPress.org, a versão gratuita atualmente tem mais de 100.000 instalações ativas, mais de 1.6 milhões de downloads e outra impressionante classificação de 5 de 5 estrelas (mais de 850 pessoas deram a ele 5 estrelas).

A screenshot of the Astra WordPress theme website homepage. The background is a gradient of purple and blue. At the top left is the Astra logo, a circle with a stylized 'A' and the word 'ASTRA' next to it. To the right of the logo are navigation links: 'STARTER SITES', 'PRO', 'FEATURES', 'WOOCOMMERCE', and 'PRICING'. Further right is a yellow 'DOWNLOAD' button. The main heading is 'Perfect Theme For Any Website' in a large, white, serif font. Below it is the tagline 'Stylish, Lightning Fast & Easily Customizable' in a smaller, white, sans-serif font. In the center is a yellow 'DOWNLOAD NOW' button with a small icon of a download arrow. At the bottom, there are five small thumbnail images showing different website designs created with the Astra theme, including a business website, a portfolio, a blog, a WooCommerce store, and a family-oriented site.

Quão rápido é isso? Fizemos uma nova instalação do Astra, fizemos cinco testes de velocidade em Pingdom e atingimos a média. O tempo total de carregamento foi de **243 ms** com um **tamanho de página total de apenas 26,6 KB**.



Em seguida, executamos outro conjunto de testes com um dos temas de demonstração da biblioteca do site do kit Astra Starter. Contém imagens, planos de fundo, novas seções e requer o plugin do construtor de páginas Elementor. Você pode ver que ainda tem clock de **700 ms**. Nota: as imagens nesta demonstração foram totalmente compactadas, mas escolheram resoluções muito altas desde o início.



É importante ter as diferenças entre os testes de velocidade com esses três temas com um pouco de sal. O problema é que é quase impossível executar uma comparação lado a lado completamente precisa. A coisa mais importante que queremos mostrar é que todos esses temas do WordPress estão em alta velocidade, tanto fora da caixa quanto demos completos!

Aviso Sobre Construtores de Página

Como você provavelmente percebeu, o OceanWP e o Astra exigiram que os construtores de páginas usassem seus temas de biblioteca de sites. Aqui estão algumas coisas que você deve ter em mente ao usar um plugin do construtor de páginas:

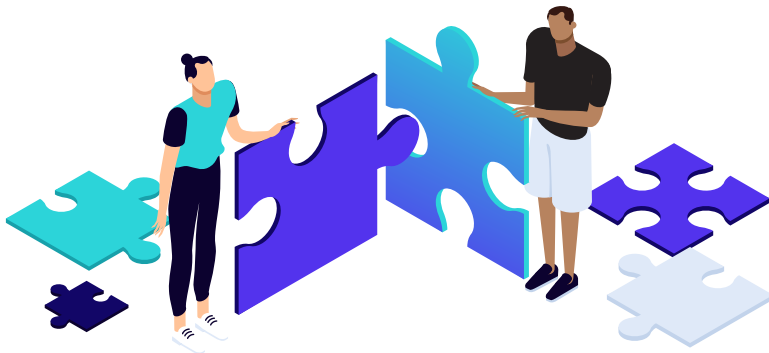
- Alguns criadores de páginas podem aumentar o tempo de carregamento no seu site. Isso ocorre porque eles precisam carregar CSS e JS adicionais para fazer as coisas funcionarem sem o código. É assim que a mágica acontece! Nós sempre recomendamos a velocidade de teste do seu site WordPress antes e depois de instalar um construtor de páginas.
- Você está cometendo e bloqueando-se no criador de páginas para o design. Certifique-se de escolher um que é atualizado regularmente e tem tudo que você precisa para o longo curso.

Com isso dito, ainda somos grandes fãs de construtores de páginas como Elementor e Beaver Builder. Na maior parte, eles são desenvolvidos com o desempenho em mente e apenas adicionam um pouco de sobrecarga. Para a maioria, a funcionalidade e a usabilidade valem a pena, já que esses plugins permitem criar qualquer coisa que você possa imaginar! Eles também podem ser mais rápidos em alguns casos, pois podem ser um substituto para outros 5 plugins que você teria que usar de outra forma.

No entanto, se você não precisar de um plugin do criador de páginas, não implante apenas um para pontapés. Também será interessante ver como o novo editor de Gutenberg irá desempenhar um papel no design do site nos próximos anos.

CAPÍTULO 06:

Fatos Sobre Plugins do WordPress



Você pode ter sido informado de que não deve instalar muitos plugins ou que isso diminuiria a velocidade do seu site do WordPress. Embora algumas vezes isso seja verdade, esse não é o fator mais crítico. O número de plugins não é tão importante quanto a qualidade dos plugins.

Assim como os temas, também é importante como o plugin é desenvolvido e se foi construído com o desempenho em mente. Temos muitos clientes na Kinsta que estão executando 30-40 plugins e seus sites ainda carregam bem menos de um segundo.

Embora seja divertido adicionar código ao seu site, isso nem sempre é prático pelos seguintes motivos:

- 1.** Você precisa manter o código por conta própria e mantê-lo atualizado à medida que os padrões mudam. As pessoas estão ocupadas, por que não confiar nos desenvolvedores fantásticos que conhecem os padrões melhor do que a maioria?
- 2.** Na maioria das vezes, um plugin bem codificado não vai introduzir muito mais overhead do que o próprio código.
- 3.** Você tem que lembrar que a maioria da comunidade WordPress não é tão experiente em tecnologia quanto a multidão de desenvolvedores. Plugins são soluções que ajudam a resolver problemas.

Com isso dito, claro que não tem bons plugins por aí que você precisa ficar longe. Confie em nós; já vimos o pior dos piores aqui na Kinsta. Muitos, mas não todos os plugins que

vimos e banimos na Kinsta, causam problemas de desempenho em primeira mão.

No entanto, não é possível ignorar que uma das coisas que as pessoas adoram no WordPress é sua enorme biblioteca de plugins de terceiros. Mas com mais de 56.000 plugins listados apenas no WordPress.org e outros milhares listados em outros lugares, pode ser difícil encontrar o plugin que você precisa. Fale sobre uma agulha num palheiro!

Nós tentamos apenas compartilhar coisas que usamos diariamente. E sim, usamos plugins do WordPress em nosso site como o resto de vocês. Muitos dos membros da equipe da Kinsta até desenvolvem e vendem plugins.

Um Grande Problema com os Plugins do WordPress

Um grande problema com os plugins do WordPress é o processo de desinstalação. Sempre que você instala um plugin ou tema do WordPress, ele armazena os dados no banco de dados. O problema é que, quando você exclui um plugin usando um dos métodos padrão, ele normalmente deixa para trás tabelas e linhas em seu banco de dados. Com o tempo, isso pode adicionar muitos dados e até começar a diminuir o ritmo do seu site. No nosso exemplo, nós desinstalamos o plugin de segurança do Wordfence, e ele deixou para trás 24 tabelas em nosso banco de dados (como visto abaixo). É ainda pior se eles estiverem por trás de dados em sua tabela `wp_options`.



| | | | | | | | | | | | |
|--------------------|---|--------|-----------|--------|--------|-------|------|---------|--------|-------------------|----------|
| wp_wfBadLeechers | ★ | Browse | Structure | Search | Insert | Empty | Drop | 9 | InnoDB | latin1_swedish_ci | 18 418 |
| wp_wfBlockedIPLog | ★ | Browse | Structure | Search | Insert | Empty | Drop | 75 | InnoDB | utf8_general_ci | 18 418 |
| wp_wfBlocks | ★ | Browse | Structure | Search | Insert | Empty | Drop | 12 | InnoDB | utf8_general_ci | 18 418 |
| wp_wfBlocksAdv | ★ | Browse | Structure | Search | Insert | Empty | Drop | 6 | InnoDB | utf8_general_ci | 18 418 |
| wp_wfConfig | ★ | Browse | Structure | Search | Insert | Empty | Drop | 156 | InnoDB | utf8_general_ci | 18 418 |
| wp_wfCrawlers | ★ | Browse | Structure | Search | Insert | Empty | Drop | 18 | InnoDB | latin1_swedish_ci | 18 418 |
| wp_wfFileMods | ★ | Browse | Structure | Search | Insert | Empty | Drop | 6 | InnoDB | utf8_general_ci | 18 418 |
| wp_wfHits | ★ | Browse | Structure | Search | Insert | Empty | Drop | 1,367 | InnoDB | latin1_swedish_ci | 113 969 |
| wp_wfHoovers | ★ | Browse | Structure | Search | Insert | Empty | Drop | 6 | InnoDB | utf8_general_ci | 18 418 |
| wp_wfIssues | ★ | Browse | Structure | Search | Insert | Empty | Drop | 6 | InnoDB | utf8_general_ci | 18 418 |
| wp_wfKnownFileList | ★ | Browse | Structure | Search | Insert | Empty | Drop | 6 | InnoDB | utf8_general_ci | 18 418 |
| wp_wfLeechers | ★ | Browse | Structure | Search | Insert | Empty | Drop | 156,312 | InnoDB | latin1_swedish_ci | 23,3 969 |
| wp_wfLockedOut | ★ | Browse | Structure | Search | Insert | Empty | Drop | 6 | InnoDB | utf8_general_ci | 18 418 |
| wp_wfLocs | ★ | Browse | Structure | Search | Insert | Empty | Drop | 4 | InnoDB | utf8_general_ci | 18 418 |

| | | | | | |
|--------------------|-----|--------|-------------------|----|-----|
| wp_wfLogins | 5 | InnoDB | utf8_general_ci | 46 | KiB |
| wp_wfNet64s | 1 | InnoDB | utf8_general_ci | 12 | KiB |
| wp_wfNotifications | 2 | InnoDB | utf8_general_ci | 18 | KiB |
| wp_wfPendingIssues | 6 | InnoDB | utf8_general_ci | 14 | KiB |
| wp_wfReverseCache | 2 | InnoDB | latin1_swedish_ci | 24 | KiB |
| wp_wfScanners | 100 | InnoDB | latin1_swedish_ci | 14 | KiB |
| wp_wfSNIPCache | 6 | InnoDB | utf8_general_ci | 14 | KiB |
| wp_wfStatus | 6 | InnoDB | utf8_general_ci | 46 | KiB |
| wp_wfThrottleLog | 6 | InnoDB | utf8_general_ci | 32 | KiB |
| wp_wfVulnScanners | 6 | InnoDB | latin1_swedish_ci | 14 | KiB |

WordFence tables

Além do banco de dados, muitos plugins também deixam pastas e arquivos adicionais. Em nossa experiência, isso é comumente visto com plugins de segurança e cache que criam diretórios adicionais para o registro. Por exemplo, depois que o plugin Wordfence foi excluído, ficamos com uma pasta “wflogs” em nosso diretório wp-content. E não estamos tentando usar o Wordfence, a maioria dos plugins e temas do mercado funciona dessa maneira.

| Filename | File... | Filetype | Last modified |
|-----------------|---------|---------------|-----------------------|
| .. | | | |
| .htaccess | 133 | HTACCESS File | 7/24/2017 10:54:49 AM |
| attack-data.php | 40,0... | PHP File | 7/24/2017 10:54:49 AM |
| config.php | 594,... | PHP File | 7/24/2017 10:57:30 AM |
| ips.php | 51 | PHP File | 7/24/2017 10:54:50 AM |
| rules.php | 101,... | PHP File | 7/24/2017 10:54:59 AM |
| wafRules.rules | 44,2... | RULES File | 7/24/2017 10:54:59 AM |

WordFence logs

Por Que os Desenvolvedores Fazem Isso?

Então, você provavelmente está se perguntando: por que os desenvolvedores não têm opções de autolimpeza quando você desinstala e exclui um plugin? Bem, eles fazem. Mas, aqui estão algumas razões pelas quais eles provavelmente não são tão óbvios logo de cara.

- 1. Eles querem manter as configurações do usuário.** Se você excluir um plugin do WordPress e decidir tentar novamente mais tarde, todas as suas configurações e dados ainda estarão lá. Embora isso seja super conveniente, não é a maneira mais eficiente.
- 2. Eles não se importam com o desempenho.** Alguns desenvolvedores podem argumentar que deixar as tabelas para trás não afeta o desempenho. Mas imagine um site ao longo de dez anos, tendo usado centenas de plugins, que geraram possivelmente milhares de linhas ou tabelas. As consultas de banco de dados têm um impacto significativo no desempenho do seu site WordPress, e os plugins podem fazer muitas dessas solicitações se o desenvolvedor não tiver cuidado. Geralmente, um plugin bem escrito deve consultar apenas as tabelas ou linhas nas quais está vinculado, no entanto, nem sempre é esse o caso. Vimos isso em primeira mão na Kinsta, consultas longas de banco de dados que traziam um site para rastrear devido a dados carregados automaticamente desnecessários na tabela `wp_options` que foi deixada para trás.
- 3. Eles cometeram um erro.** O manual do plugin WordPress até mesmo diz que “desenvolvedores menos experientes, por vezes, cometem o erro de usar o gancho de desativação para essa finalidade.”

As boas notícias? Existem maneiras de limpar e se livrar de um plugin corretamente. Links para os seguintes tutoriais detalhados estão disponíveis em nosso website em <https://kinsta.com/speed-up-wordpress>.

- Como Desinstalar um Plugin do WordPress (O Caminho Adequado)
- Como Limpar Manualmente as Tabelas Deixadas para Trás

CAPÍTULO 07:

Ótimas Configurações do WordPress



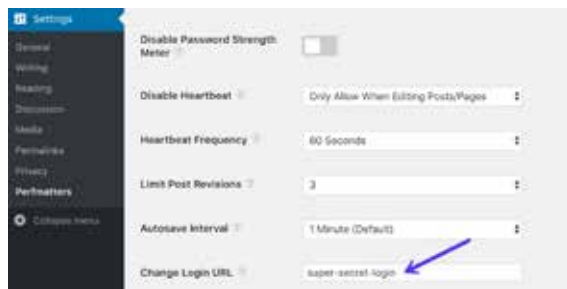
Agora, vá para as configurações ideais do WordPress. Aqui estão algumas mudanças que você pode fazer para ajudar a acelerar o seu site WordPress. Muitas dessas mudanças são muito sutis, mas tudo ajuda!

Alterar seu URL de login do WordPress

Por padrão, o URL de login do seu site WordPress é `domain.com/wp-admin/`. Um dos problemas com isso é que todos os bots, hackers e scripts por aí também sabem disso. Ao alterar o URL, você pode se tornar menos alvo, proteger-se melhor contra ataques de força bruta e diminuir a largura de banda usada pelos bots que atingem essa URL repetidamente.

Alterar seu URL de login do WordPress também pode ajudar a evitar erros comuns, como “429 Solicitações em excesso”. Essa não é uma solução para todas as correções, é apenas um pequeno truque que pode ajudar a proteger você e diminuir a carga nessa página.

Para alterar o seu URL de login do WordPress, recomendamos o uso de um dos seguintes plugins:



Change WordPress login URL in Perfmatters

- **WPS Hide Login** (Grátis, disponível no WordPress.org)
- **Perfmatters** (Premium, mas inclui outras configurações de otimização de desempenho. Desenvolvido por um membro da equipe da Kinsta e disponível em perfmatters.io)

Desativar ou Ajustar Atualizações de Plugins e Temas

Painéis de administração lentos do WordPress podem ser afetados pela rede, pela localização do data center e até mesmo pelas versões do PHP. Mas outro fator que poucas pessoas falam é o verificador de atualizações do WordPress que roda em segundo plano. Este é um exemplo em que ter muitos plugins e temas do WordPress pode prejudicá-lo. A equipe do WeFoster cunhou uma ótima frase para isso, chamando-a de “Síndrome de verificação de atualização de plugins de terceiros” ou TPPUCS.

Essencialmente, o problema é que o verificador de atualizações embutido do WordPress faz uma solicitação GET externa nos bastidores (<https://third-party-plugin/update-check.php>). Às vezes isso pode ser periódico ou muito frequente. Se isso estiver acontecendo o tempo todo, isso pode levar seu painel de administração a um rastreamento.

Este é mais um problema de como o verificador de atualizações no WordPress é construído. Se você está sofrendo com lentidão nos tempos de carregamento do painel de administração do WordPress, experimente. A solução é desativar as atualizações automáticas. Aviso: faça isso apenas se você pretende verificar as atualizações manualmente. Muitas atualizações incluem segurança e correções de bugs.

Para desabilitar as atualizações, recomendamos o uso de um dos seguintes plugins (ambos disponíveis no WordPress.org):

- **Disable All WordPress Updates:** Completamente gratuito e sem configurações.

- **Easy Updates Manager:** Fornece mais controle sobre atualizações seletivas. A versão principal é gratuita.

Você pode definir facilmente um lembrete de calendário, desativar o plugin uma vez por semana, verificar se há atualizações e reativá-lo.

Desativar Pingbacks

Um pingback é um comentário automatizado que é criado quando outro blog é direcionado para você. Também pode haver self-pingbacks criados quando você vincula um artigo ao seu próprio blog.

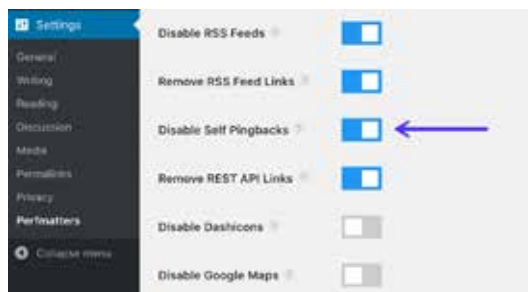
Recomendamos simplesmente desativá-los, pois eles geram consultas inúteis e spam adicional em seu site. Lembre-se, menos chamadas em seu site WordPress tem que fazer, melhor é, especialmente em sites de alto tráfego. Sem mencionar o fato de que um pingback em seu próprio site é simplesmente chato. Siga os passos abaixo para desativar os pingbacks.

Passo 1 – Desativar Pingbacks de Outros Blogs

Em seu painel do WordPress, clique em “Configurações → Discussão”. Na seção Configurações de discussão, desmarque a opção “Permitir notificações de link de outros blogs (pingbacks e trackbacks) em novos artigos”.



Etapa 2 – Desativar os auto-pingbacks



Disable self-pingbacks with Perfmatters

Quando se trata de desabilitar os auto-pingbacks, você tem algumas opções. Você pode usar o plugin gratuito No Self Pings (disponível no WordPress.org). Ou você pode usar um plugin premium como o Perfmatters (disponível em perfmatters.io).

Alternativamente, você também pode desabilitar auto-pingbacks adicionando o seguinte código ao arquivo `functions.php` do tema WordPress. Aviso, editando a fonte de um tema do WordPress pode quebrar seu site se não for feito corretamente. Dica, você pode facilmente adicionar snippets PHP como este com o plugin Code Snippets (disponível no WordPress.org). Isso significa que você nunca precisa tocar em seu tema.

```
function wpsites_disable_self_pingbacks( &$links ) {  
    foreach ( $links as $l => $link )  
        if ( 0 === strpos( $link_get_option( 'home' ) ) )  
            unset( $links[ $l ] );  
}  
  
add_action( 'pre_ping', 'wpsites_disable_self_pingbacks' );
```

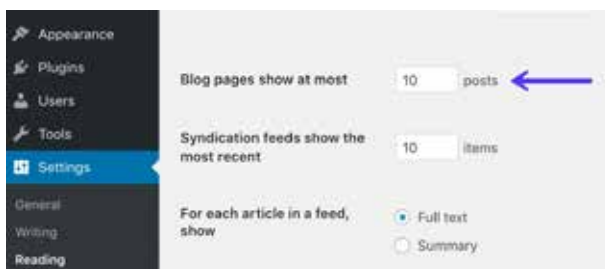
Limite de Postagens no Seu Feed de Blog

Se o seu feed de blog é definido como sua página inicial ou é outra página do seu site, você não precisa de 50 miniaturas todas carregando ao mesmo tempo. Para aqueles que executam blogs de alto tráfego, sua página inicial é a página mais importante do seu site e você deseja que isso seja carregado rapidamente. Quanto menos pedidos e mídia, melhor em termos de desempenho.

Além disso, é precisamente por isso que a paginação foi inventada (como visto abaixo). A paginação é o que você vê no final dos feeds do blog que permite navegar até a próxima página. Normalmente, são números ou podem usar postagens “anteriores / anteriores”. Seu tema WordPress provavelmente já terá paginação personalizada embutida.



Por padrão, o WordPress define o limite de novas instalações do WordPress para 10, mas já vimos isso ter mudado tantas vezes que perdemos a conta. Portanto, verifique o valor que você está usando. Recomendamos algo entre 8 e 12. Se você está curioso, estamos usando 12 em nossa página inicial do blog Kinsta.



Você pode encontrar essa opção em seu painel de administração do WordPress em “Configurações → Leitura”. Você pode alterar o valor para “Mostrar as páginas do blog no máximo”.

CAPÍTULO 08:

Por Que o Cache é Tão Importante



O cache é de longe uma das maneiras mais importantes e fáceis de acelerar o WordPress! Mas antes de mostrarmos como usar o armazenamento em cache, é essencial primeiro entender como ele funciona e os diferentes tipos de armazenamento em cache disponíveis.

O Que é Cache?

Em suma, cada página visitada em seu site WordPress requer uma solicitação ao servidor, processando por esse servidor (incluindo consultas de banco de dados) e em seguida, um resultado final enviado do servidor para o navegador do usuário. O resultado é o seu site, completo com todos os arquivos e elementos que o fazem parecer do jeito que faz.

É aí que entra um plugin de cache do WordPress! O armazenamento em cache instrui o servidor a armazenar alguns arquivos em disco ou RAM, dependendo da configuração. Portanto, ele pode lembrar e duplicar o mesmo conteúdo que está sendo veiculado no passado. Basicamente, reduz a quantidade de trabalho necessária para gerar uma exibição de página. Como resultado, **suas páginas da Web são carregadas muito mais rapidamente, diretamente do cache.**

Alguns outros benefícios do armazenamento em cache incluem:

- **Seu servidor usa menos recursos** - Isso se torna mais rápido, já que menos recursos geram um site mais rápido. No entanto, também coloca menos pressão no seu servidor. Isso é muito importante quando se trata de sites altamente dinâmicos, como sites de membros, e determinar o que você pode e não pode servir de cache.
- **Você verá um TTFB mais baixo** - o armazenamento em cache é uma das maneiras mais fáceis de reduzir seu TTFB. De fato, em nossos testes, o cache normalmente reduz o TTFB em até 90%!

Tipos de Cache

Quando se trata de tipos de cache, existem duas abordagens diferentes comumente usadas:

1. Cache no Nível do Servidor
2. Cache com um Plugin

1. Cache no Nível do Servidor

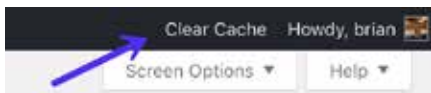
O armazenamento em cache no nível do servidor é de longe, uma das abordagens mais fáceis para o usuário final. O que isto significa é que o provedor de hospedagem WordPress lida com isso para você. Na Kinsta, utilizamos **quatro tipos de cache** a seguir, que são todos feitos automaticamente no software ou no nível do servidor:

- Cache Bytecode
- Cache de objetos
- Cache de páginas
- Cache de CDN

Isso significa que você não precisa se preocupar em lidar com plugins de cache complicados e confusos. Você pode parar de pesquisar os “melhores plugins de cache” e se concentrar em tarefas mais produtivas.

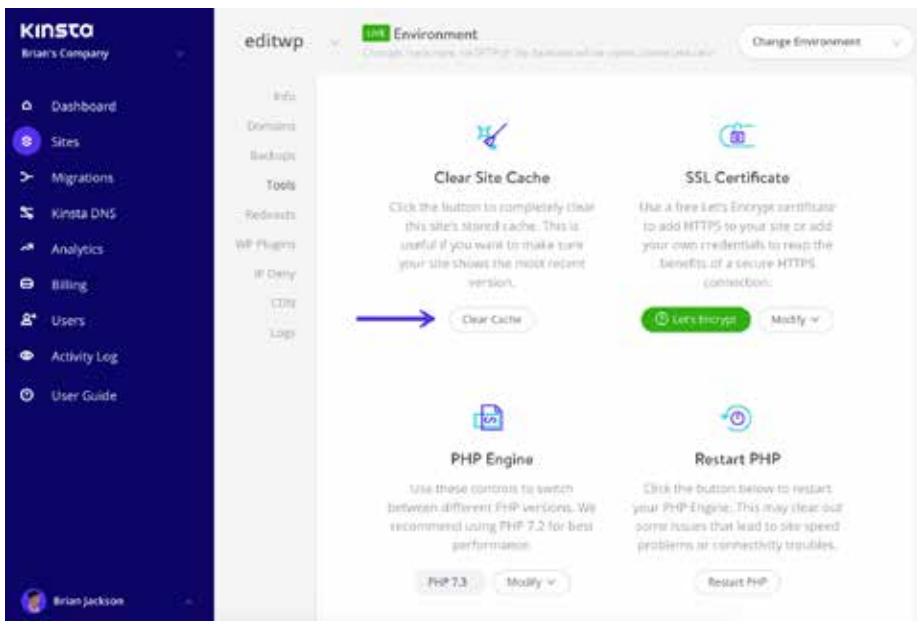
O cache de páginas é configurado para funcionar imediatamente com o WordPress padrão. Você não precisa fazer nada! Basta iniciar o seu site WordPress e o cache de páginas começará a acontecer.

Também temos regras de armazenamento em cache para sites de comércio eletrônico, como o WooCommerce e Easy Digital Downloads. Por padrão, determinadas páginas que nunca devem ser armazenadas em cache, como cart, my-account e checkout, são excluídas do armazenamento em cache. Os usuários ignoram automaticamente o cache quando o cookie `woocommerce_items_in_cart` ou `edd_items_in_cart` é detectado para garantir um processo de check-out tranquilo e sincronizado.



Você pode limpar facilmente o cache do seu site WordPress a qualquer momento na barra de ferramentas do administrador.

Também está integrado ao nosso painel MyKinsta. Basta clicar em Ferramentas e clicar em “Limpar Cache”.



2. Cache com um Plugin

Se o provedor de hospedagem não fornecer o cache, você poderá usar um plugin de armazenamento em cache do WordPress de terceiros. Com base em nossa experiência, recomendamos um dos seguintes:

- **WP Rocket** (premium, disponível em wp-rocket.me)
- **Cache Enabler** (gratuito, disponível no WordPress.org)
- **W3 Total Cache** (gratuito, disponível no WordPress.org)

Também apoiamos totalmente o WP Rocket na Kinsta! Geralmente, não permitimos plugins de armazenamento em cache em nosso ambiente porque eles entram em conflito com nossa solução de armazenamento em cache integrada. No entanto, a partir do WP Rocket 3.0, sua funcionalidade de cache de páginas será automaticamente desativada quando executada nos servidores Kinsta.

Isso permite que os clientes da Kinsta usem o nosso cache de nível de servidor rápido, mas ainda aproveitem os fantásticos recursos de otimização que o WP Rocket tem a oferecer

Sem Cache vs Cache

How much does caching help? The proof is in the pudding.

We ran a few speed tests with Kinsta's server-level caching so you can see the difference it makes, both in terms of overall speed and TTFB.

