

Влияние скорости загрузки сайтов на конверсию трафика

Введение

Скорость загрузки сайтов один из ключевых параметров, которые влияют на процент отказов и конверсию пользователей. Примером тому является типичное поведение при посещении сайта. Если пользователь переходит на страницу, и вынужден ждать длительное время, чтобы получить необходимую информацию, то он, вероятнее всего, покинет страницу и перейдет на сайт, который сразу предоставит ему необходимую информацию.

Наиболее актуальна скорость загрузки для коммерческих сайтов. Особо остро стоит этот параметр, если сайт продвигается при помощи контекстной рекламы в поисковых системах. Опираясь на исследования и собственный опыт, выделяется следующее поведение пользователя:

1. Пользователь вводит запрос в поисковой строке Google.
2. Пользователь получает результаты поиска. Первые 3-5 результатов выдачи рекламные и помечены соответствующим маркером.
3. Пользователь открывает в новых вкладках первые несколько сайтов из поисковой выдачи.
4. Пользователь начинает поочередно изучать открытые страницы.

В данном контексте, если ваш сайт будет долго загружаться, то пользователь сможет быстрее перейти к следующему сайту, нежели ожидать загрузки вашего.

Особую актуальность скорость загрузки имеет на мобильных устройствах. Мобильный интернет не имеет стабильной быстрой скорости загрузки, а также контекст и обстановка пользователя не располагает к размеренному поиску информации. Если хочет получить информацию в дороге, в спешке с низкой скоростью интернета, то скорость загрузки страницы играет важнейшую роль.

Скорость загрузки сайта для бизнеса

Не для кого не секрет, что современный бизнес в большей степени сейчас зависит от продвижения своих услуг в интернете. Сайт, как один из инструментов продвижения своего бренда и услуг, должен соответствовать многим критериям, чтобы удовлетворить ожидания пользователя. К таким критериям относятся: UX, информационная архитектура, актуальность предложения, маркетинговая составляющая контента и прочие. Первое с чем сталкивается пользователь - как быстро он может добраться до информации на вашем сайте. В среднем конверсия в целевое действие составляет 1/10. Иными словами, 10 из 100 человек посетивших ваш ресурс вероятнее всего совершит покупку или иное целевое действие. Из за низкой скорости загрузки вы рискуете потерять пользователей на начальном этапе - посещении вашего сайта. Предположим, из за долгой скорости загрузки вместо 100 пользователей к вам перешло 60. Средняя конверсия из 60 человек будет 6 пользователей, совершивших целевое действие. Если перевести это в плоскость денег и дохода, то предположим, что каждый пользователь приносит вам 50 грн. Получаем следующее:

- 100 пользователей - 10 совершило покупку - $10 \cdot 50 = 500$ грн
- 60 пользователей - 6 совершило покупку - $6 \cdot 50 = 300$ грн

Только из за скорости загрузки вы как предприниматель рискуете потерять 40% дохода в данном случае.

Исключения

Исключением из этих правил являются большие и популярные ресурсы, такие как Facebook, OLX, Rozetka и.пр. Данный феномен обусловлен тем, что мотивация пользователя и желания получить контент выше, чем раздражение от времени ожидания. Выгода от ожидания работает только в том случае, если ваш сервис достаточно известен и помогает пользователю решить его проблему. Пользователь, заранее знает что ожидать от ресурса и что он получит. Взамен он готов пойти на ожидания.

Исследования

Крупные компании, включая Google, проводили многочисленные исследования на тему влияния скорости загрузки. Далее описаны результаты этих исследований.

Google: *«Если сайт загружается дольше трех секунд, 53% пользователей покинут его».*

Ссылка: <https://www.thinkwithgoogle.com/consumer-insights/consumer-mobile-brand-content-interaction/>

Скорость загрузки сайта - важный параметр. Для того чтобы проверить скорость загрузки, можно воспользоваться сервисом от Google - Test My Site.

Ссылка: <https://www.thinkwithgoogle.com/intl/en-gb/feature/testmysite/>

SOASTA: *«Каждая секунда задержки мобильной страницы снижает конверсию на 20%».*

По результатам совместного исследования аналитического агентства SOASTA конверсия напрямую зависит от того, как быстро у пользователя целевой сайт. Так, те страницы, которые чаще приводили к конверсии, были в среднем на 26% быстрее остальных. В агентстве заключили, что быстрые страницы делают пользование комфортным, создают условия для дальнейших действий: заказа, покупки, звонка.

Google: *«У мобильной страницы, скорость которой снизилась с 1 до 10 секунд, показатель отказа вырос на 123%».*

Наибольший разрыв в конверсии наблюдается в интервале 2-5 секунд. По данным исследований, конверсия падает в 9 раз.

https://leonardo.osnova.io/f98a1653-771c-2d90-c25b-35d43b6a5ef0/-/scale_crop/700x488/center/-/format/webp/

Google: *«Низкая скорость загрузки — негативный фактор ранжирования в Google».*

Один из принципов ранжирования в поисковой выдаче Google - Mobile First Index. Этот принцип работает следующим образом: если у сайта медленная и не оптимизированная мобильная версия, то и в десктопной выдаче он тоже будет пессимизирован (то есть уйдет вниз).

Google: *«Низкая скорость загрузки является самым раздражающим фактором для мобильных пользователей»*

По опросу, проведенному Google в 2015-ом году, она обошла рекламу, невозпроизводимые видео и редирект на главную страницу сайта. Важно отметить, что это касается исключительно мобильных пользователей — но именно они являются самой активной аудиторией интернета в последние

несколько лет. https://leonardo.osnova.io/3e32ea49-af43-5e03-b119-3da6ce421daf/-/scale_crop/700x378/center/-/format/webp/

По закону на каждое исследование найдется другое исследование, ставящее первое под сомнение. Далее приводится альтернативное мнение относительно скорости загрузки.

