Анатолий Будько @atomlib Пользователь

> 1 Профиль

Geektimes

56 Публикации

карма

313 Комментарии

рейтинг

Фрилансим

7

Избранное

80

Подписчики

Спецпроекты: Собери стартап

Кризис ожирения сайтов

Примечание переводчика: визит под кат этого поста означает большие объёмы потреблённого трафика. И это иронично, если учесть поднятую тему. Но всё сразу встаёт на свои места, если помнить, что в оригинале это было выступление в прошлом ноябре в Сиднее длиной почти в час. Чтение поста занимает куда меньше времени. В форме видеозаписи (1280×720) речь занимает два гигабайта. Пост же занимает всего лишь 12 МиБ. Рекомендуется просмотр на широких мониторах.

THE WEBSITE OBESITY MACIEJ CEGLOWSKI

Перед началом тирады я хотел бы обратить внимание на то, что прекрасные сайты бывают любых форм и размеров. И я здесь не собираюсь кого-то стыдить за количество использованных бит, объём использованных ресурсов и так далее. Я люблю большие сочные галереи изображений, мне нравятся огромные эксперименты на JavaScript, я смотрю онлайн-видео в высоком разрешении, как и все вы. Я считаю, что подобное замечательно.

Выступление совсем не об этом. Я хотел бы поговорить об этом общественном кризисе здоровья, этом ожирении сайтов. Отличные дизайнеры, которые задумываются о вебе как я или даже больше, почему-то делают страницы, которые становятся больше. Речь пойдёт о текстовых в своей основе сайтах, которые по каким-то непостижимым причинам с каждым годом становятся всё больше и больше.

В этом выступлении я дам примеры, поскольку иначе получилось бы слишком абстрактно. И не ради того, чтобы пристыдить кого-то, кроме нескольких больших компаний, которые я покритикую за то, что они ломают веб. Но я понимаю отдельных людей, которые работают над этими проектами под давлением и принуждением.

## Кризис

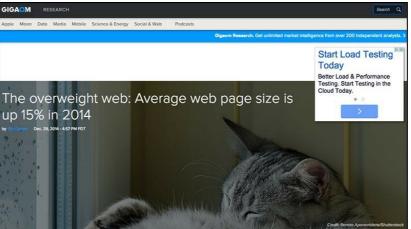
Что такое кризис ожирения сайтов? Вот статья 2012 года с сайта Gigaom «Растущая эпидемия раздутия страниц». Она предупреждает, что средний размер страницы вышел за мегабайт. Сама статья имеет размер 1,8 мегабайта.

Вот почти такая же статья «Веб с избытком веса» с того же сайта от 2014 года, два года спустя. Статья предупреждает, что средний размер страницы приближается к 2 мегабайтам.

Сама статья имеет размер 3 мегабайта.

Если такой тренд сохранится, то статьи с предупреждениями о размерах страниц выйдут за 5 мегабайтов к 2020 году.

The growing epidemic of page bloat 1042 KB 2012



нормой Т. То, что сегодня считается жутко раздутым сайтом, завтра становится типичной страницей, а в следующем году это уже элегантный изящный дизайн.

Проблема с выбором конкретного порога размера состоит в том, что подобное заставляет аномалии становиться

Я хотел бы предложить закрепить размер к чему-то более постоянному.

Повторю предложение, которое сделал в Twitter: ваш сайт, если он текстовый (типа статьи), не должен превышать в файловом размере главные работы русской литературы.

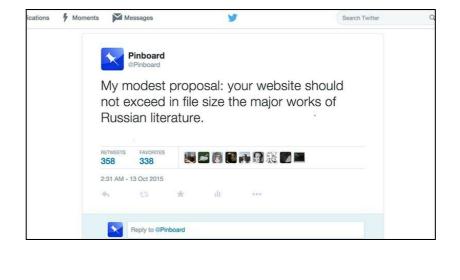
Это очень щедрое мерило. Я бы мог выбрать французскую, где романы тонкие. Но я выбрал русскую с её репутацией тяжеловесности большие кирпичи цельной прозы.

К примеру, в «Обломове»<sup>2</sup> Гончарова заглавный герой первую сотню страниц встаёт из кровати.