Sem Cache

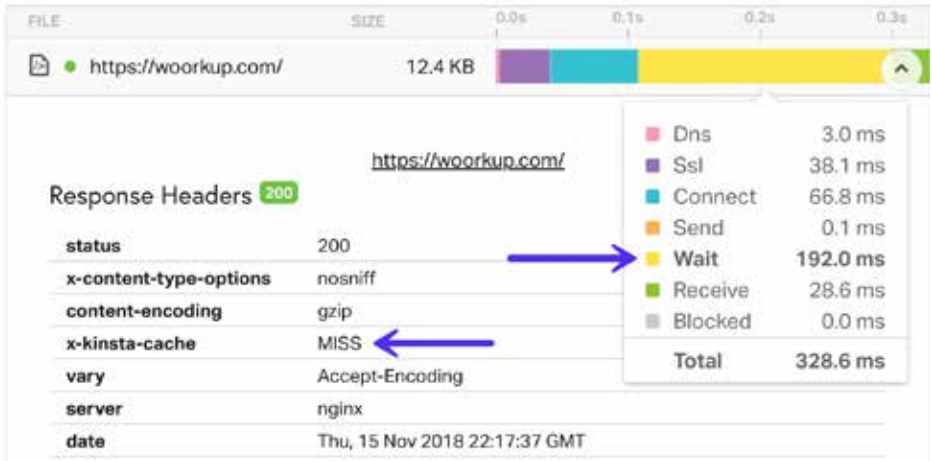


| | |
|----------------------------------|------------------------------|
| Performance grade B 89 | Page size 230.6 KB |
| Load time 507 ms | Requests 40 |

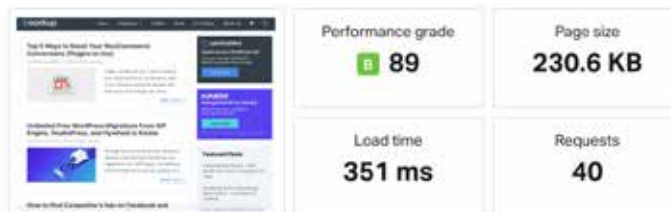
Nós primeiro executamos 5 testes no Pingdom sem o cache ativado e pegamos a média.

TTFB Sem Cach

Também é importante observar a diferença no TTFB sem e com o armazenamento em cache. O TTFB no Pingdom é representado pela barra amarela “waiting”. Como você pode ver o TTFB sem cache é de 192 ms. Você pode ver que não está sendo veiculado do cache, pois o cabeçalho `x-kinsta-cache` está exibindo uma MISS.



Com o Cache Ativado

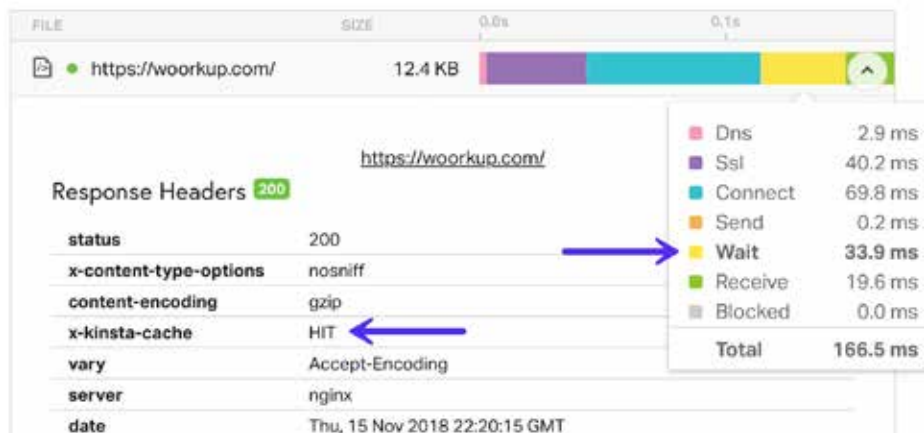


Em seguida, ativamos o cache no nível do servidor e executamos 5 testes no Pingdom e obtivemos a média.

Como você pode ver o **cache no nível do servidor diminuiu o tempo de carregamento da página em 33,77%**! E isso sem nenhum trabalho extra envolvido. Este site que testamos também é bastante otimizado, portanto sites maiores e não otimizados estão fadados a ver diferenças ainda maiores.

TTFB com Cache Ativado

Agora, se analisarmos o TTFB com o cache ativado, podemos ver que ele está abaixo de 35 ms. Você pode ver que ele está sendo veiculado do cache, pois o cabeçalho `x-kinsta-cache` está exibindo um HIT.



O cache CDN também é tão importante quanto o cache do seu host WordPress. Vamos mergulhar mais nos CDNs mais abaixo.

O armazenamento em cache do WordPress pode facilmente diminuir os tempos de carregamento de sua página em mais de 33%!

Problemas com Sites de Armazenamento em Cache e Membros

Os sites de membros contêm muitos conteúdos páginas que estão mudando continuamente. Coisas como a página de login para membros da comunidade (que podem ser atingidas constantemente dependendo do tamanho do site), páginas de checkout de produtos ou cursos digitais e fóruns de discussão são culpados e pontos problemáticos comuns, pois normalmente não podem ser armazenados em cach.

No entanto, não termina aí. Em sites padrão do WordPress, o painel do WordPress também não é armazenado em cache para **usuários “logados”**. Isso é bom quando você tem apenas alguns autores e administradores, mas quando, de repente, você tem milhares de membros usando o painel, isso imediatamente causa problemas de desempenho, já que nenhum deles pode ser exibido a partir do cache no servidor. Isso significa que você precisa do poder e da arquitetura nos bastidores para fazer backup. Os provedores de hospedagem compartilhada normalmente ficam paralisados nessas circunstâncias.

Armazenamento em Cache de Objetos para Sites Altamente Dinâmicos

Quando se trata de sites de membros do WordPress, suas configurações comuns de armazenamento em cache geralmente não são suficientes, pois nem sempre aproveitam ao máximo. É aqui que o cache de objetos entra em ação.

O cache de objetos armazena os resultados das consultas do banco de dados para que, na próxima vez em que um determinado bit de dados seja necessário, ele possa ser entregue a partir do cache sem consultar o banco de dados. Isso acelera os tempos de execução do PHP e reduz a carga no seu banco de dados. Isso se torna extremamente importante com sites de associação! Com o WordPress, você pode implementar o cache de objetos de duas maneiras diferentes:

1. Uma solução de armazenamento em cache de terceiros, como o W3 Total Cache
2. **Redis (recomendado)**
3. Memcached

Oferecemos o Redis como um complemento na Kinsta para que você possa aproveitar ao máximo o armazenamento em cache de objetos persistentes em seus sites de associação.

Analisando o Cache

Lembre-se que o cabeçalho x-kinsta-cache que mencionamos acima? Dependendo do seu provedor de hospedagem ou da solução de armazenamento em cache, o cabeçalho pode ter um nome ligeiramente diferente. Toda vez que uma solicitação é

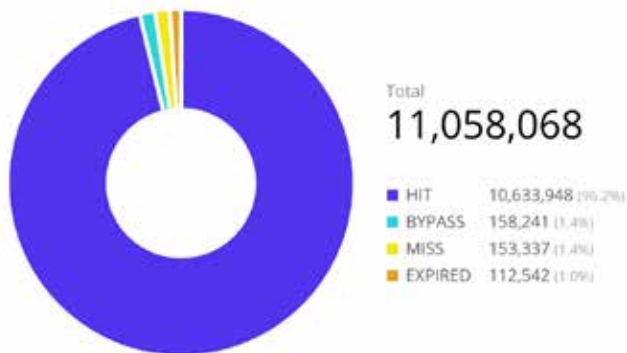
feita a partir do seu site WordPress, esse cabeçalho tem um valor, como HIT, BYPASS, MISS e EXPIRED. Isso permite que você veja o desempenho do seu cache.

Aumentar a taxa de acertos do cache do seu site WordPress é importante porque você deseja que o máximo de seu site seja veiculado do cache possível. Na Kinsta, você pode analisar os dados em nossa ferramenta de análise MyKinsta e os logs de cache kinsta para determinar se há solicitações BYPASS GET em cache que podem ser armazenadas em cache ou solicitações POST que podem ser eliminadas.

A pilha do componente de cache (como mostrado abaixo) permite ver o status de cada solicitação, seja HIT, BYPASS, MISS ou EXPIRED. Você pode filtrar os dados nas últimas 24 horas, 7 dias ou 30 dias.



O gráfico do componente de cache te ajuda você ver sua taxa de cache. Quanto mais solicitações você veicular no cache, melhor. Como você pode ver no exemplo abaixo, este site WordPress está em uma proporção de cache de 96,2% HIT. Que é bom!

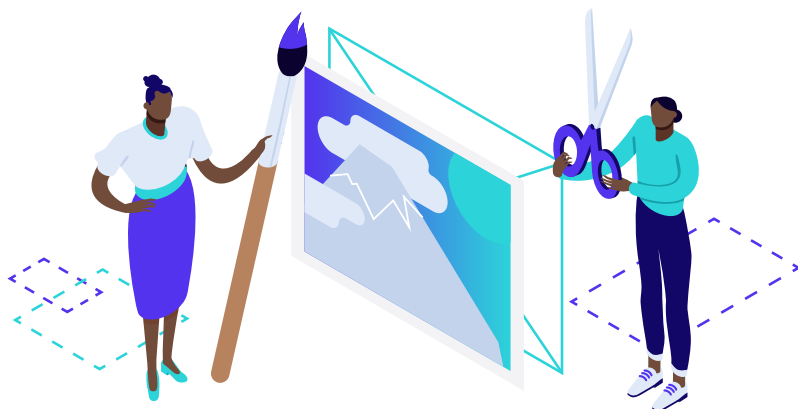


A seção principal de ignorar o cache permite ver quais solicitações não estão sendo atendidas no cache. Geralmente, eles podem incluir tarefas CRON, solicitações admin-ajax, páginas de checkout de comércio eletrônico, strings de consulta e parâmetros UTM, etc.

| PATH | #REQUESTS |
|--|-----------|
| /wp-cron.php?server_triggered_cronjob | 28,977 |
| /wp-admin/admin-ajax.php?action=edd_jilt_get_cart_data | 22,279 |
| /?ref=1 | 796 |
| / | 667 |
| /sitemap_index.xml | 490 |
| /?ajax=1&ht-kb-search=1&sc= | 435 |
| /account/ | 372 |
| /checkout/ | 329 |
| /?ref=19&campaign=bw | 292 |

CAPÍTULO 09:

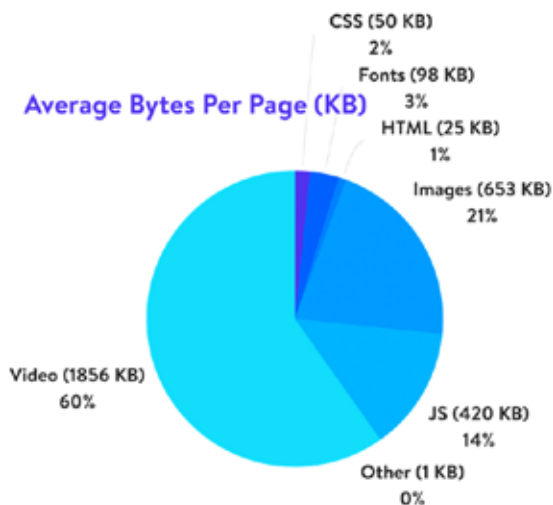
Otimização de Imagem é Uma Obrigação



A otimização de imagem é outra coisa simples que você pode fazer, que tem um impacto significativo nos tempos totais de carregamento da página. Isso não é opcional; Todo site deveria estar fazendo isso!

Imagens grandes tornam as suas páginas mais lentas, o que cria uma experiência de usuário que não é ideal. Otimizar imagens é o processo de diminuir o tamanho do arquivo, usando um plugin ou script, o que, por sua vez, acelera o tempo de carregamento da página. Compressão com perdas e sem perdas são dois métodos comumente usados.

De acordo com o HTTP Archive, em novembro de 2018, as imagens fazem em média, **21% do peso total de uma página da web**. Então, depois dos vídeos, que são muito mais difíceis de otimizar, as imagens de longe são o primeiro lugar que você deve começar! É mais importante que JavaScript, CSS e Fonts. E, ironicamente, um bom fluxo de trabalho de otimização de imagem é uma das coisas mais fáceis de implementar, mas muitos proprietários de sites ignoram isso.



Em dezembro de 2017, as imagens representavam, em média, 54% do peso total de uma página. Por isso, parece que a Web como um todo está melhorando na otimização de imagens! Mas 21% ainda é um número que não pode ser ignorado. Se você não tiver nenhum conteúdo de vídeo em seu website, as imagens provavelmente ainda serão seu ponto de dor número 1 para o peso da página.

As imagens fazem em média, 21% do peso total de uma página web. Otimize-os!

Encontrando o Equilíbrio (Tamanho e Qualidade do Arquivo)

O principal objetivo da formatação de suas imagens é encontrar o **equilíbrio entre o menor tamanho de arquivo e a qualidade aceitável**. Há mais de uma maneira de executar quase todas essas otimizações. Uma das formas mais básicas é comprimi-las antes de fazer o upload para o WordPress. Geralmente, isso pode ser feito em uma ferramenta como o Adobe Photoshop ou o Affinity Photo. Ou usando o novo aplicativo online Squoosh do Google. No entanto, essas tarefas também podem ser executadas automaticamente usando plugins, que serão abordados mais abaixo.

As duas principais coisas a considerar é o **formato do arquivo** e o **tipo de compressão** que você usa. Ao escolher a combinação certa de formato de arquivo e tipo de compressão, você pode reduzir o tamanho da sua imagem em até 5 vezes. Você terá que experimentar cada imagem ou formato de arquivo para ver o que funciona melhor.

Antes de começar a modificar suas imagens, verifique se você escolheu o melhor tipo de arquivo. Existem vários tipos de arquivos que você pode usar:

- **PNG** – produz imagens de maior qualidade, mas também possui um tamanho de arquivo maior. Foi criado como um formato de imagem sem perdas, embora também possa ser com perdas.
- **JPEG** – usa otimização com perdas e sem perdas. Você pode ajustar o nível de qualidade para um bom equilíbrio de qualidade e tamanho do arquivo.

Idealmente, você deve usar JPEG (ou JPG) para imagens com muita cor e PNG para imagens simples.

E quanto aos GIFs? Os GIFs animados são sempre divertidos, mas matam o desempenho da web. Muitos GIFs têm mais de 1 MB de tamanho. Recomendamos mantê-los para mídias sociais e Slack. Se houver um que você não pode viver sem em sua postagem do blog, você deve sempre compactá-los

Qualidade de Compressão vs. Tamanho

Aqui está um exemplo do que pode acontecer de você comprimir demais uma imagem. O primeiro é usar uma taxa de compressão muito baixa, o que resulta na mais alta qualidade (mas em tamanho de arquivo maior). A segunda é usar uma taxa de compressão muito alta, o que resulta em uma imagem de baixa qualidade (mas com tamanho de arquivo menor). Nota: A imagem original intocada é de 2,06 MB.

Como você pode ver, a primeira imagem acima é de 590 KB. Isso é muito grande para uma foto! Geralmente, é melhor se você puder manter o peso total de uma página da web com tamanho de 1 ou 2 MB. 590 KB seria um quarto disso já. A segunda imagem

parece horrível, mas é apenas 68 KB. O que você quer fazer é encontrar um meio feliz entre sua taxa de compressão (qualidade) e o tamanho do arquivo.



Baixa compressão (alta qualidade) JPG - 590 KB



Alta compressão (baixa qualidade) JPG - 68 KB



Média compressão (grande qualidade) JPG - 151 KB

Então pegamos a imagem novamente em uma taxa média de compressão, e como você pode ver abaixo, a qualidade parece boa agora, e o tamanho do arquivo é de 151 KB, o que é aceitável para uma foto de alta resolução. Isso é quase 4x menor que a foto original com baixa compressão. Tentamos manter a maioria das imagens abaixo da marca de 100 KB para o melhor desempenho.

Com perdas vs. Compressão sem Perdas

Também é importante entender que existem dois tipos de compressão que você pode usar, com perdas e sem perdas.

A otimização com perdas envolve a **eliminação de alguns dados** na sua imagem. Por causa disso, significa que você pode ver degradação (redução na qualidade ou o que alguns chamam de pixelizada). Então você tem que ter cuidado com o quanto você está reduzindo sua imagem. Não só devido à qualidade, mas também porque você não pode reverter o processo. É claro que um dos grandes benefícios da compressão com perdas e por que é um dos métodos de compressão mais populares é que você pode **reduzir o tamanho do arquivo em uma quantidade considerável**.



A compressão sem perdas, diferentemente da com perda, **não reduz a qualidade da imagem**. Como isso é possível? Geralmente é feito removendo metadados desnecessários (dados gerados automaticamente pelo dispositivo capturando a imagem). No entanto, a maior desvantagem desse método é que você **não verá uma redução significativa no tamanho do arquivo**. Em outras palavras, vai ocupar muito espaço em disco ao longo do tempo.

Você vai querer experimentar o que funciona melhor para você. Mas para a maioria dos usuários, **recomendamos o uso da compressão com perdas**, pois você pode facilmente compactar uma imagem bem acima de 70% (às vezes até acima de 90%!) Sem muita perda de qualidade. Multiplique isso por 15 imagens em uma página e isso terá um papel importante na redução do tempo de carregamento de seu site.

Plugins de Compressão de Imagem

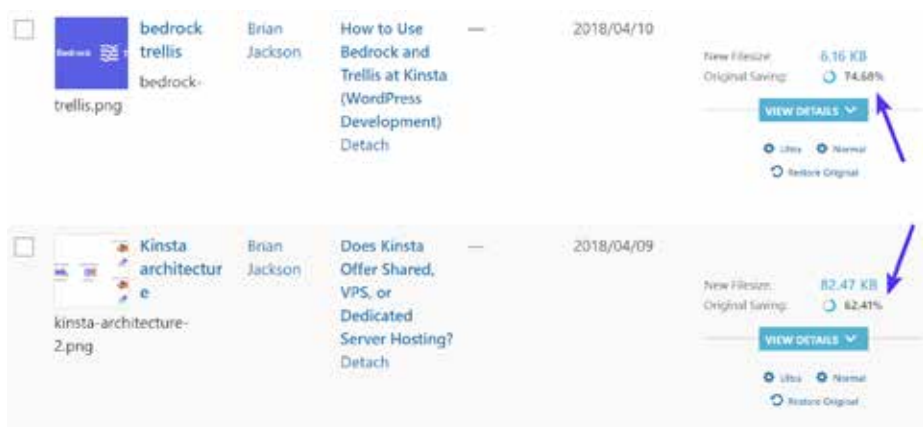
A grande notícia é que existem alguns plugins de compressão de imagem incríveis do WordPress que você pode usar para automatizar todo o processo. Aqui estão alguns plugins que recomendamos (todos disponíveis no WordPress.org):

- **Imagify** (com perdas e sem perdas - otimiza imagens externamente)
- **WP Smush** (com perdas e sem perdas - otimiza imagens externamente)
- **Optimole** (com perdas e sem perdas - otimiza imagens externamente)
- **EWWW Cloud** (com perdas e sem perdas - otimiza imagens externamente)
- **ShortPixel** (com perdas e sem perdas - otimiza imagens externamente)

O mais importante ao escolher um plugin de compressão de imagem é **usar um que comprima e otimize imagens externamente em seus servidores**. Isso, por sua vez, reduz a carga no seu site. Todos os acima fazem isso.

Se você estiver curioso, usamos o plugin Imagify no site da Kinsta. Ele compacta imagens automaticamente quando as carregamos para a biblioteca de mídia do WordPress. Então, nunca precisamos nos preocupar com nada. Com o tempo, você pode ter uma ideia do nível de compressão da imagem que deseja usar. Oferece Normal, Agressivo e Ultra.

Usamos o modo Agressivo na Kinsta e geralmente vemos **60-70% de economia** dependendo da imagem. Nota: nós usamos muito mais PNGs do que JPEGs devido ao fato de que a maioria das nossas imagens são ícones e ilustrações, não fotos.



Quanto mais rápido será o seu site WordPress se você usar compressão de imagem? Tudo depende do tamanho das suas imagens originais e do que elas são após a compressão. Entretanto, fizemos alguns testes de velocidade e descobrimos que uma solução de compressão de imagem de qualidade pode **diminuir o tempo de carregamento de páginas em mais de 80%!**

Lazy Loading

Se você tem muitas imagens, terá de considerar um carregamento preguiçoso “lazy loading”. Essa é uma técnica de otimização que carrega conteúdo visível, mas atrasa o download e a renderização do conteúdo que aparece abaixo da dobra.

Lazy loading funciona assim:

- O navegador cria a página da Web DOM sem baixar imagens e pré-carrega vídeos.
- O JavaScript é usado para determinar quais imagens baixar e quais vídeos serão pré-carregados com base no conteúdo inicialmente visível quando a página é carregada. Essas imagens e vídeos são baixados e pré-carregados, conforme apropriado.
- O download e a renderização de vídeos adicionais são atrasados até que um visitante do site role a página e um conteúdo adicional seja exibido.

O resultado final é que as imagens não são baixadas e os vídeos são carregados até que sejam realmente necessários. Isso pode fornecer um aumento significativo no desempenho de sites que incluem muitas imagens de alta resolução e vídeos incorporados.

Felizmente, existem muitos plugins WordPress gratuitos que facilitam bastante o carregamento lento (todos estão disponíveis no WordPress.org):

- **a3 Lazy Load**
- **Lazy Load by WP Rocket**
- **Lazy Load for Videos**

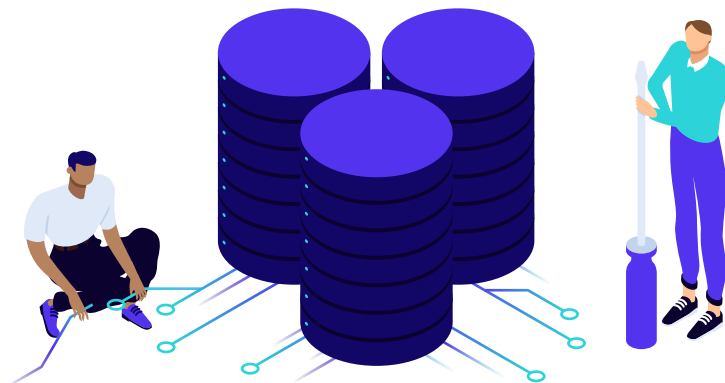
Dicas Adicionais de Otimização de Imagem

Aqui estão algumas dicas finais de otimização de imagens para você.

- Os dias de envio de imagens dimensionadas apenas para a largura da coluna ou DIV terminaram. As imagens responsivas funcionam prontas no WordPress (desde a versão 4.4) e exibem automaticamente tamanhos de imagem menores para usuários móveis.
- SVGs podem ser outra alternativa incrível ao uso de imagens. Todas as ilustrações desenhadas à mão que você vê no site da Kinsta são SVGs (vetores). SVGs são normalmente muito menores em tamanho de arquivo, embora nem sempre.
- Use fontes de ícones em vez de colocar texto nas imagens - elas ficam melhores quando dimensionadas e ocupam menos espaço. E se você usar um gerador de fontes, poderá otimizá-las ainda mais. Diminuímos o tamanho do nosso arquivo de fontes de ícones **em 97,59%** usando um gerador de fontes.
- Você também pode querer olhar para o WebP, um novo formato de imagem do Google que está rapidamente ganhando força.

CAPÍTULO 10:

Ajuste o seu Banco de Dados



Em seguida, há algumas dicas sobre como ajustar seu banco de dados do WordPress. Assim como um carro, seu banco de dados precisa de manutenção, pois com o tempo ele pode ter mais recursos do que você usaria até o ponto de desacelerar o sistema

Os sites de membros tornam isso mais complicado, pois geralmente **geram consultas mais complexas**, o que, por sua vez, adiciona latência adicional na recuperação das informações do banco de dados MySQL. Muito disso se deve a todas as partes móveis adicionais e a grandes quantidades de sites de dados como esses. Isso também pode ser causado por sites que dependem muito de consultas de pesquisa para navegação ou usam `WP_Query`.

Sem mencionar que você também tem grandes quantidades de usuários simultâneos consultando continuamente o banco de dados.

Use o InnoDB MySQL Storage Engine

Muitos sites antigos ainda estão usando o mecanismo de armazenamento MyISAM em seu banco de dados. Nos últimos anos, o InnoDB mostrou um desempenho melhor e mais confiável.

O InnoDB é como o óleo sintético, enquanto o MyISAM é regularizado.

Aqui estão algumas vantagens do InnoDB sobre o MyISAM:

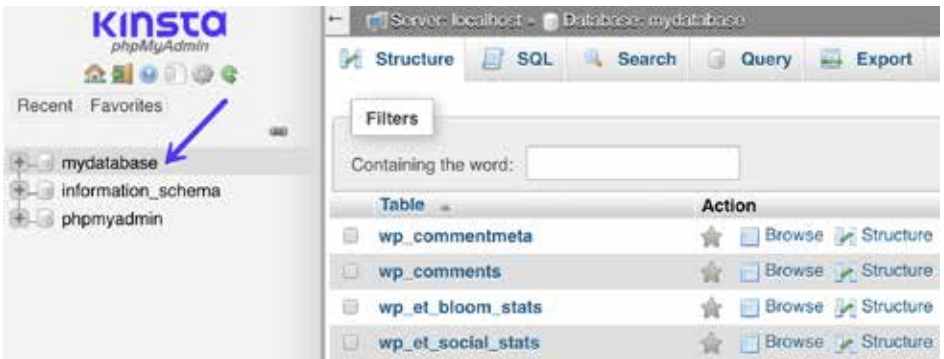
- InnoDB possui **bloqueio em nível de linha**. O MyISAM só tem bloqueio total no nível de tabela. Isso permite que suas consultas sejam processadas mais rapidamente.
- InnoDB possui o que é chamado de integridade referencial que envolve o suporte a **chaves estrangeiras** (RDBMS) e restrições de relacionamento, o MyISAM não (DMBS).
- InnoDB suporta **transações**, o que significa que você pode confirmar e reverter. MyISAM não.
- O InnoDB é mais confiável, pois usa logs transacionais para recuperação automática. MyISAM não.

Então, agora você pode estar se perguntando: você está executando o InnoDB ou o MyISAM? Se você estiver executando em um site WordPress relativamente novo, provavelmente já está usando o mecanismo de armazenamento InnoDB MySQL. Mas com sites mais antigos do WordPress você pode querer fazer uma verificação rápida. Alguns sites podem até mesclar e combinar tabelas MyISAM e InnoDB, nas quais você pode ver melhorias convertendo-as em todas as partes.

Siga estes passos simples abaixo para verificar.

Passo 1

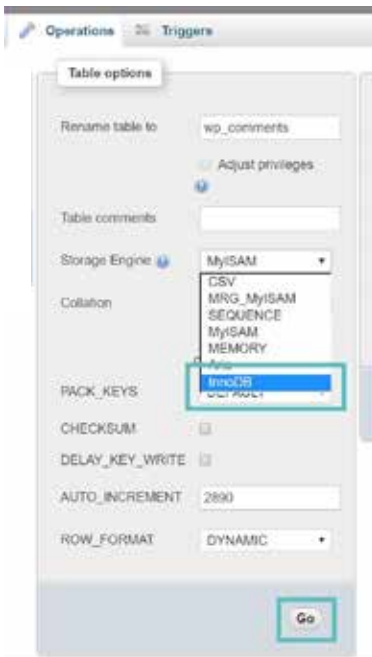
Entre no phpMyAdmin e clique no seu banco de dados MySQL.