Search Engine Land: *«Самые популярные сайты с большим трафиком не соответствуют критериям Google».*

Аналитический сайт Search Engine Land провел собственное исследование, в котором проанализировал 1000 самых популярных сайтов с высоким трафиком. Он протестировал каждый через сервис PageSpeed Insight и получил интересные результаты. Выяснилось, что только 2% мобильных сайтов получили 100 из 100 баллов. На десктопе процент не сильно изменился: всего 4% преодолели планку в 90 баллов. Средний показатель держался в районе 55 баллов, а средняя скорость страницы — 11 секунд. Это подвергает сомнению тезис о том, что 53% пользователей покидают сайт, который загружается более трех секунд

Google: *«Показатель скорости загрузки влияет на выдачу менее чем 1% результатов поиска Google».*

Google подтверждает в своем официальном блоге. Это также подтверждает, что гнаться за слишком высокой скоростью загрузки не имеет смысла.

User Interface Engineering: *«Если пользователи жалуются на скорость загрузки, то на самом деле проблема в другом».*

Именно к такому парадоксальному выводу пришли исследователи с сайта User Interface Engineering. Специалисты обнаружили, что нет корреляции между фактической скоростью загрузки сайта и тем, как ее воспринимают пользователи. Как оказалось, если сайт не удовлетворяет их запросам, если они не могут найти на нем то, что они ищут, они будут воспринимать сеанс как пустую трату времени. А скорость загрузки будет казаться им медленнее, чем она является на самом деле.

WP Faster: *«Скорость загрузки не входит в восьмерку главных показателей эффективности сайта».*

Многие веб-разработчики, которые занимаются оптимизацией сайтов, делают упор не на скорость, а на комфортный пользовательский опыт.

Подтверждает это и инженер по веб-производительности Google Илья Григорик: «Время загрузки — это плохая метрика производительности. Современные сайты должны быть нацелены на то, чтобы доставить качественный прогрессивный рендеринг».

Итого

На основании вышеописанных исследований, можно сделать следующие выводы:

- Инженеры Google выяснили, что пользователь замечает даже пустяковую задержку загрузки — 0,4 секунды;
- Пользователь с большой вероятностью покинет страницу, если та загружается 3+ секунд;
- Мобильные пользователи готовы ждать немного дольше — 6-10 секунд;
- 79% пользователей интернет-магазинов не сделают повторную покупку, если в первый раз сайт загружался долго.

На какие показатели скорости загрузки стоит ориентироваться?

- 1 секунда — отличный, но труднодостижимый результат.
- 2–3 секунды — стандартное время отклика большинства успешных ресурсов.
- 3–5 секунд — маленькая, но для некоторых проектов терпимая скорость;
- 6–10 секунд — столь низкий показатель негативно влияет на продвижение;
- от 11 секунд — плачевный результат, при котором надо срочно ускорять сайт.



Рисунок 1. Показатели скорости загрузки

От чего зависит скорость загрузки?

Скорость загрузки сайта можно условно разделить на серверную и клиентскую часть. Именно эти две составляющие впоследствии попадают под оптимизацию.

Полный цикл загрузки при первом посещении выглядит следующим образом:

- DNS-запрос по названию ресурса.
- Подключение к серверу по IP-адресу.
- Установка защищенного соединения при использовании HTTPS.
- Запрос HTML-страницы по url и ожидание ответа сервера.
- Загрузка HTML.
- Разбор файла на стороне браузера, создание очереди запросов.
- Загрузка и анализ CSS-стилей.
- Загрузка и запуск JS-кода.
- Начало отрисовки страницы, выполнение JavaScript (JS).
- Загрузка web-шрифтов.
- Загрузка картинок и других медиа.
- Окончание отрисовки, работа отложенного JS-кода.

Отдельные этапы могут меняться местами или осуществляются параллельно, однако суть остается неизменной.

Пункты с первого по четвертый относятся к серверной оптимизации, остальные — к клиентской.

Инструменты для мониторинга скорости загрузки

Google PageSpeed Insights

Удобный анализатор от компании Google для быстрой оценки клиентской оптимизации.

Чтобы проверить скорость загрузки перейдите по ссылке <https://developers.google.com/speed/pagespeed/insights/>, и введите в поле адрес интересующего вас сайта, нажмите «Анализировать».

PageSpeed позволяет узнать скорость, а также уровень оптимизации для компьютеров и мобильных устройств. Быстродействие сайта определяется двумя метриками: **FCP** и **DCL**.

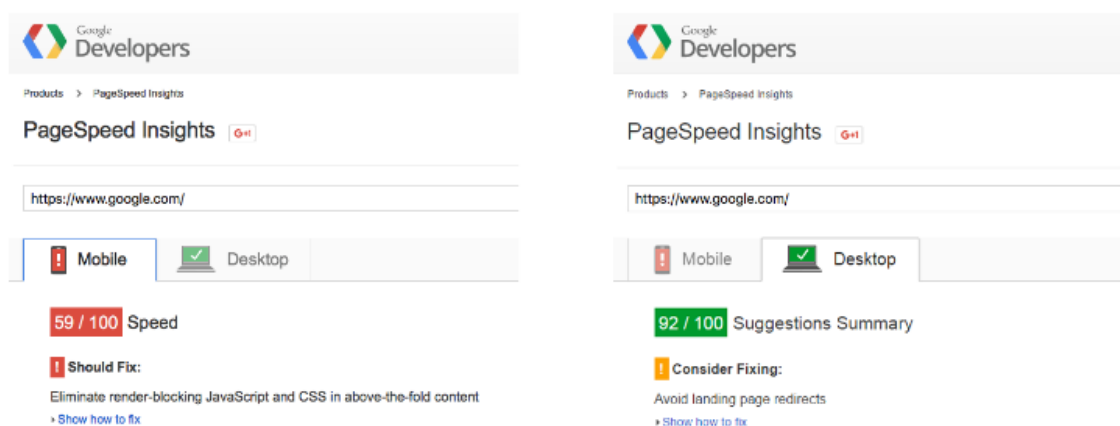


Рисунок 2. Интерфейс сервиса PageSpeed

FCP (First Contentful Paint) — первая отрисовка контента. Низкий показатель указывает на вероятные проблемы с самим соединением либо размерами используемых ресурсов.

