

УДК 004

УСКОРЕНИЕ ЗАГРУЗКИ СТРАНИЦ САЙТА

Солобуев А.И., студент гр. ПИБ-171, IV курс,
Девянин Е.С., студент гр. ПИБ-171, IV курс,
Болдырев Д.А., студент гр. ПИБ-171, IV курс,
Научный руководитель – С.А. Лимаков, преподаватель.
Кузбасский государственный технический университет
имени Т.Ф. Горбачёва,
г. Кемерово

От скорости загрузки сайта зависит, будет ли посетитель ждать загрузки вашего сайта, а также оптимизация SEO. Фактически, многие поисковые системы сегодня при ранжировании сайтов начинают обращать больше внимания на скорость загрузки страниц.

Следовательно, чем быстрее загружается ваш сайт, тем больше посетителей вы можете заполучить из поисковых систем и, следовательно, больше клиентов, больше денег, что даёт вам возможность развиваться быстрее и масштабнее.

Ни для кого не является секретом, что на скорость загрузки страницы влияет множество факторов. Поэтому в данной статье собрано несколько способов увеличения скорости загрузки своей веб-страницы и сайта:

- Уменьшение количества HTTP-запросов
- Перемещение CSS файлов в начало страницы
- Перемещение JavaScript в конец страницы
- Минимизация CSS и JavaScript
- Использование кэша браузера
- Использование CDN
- Оптимизация изображений
- Использование Gzip- сжатия

Уменьшение количества HTTP-запросов

Примерно 80% от загрузки страницы приходится на загрузку компонентов страницы: скриптов, изображений, CSS файлов. Спецификация[1] HTTP / 1.1 рекомендует, чтобы браузеры загружали не более двух компонентов на веб-страницу параллельно с одного хоста. Уменьшая количество этих компонентов, мы уменьшаем количество HTTP-запросов к серверу и, как следствие, увеличиваем скорость загрузки страницы.

Есть несколько способов уменьшить количество запросов к серверу, не прибегая к понижению качества внешнего вида страницы.

1. Использование CSS-спрайтов.

Спрайт CSS – это составное изображение, которое содержит несколько небольших изображений, которые обрезаются в нужный момент для нужного элемента страницы с помощью свойств background-image и background-position.

2. Использование встроенных изображений.

Встроенные изображения используют схему URL data: для встраивания изображения в саму страницу. Однако это увеличит размер HTML-документа. Добавление встроенных изображений в таблицы стилей приведет к меньшему количеству запросов на сервер, при неизменном размере HTML.

3. Объединение нескольких файлов в один.

Если с вашей страницей связано несколько файлов css или js, вы можете объединить их в один. Это очень простой, но эффективный способ уменьшить количество HTTP-запросов к серверу.

Перемещение CSS файлов в начало страницы

Размещая ссылку на файлы CSS в шапке страницы, мы получаем постепенное отображение страницы, то есть страница будет загружаться постепенно: сначала заголовок, затем логотип вверху, навигация и т. д. - а это, в свою очередь, служит отличным индикатором для пользователей о статусе загрузки страниц, а также улучшает общее впечатление о работе сайта.

Размещение файлов CSS внизу страницы предотвращает постепенное отображение во многих браузерах. Это связано с тем, что отображать элементы несколько раз, нерезонно, так как некоторые могут изменить свой стиль после загрузки страницы. Поэтому всегда стоит включать все свои файлы CSS вверху страницы в разделе HEAD.

Перемещение JavaScript в конец страницы

Перемещая файлы JavaScript в нижнюю часть страницы, мы позволяем браузеру сначала загрузить страницу содержимого, а затем начать загрузку файлов JavaScript. Если ваш сайт обновлен и содержит все возможные интерактивные «виджеты», то таких файлов JavaScript может быть несколько, и они весят несколько сотен килобайт, поэтому перед загрузкой страницы пользователю приходится ждать, пока все ваши файлы JavaScript не будут загружены.

Кроме того, внешние файлы .js блокируют параллельную загрузку. Спецификация HTTP / 1.1 рекомендует, чтобы браузеры загружали не более двух компонентов на веб-страницу параллельно с одного хоста. Таким

образом, если изображения для вашего сайта находятся на разных хостах, вы получите более двух параллельных загрузок. И когда скрипт загружается, браузер не будет запускать другие загрузки, даже с других хостов.

Минимизация CSS и JavaScript

Минимизация файла – это удаление из кода всех «лишних» символов для уменьшения размера файла и ускорения его загрузки. В свернутом файле удалены все комментарии и незначительные интервалы, разрывы строк, разделы. Здесь все просто. Чем меньше размер файла, тем меньше времени потребуется браузеру для его загрузки.