Passo 2

Faça uma varredura rápida ou uma espécie de coluna “Tipo” e você poderá ver quais tipos de mecanismo de armazenamento suas tabelas estão usando. Neste exemplo abaixo, você pode ver que duas das tabelas ainda estão usando o MyISAM.

| | | | | | | | | | |
|--------|-----------|--------|--------|-------|------|-----|--------|--------------------|---------|
| Browse | Structure | Search | Insert | Empty | Drop | 0 | InnoDB | utf8_unicode_ci | 64 K B |
| Browse | Structure | Search | Insert | Empty | Drop | 0 | InnoDB | utf8_unicode_ci | 48 K B |
| Browse | Structure | Search | Insert | Empty | Drop | 0 | InnoDB | utf8_unicode_ci | 64 K B |
| Browse | Structure | Search | Insert | Empty | Drop | 0 | InnoDB | utf8_unicode_ci | 80 K B |
| Browse | Structure | Search | Insert | Empty | Drop | 0 | InnoDB | utf8_unicode_ci | 48 K B |
| Browse | Structure | Search | Insert | Empty | Drop | 0 | InnoDB | utf8mb4_unicode_ci | 32 K B |
| Browse | Structure | Search | Insert | Empty | Drop | 113 | MyISAM | utf8mb4_unicode_ci | 80 K B |
| Browse | Structure | Search | Insert | Empty | Drop | 471 | MyISAM | utf8mb4_unicode_ci | 208 K B |
| Browse | Structure | Search | Insert | Empty | Drop | 0 | InnoDB | utf8mb4_unicode_ci | 48 K B |
| Browse | Structure | Search | Insert | Empty | Drop | 1 | InnoDB | utf8mb4_unicode_ci | 16 K B |



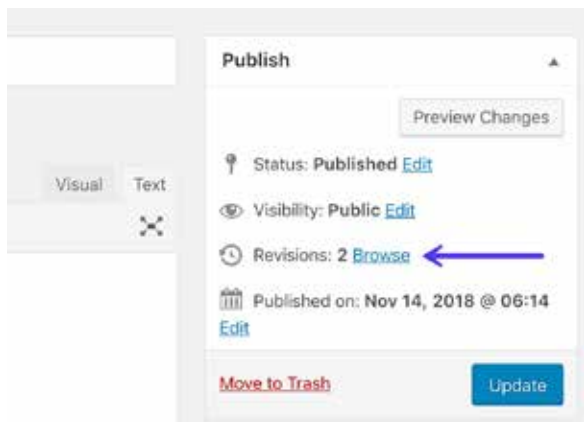
Se você encontrou alguns, provavelmente é hora de movê-los para o InnoDB. Sempre recomendamos entrar em contato com seu host e perguntar se ele pode fazer isso por você. Na Kinsta, as tabelas de banco de dados de cada cliente são automaticamente convertidas para InnoDB por nossa equipe de migração.

Mas você também pode convertê-los manualmente com o phpMyAdmin. Basta clicar na tabela myISAM, clicar na guia “Operações” e alterar o mecanismo de armazenamento.

Se você é um desenvolvedor, outra alternativa seria convertê-los com o WP-CLI.

Excluir e Limitar as Revisões de Página e Postagem

Sempre que você salva uma página ou publica no WordPress, ela cria o que é chamado de revisão. Isso ocorre em rascunhos e postagens já publicadas atualizadas. As revisões podem ser úteis caso você precise reverter para uma versão anterior do seu conteúdo.



No entanto, as revisões também podem prejudicar o desempenho do seu site WordPress. Em sites grandes, isso pode aumentar muito rapidamente milhares de linhas no banco de dados que não são necessariamente necessárias. E quanto mais linhas você tiver, maior será o tamanho do banco de dados, o que ocupará espaço de armazenamento. Embora os índices tenham sido criados para esse propósito, ainda vimos esse problema aleijando os sites do WordPress. Há algumas coisas que você pode fazer.

1. Excluir Revisões Antiga

Se você tiver um site WordPress antigo com muitas páginas e postagens, talvez seja hora de fazer uma limpeza rápida e excluir essas revisões antigas. Você pode fazer isso com o MySQL, mas com todos os fragmentos de código flutuando na web, recomendamos fazer um backup do seu site e usar um plugin gratuito como o WP-Sweep.

Outro dos nossos plugins favoritos, o WP Rocket, também possui um recurso de otimização de banco de dados para limpar as revisões.

Se você receber um erro, talvez seja necessário primeiro instalar o pacote wp-revisions-cli com o seguinte comando:

```
wp package install trepmal/wp-revisions-cli
```

Você pode então executar o seguinte comando para limpar as revisões:

```
wp revisions clean
```

2. Limite de Revisões

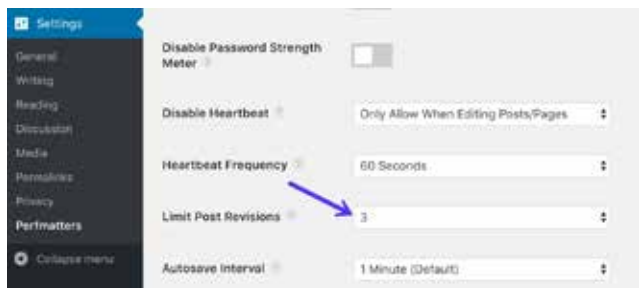
Outra boa estratégia é uma que usamos na Kinsta é limitar o número de revisões que podem ser armazenadas por postagem ou página. Mesmo configurando para algo como dez, as revisões não ficarão fora de controle, especialmente se você fizer muitas atualizações.

Para limitar as revisões, você pode adicionar o seguinte código ao seu arquivo `wp-config.php`. O código abaixo precisa ser inserido acima do “ABSPATH”; caso contrário, não funcionará. Você pode alterar o número para muitas revisões que deseja manter armazenadas no banco de dados.

```
define('WP_POST_REVISIONS', 10);
```

```
60
61  /**
62   * WordPress Database Table prefix.
63   * You can have multiple installations in one database if you give each
64   * a unique prefix. Only numbers, letters, and underscores please!
65   */
66  $table_prefix = 'wp_';
67
68  define('WP_POST_REVISIONS', 10); ←
69
70  /* That's all, stop editing! Happy blogging. */
71
72  /** Absolute path to the WordPress directory. */
73  if ( ! defined( 'ABSPATH' ) )
74      define( 'ABSPATH', dirname( __FILE__ ) . '/' );
75
76  /** Sets up WordPress vars and included files. */
77  require_once ABSPATH . 'wp-settings.php';
78
```

Line 28, Column 1



Limit revisions with Perfmatters

Ou você pode utilizar um plugin premium como o Perfmatters (disponível em perfmatters.io) para limitar as revisões.

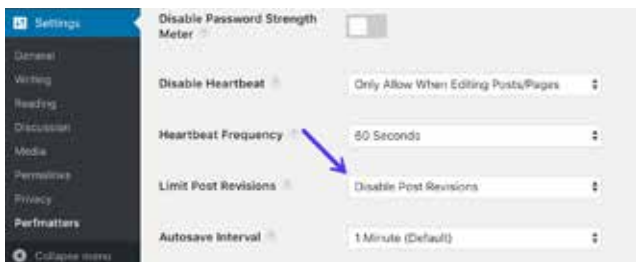
3. Desativar Revisões

E por último, mas não menos importante, você também pode desativar completamente as revisões em seu site. Se você estiver seguindo esse caminho, é altamente recomendável seguir a primeira opção acima para excluir as revisões e desativá-las depois. Desta forma, a sua base de dados está completamente livre de todas as revisões antigas e nenhuma nova será adicionada daqui para frente.

Para desabilitar revisões, você pode adicionar o seguinte código ao seu arquivo `wp-config.php`. O código abaixo precisa ser inserido acima do “ABSPATH”, caso contrário, não funcionará.

```
define('WP_POST_REVISIONS', false);
```

```
60  /**
61   * WordPress Database Table prefix.
62   *
63   * You can have multiple installations in one database if you give each
64   * a unique prefix. Only numbers, letters, and underscores please!
65   */
66  $table_prefix = 'wp_';
67
68  define('WP_POST_REVISIONS', false); ←
69
70  /* That's all, stop editing! Happy blogging. */
71
72  /** Absolute path to the WordPress directory. */
73  if ( ! defined( 'ABSPATH' ) )
74      define( 'ABSPATH', dirname( __FILE__ ) . '/' );
75
76  /** Sets up WordPress vars and included files. */
77  require_once ABSPATH . 'wp-settings.php';
78
```



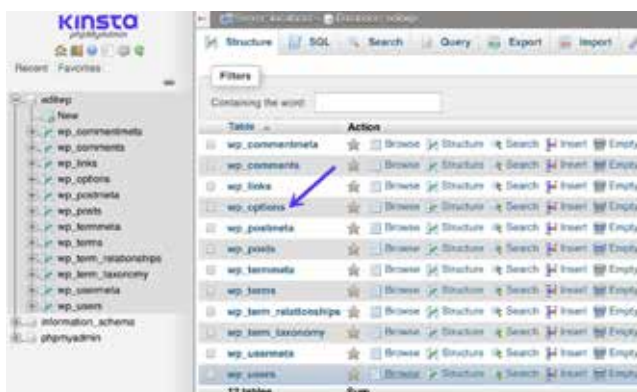
Disable revisions with Perfmatters

Ou, novamente, você pode utilizar um plugin como o Perfmatters para desabilitar revisões.

Limpar Sua Tabela `wp_options` e Dados Carregados Automaticamente

A tabela `wp_options` geralmente é ignorada quando se trata do desempenho geral do WordPress e do banco de dados. Especialmente em sites antigos e grandes, isso pode ser facilmente o culpado por tempos de consulta lentos em seu site, devido aos dados carregados automaticamente que são deixados para trás dos plugins e temas de terceiros. Confie em nós; vemos isso todos os dias!

A tabela `wp_options` contém todos os tipos de dados para o seu site WordPress, como:



- URL do site, URL inicial, email do administrador, categoria padrão, postagens por página, formato de hora, etc.
- Configurações para plugins, temas, widgets
- Dados temporariamente armazenados em cache

Esta tabela contém os seguintes campos (colunas):

- option_id
- option_name
- option_value
- **autoload** (este é o que nos preocupamos quando se trata de performance)



| | option_id | option_name | option_value | autoload |
|---|-----------|-----------------|-----------------------------|----------|
| <input type="checkbox"/> Edit Copy Delete | 1 | siteurl | https://editwp.com | yes |
| <input type="checkbox"/> Edit Copy Delete | 2 | home | https://editwp.com | yes |
| <input type="checkbox"/> Edit Copy Delete | 3 | blogname | editwp | yes |
| <input type="checkbox"/> Edit Copy Delete | 4 | blogdescription | Just another WordPress site | yes |

Uma das coisas importantes para entender sobre a tabela `wp_options` é o campo de **autoload**. Isto contém um sim ou um valor no (flag). Isso essencialmente controla se é ou não carregado pela função `wp_load_all_options()`. Dados carregados automaticamente **são dados carregados em todas as páginas** do seu site WordPress. Assim como mostramos a você como desativar determinados scripts do carregamento em todo o site, a mesma ideia se aplica aqui. O atributo `autoload` é definido como “yes” por padrão para os desenvolvedores, mas nem todos os plugins devem, teoricamente, carregar seus dados em todas as páginas.

O problema que os sites WordPress podem encontrar é quando há uma grande quantidade de dados carregados automaticamente na tabela `wp_options`. Isso geralmente é resultado de:

- Dados que estão sendo carregados automaticamente por um plugin quando deveria ser definido como “não”. Um bom exemplo disso seria um plugin de formulário de contato. Precisa carregar dados em todas as páginas ou apenas na página de contato?
- Plugins ou temas foram removidos do site WordPress, mas suas opções ainda são deixadas para trás na tabela `wp_options`. Isso pode significar que dados `autoloaded` desnecessários estão sendo consultados em cada solicitação.
- Desenvolvedores de plugins e de temas estão carregando dados na tabela `wp_options` em vez de utilizar suas próprias tabelas. Há argumentos para ambos os lados disso, pois alguns desenvolvedores preferem plugins que não criam tabelas adicionais.

No entanto, a tabela `wp_options` também não foi projetada para armazenar milhares de linhas.

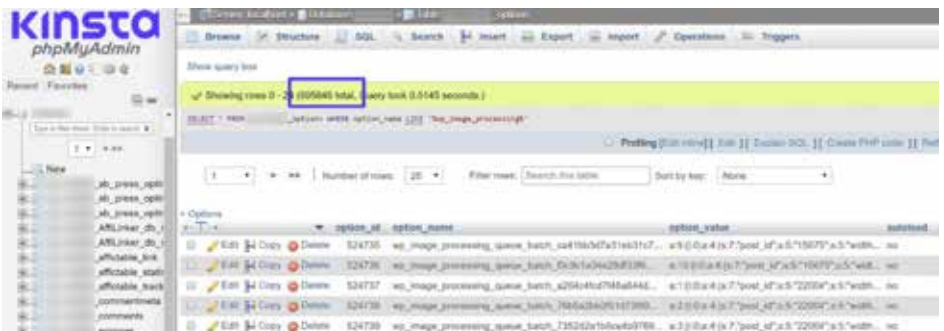
Quando sei que estou auto-carregado de dados? Isso pode variar, claro, mas, idealmente, você quer que isso esteja entre 300 KB e 1 MB. Uma vez que você comece a se aproximar da faixa de 3 a 5 MB ou mais, provavelmente há coisas que podem ser otimizadas ou removidas do autoloading. E qualquer coisa acima de 10 MB deve ser endereçada imediatamente. Isso nem sempre significa que isso vai causar um problema, mas é um bom ponto de partida.

Limpar os dados carregados automaticamente é algo que ajudamos os clientes todo o tempo na Kinsta.

Quando foi a última vez que você limpou sua tabela `wp_options`? Ya ... nós pensamos assim. Vá em frente!

Limpar Transientes

A menos que você esteja usando um cache de objetos, o WordPress armazena registros temporários na tabela `wp_options`. Normalmente, eles recebem um tempo de expiração e devem desaparecer com o tempo. No entanto, esse não é sempre o caso. Vimos alguns bancos de dados onde existem milhares de antigos registros temporários. De fato, em um site, lidamos com alguns registros transientes corromptos nos quais mais de **695.000 linhas foram geradas** na tabela `wp_options`. Caramba!

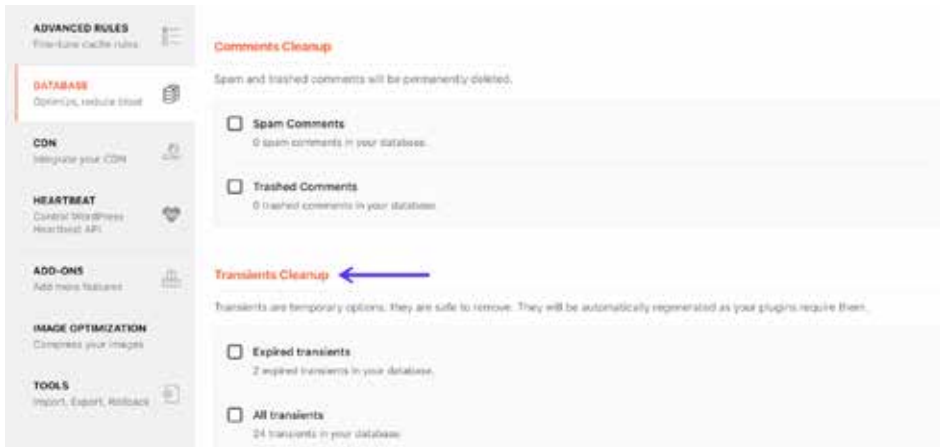


Também é importante observar que os transientes não são carregados automaticamente por padrão. Você poderia usar uma consulta como abaixo para ver se há dados temporários carregados automaticamente.

```
SELECT *  
FROM `wp_options`  
WHERE `autoload` = 'yes'  
AND `option_name` LIKE '%transient%'
```

Uma opção melhor e mais segura seria utilizar um plugin gratuito como Transient Cleaner ou Delete Expired Transients (ambos disponíveis no WordPress.org), que pode limpar apenas os transientes expirados da sua tabela `wp_options`. No entanto, parece que agora há uma função no WordPress, adicionada em 4.9, que limpa os transientes expirados. Então esperamos que isso esteja acontecendo automaticamente no seu site agora.

O WP Rocket também tem a capacidade de limpar transientes em suas opções de otimização de banco de dados.



Limpar Sessões WordPress

Outro problema comum que vemos é que, às vezes, as tarefas do cron ficam fora de sincronia ou não são disparadas corretamente e portanto, as sessões não são limpas.

Você pode acabar obtendo toneladas de linhas `_wp_session_` no seu banco de dados. Neste exemplo abaixo, o site em questão acabou com mais de 3 milhões de linhas em sua tabela `wp_options`. E a mesa cresceu para mais de 600 MB.



Você pode usar uma consulta como descrito abaixo para ver se você está encontrando esse problema:

```
SELECT *
FROM `wp_options`
WHERE `option_name` LIKE '_wp_session_%'
```

A screenshot of a database query result. The top part shows a filter for 6491 rows. Below is a table with columns: `option_id`, `option_name`, `option_value`, and `autoload`. The rows show various `_wp_session_` entries with their IDs and values. The `autoload` column is set to 'no' for all entries.

| option_id | option_name | option_value | autoload |
|-----------|--|---|----------|
| 3301835 | <code>_wp_session_254a6d16b4400d5aabf3c43b41b72d2b</code> | <code>a:1:{s:10:"edd_errors";a:0:''}</code> | no |
| 3301836 | <code>_wp_session_expires_254a6d16b4400d5aabf3c43b41b72d2b</code> | 1520801306 | no |
| 3301837 | <code>_wp_session_747c86aefae52140d1068e90f240249f</code> | <code>a:1:{s:10:"edd_errors";a:0:''}</code> | no |
| 3301838 | <code>_wp_session_expires_747c86aefae52140d1068e90f240249f</code> | 1520801306 | no |
| 3301839 | <code>_wp_session_17aa53659273873116d63993bae7cfdf8</code> | <code>a:1:{s:10:"edd_errors";a:0:''}</code> | no |
| 3301840 | <code>_wp_session_expires_17aa53659273873116d63993bae7cfdf8</code> | 1520801306 | no |
| 3301841 | <code>_wp_session_f3f29822e0c97951504e4c096af0c81</code> | <code>a:1:{s:10:"edd_errors";a:0:''}</code> | no |
| 3301842 | <code>_wp_session_expires_f3f29822e0c97951504e4c096af0c81</code> | 1520801306 | no |
| 3301843 | <code>_wp_session_264be94abb1e756ee229506e1c26e50e</code> | <code>a:1:{s:10:"edd_errors";a:0:''}</code> | no |

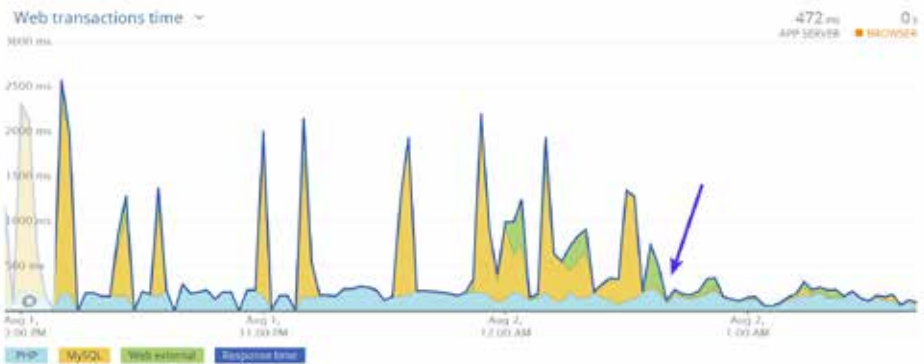
Na maioria dos casos você pode então apagar estes (como um cron job deve ter) com o seguinte comando:

```
DELETE FROM `wp_options`
WHERE `option_name` LIKE '_wp_session_%'
```

Depois de limpar todas as sobras do `_wp_session_rows` a tabela tinha menos de 1.000 linhas e foi reduzida para 11 MB de tamanho.



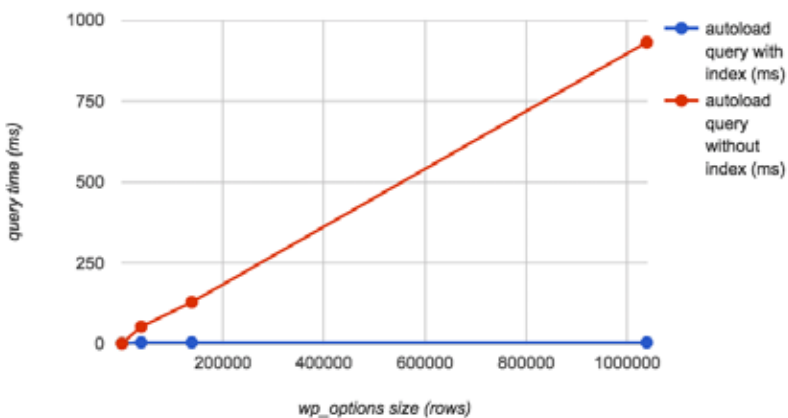
Ele também corrigiu os picos que o site estava recebendo no MySQL.



Adicionar um índice ao Autoload

Se a limpeza da sua tabela `wp_options` não fosse suficiente, você poderia tentar adicionar um “índice” ao campo de carga automática. Isto essencialmente pode ajudá-lo a ser pesquisado de forma mais eficiente. A incrível equipe do 10up executou alguns cenários de teste em uma tabela `wp_options` com um número típico de registros carregados automaticamente para mostrar como adicionar um índice de carga automática às consultas `wp_options` pode aumentar o desempenho.

tamanho das opções `wp_options` vs tempo de consulta



Fonte: 10up

Use o Redis Como um Cache de Objeto Persistente para WordPress

Redis é uma loja de estrutura de dados de código aberto, na memória. No contexto do WordPress, o Redis pode ser usado para armazenar os valores gerados pelo cache de objetos nativos do WordPress de forma persistente para que os objetos em cache possam ser reutilizados entre as cargas de páginas.

Usar um cache de objeto persistente como o Redis permite a reutilização de objetos em cache em vez de exigir que o banco de dados MySQL seja consultado uma segunda vez para o mesmo objeto. O resultado é que o Redis pode reduzir a carga no banco de dados MySQL de um site, diminuindo simultaneamente o tempo de resposta do site e aumentando a capacidade do site de escalar e lidar com tráfego adicional.



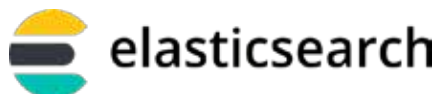
Sites altamente dinâmicos (WooCommerce, sites de membros, fóruns, grupos de discussão, blogs com sistemas de comentários extremamente ativos) que não podem fazer bom uso do cache de páginas são candidatos potenciais para uma opção de cache de objetos persistentes, como o Redis.

Se você é um cliente Kinsta, nós oferecemos um add-on Redis.

Utilize o Elasticsearch para Acelerar a Pesquisa do WordPress

O Elasticsearch é um mecanismo de busca de texto completo de código aberto. Ele é usado para indexar dados e pesquisar esses dados incrivelmente rapidamente.

No contexto do WordPress, o Elasticsearch pode ser usado para acelerar a consulta do banco de dados do WordPress. Isso é feito criando um índice do conteúdo do banco de dados do seu site e, em seguida, usando o Elasticsearch para pesquisar esse índice muito mais rapidamente do que uma consulta MySQL é capaz de realizar a mesma pesquisa.



Se você tiver tempo e habilidade, o Elasticsearch pode ser integrado com um site do WordPress por um desenvolvedor altamente experiente do WordPress e do Elasticsearch. Se o seu site usa relativamente o WP_Query como padrão, o Elasticsearch também pode ser integrado com a instalação do ElasticPress, um plugin WordPress gratuito de 10up, disponível no WordPress.org, que se integra automaticamente com o objeto WP_Query para gerar resultados de consultas com o Elasticsearch em vez do MySQL.

Qualquer site que faça uso intensivo do WP_Query pode se beneficiar do Elasticsearch. Exemplos de sites que podem se beneficiar do Elasticsearch:

- Sites onde a busca é o principal meio de navegação.
- WooCommerce sites com um grande número de ordens onde os administradores do site precisam ser capazes de pesquisar a lista de ordens regularmente.
- Qualquer site com um grande número de posts onde as consultas MySQL estão produzindo resultados inaceitavelmente lentos.

Assim como com o Redis, também temos um add-on do Elasticsearch.

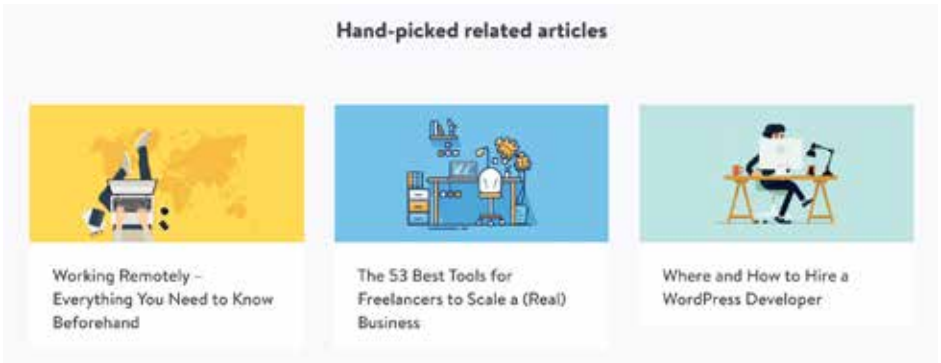
Desativar Recursos Não Críticos que São Intensivos em Banco de Dados

Isso pode parecer um pouco óbvio, mas pode fazer uma grande diferença se você desabilitar plugins não-críticos e recursos temáticos que são intensivos em banco de dados.

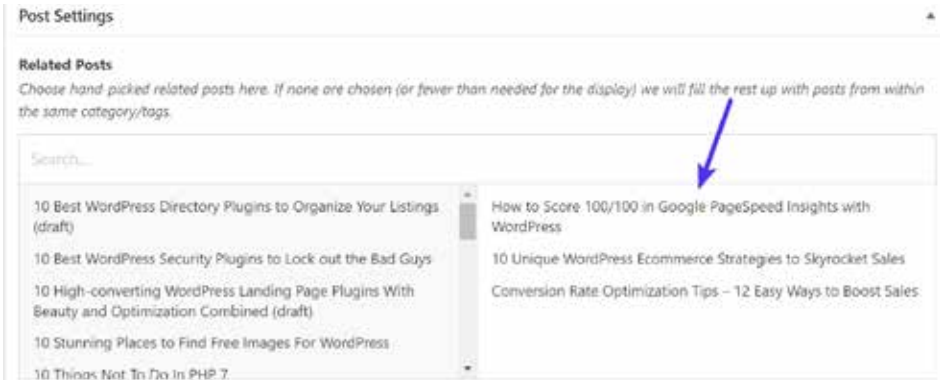
- Widgets e plugins populares e ou relacionados são horríveis. Eles tipicamente têm consultas pesadas em todo o site.
- Plugins de otimização de imagem que comprimem imagens usando seu servidor. Você deve sempre usar um plugin de otimização de imagem que otimize as imagens externamente.

Se você visitar o blog Kinsta e descer até o final de um post, você vai notar que temos o que chamamos de artigos relacionados “escolhidos a dedo”. Estes são selecionados por nós manualmente e atribuídos ao post. Isso reduz a consulta a quase nada e não

prejudica o desempenho de todo o seu site. É preciso mais trabalho? Sim, mas pode ser ainda melhor porque pode escolher o que quer que os leitores vejam.



Então, como é que conseguimos isto? Usamos o incrível plugin Advanced Custom Fields (disponível no WordPress.org) e depois atribuímos esses campos ao tipo de post do nosso blog. Isso nos permite pesquisar e atribuir qualquer conteúdo relacionado que queiramos para cada um dos posts do nosso blog (como visto abaixo)



Também recomendamos ficar longe de plugins que adicionam um contador de visualizações/posts ao seu site, a menos que você precise absolutamente dele. Por exemplo, evite coisas como “792 mensagens” ao lado do avatar de um usuário em mensagens do fórum ou “5.243 visualizações” ao listar mensagens do fórum. Quando você tem uma discussão longa, estes contadores terão um enorme custo na sua base de dados. Em geral, minimize o uso de contadores e só os use se necessário.y.