DCL (DOMContentLoaded) — параметр означающий полную загрузку HTML и построение DOM-дерева.

Преимущество данного инструмента в том, что он позволяет посмотреть скорость сайта и определить проблемные зоны, а также оценить качество его оптимизации. Google PageSpeed Insights дает наглядные рекомендации по поводу улучшения ситуации.

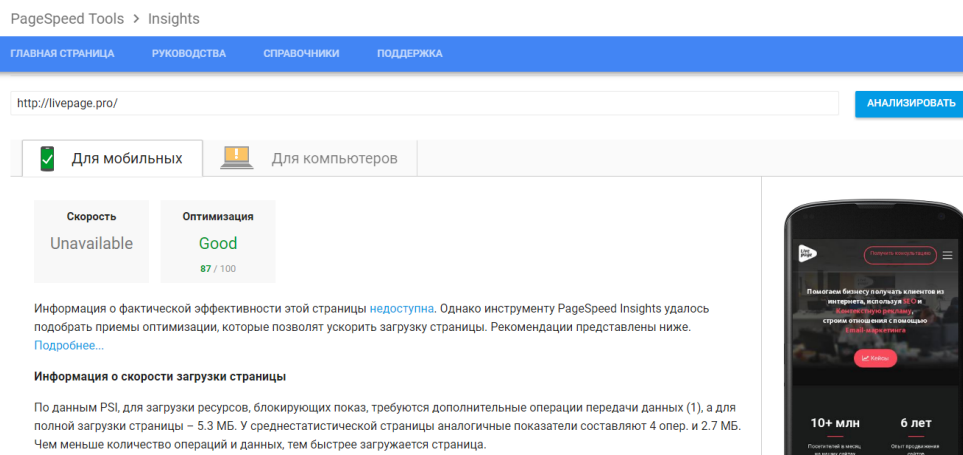


Рисунок 3. PageSpeed - рекомендации по улучшению

Pingdom. Сервис для расчета среднего времени загрузки и количества обращений к серверу. Сервис в деталях отображает важную информацию о быстродействии изображений, стилей, JavaScript, скриптов и других элементов страницы — позволяет сразу обнаружить проблемные участки, тормозящие ваш сайт, а также получите советы по их улучшению. Сервис сохраняет каждый тест, позволяя таким образом следить за историей производительности.

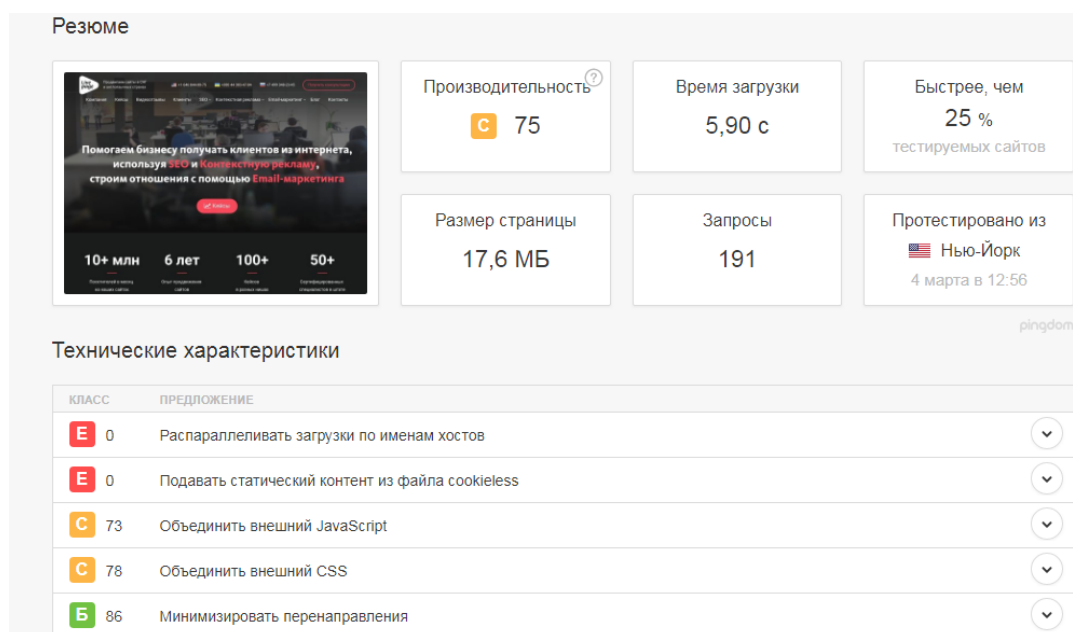


Рисунок 4. Интерфейс сервиса Pingdom

Отчет Google Аналитики

Отчеты Google аналитики позволяют получить полный спектр данных по вашему ресурсу. В том числе отчеты производительности вплоть до каждой страницы и сеанса.

В результате вы получите общую сводку по средней скорости ответа сервера, поиска домена, переадресации и загрузки страницы. Есть отдельная колонка для данных о производительности сайта с учетом браузера и страны. Также Google Аналитика предоставляет информацию о пользовательской скорости загрузки, резвости отдельных страниц по сравнению с общим быстродействием проекта.

Вы сможете лично оценить все преимущества веб-аналитики пройдя авторизацию, открыв необходимое представление. После этого вам нужно перейти в раздел «Отчеты», выбрать «Поведение», а затем «Скорость загрузки сайта».