Использование кэша браузера

Кэширование становится все более важным для современных веб-сайтов, которые в значительной степени полагаются на JavaScript и CSS. Дело в том, что, когда посетитель посещает ваш сайт впервые, браузер загружает все файлы JavaScript и CSS, загружает всю графику и флэш, но, правильно установив заголовок HTTP Expires[2], вы сделаете компоненты страницы кэшированными. Таким образом, когда посетитель снова посещает ваш сайт или переходит на следующую страницу вашего сайта, некоторые необходимые файлы теперь будут в кэше их браузера, и браузеру не нужно будет загружать их снова. Отсюда получаем прирост в скорости загрузки сайта.

Использование CDN

Если на вашем сайте используется популярный фреймворк JavaScript, например jQuery, то для его подключения лучше использовать CDN [3].

CDN (Content Delivery Network) – это набор веб-серверов, которые географически рассредоточены, чтобы максимизировать скорость доставки контента клиенту. Сервер, который будет напрямую обслуживать контент пользователя, выбирается по определенным показателям. Например [3], выбирается сервер с наименьшим количеством промежуточных звеньев или наименьшим временем отклика. Кроме того, браузер кэширует файлы JavaScript, и, если вы посетили сайты, использующие этот метод, эта библиотека уже кэшируется в вашем браузере и не будет перезагружаться ее.

Один из таких CDN – это библиотеки Google. Это CDN для популярных библиотек JavaScript с открытым исходным кодом. Загрузка популярных фреймов JavaScript из библиотек Google может увеличить скорость загрузки страницы и уменьшить трафик на ваш сервер.

Оптимизация изображений

Вам необходимо определить подходящий формат для ваших изображений. Выбор неправильного формата изображения может значительно увеличить размер файла.

- GIF – файлы идеально подходят для разноцветных изображений, например логотипов.
- JPEG – отлично подходит для детализированных изображений с множеством цветов, таких как фотографии.
- PNG – ваш выбор, когда вам нужно высококачественное изображение с прозрачностью.

Есть два способа оптимизировать изображение:

1. использование специализированных программ;
2. использование онлайн-сервисов в Интернете для сжатия изображений.

В первом случае вам потребуются определенные знания для работы с определенной программой, но пользоваться онлайн-сервисами может любой желающий. Все, что вам нужно сделать, это загрузить необходимые изображения, а сервис сам их оптимизирует и предоставит вам ссылку для загрузки уже сжатых файлов. Однако для большого объема изображений желательно использовать первый способ, всё из-за большей скорости оптимизации.

Главное не изменять размер изображения, используя атрибуты ширины и высоты тега или CSS. Это сильно сказывается на скорости загрузки страницы. Если у вас есть изображение размером 600x400 пикселей, и вы хотите вставить на сайт изображение размером 150x100 пикселей, то лучше изменить размер исходного изображения с помощью графического редактора. Так как при загрузке изображения 600x400 нужно будет загрузить сначала его, а затем отмасштабировать.

Чем меньше вес изображения, тем меньше времени потребуется на его загрузку.

Использование Gzip- сжатия

Исследования показывают, что сжатие Gzip в текстовый файл «на ходу» может сократить время, необходимое для загрузки файла в браузер в 95-98% случаев [4]. Если вы храните резервные копии файлов на сервере (в памяти прокси-сервера или просто на диске), то соединение обычно может быть освобождено в 3-4 раза быстрее.

Начиная с версии протокола HTTP / 1.1, веб-клиенты указывают, какие типы сжатия они поддерживают, задавая заголовок Accept-Encoding в HTTP-запросе [1].

Если веб-сервер видит такой заголовок в запросе, он может сжать ответ, используя один из методов, перечисленных клиентом. При выдаче

ответа с использованием заголовка Content-Encoding сервер уведомляет клиента о том, какой метод был использован для сжатия ответа.

Объем передаваемых таким образом данных примерно в 5 раз меньше исходных, что значительно ускоряет их доставку. Однако есть один недостаток: увеличивается нагрузка на веб-сервер. Но проблему с сервером всегда можно решить. Так что не будем обращать на него внимания.

В этой статье были перечислены самые распространенные методы оптимизации со стороны заказчика для увеличения скорости загрузки веб-страницы.

Список литературы:

1. Hypertext Transfer Protocol (HTTP/1.1) – Режим доступа: <https://tools.ietf.org/html/rfc7230>
2. HTTP-заголовок Expires – Режим доступа: <https://siteactiv.ru/terminy/http-expire/>
3. Как увеличить скорость загрузки страницы – Режим доступа: <https://medium.com/webbdev/speed-31716013b5e2>
4. GNU zip – Режим доступа: <https://wiki.dieg.info/gzip>