Isto também se aplica a muitos contadores sociais. Por exemplo, neste site abaixo você pode ver o tempo de resposta do popular plugin Social Warfare é 30x mais do que o próximo plugin abaixo dele. Caching é ativado, mas, obviamente, este plugin tem um custo de desempenho considerável. Depois de desativar o plugin no site, os tempos de carga melhoraram instantaneamente e a capacidade de resposta do painel de administração do WordPress melhorou.

Sort by Most time consuming

| | |
|--------------------|----------|
| social-warfare | 18.9 sec |
| wordpress-seo | 624 ms |
| social-warfare-pro | 449 ms |
| generatepress | 425 ms |
| wp-show-posts | 389 ms |
| aaawp | 356 ms |
| gp-premium | 229 ms |
| nogluten | 160 ms |
| imagify | 115 ms |
| wp-review | 107 ms |

Top 5 plugins and themes by response time



CAPÍTULO 11:

Utilizar uma Rede de Entrega de Conteúdo (CDN)



CDN é a abreviatura de rede de distribuição de conteúdo. Estes são uma rede de servidores (também conhecidos como POPs) localizados ao redor do globo. Eles são projetados para hospedar e fornecer cópias do conteúdo estático (e às vezes dinâmico) do seu site WordPress, como imagens, CSS, JavaScript e fluxos de vídeo.

Em primeiro lugar, você não quer confundir uma CDN com o seu host WordPress. Estes são serviços totalmente separados. Uma CDN não é um substituto para o seu provedor de hospedagem, mas sim uma maneira adicional de aumentar a velocidade do seu site. Enquanto a nossa hospedagem aqui na Kinsta é muito rápida, uma CDN pode tornar o seu site ainda mais rápido

Como uma CDN Funciona

Como funciona exactamente uma CDN? Bem, por exemplo, quando você hospeda seu site com Kinsta você tem que escolher um local físico de data center, como EUA, Europa, Ásia-Pacífico ou América do Sul.

Digamos que você escolha o US Central. Isso significa que seu site está fisicamente localizado em um “host server” em Council Bluffs, Iowa. Quando as pessoas na Europa visitam o seu site, vai demorar mais tempo para que ele seja carregado versus alguém visitá-lo de Dallas, TX.

Por que? Porque os dados têm de viajar mais longe. Isso é o que é conhecido como latência. Latência refere-se ao tempo e ou atraso que está envolvido na transmissão de dados através de uma rede. Quanto maior a distância, maior a latência.

Tipos de CDNs

Existem dois tipos diferentes de redes de distribuição de conteúdo:

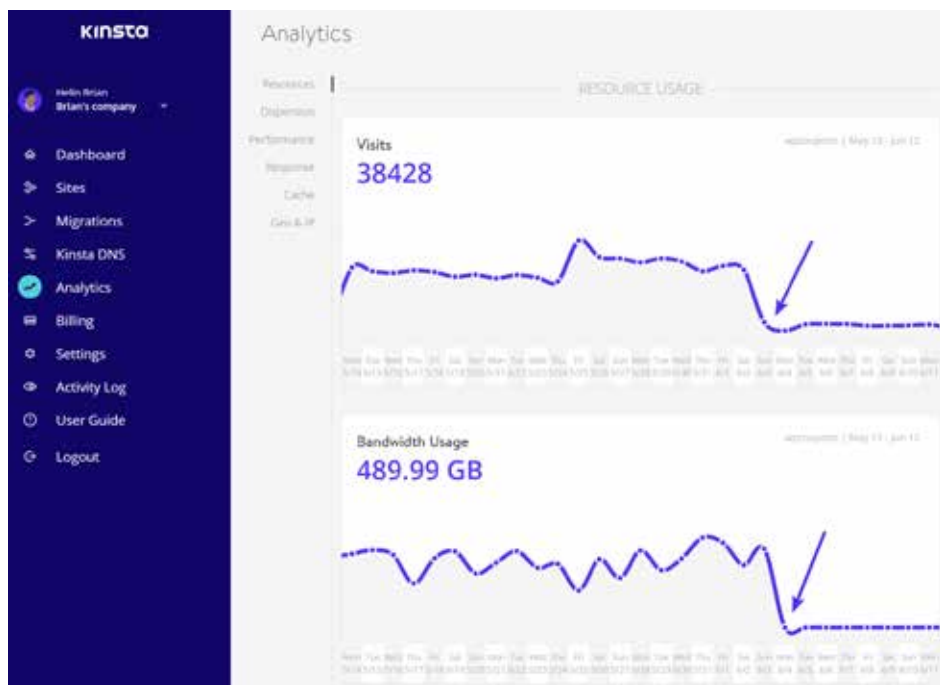
1. CDN Pull Tradicional
2. CDN Proxy Reverso

Os CDNs tradicionais pull armazenam uma cópia de todo o seu conteúdo e mídia, mas um pedido do cliente ainda é feito diretamente ao seu provedor de hospedagem. KeyCDN e CDN77 são exemplos de CDNs tradicionais.

Um CDN proxy reverso é ligeiramente diferente. Enquanto ele ainda atua como um CDN, ele intercepta todas as solicitações de entrada e atua como um servidor intermediário entre o cliente e seu host. Cloudflare e Sucuri são exemplos de CDNs de proxy reverso. Esta é uma razão pela qual você tem que apontar seu DNS diretamente para estes provedores ao invés de seu host.

O benefício destes é porque eles agem como um servidor intermediário, eles podem fornecer forte aplicação web firewalls que podem ajudar a bloquear o tráfego ruim de sempre bater o seu site WordPress e ou provedor de hospedagem. Uma queda a este é que vêm com uma sobrecarga adicional pequena nos termos do desempenho comparado a uma tração CDN tradicional. Mas com desempenho adicional e características de segurança, isto poderia ser discutido como negligenciável.

Abaixo está um exemplo do que aconteceu depois de habilitar o Sucuri no site de um cliente. Como você pode ver, ele teve um impacto dramático na quantidade de tráfego ruim que estava chegando. No final, esses tipos de serviços podem ajudá-lo a economizar em seus custos de hospedagem.



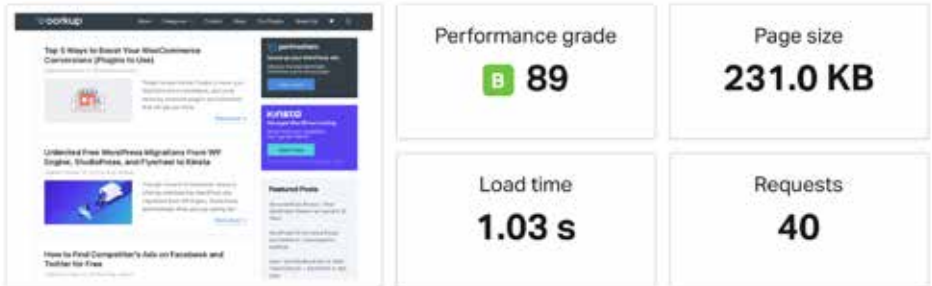
Testes de Velocidade de CDN

Há pouco falamos sobre os enormes benefícios do cache do WordPress. Bem, caching em CDN também é super poderoso. Isto é porque as CDNs tipicamente têm muito mais localizações de servidores do que os provedores de hospedagem. Isto significa que eles podem armazenar todos os seus ativos (imagens, JS, CSS) mais perto de seus visitantes e servi-los em velocidades rápidas de relâmpago.

Deixe-nos fazer alguns testes rápidos para ver apenas quanto mais rápido seu local poderia ser com um CDN.

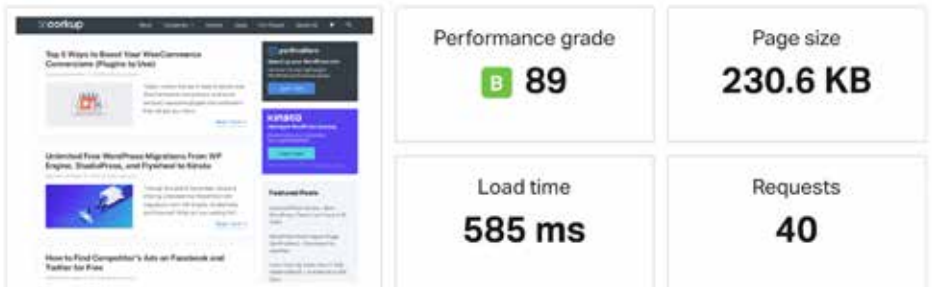
Sem CDN

Nosso site de testes está hospedado na Kinsta e está fisicamente localizado no data center de Iowa, EUA. Primeiro fizemos cinco testes de velocidade em Pingdom (sem a CDN habilitada), e fizemos a média. Importante: Estamos usando a localização Europa - Reino Unido - Londres em Pingdom para demonstrar o poder real de uma CDN. O tempo total de carga foi de **1,03 ms**.



Com CDN

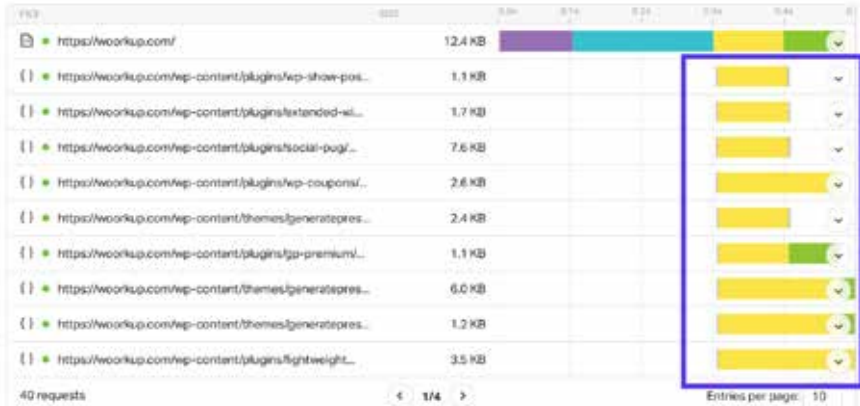
Depois, ativamos a nossa CDN e fizemos cinco testes de velocidade adicionais em Pingdom. Nosso tempo total de carga é agora de **585 ms** do local de testes Europa - Reino Unido - Londres. Assim, ao utilizar a CDN, conseguimos **diminuir o tempo de carregamento de páginas em 43,2%**! Isto é enorme.



A razão para uma diferença tão drástica é porque a CDN tem um centro de dados em Londres. Isto significa que todos os ativos são armazenados em cache naquele local e prontos para serem servidos com latência mínima.

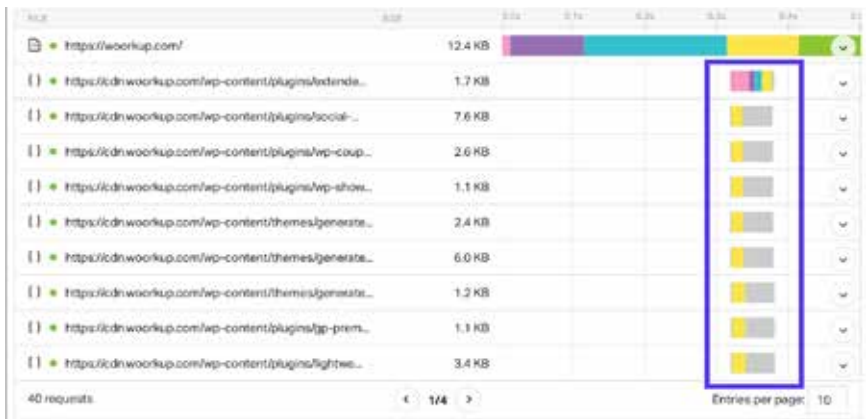
TTFB sem CDN

Lembre-se que a barra amarela em Pingdom significa tempo de espera, que é tempo para o primeiro byte (TTFB). Em nossos testes de velocidade sem a CDN rodando o TTFB médio em ativos foi de cerca de 98 ms.



TTFB com CDN

Uma vez habilitada a CDN, a média do TTFB dos ativos caiu para uma média de **15 ms**. Então, usando uma CDN, nosso TTFB **médio caiu 84,69%**. Isso ocorre principalmente porque os ativos estavam sendo servidos diretamente do cache da CDN.



*A CDN diminuiu o nosso tempo de carregamento de páginas em 43,2%!
Confira por que você deveria estar usando um.*

Como Habilitar uma CDN

Habilitar uma CDN no seu site WordPress não precisa ser difícil, é muito fácil! Basta seguir estes passos

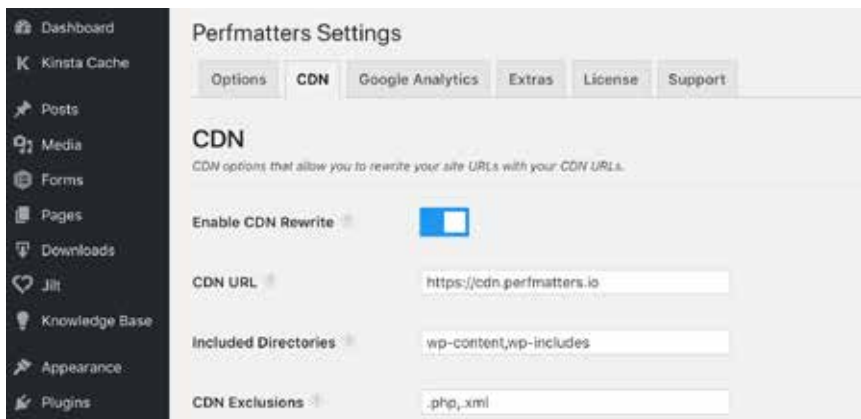
Passo 1

Selecione um provedor CDN e assine seu serviço. Estes são normalmente cobrados mensalmente ou por uso de dados. A maioria dos provedores terá uma calculadora para estimar seus custos.

- Cada provedor de CDN deve ter documentação em seus sites para ajudá-lo a começar.
- Também temos tutoriais detalhados sobre como instalar o Cloudflare e o Sucuri disponíveis em nosso site em kinsta.com.

Passo 2

Se você está usando uma CDN pull tradicional, você pode utilizar plugins gratuitos como CDN Enabler (disponível em WordPress.org), WP Rocket (disponível em wp-rocket.me), ou Perfmatters (disponível em perfmatters.io) para integrá-lo ao seu site WordPress. Estes plugins ligam automaticamente os seus ativos ao CDN. Não é necessário nenhum trabalho da sua parte para obter o seu conteúdo na CDN; isto é tudo mãos-livres! Os CDNs com Proxy Reverso normalmente não requerem plugins, embora às vezes eles os tenham para habilitar recursos adicionais.



Enable CDN in WordPress with Perfmatters

Como Ativar Kinsta CDN

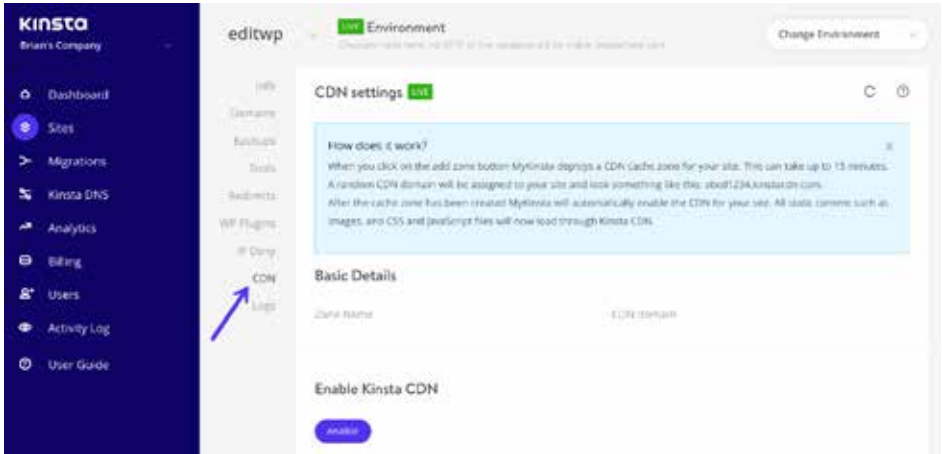
Gostou dos testes de velocidade da CDN acima? Estávamos usando KeyCDN nesses testes. A grande notícia é que a Kinsta CDN é alimentada por KeyCDN. É uma rede de distribuição de conteúdo HTTP/2 e IPv6 com mais de 34 localizações, para turbocar seus ativos e mídias ao redor do globo. As regiões atualmente servidas incluem América, América do Sul, Europa, África, Ásia e Austrália.



Se você é um cliente Kinsta, nós incluímos gratuitamente a largura de banda CDN em todos os nossos planos de hospedagem. Você pode habilitar a Kinsta CDN em dois simples passos.

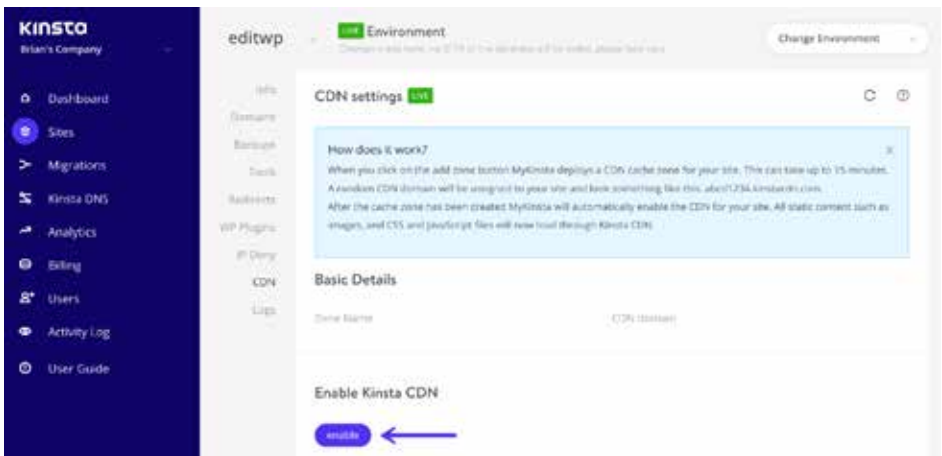
Passo 1

Primeiro, inicie sessão no seu painel MyKinsta. Clique no seu site e depois no separador CDN.



Passo 2

Depois clique em “Ativar”. Após alguns minutos, a CDN é implantada automaticamente, e seus ativos serão servidos pelo cache ao redor do mundo. Isso é tudo o que há para ele.



Otimizações CDN adicionais

Aqui estão algumas otimizações adicionais de CDN que você pode querer verificar ou pensar.

- Se você tem muitos comentários, os gravatares podem gerar muitos pedidos e carregar a partir de secure.gravatar.com. Em vez disso, pode carregar os gravatares a partir da sua CDN. Fazemos isto no site da Kinsta. Se você é um cliente Kinsta, entre em contato com nossa equipe de suporte para obter esta configuração.
- Você também pode hospedar fontes personalizadas, fontes do Google e até mesmo seu favorito em seu CDN. Links para tutoriais sobre como fazer isso estão disponíveis em nosso site em <https://kinsta.com/speed-up-wordpress>

CAPÍTULO 12:

Transfira Mídia e E-mail Quando Necessário



Tudo o que gera uma solicitação tem um impacto no desempenho do seu site de uma forma ou de outra. Para sites com hospedagem de centenas de milhares de arquivos ou grandes mídias, pode ser sábio descarregar isso completamente. Descarregar é diferente de o servir através de uma CDN. Com uma CDN os dados originais ainda residem no seu host, a CDN simplesmente tem várias cópias dela.

Quando o cache expira em seus ativos CDN, ele consulta novamente o host para obter as últimas cópias dos arquivos. Os CDNs são feitos para armazenar arquivos em cache por longos períodos de tempo. Mas devido ao fato de que eles têm tantos POPs, poderia haver um monte de re-querying acontecendo como cache expira em diferentes regiões.

Quando você descarrega mídia ou arquivos, isso significa realmente mover a localização física original deles para fora do seu provedor de hospedagem. Assim, embora possa parecer que os arquivos são servidos a partir do seu site, eles estão realmente localizados em outro lugar inteiramente. Além de reduzir as consultas adicionais de volta ao host, a razão número um, obviamente, é também economizar espaço em disco.

Transferir Mídia para o Amazon S3

Uma das soluções de descarregamento mais populares é o Amazon S3. O Amazon S3 é uma solução de armazenamento e faz parte da Amazon Web Services muitos produtos.

Normalmente, isso é usado para grandes sites que precisam de backups adicionais ou estão servindo arquivos grandes (downloads, software, vídeos, jogos, arquivos de áudio, PDFs, etc.). Amazon tem um histórico comprovado de ser muito confiável, e por causa de sua infraestrutura maciça, eles podem oferecer custos de armazenamento muito baixos. Alguns dos clientes do S3 incluem Netflix, Airbnb, SmugMug, Nasdaq, etc.

Porque eles lidam inteiramente com armazenamento em massa, você pode quase garantir que o preço será mais barato do que a sua hospedagem WordPress. O download de mídia



para a AWS pode ser uma ótima maneira de economizar dinheiro e é gratuito no primeiro ano (até 5 GB de armazenamento). Além disso, como as solicitações de mídia são atendidas diretamente da Amazon, isso coloca menos carga no site do WordPress, o que significa tempos de carregamento mais rápidos.

Recomendamos o uso do plugin WP Offload Media Lite for Amazon S3 (disponível no WordPress.org) para isso. O plugin copia automaticamente os ficheiros do seu site WordPress para o Amazon S3 à medida que são carregados para a biblioteca de multimídia. Você também pode usar uma CDN com a mídia transferida para o melhor dos dois mundos.

Transferir Mídia para o Google Cloud Storage

Outra solução de descarregamento popular é o Google Cloud Storage. Desde que Kinsta é



alimentado pelo Google Cloud Platform, somos grandes fãs de sua tecnologia e infra-estrutura. Devido à infraestrutura massiva do Google e ao fato de que eles lidam com armazenamento em massa, eles podem oferecer custos de armazenamento muito baixos. Alguns de seus clientes incluem Spotify, Vimeo, Coca-Cola, Philips, Evernote e Motorola.

Para tal, recomendamos a utilização do plugin gratuito WP-Stateless (disponível em WordPress.org). O plugin copia arquivos do seu site do WordPress para o Google Cloud Storage automaticamente conforme eles são carregados na biblioteca de mídia. Em seguida, ele oferece diferentes modos, como backup de uma cópia de segurança da sua mídia ou até mesmo excluir a mídia do WordPress e servir a partir do Armazenamento em nuvem do Google.

Descarregar E-mails Transacionais e de Marketing

Não importa se você pensa assim ou não, os e-mails têm um impacto sobre os recursos do seu servidor e do servidor. Com alguns hosts, especialmente hosts compartilhados, abusar disso pode até mesmo fazer com que você seja suspenso. Isto torna-se especialmente um problema com aqueles que tentam enviar e-mails em massa. Esta é a razão pela qual os provedores de e-mail transacionais de terceiros existem e por que muitos provedores de hospedagem bloqueiam a entrega de e-mail em portas padrão completamente. **Nós nunca recomendamos usar o seu provedor de hospedagem para e-mail.**

Se você está enviando newsletters ou e-mails em massa, nós sempre recomendamos as seguintes alternativas para obter os melhores resultados:

- Use um software de e-mail marketing profissional de terceiros que não faça parte do WordPress
- Use um provedor de serviços de e-mail transacional (API HTTP ou SMTP) junto com o WordPress

Outras vantagens da utilização de um serviço de terceiros incluem:

- Melhor entregabilidade de e-mail. Permita que os provedores de e-mail façam o que fazem de melhor!
- Menos hipóteses de ficar na lista negra.
- Pode nem sempre ser possível configurar registros DMARC com seu provedor de hospedagem.

Ferramentas de E-mail Marketing

Alguns exemplos de e-mails de marketing incluem newsletters, anúncios de produtos e recursos, vendas, convites para eventos, lembretes, etc. Aqui estão algumas ferramentas de e-mail marketing que recomendamos:

- **MailChimp** – Usamos o MailChimp na Kinsta.
- **MailerLite**
- **Drip**

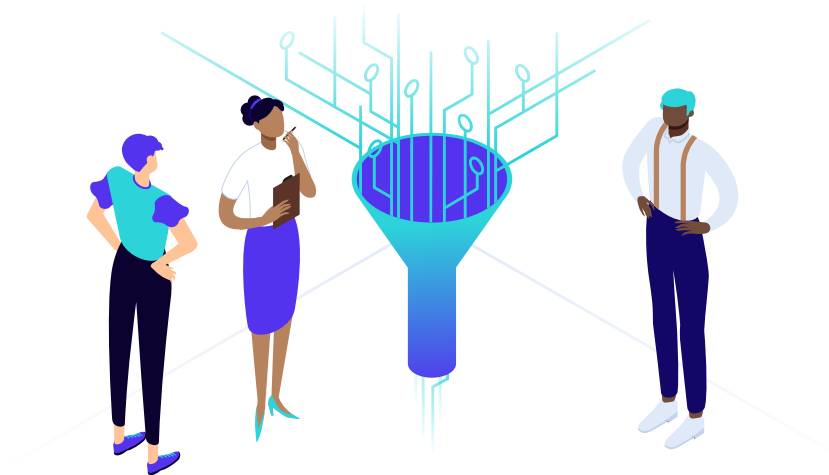
Serviços de E-mail Transacionais

Alguns exemplos de e-mails transacionais incluem recibos de compra do WooCommerce ou EDD, notificações de criação de conta, notificações de envio, mensagens de erro do aplicativo, redefinições de senha, etc. Se você é um cliente Kinsta, contamos com um provedor SMTP terceirizado para garantir alta entregabilidade. Mas, dependendo do seu volume, sempre recomendamos que você mova esse provedor externo. Aqui estão alguns serviços de e-mail de transação que recomendamos:

- **SendGrid** – Usamos a SendGrid na Kinsta.
- **Mailgun**
- **SparkPost**

CAPÍTULO 13:

Como Encontrar Restrições e Plugins Lentos

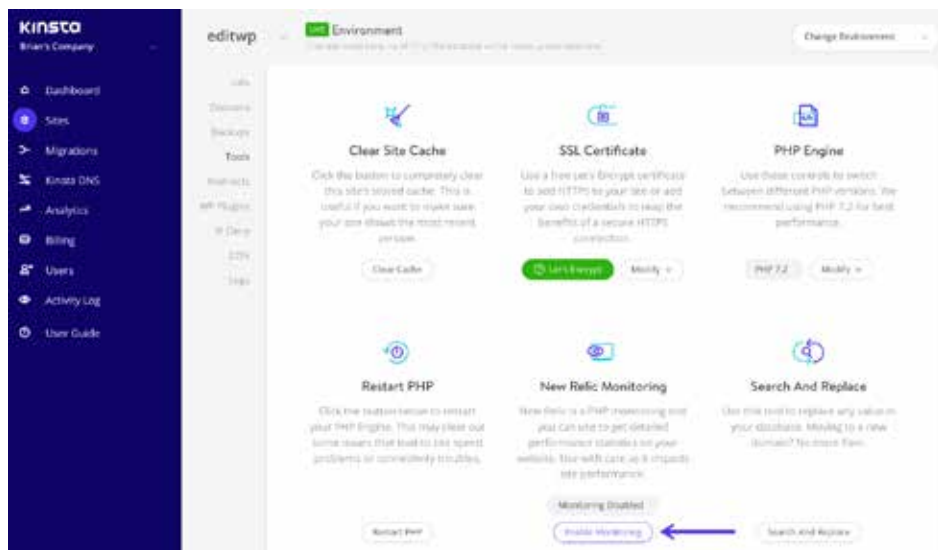


Agora vamos mergulhar em algumas dicas sobre como encontrar restrições no seu site WordPress e o que você pode fazer a respeito.