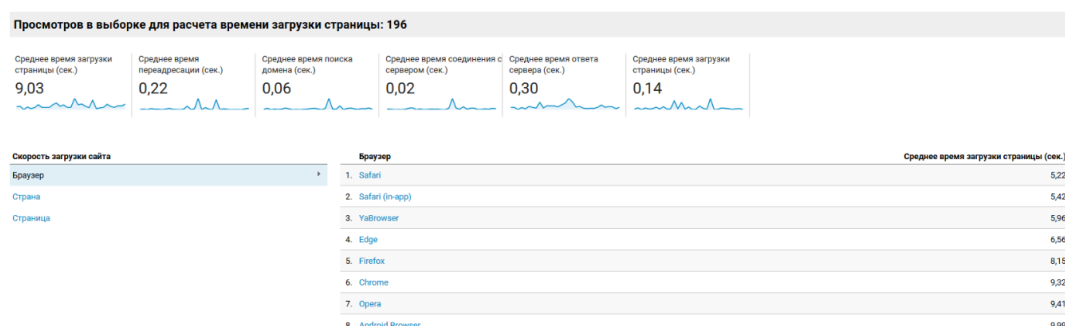


Рисунок 5. Отчет Google Analytics

GTmetrix

Сервис тестирует производительность ресурса, предоставляя много полезной информации. Благодаря сохранению истории этот сервис дает возможность сравнить, как ухудшилась или улучшилась скорость после внесения правок. GTmetrix сообщает об ошибках, подгружая и сортируя по приоритету рекомендации от Google и Yahoo.

Пользоваться им просто: надо перейти на <https://gtmetrix.com/> и ввести в поле адрес сайта, а затем нажать на кнопку Analyze.

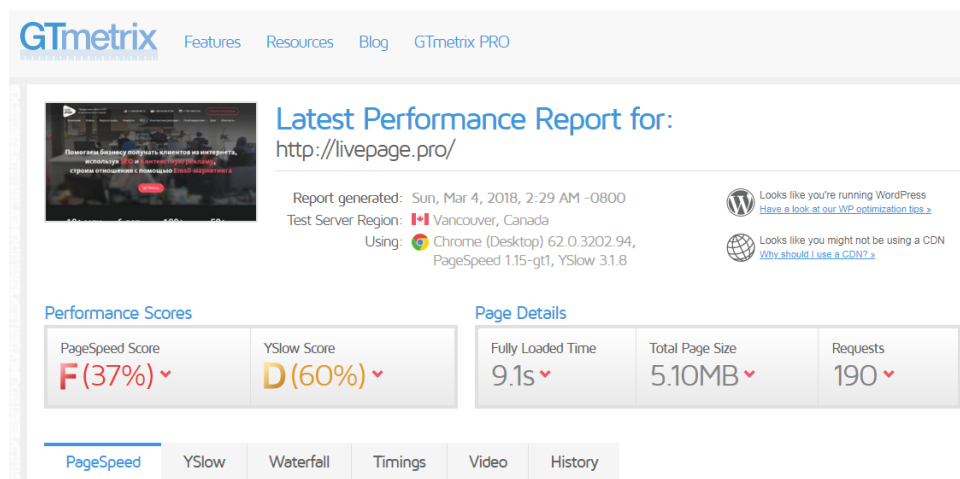


Рисунок 6. Интерфейс сервиса GTmetrix

PR-CY

Сервис позволяет оценить быстродействие проекта на телефонах и компьютерах. К тому же сервис предоставляет подробную информацию, способную помочь в оптимизации.

PR-CY пользуется популярностью среди вебмастеров, благодаря продуманному интерфейсу и точным сведениям.

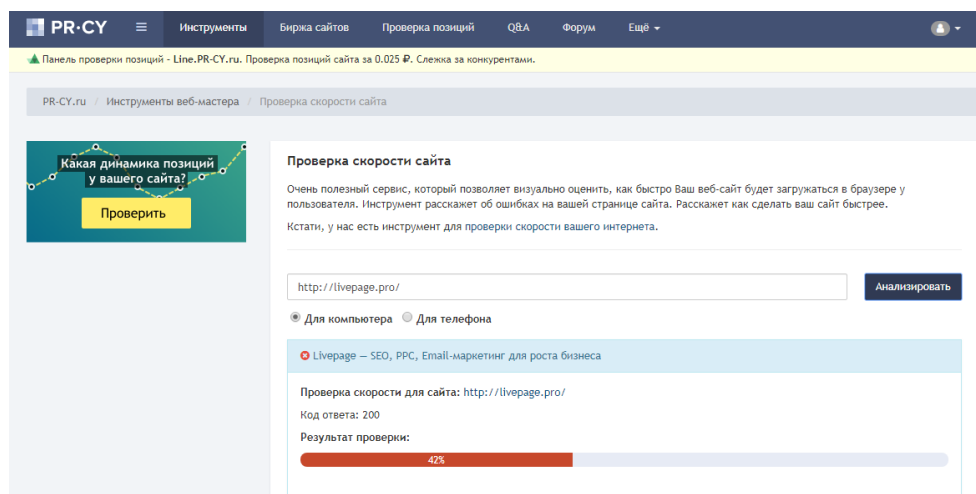


Рисунок 7. Интерфейс сервиса PR-CY

Ускорение скорости загрузки сайта

Серверная часть

Серверная часть легко тестируется и координируется системным администратором. К тому же медленный ответ сервера создает трудности всем пользователям, независимо от типа их устройства и скорости подключения.

Улучшение хостинга

Нехватка ресурсов хостинга — проблема № 1 для небольших блогов. Иногда рентабельнее увеличить скорость дисковой системы и CPU, перейдя на тариф с расширенными возможностями. Если же цена получения дополнительных ресурсов значительно выше стоимости работ по улучшению производительности, стоит воспользоваться следующими методами.

Настройка сервера Nginx

Выделенный веб-сервер Nginx — отличный способ повысить производительность сайта. Особенно хорош он при обработке статического контента (до 600 тысяч запросов в секунду на Linux), более того даже такой показатель можно улучшить с помощью правильной настройки

Удаление неиспользуемых nginx-модулей

Это не только ускорит работу, но и уменьшит уязвимость сервера.

Вам нужно выполнить сборку с такими командами

```
# ./configure --without-http_autoindex_module --without-http_ssi_module  
# make  
# make install
```

В итоге вы получите Nginx с уже отключенными модулями Autoindex и SSI, которые в большинстве случаев бесполезны. Запустив скрипт configure с флагом -help, вы узнаете, какие еще модули можно безболезненно исключить из веб-сервера.

Работа с секцией Server

Для ускорения сайта рекомендуется включить в секцию Server следующие строки:

```
# vi /etc/nginx/nginx.conf  
# Таймаут при чтении тела запроса клиента  
client_body_timeout 10;
```

```
# Таймаут при чтении заголовка запроса клиента
client_header_timeout 10;
```

```
# Таймаут, по истечению которого keep-alive соединение с клиентом не
будет закрыто со стороны сервера
keepalive_timeout 5 5;
```

```
# Таймаут при передаче ответа клиенту
send_timeout 10;
```

Сжатие данных

Обязательная процедура для тех, кто желает повысить скорость загрузки сайта. Так вы сможете уменьшить размер пересылаемого трафика.

Откройте файл конфигурации, который находится по адресу `/etc/nginx/nginx.conf` и добавьте в секцию `http {...}` следующую информацию:

```
gzip on;
gzip_static on;
gzip_comp_level 5;
gzip_min_length 1024;
gzip_proxied any;
gzip_types text/plain application/xml application/x-javascript text/javascript
text/css text/json;
```

Кэширование статических файлов

В том же файле необходимо найти конструкцию `Server` и дописать туда:

```
location ~* ^.+.(jpg|jpeg|gif|png|ico|css|pdf|ppt|txt|bmp|rtf|js)$ {
root /var/www/user/data/www/site.ru;
expires 7d;
```

— (`expires 7d` — количество дней хранения кэша на ПК пользователя).

Настройки сервера Apache

Самая быстрая возможность оптимизировать веб-сервер — воспользоваться модулем `mod-pagespeed` от Google. После его установки, соединитесь через SSH с сервером и введите такие команды:

```
wget https://dl-ssl.google.com/dl/linux/direct/mod-pagespeed-  
stable_current_amd64.deb  
dpkg -i mod-pagespeed-*.deb  
apt-get -f install  
/etc/init.d/apache2 restart
```

Зайдите в директорию `/var/cache/mod_pagespeed` и удостоверьтесь, что модуль работает корректно. В ней должны появиться папки сайта с кэшированием запросов к Apache — все настроено автоматически.

Что делает данный инструмент? Он удаляет лишние пробелы и атрибуты HTML-документов, объединяет несколько JS, CSS и HTML файлов в один, конвертирует картинки под установленный в тегах `` размер, а также способствует оптимизации другими способами.

Gzip — сжатие на стороне сервера

Этот способ архивирования файлов сервером с последующей их распаковкой на стороне пользователя поможет вам существенно увеличить скорость загрузки страницы. Важно, что технологию компрессии поддерживают все без исключения браузеры.

Работа осуществляется на стороне сервера без задействования клиентского оборудования. Единственный минус подобного решения — рост нагрузки на сервер, однако он незначительный.

Для включения gzip-кэширование можно обратиться в техподдержку хостинга или же активировать его самостоятельно — это не требует каких-либо специальных знаний.

Откройте файл `.htaccess` и добавьте в него код:

```
<IfModule mod_deflate.c>  
AddOutputFilterByType DEFLATE text/html  
AddOutputFilterByType DEFLATE application/javascript  
AddOutputFilterByType DEFLATE text/javascript  
AddOutputFilterByType DEFLATE text/css  
<IfModule mod_setenvif.c>  
BrowserMatch ^Mozilla/4 gzip-only-text/html
```

```
BrowserMatch ^Mozilla/4.0[678] no-gzip
BrowserMatch bMSIE !no-gzip !gzip-only-text/html
</IfModule>
</IfModule>
```

CDN для загрузки популярных JavaScript библиотек

Content Delivery Network (CDN) представляет собой своеобразную сеть доставки содержимого — большое количество серверов, расположенных по всему миру. Веб-сервер, предназначенный для отдачи контента клиенту на максимально возможной скорости, выбирается на основе определенных показателей. Чаще всего он отличается наименьшим временем отклика. К тому же браузер кэширует JavaScript, позволяя не загружать библиотеку повторно.

Благодаря сокращению расстояния от контента до посетителя вы сможете улучшить производительность сайта и снизить нагрузку на свой сервер. Еще одно достоинство — CDN способен не только раздавать изначальное содержимое, но и оптимизировать контент: компрессировать тексты и сжимать картинки.

На Content Delivery Network часто переносят все файлы JavaScript, изображения и CSS, оставляя на основном сервере лишь файл HTML.

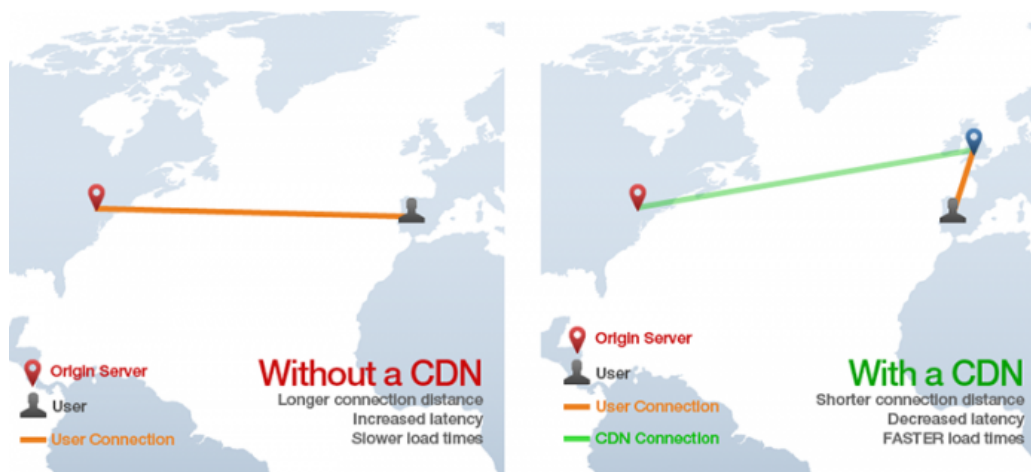


Рисунок 8. Иллюстрация принципа работы CDN

Настройка кэширования на стороне сервера

Суть операции заключается в том, чтобы записать в файл все, что происходит на сервере, для дальнейшей быстрой выдачи статичной

версии. Таким образом, вы сможете увеличить скорость загрузки сайта, а также уменьшить нагрузку на базу данных и сервер.