Use a New Relic para Identificar Plugins Lentos e Consultas ao Banco de Dados

Existem algumas grandes ferramentas no mercado que podem ajudá-lo a localizar e identificar consultas de banco de dados lento e plugins que estão consumindo muito tempo. Nós somos grandes fãs da New Relic em Kinsta e usá-lo em uma base diária. A New Relic é uma ferramenta de monitoramento PHP que você pode usar para obter estatísticas de desempenho detalhadas em seu site.

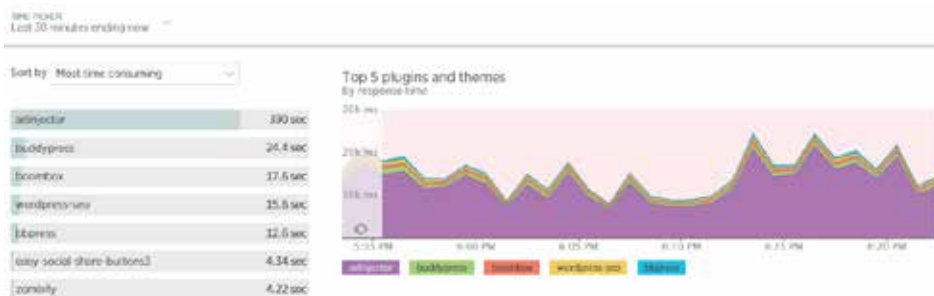
Se você é um cliente Kinsta, você pode até mesmo adicionar sua própria chave de licença da New Relic em nosso painel MyKinsta.



No entanto, use a New Relic com cuidado, pois ela afeta o desempenho do local. Ele adiciona JavaScript ao seu site. Recomendamos ativá-lo quando você precisar solucionar problemas de desempenho e depois desativá-lo.

Encontrando Plugins Lentos

Quando um plugin do WordPress está causando lentidão geral, os sintomas variam de acordo com a atividade que o plugin está executando. No entanto, em muitos casos, você verá que um plugin lento afetará todas as páginas de um site do WordPress. No caso do site cujos dados você vê na imagem abaixo, a lentidão geral foi observada em todas as páginas front-end do site. Aqui está o que a New Relic tinha a dizer sobre o desempenho dos plugins no site.



Imediatamente você pode ver que o plugin do adinjetor está consumindo mais de 15 vezes a quantidade de tempo como o próximo plugin mais lento.

Quando você vê dados como este, pode ser tentador dispensar imediatamente o plugin como mal codificado ou, de alguma forma, ineficaz. Enquanto isso às vezes é o caso, nem sempre é o caso. A má configuração do plugin, a lentidão da base de dados ou recursos externos que são lentos a responder podem fazer com que um plugin consuma muito tempo.

Então, quando você vê um plugin que está respondendo lentamente, é uma boa idéia verificar várias outras telas na New Relic para encontrar informações adicionais. As transações, bancos de dados e recursos externos devem ser todos verificados antes de decidir que desativar o plugin é o melhor ou único caminho a seguir.

Lentidão Geral Causada por um Banco de Dados Sobrecarregado

Um banco de dados mal otimizado pode causar lentidão geral em um site WordPress. Anteriormente discutimos um monte de coisas diferentes que você pode fazer para corrigir isso. Em New Relic, essa lentidão relacionada ao banco de dados provavelmente aparecerá em dois lugares:

- Primeiro, você verá uma quantidade enorme de atividade MySQL na visão geral.
- Em segundo lugar, você verá uma ou mais tabelas de banco de dados consumindo muito tempo na guia bancos de dados.

Começando com a tela de visão geral, um site com um banco de dados em dificuldades pode parecer algo parecido com isso:



Use o Plugin Gratuito Query Monitor

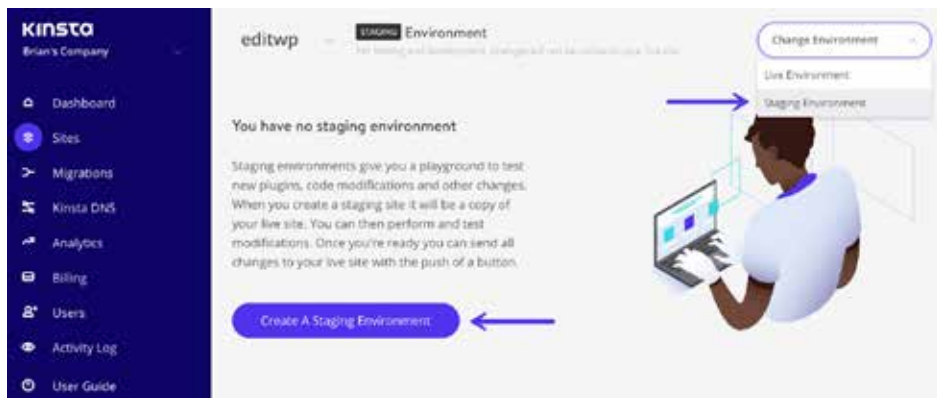
Você também pode usar o plugin gratuito Query Monitor WordPress (disponível no WordPress.org). Use-o para identificar e depurar consultas de banco de dados lentas, chamadas AJAX, solicitações de API REST e muito mais. Além disso, o plugin relata detalhes do site como dependências e dependências de script, ganchos do WordPress que dispararam durante a geração da página, detalhes do ambiente de hospedagem, tags de consultas condicionais encontradas pela página atual e muito mais.

| Query Monitor | | | | | |
|---|---|------------------------------|------------------|----------------|-----------------------------|
| Page generation time | Peak memory usage | Database query time | Database queries | Object cache | |
| 0.0321 0.0% of 0.1301 | 2,048 kb 0.0% of 10,777,220 kb 13017 | 0.0022 | SELECT: 3 | 99.9% hit rate | External object cache: true |
| SQL Queries | | | | | |
| Query | Caller | Component | Rows | Time | |
| 1. SELECT user_id, meta_key, meta_value FROM wp_usermeta WHERE user_id IN (1) ORDER BY umeta_id ASC | update_meta_cache() wp-includes/meta.php:805 | * Plugin: saming-smm-builder | 19 | 0.0004 | |
| 2. SELECT wp_posts.* FROM wp_posts WHERE 1=1 AND (wp_posts.ID = '2') AND wp_posts.post_type = 'page' ORDER BY wp_posts.post_date DESC | wp_query->get_posts() wp-includes/query.php:1051 | * Core | 1 | 0.0017 | |
| 3. SELECT * FROM wp_posts WHERE (post_type = 'page' AND post_status = 'publish') AND post_parent = 1 ORDER BY wp_posts.post_title ASC LIMIT 0,1 | get_pages() wp-includes/post.php:439 | * Core | 0 | 0.0001 | |
| Total Queries: 3 | | | | | 0.0022 |

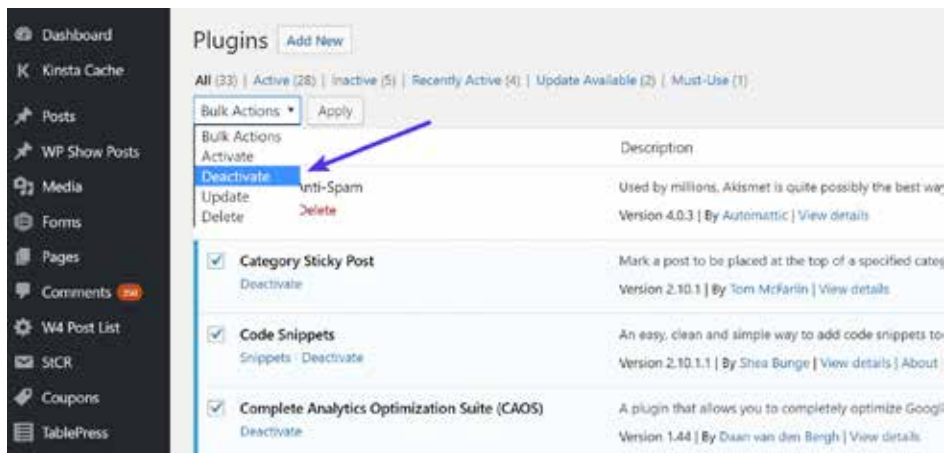
O plugin foi desenvolvido por John Blackbourn, um committer WordPress central que é atualmente um desenvolvedor na Human Made e foi anteriormente empregado pelo WordPress.com VIP - em outras palavras, alguém que conhece WordPress extensivamente. Query Monitor foi adicionado ao diretório de plugins do WordPress em 2013 e atualmente possui mais de 10.000 instalações ativas - uma soma impressionante para um plugin de desenvolvimento. A avaliação do usuário do plugin de cinco entre cinco estrelas ajuda a explicar sua popularidade entre os desenvolvedores

Utilize Locais de Testes sem Tocar na Produção (Ao Vivo)

Não sabemos o que fariamos sem encenar ambientes. Estes podem ser inestimáveis quando se trata de resolver problemas de desempenho. Felizmente, Kinsta tem ambientes de testes com um clique. Se o seu host WordPress não oferece ambientes de testes, você também pode usar um plugin como o WP Staging (disponível no WordPress.org), embora não seja tão fácil.



Depois de ter um site de teste em execução, a primeira coisa que você pode fazer é desativar todos os seus plugins. Uma vez que esta é uma cópia de seu site de produção (ao vivo), você não tem que se preocupar em quebrar qualquer coisa. É de longe uma das maneiras mais fáceis de restringir problemas. Basta ir a Plugins, selecionar todos eles e escolher “Desativar” a partir das opções em massa.



Após fazer isso, você pode monitorar os tempos de resposta na New Relic ou no Query Monitor e ver o que acontece. Neste exemplo abaixo, os tempos de resposta voltaram imediatamente ao normal no site, então sabíamos que era um dos plugins que causava um problema. Você pode então reativá-los um a um, repetindo o mesmo processo até encontrar o culpado.



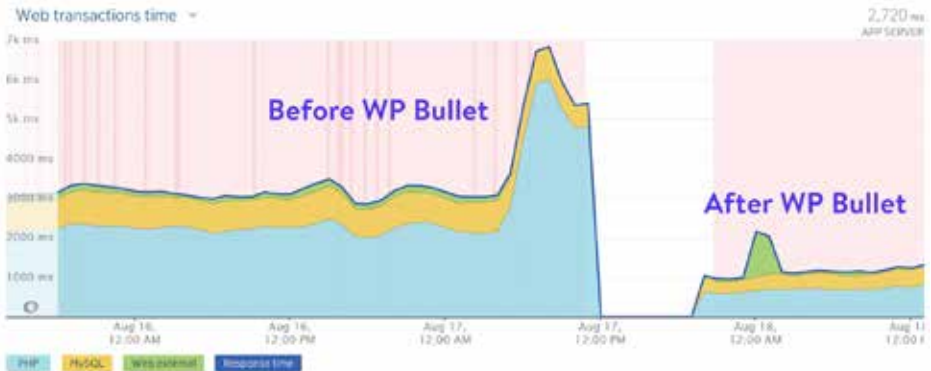
Aqui está um exemplo do que aconteceu quando ativamos o plugin que estava causando o problema. Os tempos de carga (tempos de transação web) voltaram a subir imediatamente.



Aqui está um exemplo do que aconteceu quando ativamos o plugin que estava causando o problema. Os tempos de carga (tempos de transação web) voltaram a subir imediatamente.

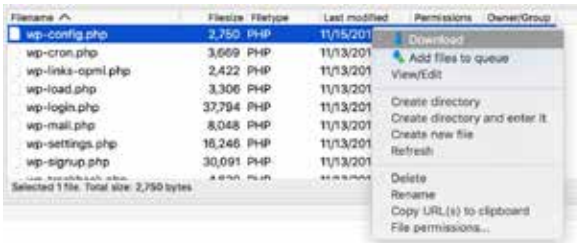
1. Atualize seus plugins e temas para a versão mais recente, caso ainda não o tenha feito.
2. Entre em contato com o desenvolvedor do plugin ou tema e peça ajuda.
3. Encontre um plugin alternativo que possa oferecer a mesma funcionalidade.
4. Talvez sua versão do PHP esteja causando um problema. Mude seu mecanismo PHP para uma versão mais baixa e veja se o plugin ou tema então funciona.

Você também pode contratar um desenvolvedor do WordPress para corrigir o problema. Se for relacionado ao desempenho, temos que dar um grito pessoal ao Mike Andreason em wp-bullet.com. Ele é um desenvolvedor Codeable em tempo integral especializado em otimização de performance, que tem ajudado muitos clientes aqui em Kinsta com instalações complexas a levar seu site para o próximo nível.



Verifique Seus Registros de Erros

Verificar logs de erros nunca é divertido, mas pode revelar muito sobre problemas de desempenho com plugins do WordPress. Se você é um cliente Kinsta, você pode facilmente visualizar seus logs de erros, logs de cache e logs de acesso diretamente do painel MyKinsta.

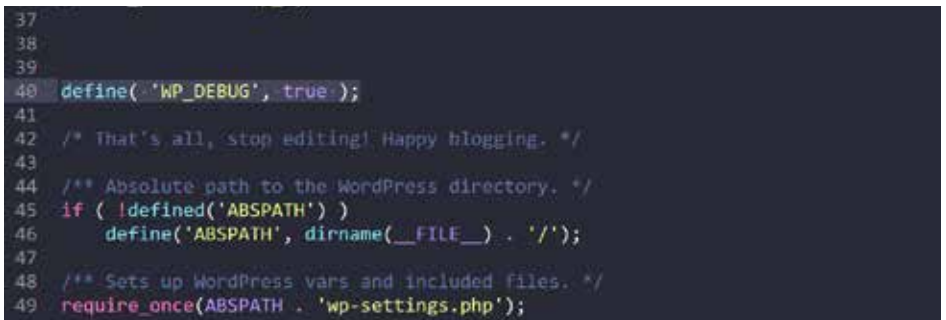


Você também pode habilitar logs de erros adicionando algum código ao seu arquivo wp-config.php. Primeiro, você vai querer se conectar ao seu site via SFTP. Depois faça o download do seu wp-config.php para que possa editá-lo.

Nota: Faça sempre uma cópia de segurança deste ficheiro primeiro!

Encontre a linha que diz `/* That's all, stop editing! Happy blogging. */` e logo antes disso, adicione o seguinte (como visto abaixo):

```
define('WP_DEBUG', true);
```



Se o código acima já existe no seu arquivo `wp-config.php` mas está definido como “false”, simplesmente mude-o para “true”. Isso habilitará o modo de depuração. Nota: Você também verá avisos ou erros no seu administrador do WordPress, se eles existirem.

Você pode então habilitar o log de depuração para enviar todos os erros para um arquivo adicionando o seguinte código logo após a linha `WP_DEBUG` (como visto abaixo):

```
define('WP_DEBUG_LOG', true);
```

```
38
39
40 define( 'WP_DEBUG', true );
41 define( 'WP_DEBUG_LOG', true );
42
43 /* That's all, stop editing! Happy blogging. */
44
45 /** Absolute path to the WordPress directory. */
46 if ( !defined('ABSPATH') )
47     define('ABSPATH', dirname(__FILE__) . '/');
48
49 /** Sets up WordPress vars and included files. */
50 require_once(ABSPATH . 'wp-settings.php');
```

Salve suas alterações e faça o upload novamente para o seu servidor. Os erros serão então registrados no arquivo `debug.log` dentro da pasta `/wp-content/`. Se por algum motivo você não vê este arquivo, você pode sempre criar um.

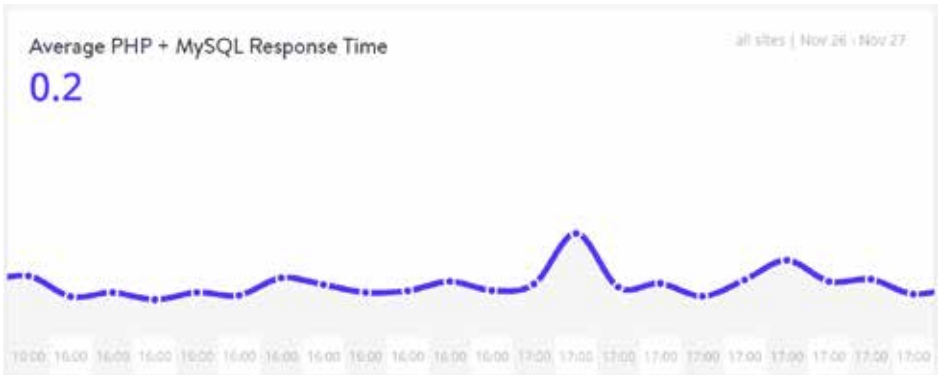
Use o MyKinsta Analytics

Se você é um cliente Kinsta, você pode aproveitar os insights de desempenho que incorporamos em nossa ferramenta MyKinsta Analytics.

Na seção de monitoramento de desempenho, você pode visualizar seu tempo médio de resposta PHP + MySQL, throughput PHP, uso de AJAX, tempo médio de upstream top e tempo máximo de upstream top.

Tempo Médio de Resposta PHP + MySQL

Sempre que você visita seu site WordPress, PHP e MySQL são usados para compilar e consultar os dados que você vê na página. Este gráfico mostra o tempo médio de resposta do motor PHP e do motor MySQL para cada solicitação dinâmica não armazenada em cache. Conhecer esses tempos de resposta pode ajudá-lo a solucionar problemas de lentidão.



Taxa de Transferência PHP

O rendimento indica o número de transações por segundo que um aplicativo pode manipular, e neste relatório, ele está se referindo ao rendimento do PHP do seu site do WordPress. Em outras palavras, ele mostra quantas vezes um recurso do PHP foi solicitado.

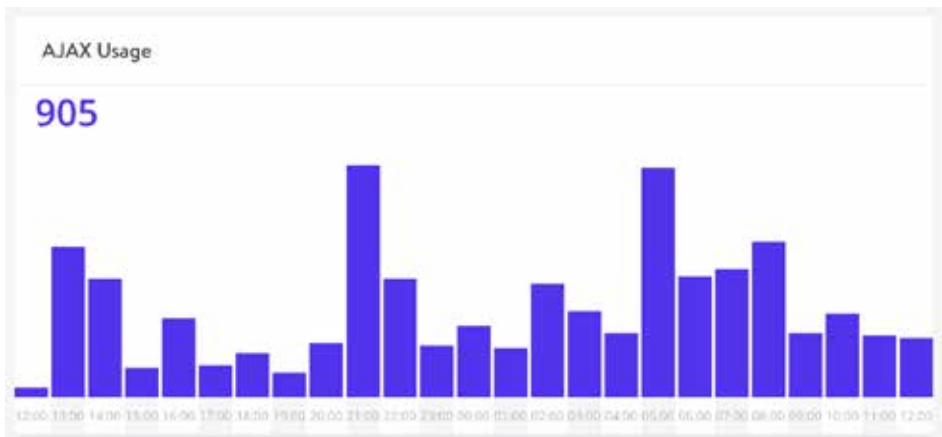


Uso do AJAX

AJAX é um script client-side que se comunica de e para um servidor/banco de dados sem a necessidade de um postback ou uma atualização completa da página. Quando se trata de WordPress, muitos de vocês provavelmente já viram isso em seus testes de velocidade. Os dois principais problemas com AJAX incluem plugins fazendo com que ele tenha pico e problemas de CPU no back-end.



O relatório de uso do AJAX no MyKinsta Analytics pode ser uma ótima maneira de ajudá-lo a solucionar esses tipos de problemas, pois você pode ver se você está vendo certos picos de AJAX durante certos períodos. Este gráfico mostra a contagem das solicitações admin-ajax. Você pode então utilizar algumas das dicas no post que mencionamos acima para estreitar onde eles podem estar vindo.



Média Máxima de PHP + Tempo de Resposta MySQL

Esta lista mostra os principais tempos médios de resposta do PHP e MySQL. Estes números podem ser picos únicos, por isso é sugerido comparar esta lista com “Top Maximum Upstream Time”.

| PATH | REQUESTS | TIME |
|--|----------|--------|
| /wp-admin/update.php?action=upload-plugin | 1 | 2.19 s |
| /wp-cron.php?server_triggered_cronjob | 93 | 1.97 s |
| /wp-admin/plugins.php?action=activate&plugin=wp-rocket%2Fwp-rocket.php&plugin_status=all&paged=1&s&_wpnonce=70f7d2faca | 1 | 1.47 s |
| /wp-admin/plugins.php?activate=true&plugin_status=all&paged=1&s= | 2 | 1.37 s |
| /wp-admin/plugins.php?action=deactivate&plugin=wp-rocket%2Fwp-rocket.php&_wpnonce=729c628974 | 1 | 1.07 s |
| /wp-admin/plugins.php?action=activate&plugin=wp-rocket%2Fwp-rocket.php&_wpnonce=70f7d2faca | 1 | 0.91 s |
| /wp-cron.php?doing_wp_cron=1542348999.8110280036926269531250 | 1 | 0.67 s |
| /wp-admin/plugins.php?action=deactivate&plugin=wp-rocket%2Fwp-rocket.php&_wpnonce=113bcb7711 | 1 | 0.53 s |
| /wp-admin/plugins.php | 3 | 0.49 s |
| /wp-admin/plugin-install.php | 1 | 0.48 s |

Tempo Máximo de Montante

Upstream time é o tempo total que leva para NGINX (e servidores upstream) processar um pedido e enviar uma resposta. O tempo é medido em segundos, com resolução em milissegundos.

| PATH | REQUESTS | TIME |
|--|----------|----------|
| /wp-cron.php?server_triggered_cronjob | 93 | 180.00 s |
| /wp-admin/plugins.php?activate=true&plugin_status=all&paged=1&s= | 2 | 2.60 s |
| /wp-admin/update.php?action=upload-plugin | 1 | 2.19 s |
| /wp-admin/plugins.php?action=activate&plugin=wp-rocket%2Fwp-rocket.php&plugin_status=all&paged=1&s&_wpnonce=70f7d2faca | 1 | 1.47 s |
| /hello-world/ | 5 | 1.46 s |
| /wp-admin/plugins.php?action=deactivate&plugin=wp-rocket%2Fwp-rocket.php&_wpnonce=729c628974 | 1 | 1.07 s |
| / | 83 | 0.93 s |
| /wp-admin/plugins.php?action=activate&plugin=wp-rocket%2Fwp-rocket.php&_wpnonce=70f7d2faca | 1 | 0.91 s |
| /wp-admin/plugins.php | 3 | 0.85 s |
| /wp-cron.php?doing_wp_cron=1542348999.8110280036926269531250 | 1 | 0.67 s |

Seu Site Pode ser Hackeado

Se você estiver tendo problemas para rastrear um problema de desempenho, pode muito bem ser que seu site esteja hackeado, infectado por malware ou passando por um ataque DDoS. Isso pode afetar a velocidade do seu site e até mesmo a capacidade de resposta do

seu painel de administração do WordPress. Nesses casos, recomendamos o seguinte:

1. Implementar um servidor proxy e WAF, como Cloudflare ou Sucuri.
2. Bloqueie endereços IP ruins usando os serviços acima ou se você for um cliente Kinsta, você também pode bloquear endereços IP do nosso painel MyKinsta.
3. Também é possível implementar o bloqueio geográfico. Alguns países são realmente maus quando se trata da qualidade do tráfego que geram. Se você estiver sob ataque, você pode precisar bloquear o país inteiro, temporária ou permanentemente.

Solução de Problemas com Códigos de Erro (Códigos de Status HTTP)

Os códigos de status HTTP são como uma nota curta do servidor web que é colocada no topo de uma página web. Não faz parte da página web. Em vez disso, é uma mensagem do servidor informando como as coisas correram quando a solicitação para visualizar a página foi recebida pelo servidor. Estes podem ser inestimáveis quando se trata de resolução de problemas!

Embora existam mais de 40 códigos de status diferentes, abaixo estão os mais comuns com os quais vemos os usuários do WordPress lutando. Temos tutoriais disponíveis na kinsta.com com instruções sobre como corrigir cada um dos seguintes códigos de erro:

429: “Demasiados pedidos.” Gerado pelo servidor quando o utilizador enviou demasiados pedidos num determinado período de tempo (limitação de taxas). Isso às vezes pode ocorrer a partir de bots ou scripts tentando acessar seu site. Neste caso, você pode tentar mudar sua URL de login do WordPress.

429 Too Many Requests

nginx

500: “Houve um erro no servidor e o pedido não pôde ser concluído.”
Um código genérico que significa simplesmente “erro interno do servidor”. Algo deu errado no servidor, e o recurso solicitado não foi entregue. Este código é tipicamente gerado por plugins de terceiros, PHP defeituoso, ou mesmo a conexão com o banco de dados quebrando.

Error establishing a database connection

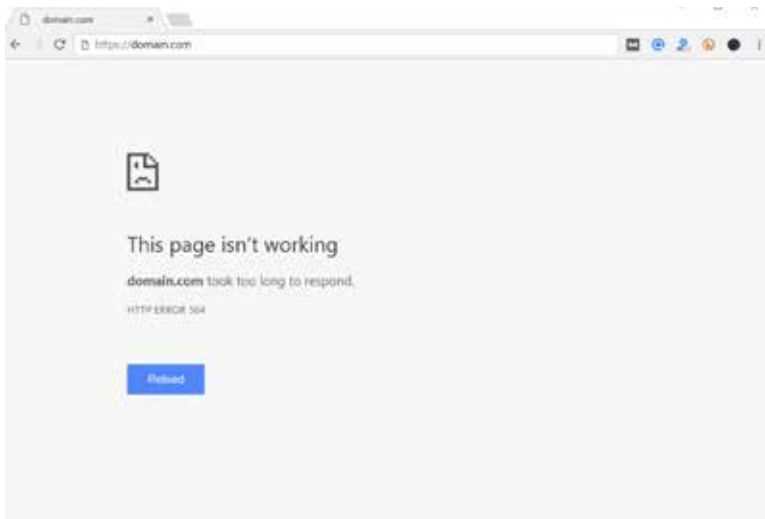
502: “Bad Gateway.” Este código de erro tipicamente significa que um servidor recebeu uma resposta inválida de outro. Às vezes, uma consulta ou requisição leva muito tempo, e por isso é cancelada ou morta pelo servidor e a conexão com o banco de dados é interrompida.



503: “O servidor não está disponível para lidar com este pedido agora.”

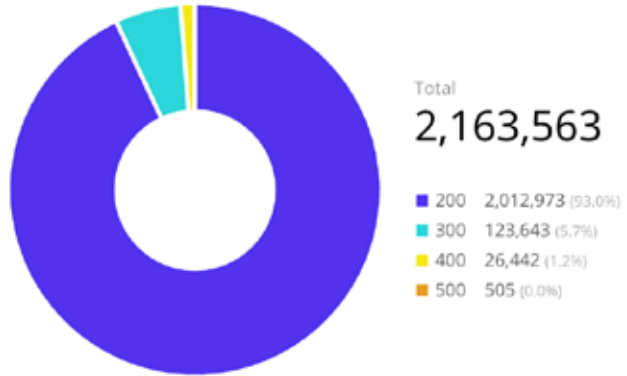
O pedido não pode ser concluído neste momento. Este código pode ser devolvido por um servidor sobrecarregado que não é capaz de lidar com pedidos adicionais.

504: “O servidor, agindo como um gateway, ficou sem tempo à espera que outro servidor respondesse.” O código é retornado quando há dois servidores envolvidos no processamento de uma solicitação, e o primeiro servidor acaba esperando que o segundo servidor responda.



Você também pode pesquisar esses códigos de resposta HTTP em nossa ferramenta MyKinsta Analytics. Nosso relatório de detalhamento de código de resposta permite que você veja uma visão geral da distribuição de códigos de status HTTP servidos para os recursos solicitados.