Создайте на сервере папку с любым допустимым названием (например, cache) и напишите в начале PHP-страницы следующее:

```
<?php
if (file_exists('cache/index_'.$_GET["page"].'_'.$_GET["id"].'.cache'))
{
    readfile('cache/index_'.$_GET["page"].'_'.$_GET["id"].'.cache'); exit();
}
ob_start();
// далее идет любой ваш HTML-PHP код
include("includes/bd.php");
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

При наличии файла index_xx_yy.cache просто считайте его содержимое и выведите на экран (здесь xx — это переменная \$_GET["page"], а yy — значение переменной \$_GET["id"]). Если вышеупомянутого файла нет, включите в папке кэш буфер и запишите туда все процессы, запущенные на странице.

В конце после тэга не забудьте добавить:

```
<?php
$buffer = ob_get_contents();
ob_end_flush();
$fp = fopen('cache/index_'.$_GET["page"].'_'.$_GET["id"].'.cache', 'w');
fwrite($fp, $buffer);
fclose($fp);
```

Запишите в \$buffer его содержимое, затем остановите буферизацию и очистите контент. Следующее действие предполагает открытие файла index_xx_yy.cache (если его на сервере нет, то надо создать) и запись в файл \$fp значения \$buffer.

Оптимизация базы данных

По мере роста сайта объем базы данных будет увеличиваться, что приведет к ее торможению. Соответственно, необходимо регулярно избавляться от мусора в виде спама, ревизий, ненужных таблиц.

Это можно сделать вручную через phpMyAdmin хостинга (пункт SQL). Не хотите попусту тратить время? Воспользуйтесь специальными плагинами или настройте автоматическую оптимизацию. Сделать это несложно.

Установите на ПК программу Notepad++ и подключитесь через FTP к хостингу. Найдите в папке public_html файл wp-config.php и перетащите на рабочий стол. После этого откройте его с помощью Notepad++ и найдите строчку с названием базы данных. Под ней надо добавить следующее:

```
define('WP_POST_REVISIONS', false);  
define('EMPTY_TRASH_DAYS', 1);
```

Первая команда отключит ревизии постов, а вторая станет через день очищать корзину.

Обновите файл и перетащите его мышкой обратно на хостинг, заменив старую версию новой.

Оптимизация TCP, TLS, HTTP/2

В этом разделе объединены тонкие сетевые настройки, способствующие ускорению сервера. Они не меняют показатели разительно, зато их оптимизация не требует финансовых вложений.

TCP — популярный сетевой протокол, функционирующий на основе «окна перезагрузки». Чем выше габариты окна (количества пакетов для передачи), тем больше пропускная способность. Ваша задача — правильно рассчитать и настроить размер буфера TCP. Большинство разработчиков используют для этого параметры задержки и полосы пропускания, умноженные на два. Чтобы изменить размер буфера, используйте вызов `Setsockopt` в C или методы `setReceiveBufferSize` и `setSendBufferSize` в Java. Такая корректировка ориентирована на ускорение масштабных порталов и серверов. Главное — регулярно обновлять систему.

Что касается тюнинга TLS, он сократит время ответа и повысит уровень безопасности соединений. Найдите конфигурационный файл на Nginx и проведите следующие изменения:

- включите разделяемый кэш сессий и отключите встроенный аналог;
- уравнийте количество рабочих операций с числом процессоров;
- разрешите keep-alive соединения;
- увеличьте время жизни сессии.

Создаем отдельный Server для перенаправления с HTTP на HTTPS

```
server {  
    server_name example.com www.example.com; # можно указать любые  
    домены и поддомены  
    listen 1.2.3.4:80; # где 1.2.3.4 - айпи вашего сервера  
    rewrite ^(.*) https://$host$1 permanent; # редирект HTTP/1.1 301 Moved  
    Permanently с http на https  
}
```

Прописываем основной сервер с HTTPS

```
server {  
    server_name example.com www.example.com; # копируем из верхнего  
    сервера  
    listen 1.2.3.4:443 ssl http2; # вместо 1.2.3.4 вставляем IP своего сервера.  
    http2 включаем поддержку протокола http/2
```

```
ssl_certificate /etc/letsencrypt/live/example.com/fullchain.pem; # сертификат  
ssl_certificate_key /etc/letsencrypt/live/example.com/privkey.pem; # ключ
```

Для кэширования запросов

```
keepalive_timeout 70; # 70 секунд держим соединение открытым  
keepalive_requests 150; # 150 запросов максимум на одно соединение,  
после чего закрываем  
ssl_session_cache shared:SSL:10m; # разделяемый между всеми  
процессами кэш сессий на 10 байт с названием SSL. 1 Мб вмещает около  
4000 сессий  
ssl_session_timeout 10m; # 10 минут - максимальное время жизни сессии
```

Для улучшения безопасности соединения

ssl_prefer_server_ciphers on; # указываем, чтобы при использовании протоколов SSLv3 и TLS серверные шифры обладали большим приоритетом, чем клиентские

```
ssl_ciphers ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-DSS-AES128-GCM-SHA256:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-AES128-SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-SHA:DHE-RSA-AES256-SHA:ECDHE-RSA-DES-CBC3-SHA:ECDHE-ECDSA-DES-CBC3-SHA:AES128-GCM-SHA256:AES256-GCM-SHA384:AES128-SHA256:AES256-SHA256:AES128-SHA:AES256-SHA:AES:CAMELLIA:DES-CBC3-SHA:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!MD5:!PSK:!aECDH:!EDH-DSS-DES-CBC3-SHA:!EDH-RSA-DES-CBC3-SHA:!KRB5-DES-CBC3-SHA; # типы шифров
ssl_protocols TLSv1 TLSv1.1 TLSv1.2; # разрешённые типы протоколов
}
```

Теперь немного о HTTP/2. В этот достаточно молодой протокол заложены механизмы быстрогодействия, рассчитанные на уменьшение влияния сетевых задержек. Чтобы увеличить скорость загрузки сайта, SEO-профи рекомендуют отказаться от встраивания файлов в HTML-код, доменного шардинга, спрайтов и конкатенации файлов.

Количество HTTP-запросов

Время открытия страницы на 80% зависит от загрузки Flash, CSS, скриптов и картинок. Уменьшение числа упомянутых элементов приводит к ускорению веб-проекта. Но как сократить количество запросов без изменения внешнего вида?

Для этого достаточно применить встроенные в страницу Inline-изображения и CSS-спрайты — комбинированные картинки, которые состоят из небольших фрагментов.

Если вы используете на странице несколько JS и CSS-элементов, их можно объединить в один файл — это действенный и простой прием.

Уменьшение редиректов

Создание перенаправлений — стандартная практика продвижения, когда надо сообщить серверу, что адрес страницы был изменен. 301-редиректы помогают избежать 404-ошибок, но при этом они замедляют работу сайта. Особенно явной становится проблема для пользователей мобильных устройств, которым приходится совершать дополнительные переходы. Google советует по возможности уменьшить количество перенаправлений.

Чтобы быстро найти все редиректы на своем ресурсе, воспользуйтесь специальной программой. После установки на компьютер утилита Screaming Frog за считанные минуты определит проблемные места.

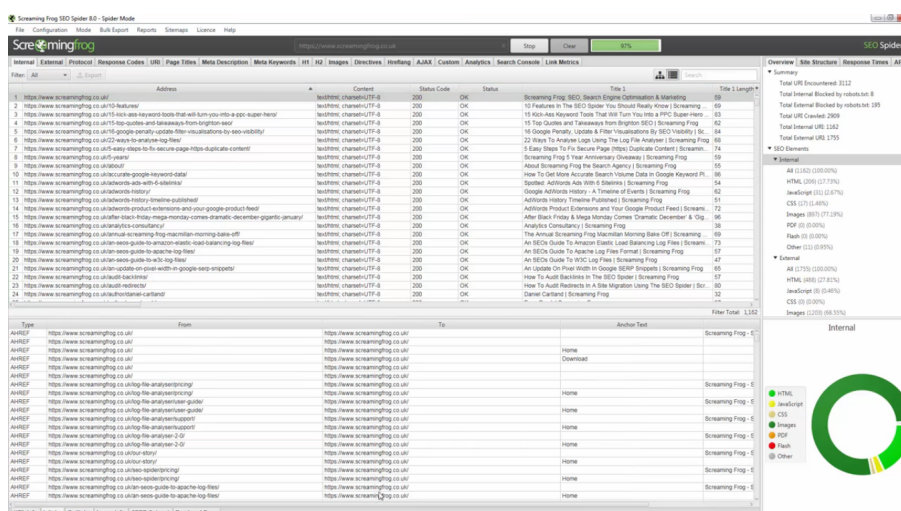


Рисунок 9. Утилита Screaming Frog

Клиентская часть

Этот сегмент менее удобен в управлении, так как направлен на процессы, которые осуществляются в браузере пользователя. Тем не менее их оптимизация порадует вас хорошим результатом, позволив заметно повысить скорость загрузки сайта.

Настройка асинхронной загрузки

Современные страницы перегружены тяжелыми файлами JavaScript, которые замедляют дальнейшее отображение страницы. Соответственно, в местах с этими строками загрузка тормозится, ведь все HTML-элементы открываются по очереди. Однако с выходом HTML5 вебмастерам стала доступна асинхронная загрузка, во время которой скрипты запускаются в последнюю очередь.

Ускорить сайт можно при помощи специального скрипта Google ExtSrcJs. Надо лишь заменить:

`<script src="...">` на: `<script extsrc="...">`

и подключить файл скрипта extsrc.js.

Удаление лишних плагинов

Большинство генераторов и визуальных редакторов легко создадут необходимый вам код, правда, он будет «мусорным» – лишние строчки кода приведут не только к торможению ресурса, но и к возникновению ошибок, мешающих загрузке. Это надо исправлять. Кроме того, опытные специалисты рекомендуют всегда удалять ненужные расширения, плагины и темы.

Оптимизация размера страницы

Уменьшив размер страницы и количество «тяжелого» контента, вы ускорите работу сайта и сэкономите деньги на оплате услуг хостинга, если платите за трафик.

Для оптимизации попросите провайдера включить опцию сжатия HTTP. Это как минимум вдвое уменьшит размер графики и других элементов страницы.

Физическое сжатие графики и смена расширения файлов

Следуя простым советам, вы сможете увеличить скорость загрузки страницы без потери ее визуальной привлекательности

- Удаляйте ненужные картинки. Где предусмотрена такая возможность, используйте текст вместо тяжелой графики.
- Отдавайте предпочтение формату JPEG, который сжимается без потери качества. Даже снизив качество на 25-40%, вы получите на выходе достойную иллюстрацию, практически не отличимую от исходника. При этом вес картинки будет минимален.
- Не злоупотребляйте PNG для создания эффекта прозрачности. Для сжатия этого и любого другого формата подойдет не только Photoshop, но и OptiPNG, Compresspng и PNGout.
- Очищайте элементы от цифрового мусора (рабочих версий, ненужных палитр, комментариев дизайнера). Сделать это можно при помощи сервиса Smush.it.

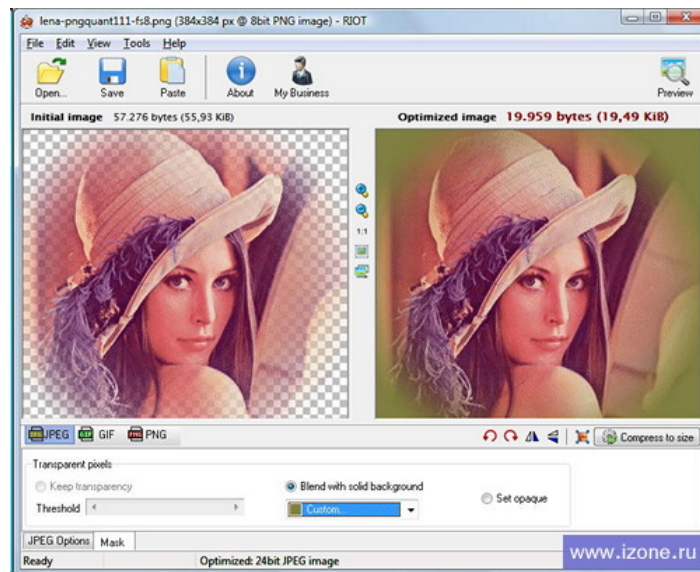


Рисунок 10. Сервис Smush.it

Для сжатия графических файлов онлайн используйте [tinypng](#).

Отказ от масштабирования картинок

Не стоит менять размер изображений с помощью CSS, `width` и `height`. Лучше предварительно внести корректировки, воспользовавшись графическим редактором.

Вместо одной масштабируемой иллюстрации, которая занимает много места и замедляет загрузку, стоит использовать несколько картинок, адаптированных под разные устройства.

Чтобы браузер каждый раз грузил нужный вариант, установите плагин **Imager** (для WordPress). Это работает так:

```
<div style="width: 240px">
  <div class="delayed-image-load" data-src="http://placeholder.it/{width}" data-
alt="alternative text"> </div>
</div>
<script>
  new Imager({ availableWidths: [200, 260, 320, 600] });
</script>
```

Оптимизация JavaScript и CSS

Специальные методы упрощения кода уменьшают размеры элементов, сокращая названия функций и удаляя мусор. Можно дополнительно улучшить CSS и JavaScript за счет изменения названий переменных (в случае обновления HTML и правильной работы селекторов).

Упростить код несложно — имеется много сервисов: JSCompress или Refresh-SF. Подойдет для этого и Google PageSpeed Insights. После выполнения на нем анализа страницы, нажмите на «Сократить HTML», а затем на «Посмотреть содержание».

Оптимизация веб-шрифтов

Подключенные веб-шрифты — стандарт оформления современных сайтов. Но, к сожалению, они негативно сказываются на производительности сайта. Проблема усугубляется при расположении указателей на файлы шрифтов в CSS и применении публичных сервисов, требующих дополнительные подключения.

Начните пользоваться новыми форматами WOFF и WOFF2, при этом исключите все наборы символов, кроме кириллицы и латиницы.

Улучшить ситуацию можно при помощи спецификации `link rel=«preload»` и CSS-свойства `font-display`. Первый показатель мгновенно указывает браузеру о необходимости загрузки файла шрифта, второй позволяет управлять поведением браузера в случае задержки.

Настройка кэширования на стороне браузера

Использование кэша предполагает то, что файлы сайта будут загружаться с сервера лишь один раз. В дальнейшем они сохраняются на компьютер и «подтягиваются» при загрузке с ПК пользователя.

Для подключения кэша браузера необходимо создать файл `.htaccess` и, воспользовавшись модулем `expires`, прописать в нем выбранные директивы:

- `Expires` — отвечает за актуальность кэша (можно установить срок от недели до года).

- Cache-Control: max-age — аналог предыдущего заголовка, выполняющий те же функции.
- Last-Modified и ETag — устанавливают в кэше уникальный id для каждого url. В Nginx для статичных файлов они включены по умолчанию.

Заключение

В совокупности все меры, описанные выше, позволяют получить наиболее оптимальную скорость загрузки сайта, тем самым напрямую влияя на конверсию и количество пользователей вашего ресурса. Все описанные выше меры были применены на сайте компании Geometrya.com.ua - интернет-магазин зеркал. Показатели роста конверсии не заставили себя долго ждать. Взяв статистику за период - 7 дней, при том же количестве трафика, конверсия в покупку сайта выросла с 7% до 15%. А число отказов при первом посещении снизилось на 30%. Что в результате сказалось на количестве покупок и прибыли ресурса.

Список литературы

1. Исследование влияния скорости загрузки Google. - URL: <https://www.thinkwithgoogle.com/consumer-insights/consumer-mobile-brand-content-interaction>
2. Исследование влияния скорости загрузки SOASTA. - URL: <https://www.soasta.com/wp-content/uploads/2017/04/State-of-Online-Retail-Performance-Spring-2017.pdf>
3. Исследование - “Соотношение показателя отказа к скорости загрузки”. - URL: <https://www.thinkwithgoogle.com/marketing-resources/data-measurement/mobile-page-speed-new-industry-benchmarks/>
4. Влияние скорости загрузки на ранжирование. - URL: <https://www.hobo-web.co.uk/your-website-design-should-load-in-4-seconds/>
5. Исследование скорости загрузки как фактора раздражения . - URL: <https://webmasters.googleblog.com/2015/04/mobilemadness-campaign-to-help-you-go.html>

6. Исследование сайтов с низкой скоростью загрузки. - URL: <https://searchengineland.com/mobile-page-speed-important-not-yet-seo-270474>
7. Влияние показателя скорости загрузки на поисковую выдачу. - URL: <https://ain.ua/2017/04/13/tak-li-vazhna-skorost-zagruzki-sajta-ili-o-polze-google-pagespeed-insights/>
8. Влияние показателя скорости загрузки на поисковую выдачу. - URL: https://articles.uie.com/download_time/
9. VC.RU “Правда о том, на сколько важна скорость загрузки сайта”. - URL: <https://vc.ru/flood/34484-pravda-o-tom-naskolko-vazhna-skorost-zagruzki-sayta>
10. Livepage “Как увеличить скорость загрузки сайта?”. - URL: <https://livepage.pro/knowledge-base/kak-uvelichit-skorost-zagruzki-sajta.html>