Repartição do Código de Resposta



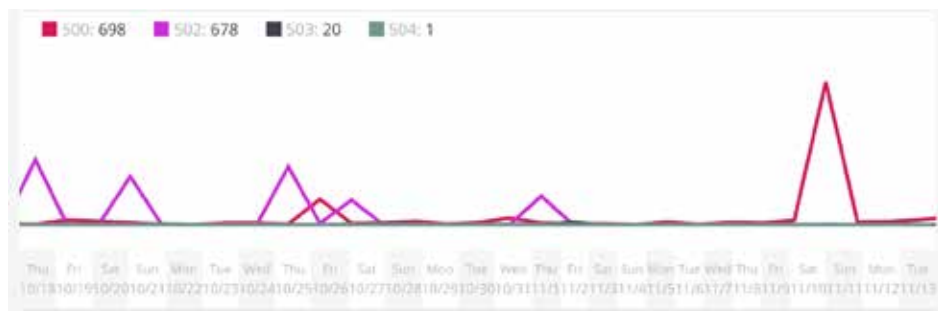
O relatório de estatísticas de resposta permite que você veja o número total de redirecionamentos acontecendo, erros, taxa de sucesso e taxa de erro. Cada site do WordPress normalmente terá uma pequena taxa de erro; isso é completamente normal.

Estatísticas de Resposta



Em seguida, há relatórios de parada para cada tipo de código de erro, como 500 errors, 400 errors, redirecionamentos, etc.

500 Error Breakdown



CAPÍTULO 14:

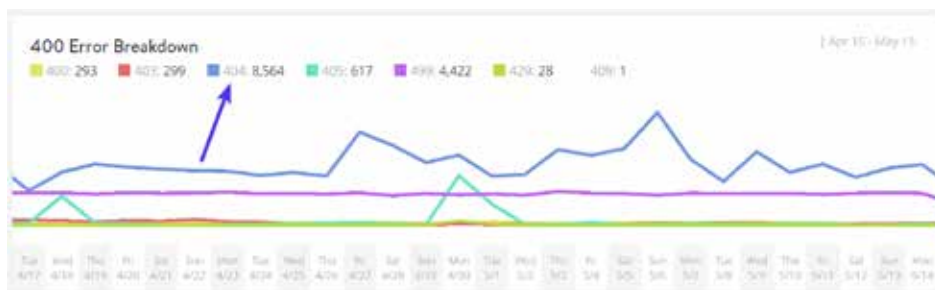
Recomendações Sobre Optimização Back-End



Agora vamos mergulhar em algumas formas de acelerar o WordPress, otimizando o back-end. O back-end tipicamente envolve qualquer coisa que é tratada inteiramente pelo servidor, como PHP, cabeçalhos de cache HTTP, compressão GZIP, etc.

Criar uma Página 404 Light

Vimos em primeira mão que sites altamente dinâmicos normalmente geram muitos erros 404. Seu site pode estar gerando mais do que você pensa! Nossa ferramenta de análise do MyKinsta pode ajudá-lo a determinar a quantidade exata (como visto abaixo).



A razão pela qual esses erros são ruins é que muitas 404 páginas são muito intensivas em recursos. Para um site WordPress altamente dinâmico, você vai querer evitar uma página 404 pesada. Crie um modelo 404 simples que evite consultar o banco de dados,

se possível. E é claro, gaste algum tempo e corrija os erros do 404, pois isso não é apenas um recurso intensivo, é simplesmente ruim para a experiência do usuário.

Aumente os PHP Workers

PHP workers pode ser um termo que você nunca ouviu falar, mas eles são quantos hosts, incluindo Kinsta, lidam com solicitações de limitação (ao invés de limitá-lo por CPU ou RAM, que é tipicamente o que os provedores de hospedagem compartilhada fazem).

PHP workers determinam **quantas requisições simultâneas seu site pode lidar em um determinado momento**. Simplificando, cada pedido não armazenado para o seu site é tratado por um PHP workers. Por exemplo, se você tem 4 requisições que chegam ao seu site exatamente ao mesmo tempo e seu site tem 2 PHP workers, duas dessas requisições serão processadas enquanto as outras duas terão que esperar na fila até que as duas primeiras tenham terminado o processamento.

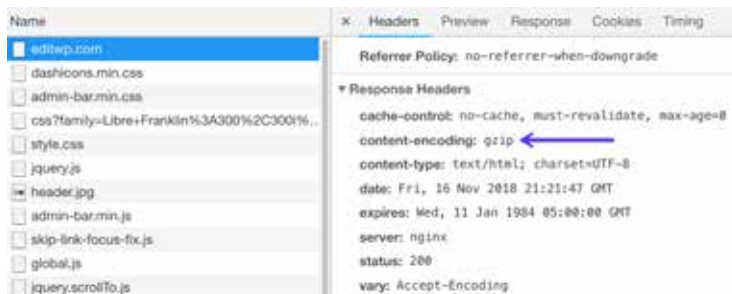
Lembre-se que discutimos anteriormente que um dos maiores problemas com os sites de membros do WordPress é todos os pedidos não armazenados em cache. É por isso que os trabalhadores do PHP se tornam muito importantes, pois eles têm que trabalhar para cada requisição. Portanto, estes sites tipicamente requerem trabalhadores PHP adicionais para garantir que cada requisição seja processada sem atrasos e completada com sucesso.

O que acontece se você continuamente maximizar seus PHP workers? Basicamente, a fila começa a empurrar pedidos mais antigos que podem resultar no erro 500 no seu site. Cada um dos planos de hospedagem da Kinsta inclui um número predefinido de PHP workers. Se você tiver problemas para estimar o que seu site pode precisar, você sempre pode conversar com nossa equipe de vendas ou suporte.

Utilize a Compressão GZIP

GZIP é um formato de arquivo e uma aplicação de software utilizada para compressão e descompressão de arquivos. A compressão GZIP está habilitada no lado do servidor e permite uma maior redução no tamanho de seus arquivos HTML, folhas de estilo e JavaScript.

Quando um navegador da Web visita um site, ele verifica se o servidor Web tem GZIP ativado, `content-encoding: gzip` HTTP-codificação de conteúdo: existe. Se o cabeçalho for detectado, ele serve para os arquivos compactados e menores. Se não, ele serve para os arquivos não compactados. Se você não tiver o GZIP ativado, provavelmente verá avisos e erros em ferramentas de teste de velocidade, como o Google PageSpeed Insights e o GTmetrix.



A ativação da compressão GZIP pode ajudar a reduzir o tamanho da sua página web, o que pode reduzir significativamente a quantidade de tempo para baixar o recurso, reduzir o uso de dados para o cliente e melhorar o tempo para a primeira renderização de suas páginas. Isto é bastante padrão agora na maioria dos provedores de hospedagem, mas nada mais nos surpreende neste ponto.

Ativar Proteção Hotlink

O conceito de hotlinking é bastante simples. Você encontra uma imagem na internet em algum lugar e usa a URL da imagem diretamente no seu site. Esta imagem será exibida em seu site, mas ele será servido a partir do local original. Isto é muito conveniente para o hotlinker, mas é realmente o roubo como ele está usando os recursos do site hotlinked. É como se nós fôssemos entrar em nosso carro e ir embora com gasolina que roubamos do carro do nosso vizinho.

*Hotlinking é como fugir com gasolina que
você tirou do carro do seu vizinho.*

Hotlinking pode ser **um grande dreno de recursos para o servidor alvo**. Imagine se você está em um host WordPress compartilhado e Huffington Post de repente links para suas imagens. Você pode ir de algumas centenas de consultas por hora no seu site para algumas centenas de milhares. Isso poderia até mesmo resultar em uma suspensão de sua conta de hospedagem. Esta é uma razão para não só usar um host de alto desempenho (que pode lidar com soluções como este), mas também para ativar a proteção hotlink, para que isso não aconteça.

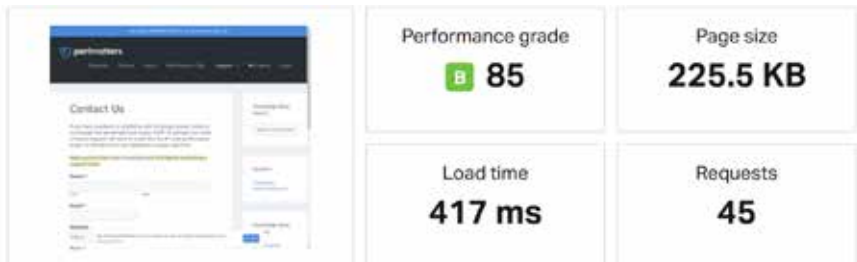
Minimizar os Redirecionamentos e Adicioná-los no Nível do Servidor

Demasiados redirecionamentos são sempre algo com que tens de ter cuidado. Redirecionamentos simples como um único redirecionamento 301, HTTP para HTTPS, ou www para non-www (vice-versa) estão bem. E muitas vezes estes são necessários em certas áreas do seu site. Entretanto, cada um tem um custo no desempenho do seu site. E se você começar a empilhar redireciona em cima um do outro, é importante perceber como eles impactam seu site. Isso se aplica a redirecionamentos de páginas e posts, redirecionamentos de imagens, tudo.

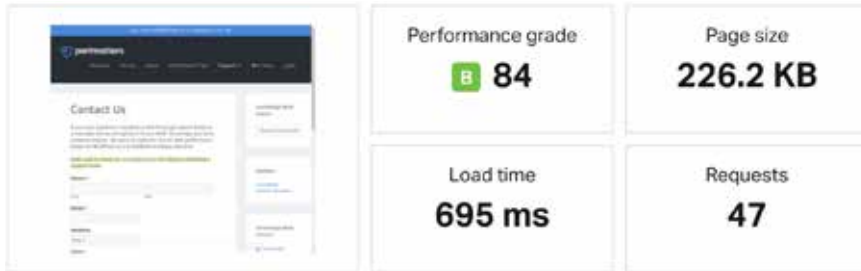
Um redirecionamento irá gerar um 301 ou 302 no status do cabeçalho de resposta.



Como os redirecionamentos impactam seu site? Vamos fazer um pequeno teste. Primeiro, fazemos um teste de velocidade em nossa página de contato: <https://perfmatters.io/contact/>. Como você pode ver abaixo, temos um tempo total de carga de **417 ms**.



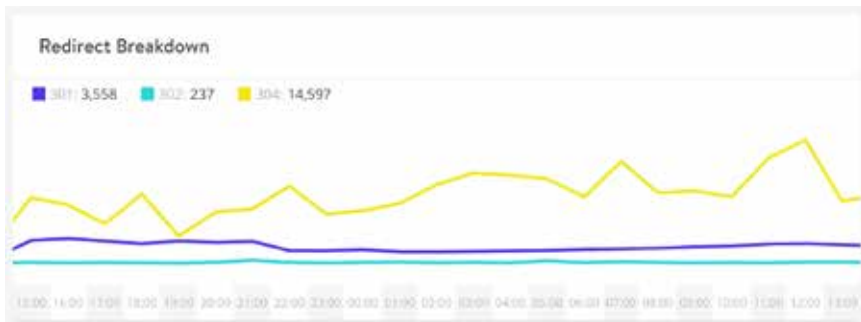
Em seguida, modificamos ligeiramente o URL e executamos outro teste de velocidade para ver o impacto de vários redirecionamentos. <http://www.perfmatters.io/contact>. Como você pode ver, a mesma página agora leva **695 ms** para carregar. Isso é um aumento de 66%. Caramba!



Usar plugins gratuitos do WordPress para implementar redirecionamentos às vezes pode causar problemas de desempenho, pois a maioria deles utiliza a função `wp_redirect`, que requer execução de código e recursos adicionais. Alguns deles também adicionam dados carregados automaticamente à sua tabela `wp_options`, o que aumenta o inchaço do banco de dados. Adicioná-los no nível do servidor é onde eles devem ser feitos. Nós permitimos que você faça isso no MyKinsta com nossa ferramenta de regras de redirecionamento.



Você também pode ver um resumo completo de quantos redirecionamentos estão acontecendo em seus sites em nossa ferramenta de análise MyKinsta. Veja o número total de 301's, 302's e 304's



Não Deixe as Tarefas Cron sair do Controle

As tarefas CRON (WP-Cron) são utilizadas para agendar tarefas repetitivas para o seu site WordPress. No entanto, ao longo do tempo, estes podem ficar fora de controle e causar problemas de desempenho. Você pode usar o plugin gratuito WP Control (disponível no WordPress.org) para verificar uma alça em todos os trabalhos Cron acontecendo em seu site.

Nós também vimos problemas de desempenho com o WordPress built-in Cron handler: WP-Cron. Se um site não tem PHP workers suficientes, às vezes uma requisição chega, o WordPress gera o cron, mas o cron tem que esperar pelo trabalhador e portanto, fica ali sentado. Uma abordagem melhor é desabilitar o WP-Cron e usar o cron do sistema. Isto é até mesmo recomendado no manual oficial do Plugin.

Para desactivar o WP-Cron, adicione o seguinte ao seu ficheiro `wp-config.php`, mesmo antes da linha que diz “Isto é tudo, edição de passos! Blogs felizes”. Nota: Isso o desativa de rodar no carregamento da página, não quando você a chama diretamente via `wp-cron.php`.

```
define('DISABLE_WP_CRON', true);
```

```
60  /**
61  * WordPress Database Table prefix.
62  *
63  * You can have multiple installations in one database if you give each
64  * a unique prefix. Only numbers, letters, and underscores please!
65  */
66  $table_prefix = 'wp_';
67
68  define('DISABLE_WP_CRON', true); ←
69
70  /* That's all, stop editing! Happy blogging. */
71
72  /** Absolute path to the WordPress directory. */
73  if ( ! defined( 'ABSPATH' ) )
74      define( 'ABSPATH', dirname( __FILE__ ) . '/' );
75
76  /** Sets up WordPress vars and included files. */
77  require_once ABSPATH . 'wp-settings.php';
78
```

Line 68, Column 33

Você precisará então agendar `wp-cron.php` partir do seu servidor. Se você é um cliente Kinsta, os crons de sistemas já estão habilitados e rodam a cada 15 minutos por padrão. Você também pode ter a frequência aumentada entrando em contato com nossa

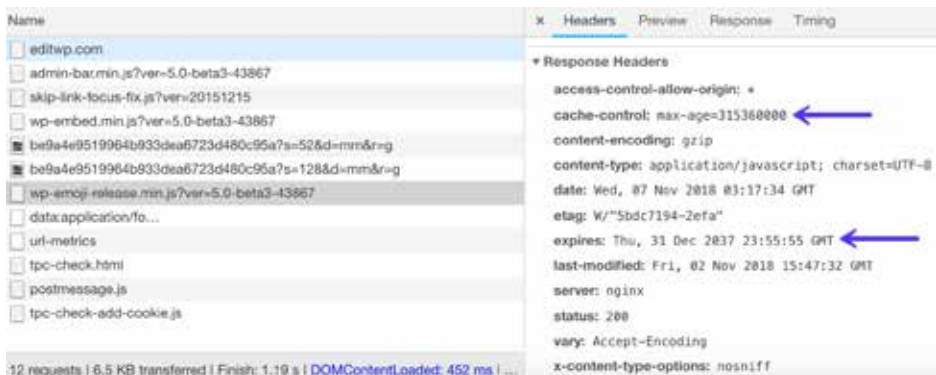
equipe de suporte. Se você estiver familiarizado com o SSH, você pode gerenciar os crons do servidor a partir da linha de comando.

Adicionar Cache-Control e Cabeçalhos Expirados (Determinar Comprimento do Cache)

Cada script no seu site WordPress precisa ter um cabeçalho de cache HTTP anexado a ele (ou deveria ter). Isso **determina quando o cache do arquivo expira**. Para corrigir isso, certifique-se de que seu host WordPress tem os cabeçalhos de `cache-control` adequados e `expires` a configuração de cabeçalhos. Se você não fizer isso, você provavelmente verá avisos sobre a necessidade de adicionar cabeçalhos expira ou alavancar o cache do navegador em ferramentas de teste de velocidade.

Enquanto o cabeçalho `cache-control` ativa o cache do lado do cliente e define a idade máxima de um recurso, o cabeçalho `expires` é usado para especificar um ponto específico no tempo em que o recurso não é mais válido. Embora ambos os cabeçalhos possam ser usados juntos, você não precisa necessariamente adicionar ambos os cabeçalhos. `cache-control` é mais recente e geralmente é o método recomendado.

Kinsta adiciona automaticamente cabeçalhos de cache HTTP em todas as requisições do servidor, e se você estiver usando um CDN, eles provavelmente adicionarão esses cabeçalhos para você também.



Se o seu servidor não tiver esses cabeçalhos, você pode adicioná-los manualmente.

Adicionando Cache-Control no Cabeçalho do Nginx

Você pode adicionar cabeçalhos de `cache-control` no Nginx adicionando o seguinte ao local ou bloco do servidor de configuração do seu servidor.

```
location ~* \.(js|css|png|jpg|jpeg|gif|svg|ico)$ {
    expires 30d;
    add_header Cache-Control "public, no-transform";
}
```

Adicionando Cabeçalho Expirados no Nginx

Você pode adicionar cabeçalhos `expires` no Nginx adicionando o seguinte ao seu bloco de servidores. Neste exemplo, você pode ver como especificar diferentes tempos de expiração com base nos tipos de arquivo.

```
location ~* \.(jpg|jpeg|gif|png|svg)$ {
    expires 365d;
}

location ~* \.(pdf|css|html|js|swf)$ {
    expires 2d;
}
```

Adicionando Cache-Control no Cabeçalho do Apache

Você pode adicionar cabeçalhos de `cache-control` no Apache adicionando o seguinte ao seu arquivo `.htaccess`. Snippets de código podem ser adicionados na parte superior ou inferior do ficheiro (antes de `# BEGIN WordPress` ou depois de `# END WordPress`).

```
<filesMatch "(ico|pdf|flv|jpg|jpeg|png|gif|svg|js|css|swf)$">
    Header set Cache-Control "max-age=84600, public"
</filesMatch>
```

Adicionando Cabeçalho Expirados no Apache

Você pode adicionar cabeçalhos que `expires` no Apache adicionando o seguinte ao seu arquivo `.htaccess` file.

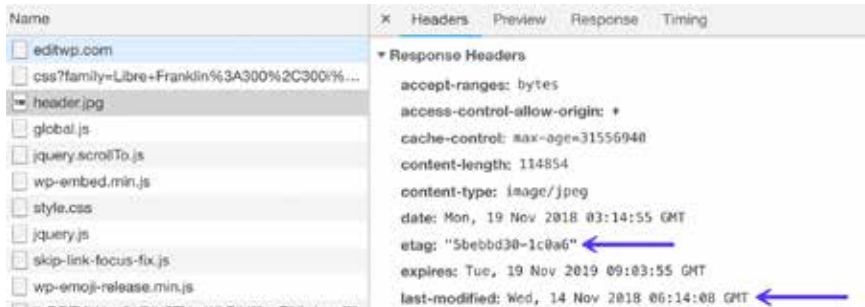
```
## EXPIRES HEADER CACHING ##  
<IfModule mod_expires.c>  
ExpiresActive On  
ExpiresByType image/jpg "access 1 year"  
ExpiresByType image/jpeg "access 1 year"  
ExpiresByType image/gif "access 1 year"  
ExpiresByType image/png "access 1 year"  
ExpiresByType image/svg "access 1 year"  
ExpiresByType text/css "access 1 month"  
ExpiresByType application/pdf "access 1 month"  
ExpiresByType application/javascript "access 1 month"  
ExpiresByType application/x-javascript "access 1 month"  
ExpiresByType application/x-shockwave-flash "access 1 month"  
ExpiresByType image/x-icon "access 1 year"  
ExpiresDefault "access 2 days"  
</IfModule>  
## EXPIRES HEADER CACHING ##
```

É também importante notar que você só pode adicionar cabeçalhos de cache HTTP em recursos no seu servidor. Se você está começando um aviso sobre que talvez você precisa alavancar o cache do navegador em um pedido de terceiros, não há nada que você possa fazer, como você não tem pedido sobre o seu servidor. Culpados comuns incluem o script do Google Analytics e pixels de marketing, como Facebook e Twitter.

Se você está tentando corrigir isso com o script do Google Analytics, você pode hospedá-lo localmente ou em sua CDN (embora isso não seja oficialmente suportado) com um plugin como Perfmatters (disponível em perfmatters.io) ou WP Rocket (disponível em wp-rocket.me).

Adicionar Cabeçalhos Last-Modified e ETag (Validar cache)

Em seguida, temos outros dois conjuntos de cabeçalhos, `last-modified` e `etag`. Enquanto o `cache-control` e os cabeçalhos `expires` ajudam o navegador a determinar **se o arquivo mudou** desde a última vez que foi solicitado (ou melhor, eles validam o cache). Os últimos cabeçalhos modificados e `etag` **validam e definem o comprimento do cache** e devem ser incluídos em cada resposta do servidor de origem. Se estes não estiverem devidamente configurados, poderá ver um aviso de que precisa de “Especificar um validador de cache”.



Se os cabeçalhos não forem encontrados, ele irá gerar uma nova requisição para o recurso toda vez que aumenta a carga no seu servidor. A utilização de cabeçalhos de cache assegura que os pedidos subsequentes não tenham de ser carregados a partir do servidor, poupando assim largura de banda e melhorando a performance para o utilizador.

Kinsta adiciona automaticamente os cabeçalhos acima em todas as requisições do servidor, e se você estiver usando uma CDN, eles provavelmente adicionarão esses cabeçalhos para você também. Assim como com o `cache-control` e `expires`, você não pode definir manualmente esses cabeçalhos HTTP em recursos externos.

Cabeçalho Last-modified

O `last-modified` é geralmente enviado automaticamente do servidor. Este é um cabeçalho que você **geralmente não precisa adicionar manualmente**. É enviado para ver se o ficheiro na cache do browser foi modificado desde a última vez que foi pedido. Você pode ver a solicitação de cabeçalho no Pingdom ou usar o Chrome DevTools para ver o valor do último cabeçalho modificado

Cabeçalho ETag

O cabeçalho **ETag** também é muito similar ao último cabeçalho modificado. Ele também é usado para validar o cache de um arquivo. Se você estiver rodando o Apache 2.4 ou superior, o cabeçalho ETag já está automaticamente adicionado usando a diretiva FileETag. E até onde vai NGINX, o cabeçalho ETag está habilitado por padrão desde 2016.

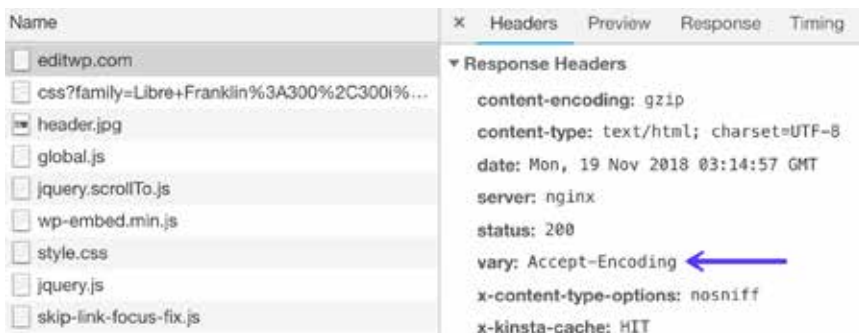
Você pode habilitar o cabeçalho ETag manualmente no Nginx usando o seguinte código.

```
etag on
```

Adicionando uma Variação: Accept-Encoding Header

A `vary: Accept-Encoding` header deve ser incluído em cada resposta do servidor de origem, pois diz ao navegador se o cliente pode ou não lidar com versões compactadas do conteúdo. Se isso não estiver configurado corretamente, você pode ver um aviso de que você precisa “Especificar uma Vary”: Aceitar-Codificação de Cabeçalho”.

Por exemplo, digamos que você tem um navegador antigo sem compressão gzip e um navegador moderno com ele. Se você não utilizar o `vary: Accept-Encoding` do cabeçalho seu servidor web ou CDN pode armazenar em cache a versão não compactada e entregá-la ao navegador moderno por engano, o que, por sua vez, prejudica o desempenho do seu site WordPress. Ao usar o cabeçalho, você pode garantir que o seu servidor web e ou CDN entrega a versão apropriada.



| Name | Headers | Preview | Response | Timing |
|--|--|---------|----------|--------|
| editwp.com | | | | |
| css?family=Libre+Franklin%3A300%2C300i%... | | | | |
| header.jpg | Response Headers | | | |
| global.js | content-encoding: gzip | | | |
| jquery.scrollTo.js | content-type: text/html; charset=UTF-8 | | | |
| wp-embed.min.js | date: Mon, 19 Nov 2018 03:14:57 GMT | | | |
| style.css | server: nginx | | | |
| jquery.js | status: 200 | | | |
| skip-link-focus-fix.js | vary: Accept-Encoding | | | |
| | x-content-type-options: nosniff | | | |
| | x-kinsta-cache: HIT | | | |

Kinsta adiciona automaticamente os cabeçalhos acima em todas as requisições do servidor, e se você estiver usando uma CDN, eles provavelmente adicionarão esses cabeçalhos para você também. Assim como com os outros cabeçalhos de cache que discutimos acima, você não pode definir manualmente este cabeçalho em recursos externos.

Adicionando uma Variação: Accept-Encoding no Cabeçalho do Apache

Você pode adicionar a `vary: Accept-Encoding` header no Apache adicionando o seguinte ao seu arquivo `.htaccess`.

```
<IfModule mod_headers.c>
  <FilesMatch "(.js|.css|.xml|.gz|.html)$">
    Header append Vary: Accept-Encoding
  </FilesMatch>
</IfModule>
```

Adicionando uma Variação: Accept-Encoding no Cabeçalho do Nginx

Você pode adicionar a `vary: Accept-Encoding` header no Nginx adicionando o seguinte código ao seu arquivo de configuração. Todos os arquivos de configuração do Nginx estão localizados no diretório `/etc/nginx/`. O arquivo de configuração primário é `/etc/nginx/nginx.conf`.

```
gzip_vary on
```

Alterar o Limite de Memória do WordPress no wp-config.php

Como indicado no WordPress Codex, com o WordPress Versão 2.5, a opção `WP_MEMORY_LIMIT` permite especificar a quantidade máxima de memória que pode ser consumida pelo PHP. Esta configuração pode ser necessária no caso de você receber uma mensagem como “Tamanho de memória permitido de bytes xxxxxx esgotados”.

Por padrão, o WordPress tentará aumentar a memória alocada ao PHP para 40MB para um único site e 64MB para multisite. Eles definem os limites de memória no arquivo `./wp-includes/default-constants.php`, nas linhas 32 - 44.

Você também tem o `memory_limit` PHP no servidor pelo seu provedor de hospedagem. Estas são duas coisas diferentes. Em Kinsta definimos o `memory_limit` padrão para 256M. Se você está correndo para o erro de esgotamento do tamanho da memória você pode tentar aumentar o limite de memória PHP no WordPress.

Adicione o seguinte ao seu ficheiro `wp-config.php` file, mesmo antes da linha que diz “Isso é tudo, edite por etapas. Feliz blogging”.

```
define('WP_MEMORY_LIMIT','256M');
```

Jan Reilink também tem um grande artigo em seu site em saotn.org que faz referência a esta edição com mais detalhes. Ele também dá uma variação no código que você pode usar. Em vez de definir a quantidade manualmente, você pode defini-la para o valor do `memory_limit` do PHP.

```
define('WP_MEMORY_LIMIT',ini_get('memory_limit'));
```

CAPÍTULO 15:

Dicas Sobre Otimização de Front-End e Serviços Externos



Agora vamos mergulhar em algumas formas de acelerar o WordPress, otimizando o front-end. O front-end normalmente envolve qualquer coisa que é tratada inteiramente pelo navegador do lado do cliente, como CSS, JavaScript, imagens, etc. Isso também inclui a análise dos serviços externos que você tem de carregar em seu site e como eles estão impactando seu tempo de carga geral.

Dois dos objetivos mais importantes que você deve ter quando se trata de otimização front-end são:

- **Reduzir o tamanho geral da sua página web.** O tamanho do seu CSS, JavaScript, imagens importa. Um website de 4 MB normalmente vai carregar muito mais lentamente do que um website de 1 MB. Entretanto, Paul Calvano tem um artigo grande no impacto do peso da página no tempo da carga e como é importante certificar-se de que não é a única coisa que seu seguimento como às vezes isto pode ser enganador.
- **Reduzir os pedidos HTTP e os serviços externos.** Com o HTTP/2, várias solicitações e respostas podem agora ser enviadas ao mesmo tempo usando uma única conexão TCP. Embora isso seja ótimo para o desempenho, reduzir as solicitações HTTP ainda pode ajudar a acelerar o seu site WordPress. Isso também inclui a redução do número total de solicitações e serviços externos. Cada um deles adiciona atrasos adicionais, como pesquisas de DNS, conexões TLS e latência de rede.

Teste a Velocidade do Seu Site WordPress para Obter uma Referência Base

Quando se trata de otimizar o front-end do seu site, é sempre bom começar com uma referência. Isso geralmente significa que você precisa executar um teste de velocidade. Existem muitas ferramentas diferentes que você pode usar para isso. Temos guias detalhados sobre como usar corretamente Pingdom e GTmetrix disponíveis em nosso site em kinsta.com.

The screenshot shows the Pingdom Website Speed Test interface. At the top, there is a navigation bar with links for Product, Pricing, Resources, Support, and Tools. The main heading is "Pingdom Website Speed Test" with a subtext "Enter a URL to test the page load time, analyze it, and find bottlenecks." Below this, there is a form with a URL input field containing "https://perfmatters.io/" and a "Test from" dropdown menu set to "North America - USA - Washington D.C.". A green "START TEST" button is visible. Below the form, the "Your Results:" section displays a preview of the test results page and a summary table:

| | | | |
|-------------------|--------|-----------|----------|
| Performance grade | 88 | Page size | 243.0 KB |
| Load time | 398 ms | Requests | 36 |

Aqui estão algumas coisas para ter em mente quando se testa a velocidade:

1. Escolha uma Ferramenta e Fique com Ela

Somos grandes fãs de Pingdom, GTmetrix, WebPageTest, PageSpeed Insights e Chrome DevTools. No entanto, não importa muito qual ferramenta de teste de velocidade você use, pois ela faz isso de forma consistente. Todos eles têm diferentes formas de medir e quantificar a velocidade, por isso escolha uma ferramenta e mantenha-se fiel a ela durante todos os seus testes e otimizações. Até o Google diz para escolher uma.

A ferramenta de teste de velocidade que você escolher não importa tanto quanto escolher um e ficar com ele durante todos os seus testes.

2. Não fique Obcecado por uma Pontuação Perfeita

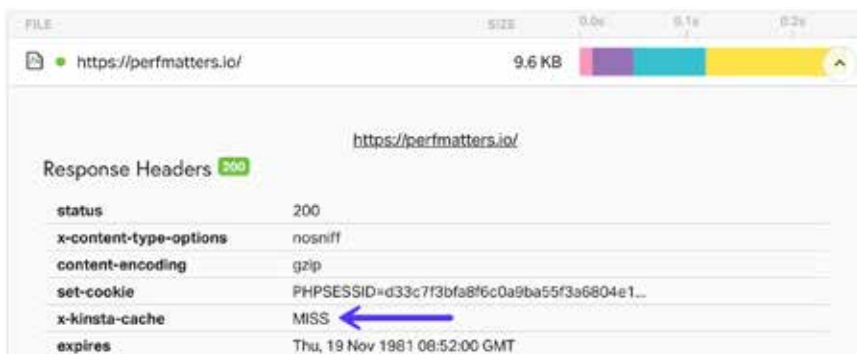
Muitas das ferramentas, como o Google PageSpeed Insights, têm algum tipo de velocidade ou pontuação de desempenho. É importante lembrar que a pontuação nem sempre importa tanto quanto a velocidade do seu site e o desempenho percebido pelo usuário. A pontuação está lá para ajudar a medir o quão bem você está fazendo. Mas obcecar-se com um 100/100 perfeito ou uma pontuação A, em alguns casos pode ser uma perda de tempo. E sites maiores com muitos scripts e anúncios externos nunca terão uma pontuação perfeita, o que é perfeitamente aceitável.

3. O Local do Seu Teste Importa

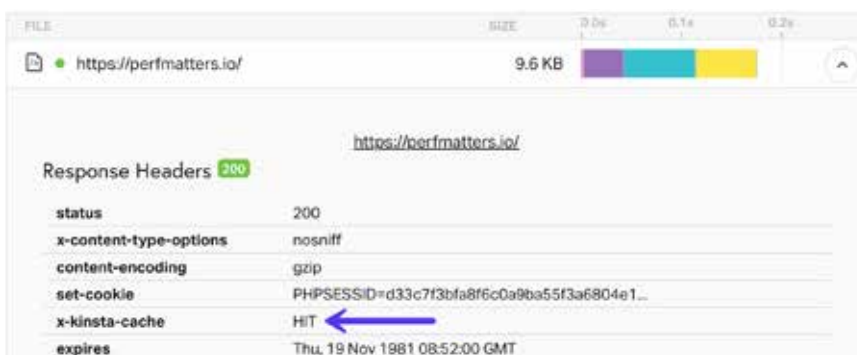
O local que você escolhe quando o teste de velocidade é muito importante. Como discutimos em uma seção anterior, a razão é que tudo isso é relativo ao local do centro de dados que você escolhe. TTFB, latência de rede, tudo isso entra em jogo. Então teste o seu site tanto de um local que esteja próximo ao seu centro de dados quanto de um que esteja distante. Isso também o ajudará a ver o impacto que um CDN pode ter no seu site WordPress.

4. Teste Várias Vezes Por causa do Caching

Como revisamos anteriormente na seção sobre cache, se o cache foi limpo recentemente ou expirou em seu host WordPress ou CDN, ele vai registrar um “MISS” no cabeçalho HTTP. Isso significa que seu site ou ativo não está servindo a partir do cache.



Para ver corretamente a velocidade de todo o seu site você precisa ver tudo carregado do cache, sua página inicial, e todos os ativos registram um “HIT”. Isso às vezes requer a execução de seu teste de velocidade várias vezes. Você pode então pegar a média.



Agora vamos avançar para algumas otimizações front-end que você pode fazer no seu site WordPress.

Remove Query Strings

Um aviso ou recomendação comum que as pessoas vêem nas ferramentas de teste de velocidade é que você deve remover os strings de consulta. O que é isso tudo? Bem, basicamente como funciona é que seus arquivos CSS e JavaScript geralmente têm a

versão do arquivo no final de suas URLs, como por exemplo **https://domain.com/file.min.css?ver=4.5.3**. Alguns servidores e servidores proxy são incapazes de fazer o cache de strings de consulta. Então, removendo-os, às vezes você pode melhorar seu cache.

Você pode removê-las manualmente, soltando o seguinte código no arquivo `functions.php` do seu tema.

```
function remove_query_strings() {  
    if(!is_admin()) {  
        add_filter('script_loader_src','remove_query_strings_split', 15);  
        add_filter('style_loader_src','remove_query_strings_split', 15);  
    }  
}  
  
function remove_query_strings_split($src){  
    $output = preg_split("/(&ver|\?ver)/", $src);  
    return $output[0];  
}  
  
add_action('init','remove_query_strings');
```

Ou você pode usar um plugin premium como o `Perfmatters` (disponível no `perfmatters.io`) para remover strings de consulta com um único clique.

Com Query Strings

| Name | Status |
|---|--------|
| <input type="checkbox"/> editwp.com | 200 |
| <input type="checkbox"/> css?family=Libre+Franklin%3A300%2C300%2C400%2C400...C800%2C600i... | 200 |
| <input type="checkbox"/> style.css?ver=4.9.8 | 200 |
| <input type="checkbox"/> jquery.js?ver=1.12.4 | 200 |
| <input checked="" type="checkbox"/> header.jpg | 200 |
| <input type="checkbox"/> skip-link-focus-fix.js?ver=1.0 | 200 |
| <input type="checkbox"/> global.js?ver=1.0 | 200 |
| <input type="checkbox"/> jquery.scrollTo.js?ver=2.1.2 | 200 |
| <input type="checkbox"/> jizDREVIHgc8qDIbSTKq4XkRIUf2zcZIVbJ.woff2 | 200 |

Aqui está um exemplo de carregamento de scripts com strings de consulta.

Sem Query Strings (Depois do Código)

| Name | Status |
|---|--------|
| editwp.com | 200 |
| css?family=Libre+Franklin%3A300%2C300%2C400%2C400...C600%2C600... | 200 |
| style.css ← | 200 |
| jquery.js | 200 |
| header.jpg | 200 |
| skip-link-focus-fix.js | 200 |
| global.js | 200 |
| jquery.scrollTo.js | 200 |
| jzDREVIIHgc8qDibSTKq4XkRIUI2zcZIVbJ.woff2 | 200 |

Aqui está um exemplo de scripts após ter removido strings de consulta.

No entanto, antes de você sair imediatamente do seu site, é importante saber por que são utilizadas as strings de consulta. A versão em arquivos é normalmente usada pelos desenvolvedores do WordPress para contornar problemas de cache.

Por exemplo, se um desenvolvedor de plugins promover uma atualização e mudar o `style.css` do `?ver=4.6` para `?ver=4.7`, será tratado como uma URL totalmente nova e não será colocado em cache. Se você remover as strings da consulta e atualizar um plugin, isso poderá resultar na versão em cache para continuar a servir. Em alguns casos, isto pode quebrar a aparência do seu site até que o recurso em cache expire ou o cache seja completamente esvaziado.

Além disso, alguns CDNs podem fazer o cache de cordas de consulta. O Kinsta CDN pode e faz por padrão. Então, se você é um cliente Kinsta, as sequências de consulta já estão em cache nos seus ativos.

Eliminar Render-Blocking JavaScript e CSS

Um aviso sobre bloqueio de renderização JavaScript e CSS pode aparecer quando você tiver arquivos impedindo que a página seja carregada o mais rápido possível. JS e CSS específicos são às vezes condicionais, o que significa que não são obrigados a exibir o conteúdo acima do dobrado. Você pode evitar que eles se tornem render-bloqueadores usando atributos de `async` e `defer`.

| Opportunity | Estimated Savings |
|---|-------------------|
| 1 Eliminate render-blocking resources | 2.24 s ^ |
| Resources are blocking the first paint of your page. Consider delivering critical JS/CSS inline and deferring all non-critical JS/styles. Learn more. | |

Para eliminar o render-blocking JavaScript e CSS, você precisa fazer o seguinte:

Limpar o JS do Caminho da Renderização Crítica

Mover o JavaScript para fora do caminho crítico de renderização é normalmente feito adicionando ou o atributo `defer` ou o atributo `async` aos elementos HTML do script que chamam recursos JavaScript.

- O **atributo `async`** diz ao navegador para começar a baixar o recurso imediatamente sem diminuir a velocidade de análise HTML. Uma vez que o recurso está disponível, o analisador HTML é pausado para que o recurso possa ser carregado.
- O **atributo `defer`** diz ao navegador para esperar o download do recurso até que a análise HTML esteja completa. Uma vez que o navegador tenha terminado com o HTML, ele então irá baixar e renderizar todos os scripts diferidos na ordem em que eles aparecem no documento.

Otimizar a Entrega de Recursos do CSS

Otimizar a entrega do CSS significa essencialmente que você precisa descobrir como torná-lo sem bloqueio de renderização.

- Identificar os estilos que são requeridos para renderizar o conteúdo acima do dobrado e entregar esses estilos em linha com o HTML.
- Use condicionalmente o CSS em dispositivos apenas quando necessário.
- Carregar o VCC restante de forma assíncrona.

Fazer tudo isso às vezes pode ser um processo complicado e definitivamente requer alguns ajustes com base nos scripts que você tem carregando em seu site. Aqui estão alguns plugins do WordPress que podem ajudar:

- Autoptimize (Gratuito), disponível no [WordPress.org](https://wordpress.org/plugins/autoptimize/)
- Async JavaScript (Gratuito), disponível no [WordPress.org](https://wordpress.org/plugins/async-javascript/)

- Hummingbird (Gratuito), disponível no WordPress.org

Para uma explicação mais detalhada e uma caminhada, recomendamos verificar nosso post sobre como eliminar o bloqueio de renderização JavaScript e CSS.

Combine CSS externo e JavaScript no WordPress

O aviso de combinação de CSS externo é normalmente visto quando se usa um CDN porque você está hospedando seus arquivos CSS em um domínio externo, tal como `cdn.domain.com`. No passado, uma maneira rápida de corrigir isso é concatenar seus arquivos CSS, ou combiná-los para que eles estejam carregando em um único pedido.

Entretanto, se você estiver rodando sobre HTTPS com um provedor que suporta HTTP/2, este aviso não é mais tão relevante quanto costumava ser. Com o HTTP/2 vários arquivos CSS podem agora ser carregados em paralelo sobre uma única conexão. E mais de 86% dos navegadores suportam HTTP/2.

Mas isso não significa necessariamente que essa otimização esteja completamente morta. Em alguns casos, nós vimos isso ainda acelerar os sites WordPress. Depende do tamanho dos arquivos e de quantos deles existem. Portanto, esta é uma otimização que recomendamos que você ainda teste em seu site.

Uma das maneiras mais fáceis de combinar seus arquivos CSS e JavaScript externos é com o plugin gratuito Autooptimize (Disponível no WordPress.org). Depois de combiná-los, você verá um arquivo “`autooptimize_xxxxx.css`” ou “`autooptimize_xxxxx.js`”. Ele também suporta o carregamento deles a partir do seu CDN. Você também pode fazer isso com o plugin WP Rocket (Disponível em `wp-rocket.me`).

| Name | Status | Domain |
|--|--------|--------------------|
| editwp.com | 200 | editwp.com |
| autooptimize_23d90d8a1eaa0e4363d9e02c50ae363.css ← | 200 | mi0editwpn6... |
| css?family=Libre+Franklin%3A300%2C300%2C400%2C400...C60... | 200 | fonts.googlea... |
| query.js?ver=1.12.4 | 200 | mi0editwpn6... |
| header.jpg | 200 | mi0editwpn6... |
| autooptimize_3f0e0f2c0341cdc5afd7cc0c0e403f92.js ← | 200 | mi0editwpn6... |
| jizDREVIHgc8qDibSTKq40XRUI2zcZVbJ.woff2 | 200 | fonts.gstatic.c... |
| jizAREVIHgc8qDibSTKq40XR24_Sl0q1vj1Oh.woff2 | 200 | fonts.gstatic.c... |
| jizAREVIHgc8qDibSTKq40XRi20-Sl0q1vj1Oh.woff2 | 200 | fonts.gstatic.c... |

Usar Minificação em HTML, CSS, e JavaScript

Podemos reduzir a quantidade de dados que o navegador tem que baixar, minificando os recursos HTML, CSS e JavaScript. Minificação é o processo de remoção de caracteres desnecessários como comentários e espaços em branco do código fonte. Estes caracteres são extremamente úteis no desenvolvimento, mas são inúteis para o navegador renderizar a página.

HTML não-minificado

Aqui está um exemplo de código HTML não-minificado.

```
1 <!DOCTYPE html>
2 <html lang="en-US" class="no-js no-svg">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1">
6 <link rel="profile" href="http://gmpg.org/xfn/11">
7
8 <script>(function(html){html.className = html.className.replace(/\bno-
9 js\b/, 'js')})(document.documentElement);</script>
10 <title>editwp &#8211; Just another WordPress site</title>
<link rel='dns-prefetch' href='//fonts.googleapis.com' />
```

HTML Minificado

Aqui está um exemplo de código HTML minificado.

```
1 <!DOCTYPE html><html lang="en-US" class="no-js no-svg"><head><meta
charset="UTF-8"><meta name="viewport" content="width=device-width,
initial-scale=1"><link rel="profile" href="http://gmpg.org/xfn/11">
<script>(function(html){html.className = html.className.replace(/\bno-
js\b/, 'js')})(document.documentElement);</script> <link type="text/css"
media="all" href="https://mk0editwpn6qbfrow5o9.kinstacdn.com/wp-
content/cache/autoptimze/css/autoptimze_499850c34eb7b4e7cbd9d8b41b2a5
7d6.css" rel="stylesheet" /><style type="text/css"
media="print">#wpadminbar{display:none}</style><style type="text/css"
media="screen">html{margin-top:32px !important}* html body{margin-
```

Você pode usar o plugin gratuito Autoptimize ou WP Rocket para facilmente minificar seus arquivos.

Use Domínios Livres de Cookies

Geralmente, quando você está servindo conteúdo como imagens, JavaScript, CSS, não há motivo para que um cookie HTTP o acompanhe, pois ele cria custos adicionais. Uma vez que o servidor define um cookie para um determinado domínio, todas as solicitações HTTP subsequentes para esse domínio devem incluir o cookie. Este aviso é normalmente visto em sites com um grande número de requisições.

Muitas vezes você pode ignorar este aviso, pois novos protocolos como o HTTP/2 agora tornam isto menos importante. O custo de uma nova conexão é normalmente mais alto do que o custo de transmitir tudo sobre a mesma conexão.

Uma maneira fácil de corrigir este aviso é usar um provedor CDN que pode ignorar cookies, bem como strip cookies que irão impedir completamente o cliente de receber o cabeçalho de resposta do Set-Cookie. O KeyCDN é um provedor de CDN que oferece este recurso. Por padrão, você pode ver as duas opções a seguir estão habilitadas. Esta é uma alternativa fácil sem ter que mexer na movimentação e configuração do seu site para entregar ativos estáticos de um subdomínio separado.

Cache Cookies *

enabled

By default, files with cookies are not cacheable. However, enabling this option will ignore the presence of cookies and therefore force the edge servers to cache these files:

Strip Cookies *

enabled

This feature strips the cookies received from the origin server. The client will not receive the `Set-Cookie` response header. **It is highly recommended to enable this directive if you enable Cache Cookies.**

Se você estiver rodando o Cloudflare, não poderá desativar os cookies nos recursos atendidos através de sua rede. O CloudFlare inclui seu próprio cookie de segurança em seu cabeçalho. Novamente estes cookies são muito pequenos e as implicações de desempenho são extremamente mínimas. Mas se você usa o CloudFlare, não há como contornar este aviso.

Uma segunda maneira de contornar isso é reconfigurando seu site WordPress para entregar os ativos estáticos de um novo domínio ou subdomínio.

Desabilitar Emblemas no WordPress

Quando eles lançaram o WordPress 4.4, eles fundiram o recurso oEmbed no núcleo. Isso permite aos usuários incorporar vídeos do YouTube, tweets e muitos outros recursos em seus sites simplesmente colando uma URL, que o WordPress converte automaticamente em uma incorporação e fornece uma visualização ao vivo no editor visual. Com a atualização, o próprio WordPress se tornou um provedor oEmbed.

Este recurso é útil para muitas pessoas, e você pode querer mantê-lo ativado. Entretanto, isso significa que ele também gera uma solicitação HTTP adicional em seu site WordPress para carregar o arquivo `wp-embed.min.js`. E isso carrega todo o site. Enquanto este arquivo é apenas 1.7 KB, coisas como estas se somam ao longo do tempo. A requisição em si é às vezes um negócio maior do que o tamanho do download do conteúdo.

| Name | Status | Domain |
|---|--------|-----------------|
| <input type="checkbox"/> editwp.com | 200 | editwp.com |
| <input type="checkbox"/> global.js | 200 | mk0editwpn... |
| <input type="checkbox"/> css?family=Libre+Franklin%3A300%2C300%2C400%2C400... | 200 | fonts.google... |
| <input type="checkbox"/> jquery.scrollTo.js | 200 | mk0editwpn... |
| <input type="checkbox"/> wp-embed.min.js ← | 200 | mk0editwpn... |
| <input type="checkbox"/> style.css | 200 | mk0editwpn... |
| <input type="checkbox"/> jquery.js | 200 | mk0editwpn... |
| <input type="checkbox"/> header.jpg | 200 | mk0editwpn... |
| <input type="checkbox"/> skip-link-focus-fix.js | 200 | mk0editwpn... |
| <input type="checkbox"/> wp-emoji-release.min.js | 200 | editwp.com |

Existem algumas formas diferentes de desabilitar as incrustações no WordPress. Você pode fazer isso com os seguintes plugins:

- Disable Embeds (Gratuito, disponível no WordPress.org)
- Perfmatters (Disponível no perfmatters.io)

Ou você também pode desabilitar as incrustações com código. Devido ao seu comprimento, ele pode ser encontrado em nosso site em <https://kinsta.com/knowledgebase/disable-embeds-wordpress/>.

Desabilitar Emojis no WordPress

Semelhante aos embeds, no WordPress 4.2, eles adicionaram suporte para emojis no núcleo

para navegadores mais antigos. O grande problema com isso é que ele gera uma solicitação HTTP adicional em seu site WordPress para carregar o arquivo `wp-emoji-release.min.js`. E isso carrega todo o site. Enquanto este arquivo é apenas 10.5 KB, ele é inútil se você não estiver usando emojis no seu site.

| Name | Status | Domain |
|---|--------|------------------|
| <input type="checkbox"/> editwp.com | 200 | editwp.com |
| <input type="checkbox"/> global.js | 200 | mk0editwpn... |
| <input type="checkbox"/> css?family=Libre+Franklin%3A300%2C300%2C400%2C400... | 200 | fonts.google... |
| <input type="checkbox"/> jquery.scrollTo.js | 200 | mk0editwpn... |
| <input type="checkbox"/> wp-embed.min.js | 200 | mk0editwpn... |
| <input type="checkbox"/> style.css | 200 | mk0editwpn... |
| <input type="checkbox"/> jquery.js | 200 | mk0editwpn... |
| <input checked="" type="checkbox"/> header.jpg | 200 | mk0editwpn... |
| <input type="checkbox"/> skip-link-focus-fix.js | 200 | mk0editwpn... |
| <input type="checkbox"/> wp-emoji-release.min.js ← | 200 | editwp.com |
| <input type="checkbox"/> jzDREVIHgc8qDlbSTKq4XkRiUf2zcZVbJ.woff2 | 200 | fonts.gstatic... |

Há algumas maneiras diferentes de desativar emojis no WordPress. Você pode fazer isso com os seguintes plugins:

- Disable Emojis (Gratuito, disponível no WordPress.org)
- Perfmatters (Disponível no perfmatters.io)

Ou você também pode desativar emojis com código. Devido ao seu comprimento, ele pode ser encontrado em nosso site em <https://kinsta.com/knowledgebase/disable-emojis-wordpress/>

Como Acelerar os Comentários do WordPress ou Desativá-los

Uma seção ocupada de comentários sobre um site pode causar muitos problemas de desempenho. Basta pensar nos recursos que vão para fazer os comentários funcionarem:

- Uma base de dados é consultada para obter os comentários existentes.
- As entradas da base de dados são criadas para cada novo comentário.
- Comentários e metadados de comentários são recebidos e processados pelo navegador de um visitante.
- Recursos externos, como o Gravatars, são solicitados, baixados e carregados (requerendo uma pesquisa DNS separada).

- Em muitos casos, grandes recursos JavaScript e jQuery têm que ser baixados e processados para que o sistema de comentários funcione da maneira que deve ser feito.

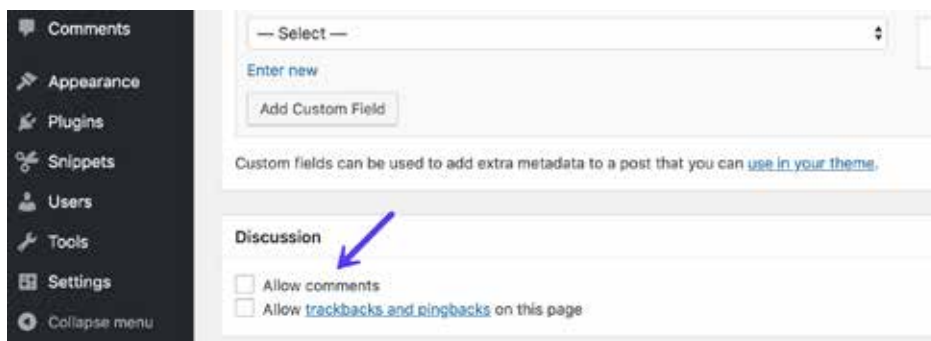
Aqui estão quatro opções diferentes que você pode fazer para acelerar os comentários do WordPress:

Opção 1 - Desabilitar Comentários

Se o seu site não está recebendo muitos comentários e você acha que eles não estão agregando nenhum valor, talvez seja melhor desativar os comentários por completo. Lembre-se, os comentários podem impactar o seu SEO, já que o Google normalmente os rastreia como conteúdo adicional na página, portanto você só deve aprovar comentários de alta qualidade. Confira estas três maneiras fáceis de desativar os comentários:

1. Desativar Comentários dentro das Opções do WordPress

Você pode desabilitar comentários individualmente em uma página ou post, editando-a e desmarcando a opção “Permitir comentários” na seção “Discussão”. Se você não ver esta seção você pode precisar rolar para cima e ativá-la sob a opção “Opções de Tela”.



Você também pode desativar globalmente as discussões. Você pode fazer isso clicando em “Discussão” em Configurações no seu painel do WordPress e depois desmarcando a opção “Permitir que as pessoas postem comentários sobre novos artigos”. O inconveniente deste método é que ele não afeta os artigos mais antigos, apenas os novos.

Se você precisa de mais controle e simplesmente quer desativar globalmente todos os comentários de uma só vez, recomendamos usar um dos seguintes métodos abaixo.



2. Desativar Comentários com Plugin

Uma das maneiras mais fáceis de desativar comentários é usar um plugin gratuito como o [Disable Comments](#) (Disponível no [WordPress.org](#)). Você pode desabilitar comentários em qualquer lugar com um clique, ou desabilitá-los em posts, páginas, ou tipos de posts de mídia.

3. Desativar Comentários com Código

Você também pode desabilitar comentários com código, simplesmente removendo a seção de comentários do seu tema WordPress com comentários HTML simples.

Em seu arquivo `page.php`, simplesmente substitua o seguinte código:

```
<?php comments_template(, true ); ?>
```

Com este código:

```
<!-- Begin Comment  
<?php comments_template(," true );?>  
End Comment -->
```

Isso o remove de suas páginas. E para removê-lo de seus posts no blog, navegue até seu arquivo single.php e substitua o seguinte código:

```
<?php comments_template();?>
```

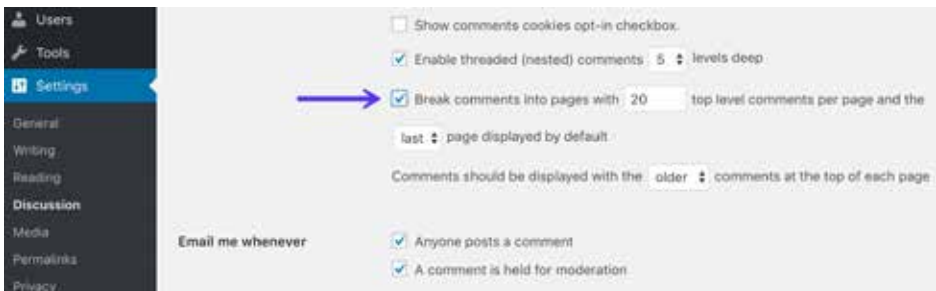
Com este código:

```
<!-- Begin Comment  
<?php comments_template();?>  
End Comment
```

Opção 2 - Otimizar Comentários do Native WordPress

Sua segunda opção seria otimizar o sistema de comentários do WordPress nativo. Uma maneira seria reduzir o número de comentários carregados com o carregamento inicial da página.

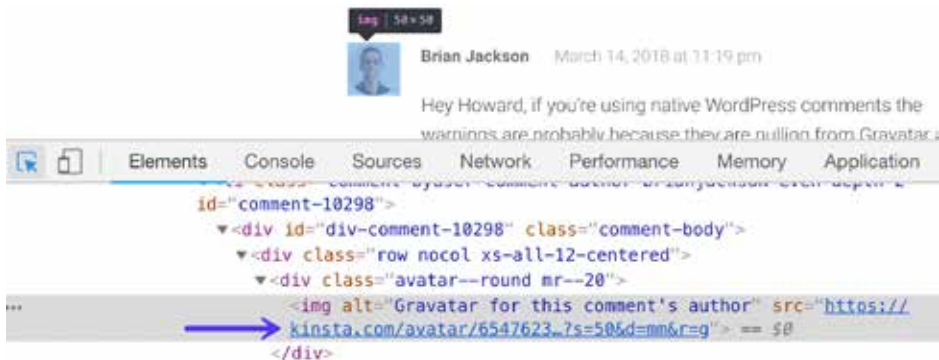
1. Vá para Configurações → Discussão na área de administração do WordPress.
2. Procure a seção Outras configurações de comentários.
3. Marque a caixa de seleção ao lado de Dividir comentários em páginas com e adicione um valor para o número de comentários que você deseja exibir com a carga inicial da página.



Outra opção que você tem é usar o Host Gravatars no seu CDN. Esta é a abordagem que fazemos na Kinsta.

Por padrão, quando os comentários do WordPress são carregados, cada Gravatar requer uma requisição HTTP. Então, se uma página for carregada com comentários de 50 comentadores diferentes, 50 requisições HTTP serão necessárias para baixar todos esses Gravatars. Como você pode imaginar, isso pode impactar a velocidade da sua página. Sem mencionar o fato de que vimos o DNS externo do gravatar.com ser lento às vezes e, em alguns casos, até mesmo o timeout.

Se você olhar o Gravatars no blog da Kinsta, você pode ver que eles estão carregando do Kinsta.com (incluindo o nosso CDN).



Opção 3 - Utilizar um Sistema de Comentários de Terceiros

Sua terceira opção é usar um sistema de comentários de terceiros. Se o seu site está hospedado em um servidor compartilhado barato e sem recursos, então usar um sistema de comentários de terceiros pode acelerar as páginas com muitos comentários. São as mesmas idéias da otimização de imagens, descarregue o trabalho. Entretanto, se você estiver hospedado com Kinsta ou outro web host de qualidade, mudar para um terceiro não fará muito para ajudar na velocidade de carregamento do seu site e pode diminuir a velocidade.

| | | | | |
|---|---|--------|---|--|
|  | vader_f0c2350b4af6d1a9874500d7a85c412... | 3.4 kB |  |  |
|  | spide_e37425042815ed495a8926870fede... | 3.6 kB |  |  |
|  | pops_71820562911bed317551e054543459... | 7.5 kB |  |  |
|  | discovery.ec3171c4e72eece074c2ac724... | 1.4 kB |  |  |
|  | data.imageof_base64_R0GODhAQABAAA... | 0 B | | |
|  | novotar59_7a2b6c410943665c0b6f0ddee... | 1.1 kB |  |  |
|  | discovery.bundle_e75d8672e9517603e31b... | 7.3 kB |  |  |
|  | novotar92.png | 2.1 kB |  |  |
|  | event=ref_embed&fimage=551... | 229 B |  |  |
|  | event=referrer_network_default_h... | 276 B |  |  |

Certifique-se sempre de acelerar o teste do sistema de comentários de terceiros que você está tentando. Dê uma olhada em todas as solicitações separadas que os Disqus geram (como mostrado abaixo). Enquanto a maioria dessas requisições está carregando assincronamente, você ainda notará algum tempo adicional de carga se estiver usando Disqus.

Opção 4 - Carregamento Preguiçoso Sobre os Comentários

Sua quarta opção é carregar preguiçosamente os comentários para que eles não abrande a renderização da página inicial. Aqui estão alguns plugins que você pode querer conferir:

- **Lazy Load for Comments** (Gratuito, disponível no WordPress.org): Este plugin permite que você carregue preguiçosamente comentários nativos do WordPress.
- **Disqus Conditional Load** (Gratuito, disponível no WordPress.org): Se você quiser usar o sistema de comentários Disqus, este é um plugin obrigatório para carregar comentários preguiçosos.

Desativar RSS Feeds WordPress

Se você não estiver usando a parte de blogs do WordPress em seu site, você pode desativar os feeds RSS do WordPress. Embora isso não tenha um grande impacto no desempenho, tudo ajuda. É também uma coisa a menos com que você tem que se preocupar.

Confira estas duas formas diferentes de desabilitar os feeds RSS no WordPress:

1. Desabilitar RSS Feed com Plugin

A primeira maneira de desativar um feed RSS WordPress é usar um plugin gratuito como o Disable Feeds (Disponível no WordPress.org). Este plugin desabilita todos os feeds RSS/Atom no seu site WordPress, redirecionando todas as solicitações.

Você também pode usar um plugin premium como o Perfmatters (desenvolvido por um membro da equipe da Kinsta, disponível no perfmatters.io), que permite que você desabilite tanto os feeds RSS quanto os links de feed RSS, juntamente com outras otimizações para o seu site WordPress.

2. Desabilitar RSS Feed com Código

Se você não quiser usar um plugin, você também pode desativar os feeds RSS com código. Solte o seguinte código no arquivo `functions.php` do seu tema.

```
function itsme_disable_feed() {  
    wp_die(__( 'No feed available, please visit the <a href="' esc_url(  
    home_url( '/' ) ).">homepage</a>!' ));  
}  
  
add_action('do_feed','itsme_disable_feed', 1);  
add_action('do_feed_rdf','itsme_disable_feed', 1);  
add_action('do_feed_rss','itsme_disable_feed', 1);  
add_action('do_feed_rss2','itsme_disable_feed', 1);  
add_action('do_feed_atom','itsme_disable_feed', 1);  
add_action('do_feed_rss2_comments','itsme_disable_feed', 1);  
add_action('do_feed_atom_comments','itsme_disable_feed', 1);
```

O WordPress também gera links para os feeds RSS dentro do cabeçalho da sua página web. Você pode ir um passo além e remover esses links de dentro do código HTML da sua página. Deixe cair o seguinte código no arquivo `functions.php` do seu tema.

```
remove_action('wp_head','feed_links_extra', 3);  
remove_action('wp_head','feed_links', 2);
```

Use o Prefetch e o Preconnect

Dicas e diretivas de recursos como prefetch e pré-conexão podem ser uma ótima maneira de acelerar o WordPress nos bastidores.

Prefetch

O prefetch DNS permite que você resolva nomes de domínio (faça uma busca DNS em segundo plano) antes que um usuário clique em um link, o que, por sua vez, pode ajudar a melhorar o desempenho. Isso é feito adicionando uma tag `rel="dns-prefetch"` no cabeçalho do seu site WordPress.

```
<link rel="dns-prefetch" href="//domain.com">
```

Algumas coisas comuns para usar o DNS prefetching é sua URL CDN, fontes do Google, Google Analytics, etc.

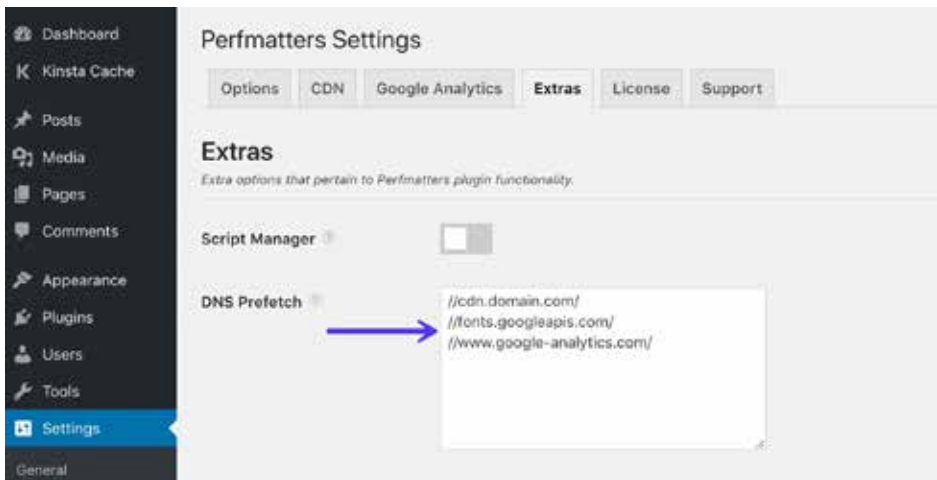
```
<link rel="dns-prefetch" href="//cdn.domain.com/">
```

```
<link rel="dns-prefetch" href="//fonts.googleapis.com/">
```

```
<link rel="dns-prefetch" href="//www.google-analytics.com">
```

O Prefetch também é suportado pela maioria dos navegadores modernos. Existem algumas maneiras de implementar isso.

- 1.** Você pode adicionar o código ao seu cabeçalho usando um plugin gratuito como Head, Footer e Post Injections (disponível no WordPress.org).
- 2.** Você pode adicionar o código manualmente ao arquivo `header.php` do seu tema.
- 3.** Implemente o prefetch DNS usando um plugin premium como o Perfmatters (disponível no perfmatters.io). Basta clicar na aba “Extras” no plugin Perfmatters e adicionar domínios. Formatar `//domain.tld` (um por linha)



Preconnect

Preconnect permite que o navegador configure conexões antecipadas antes de uma solicitação HTTP, eliminando a latência de ida e volta e economizando tempo para os usuários.

O Preconnect é uma ferramenta importante na sua caixa de ferramentas de otimização... ele pode eliminar muitas viagens de ida e volta caras do seu caminho de pedido - em alguns casos reduzindo a latência do pedido em centenas e até milhares de milissegundos.



Iya Grigorik

Isso é feito adicionando uma tag `rel="preconnect"` no cabeçalho do seu site WordPress.

```
<link rel="preconnect" href="//domain.com">
```

Alguns exemplos de coisas que você pode querer utilizar isso para incluir sua URL CDN ou fontes do Google.

```
<link rel="preconnect" href="https://cdn.domain.com">
```

```
<link rel="preconnect" href="https://fonts.gstatic.com">
```

O Preconnect é suportado pela maioria dos navegadores modernos, com exceção do Internet Explorer, Safari, IOS Safari e Opera Mini. Existem algumas maneiras de implementar isso.

1. Você pode adicionar o código ao seu cabeçalho usando um plugin gratuito como Head, Footer e Post Injections (disponível no WordPress.org).
2. Você pode adicionar o código manualmente ao arquivo `header.php` do seu tema.
3. Implemente pré-conexão usando um plugin como o Perfmatters (disponível em perfmatters.io). Basta clicar na aba “Extras” no plugin Perfmatters e adicionar domínios. Formato: `scheme://domain.tld` (um por linha).



Desativar Scripts por Página/Post

Outra maneira muito poderosa de acelerar o WordPress é pesquisar cada pedido que está carregando em suas páginas e posts. Você provavelmente acabará encontrando scripts que estão carregando em todo o site e que não deveriam estar.

Você pode usar um plugin premium como o Perfmatters (disponível no perfmatters.io) que tem um recurso “Script Manager” incorporado. Isso permite que você desabilite scripts (CSS e JavaScript) por página/post, ou mesmo em todo o site com um único

clique. Mais uma vez, este plugin é desenvolvido por um membro da equipe da Kinsta.

Alguns exemplos do que isto pode ser usado:

- O popular plugin Contact Form 7 é carregado por si próprio em todas as páginas e posts. Você pode facilmente desativá-lo em qualquer lugar com um clique e ativá-lo apenas na sua página de contato.
- Os plugins de compartilhamento de mídia social só devem ser carregados nos seus posts. Você pode facilmente desabilitá-lo em qualquer lugar e carregá-lo apenas em tipos de posts, ou mesmo em tipos de posts personalizados.
- O plugin Tabela de conteúdo (TOC) é carregado em cada página e postagem. Com o gerenciador de scripts, você pode facilmente controlar onde você quer que ele seja carregado.
- Se você atualizou para o WordPress 5.0 e não está usando o editor de blocos do Gutenberg, talvez você ainda esteja usando o editor clássico ou outro editor de terceiros, há dois scripts front-end adicionais que são adicionados em todo o site e que você pode desativar: `/wp-includes/css/dist/block-library/style.min.css` e `/wp-includes/css/dist/block-library/theme.min.css`.

Porque é que Alguns Plugins são codificados desta forma?

Você pode estar se perguntando por que todos os desenvolvedores de plugins não carregam seus scripts apenas quando o plugin é detectado na página? Bem, é um pouco mais complicado do que isso. Por exemplo, se você tem um plugin como o Formulário de Contato 7, ele também tem atalhos que permitem que você o coloque em qualquer lugar. Isto inclui deixá-lo cair em um widget. Com o WordPress, é muito mais difícil consultar dados a partir deles quando você decodifica scripts ao invés de consultar dados dos metadados do post ou da página.

Portanto, muitas vezes isso é devido a problemas de usabilidade. Quanto menos chance eles tiverem de quebrar um plugin, menos tickets e suporte eles terão. No entanto, com muitos plugins no mercado, há maneiras de contornar isso e codificar a performance se eles quiserem. Infelizmente, às vezes, o grande número de downloads e usuários torna a codificação para a usabilidade uma prioridade.

Visitando o Script Manager

Vamos dar-lhe uma pequena visita guiada ao Script Manager. Depois de clicar nele na sua barra de ferramentas você será apresentado com todos os scripts carregados naquela URL atual, tanto JavaScript como arquivos CSS. Você então terá as seguintes opções:

1. **Status On** (configuração padrão)
2. **Status Off:** Desativar em todos os lugares (você pode então escolher em que tipos de posts você quer ativá-lo, juntamente com a URL atual)
3. **Status Off:** Desativar somente na URL atual (isso é muito útil para usar em sua página inicial)
4. **Status Off:** Exceções (URL atual, tipo de post, ou arquivo)

Tudo é **agrupado pelo nome do plugin ou tema**. Isto torna super fácil desativar um plugin inteiro de uma só vez. Normalmente um plugin WordPress terá tanto um arquivo JavaScript quanto um CSS. Um tema WordPress pode ter mais de 10 arquivos. O Script

Manager suporta até mesmo regex para quando você tem uma estrutura de URL ou instalação mais complicada.

Depois de selecionar e ou modificar as configurações, certifique-se de clicar em “Salvar” na parte inferior. Você pode então testar em uma ferramenta de velocidade do site para garantir que os scripts não estejam mais carregando na página ou no post. Certifique-se de limpar o seu cache primeiro! E se alguma coisa correr mal visualmente no seu site, pode sempre voltar a activá-lo nas definições para voltar ao normal.

Em um teste de velocidade pelo woorkup.com, eles foram capazes de **diminuir o tempo total de carga em 20,2%**. Somente em sua página inicial, eles foram capazes de reduzir o número de solicitações HTTP de 46 para 30. Seu tamanho de página também diminuiu de 506.3 KB para 451.6 KB.

Analizando o Desempenho de Terceiros

Basicamente, qualquer coisa que você chamar externamente do seu site tem conseqüências em termos de tempo de carga. O que torna este problema ainda pior é que alguns deles são lentos apenas intermitentemente, tornando a identificação do problema ainda mais difícil.

Um serviço externo de terceiros pode ser considerado qualquer coisa que se comunique com o seu site WordPress de fora do seu próprio servidor. Aqui estão alguns exemplos comuns que encontramos regularmente:

- Plataformas de mídia social como Twitter, Facebook e Instagram (widgets ou pixels de conversão)
- Redes de publicidade de terceiros como Google AdSense, Media.net, BuySellAds, Amazon Associates
- Análise de sites e scripts de rastreamento como Google Analytics, Crazy Egg, Hotjar, AdRoll
- Ferramentas de teste A/B como Optimizely, VWO, Unbounce
- Sistemas de comentários WordPress como Disqus, Jetpack, comentários no Facebook
- Ferramentas de backup e segurança como VaultPress, Sucuri, CodeGuard

- Ferramentas de compartilhamento social, tais como SumoMe, HelloBar
- Redes CDN como KeyCDN, Amazon CloudFront, CDN77, e StackPath
- Javascript hospedado externamente

Qual o impacto de alguns desses rastreadores de terceiros no desempenho? Em nosso próprio estudo de caso, vimos que os scripts de terceiros **aumentaram o tempo de carregamento de páginas em 86,08%**.

Ghostery também mediu os 500 maiores domínios dos EUA no Alexa, e os resultados foram surpreendentes, embora para nós, não surpreendentemente. Os sites foram 2x mais lentos quando nenhum rastreador foi bloqueado. O que significa que esses scripts de rastreamento de terceiros são um dos principais contribuidores para a lentidão da velocidade de carregamento de páginas na web.

AVERAGE PAGE LOAD TIME WITH vs. WITHOUT TRACKERS



Average page load time (seconds) measured by Ghostery, of the top 500 US domains according to Alexa.

Fonte da imagem: Ghostery

Você tem que ter muito cuidado no seu site WordPress. Apenas uma chamada ruim de API de terceiros pode acabar com o tempo do seu site inteiro! Sim, não deve funcionar dessa forma, mas em muitos casos funciona. Nós já vimos isso mais vezes do que podemos contar.

O New Relic fornece uma maneira excelente e fácil de monitorar seus serviços externos ao longo do tempo. Neste exemplo abaixo, podemos ver chamadas externas sendo feitas para [tweetcount.com](https://www.tweetcount.com/), [graph.facebook.com](https://www.graph.facebook.com/), e [widgets.pinterest.com](https://www.widgets.pinterest.com/).

Top 5 external services

By total response time



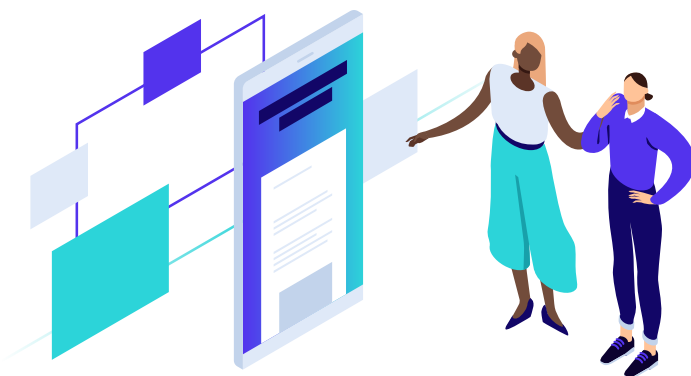
External calls per minute



É importante que sempre que você adicionar um novo recurso ou plugin ao seu site que você investigue o carregamento de recursos externos a partir dele. Quanto menos, melhor!

CAPÍTULO 16:

Otimização da Versão Mobile em Mente



O Google começou a lançar seu primeiro índice de mobilidade em 26 de março de 2018. Anteriormente, os sistemas de rastreamento, indexação e ranking do Google utilizavam a versão desktop dos sites. O Mobile-first indexing significa que o Googlebot agora utilizará a versão móvel do seu site WordPress para indexação e ranking. Isso ajuda a melhorar a experiência de busca para usuários móveis.

O Google começou a lançar seu primeiro índice de mobilidade em 26 de março de 2018. Anteriormente, os sistemas de rastreamento, indexação e ranking do Google utilizavam a versão desktop dos sites. O Mobile-first indexing significa que o Googlebot agora utilizará a versão móvel do seu site WordPress para indexação e ranking. Isso ajuda a melhorar a experiência de busca para usuários móveis.

Quando se trata de otimizar o seu site para mobile-first, a **velocidade é um dos fatores mais importantes para se focar**. A velocidade desempenha um papel importante em tudo, desde a usabilidade até a taxa de retorno e determinar se os potenciais compradores voltarão ou não ao seu site. Na verdade, a velocidade é agora um fator de página de destino para pesquisas no Google e Anúncios para pesquisas no celular.

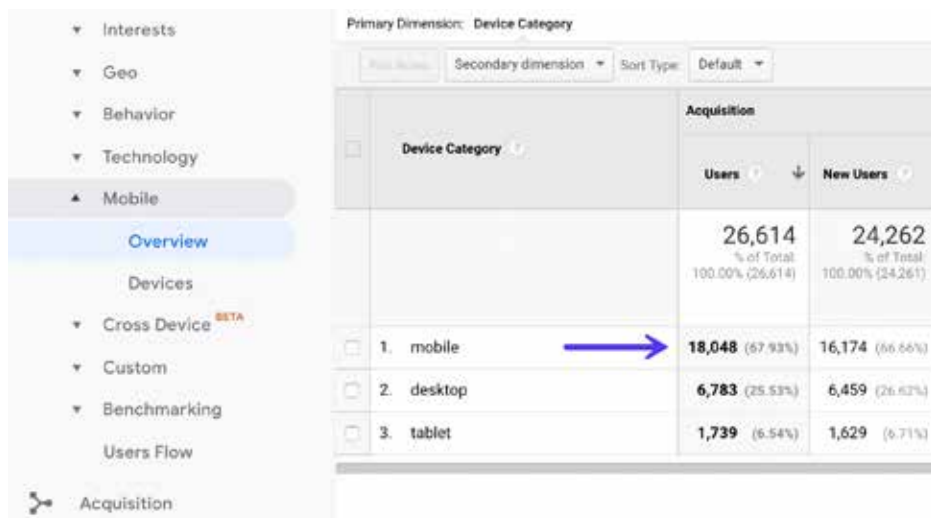
Más experiências móveis levarão a maioria dos usuários a nunca mais voltar. De acordo com o último relatório de velocidade de páginas do Google, o tempo médio que um site móvel levou para carregar em 2018 foi de 15 segundos. Você consegue imaginar esperar tanto tempo para carregar uma única página? Assombroso.

Os usuários exigem (e merecem) melhor. De acordo com o mesmo relatório de velocidade de páginas, 53% dos visitantes de um site móvel deixam páginas que demoram mais do que apenas três segundos para serem carregadas.

Experiências móveis lentas não estão matando conversões. Elas estão impedindo que você tenha até mesmo a chance de converter os potenciais clientes. Como os tempos de carregamento de páginas aumentam em apenas alguns segundos, a probabilidade de alguém saltar sobe exponencialmente. Aqui estão algumas coisas a serem consideradas ao otimizar para celular.

Confira o Seu Tráfego Móvel

É sempre importante dar uma olhada em quanto tráfego móvel você está recebendo, pois isso pode mudar um pouco suas prioridades. Você pode ver quantos dispositivos móveis estão visitando seu site no Google Analytics em “Audience → Mobile → Overview”. Como você pode ver neste site, mais de 67% de todo ele é tráfego do celular. Isso é muito!



The screenshot shows the Google Analytics interface for Audience Mobile Overview. The primary dimension is 'Device Category'. The table below summarizes the data:

| Device Category | Acquisition | |
|-----------------|--|--|
| | Users | New Users |
| | 26,614 % of Total 100.00% (26,614) | 24,262 % of Total 100.00% (24,261) |
| 1. mobile | 18,048 (67.93%) | 16,174 (66.66%) |
| 2. desktop | 6,783 (25.53%) | 6,459 (26.62%) |
| 3. tablet | 1,739 (6.54%) | 1,629 (6.71%) |

Se você é um cliente Kinsta, você também pode verificar o tráfego do seu celular vs. desktop no MyKinsta Analytics. Como você pode ver neste site, mais de 88% do tráfego é da área de trabalho. É sempre importante checar e não apenas assumir. Só porque todos dizem que as coisas estão indo para o celular, nem sempre significa que seja para o seu site. Olhe os dados.



Tenha Certeza de Que o Seu Site é Responsivo

The image shows two side-by-side screenshots. On the left is a 'Page is mobile friendly' report from Google, dated Nov 18, 2018. It includes a green heading 'Page is mobile friendly', a sub-heading 'This page is easy to use on a mobile device', and a section for 'Additional resources' with links to a site-wide report, learning more about mobile-friendly pages, and a discussion group. On the right is a screenshot of a mobile device displaying the Kinsta website. The page features a blue gradient background with the text 'Premium WordPress hosting for everyone, small or large'. Below this is a 'VIEW PLANS' button and a cookie consent banner with 'COOKIE SETTINGS' and 'ACCEPT COOKIES' buttons.

Em 2019, é melhor que seu site seja mais ágil! Isso significa que ele utiliza consultas de mídia para reduzir automaticamente a escala em dispositivos móveis. Se você ainda não fez isso, provavelmente já está atrás da sua concorrência. Todos os temas do WordPress que mencionamos anteriormente neste post são totalmente responsivos e ficam ótimos em todos os dispositivos

Use a ferramenta Mobile-Friendly do Google para testar e garantir que seu site passe todos os requisitos.

Dupla Checagem Para Ter Certeza de Que o srcset Está Funcionando

No passado, era muito importante que você carregasse as imagens em escala e não deixasse o CSS redimensioná-las. Entretanto, isso não é mais tão importante, uma vez que o WordPress 4.4 agora suporta imagens responsivas (não redimensionadas pelo CSS). O WordPress cria automaticamente vários tamanhos de cada imagem carregada para a biblioteca de mídia. Ao incluir os tamanhos disponíveis de uma imagem em um atributo srcset, os navegadores podem agora optar por baixar o tamanho mais apropriado e ignorar os outros. Veja um exemplo de como seu código se parece abaixo.

```

```

Devido a todos os plugins de imagem de terceiros e customizações por aí, houve muitas vezes em que vimos isso não funcionar corretamente. Portanto, é importante verificar duas vezes se suas imagens estão recebendo corretamente o atributo `srcset` adicionado com diferentes versões para diferentes tamanhos de tela. A otimização de imagens agora é importante para sempre.

O Google AMP Pode Ser uma Solução Para Você

O Google AMP (Accelerated Mobile Pages Project) foi lançado originalmente em outubro de 2015. O projeto se baseia no AMP HTML, um novo framework aberto construído inteiramente a partir de tecnologias web existentes, que permite aos sites construir páginas web leves. Para simplificar, ele oferece uma maneira de servir uma versão descascada de sua página web atual.

Temos uma relação de amor e ódio com o Google AMP, e muito da comunidade também tem. Nós mesmos já testamos isso e não vimos bons resultados. No entanto, isso não significa que você não vai ver. Cada site é diferente, e o Google AMP está constantemente sendo melhorado.

Você pode começar rapidamente com o Google AMP no seu site WordPress com um dos seguintes plugins:

- AMP for WordPress (Free, available on WordPress.org)
- AMP for WP (Free, available on WordPress.org)

Resumo

Como você provavelmente pode dizer, estamos obcecados com todas as diferentes formas de acelerar o WordPress. Ter um site rápido ajuda a aumentar a sua classificação, melhora a capacidade de rastreamento dos mecanismos de busca, melhora as taxas de conversão, aumenta o tempo no site e diminui a sua criação de ressaltos. Sem mencionar o fato de que todos adoram visitar um site rápido!

Esperamos que este guia de aceleração tenha sido útil e que você tenha sido capaz de tirar algumas coisas e aplicá-las no seu site WordPress.

Não deixe de conferir também nosso programa de afiliados de hospedagem WordPress em <https://kinsta.com/affiliates>. Ganhe até \$500 por cada indicação + 10% de **comissões mensais recorrentes ao longo da vida**. Este é o melhor negócio de afiliados que você encontrará na indústria. Vamos agilizar a web juntos!

Para saber mais sobre a Kinsta, visite-nos em <https://kinsta.com/pt/>

Obrigado pela leitura!



Notas