Если вы откроете этот твит в браузере, то вы увидите страницу на 900 КБ.

Это почти на 100 КБ больше, чем полный текст

«Мастера и Маргариты», наполненного мистикой и комедией романа Булгакова о Дьяволе, который наведывается в Москву со своей свитой (включая огромного кота!) во время Большого террора 1937 года. Роман прерывается параллельным повествованием о жизни Понтия Пилата, Иисуса Христа и верном, но ненадёжном апостоле Матфее.





## На один твит.

Это отличная книга, и вам стоит её прочесть. Это отличный твит, и его вы уже прочитали. Всё, что я хочу сказать: они не должны быть одного размера.

Или рассмотрите эту статью на 400 слов на Medium. Там есть такое предложение:

«Команды, которые не не понимают, для кого и для чего они создают, склонны к написанию раздутых продуктов.»

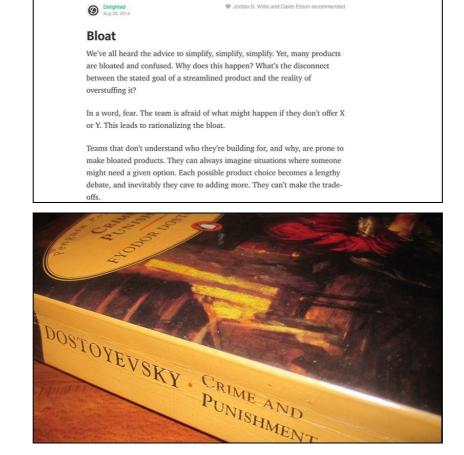
Команда Medium сделала так, что эта крупица гениальности требует 1,2 мегабайта.

Это длиннее, чем размер «Преступления и

наказания», психологического триллера Достоевского о доведённом до нищеты студенте, который забивает голову мыслями о Наполеоне и заставляет себя убить пожилую ростовщицу.

Пытаемый виной, настолько перепутанный своим

преступлением, что он даже забыл забрать деньги, Раскольников ввязывается в игру в кошки-мышки с коварным следователем, а потом находит искупление в неправдоподобной любви святой проститутки.



Sign in / Sign up

Всё это Достоевский написал в свете свечи, от руки, пером, чёрт возьми. Вот недавняя статья «(Не такая уж и) краткая

история раздутия в вебе».

Она повторяет обычные причины, по которым

раздутие плохо, но также включает эту фразу: «тяжёлые страницы часто бывают медленными, а медленные страницы означают несчастливых пользователей».

Эта цитата могла напомнить знаменитую строчку, которая открывает роман «Анна Каренина»:

каждая несчастливая семья несчастлива посвоему.»

«Все счастливые семьи похожи друг на друга,





Самое читаемое

Сейчас

Реклама

Неделя

Релиз ownCloud 9.0 — opensource альтернативы Dropbox и другим

Месяц

облачным хранилищам 🔲 11 Как Облако@mail.ru спасло все\* мои

файлы и что из этого вышло 📉 54 Как можно упростить себе жизнь с

помощью Telegram-бота 5

Кризис ожирения сайтов 70

Мы ищем тебя, чтобы помочь тебе заработать 5000\$ этим летом 🔲 0

Пример разработки игры на основе данных Google Analytics 1

Гигабитный Wi-Fi в России \_\_\_\_ 1

SQL Server 2016 RC0 5

Центр сертификации Let's Encrypt выдал миллион бесплатных сертификатов 6

Игра гомоку (крестики-нолики, 5 в ряд) 📉 15

## Вопросы по теме

словарю? 📭 1

Есть ли проверка баннеров на соответствие требованию сервисов:

Каким размером делать макеты? 🚅 3

Как реализуется проверка слов по

Имеет ли смысл использовать https во время разработки? 📮 2

Возможна ли переквалификация в разработчики после 30 без профильного высшего образования? Но не такая уж и краткая история раздутия в вебе куда длиннее «Анны Карениной». Сравнение неверно, роман длится всего лишь 1,8 МБ.

Реальная книга, с которой нужно сравнивать — это «Война и мир», примерно 3 МБ. Это размышление Толстого на тему, определяют ли отдельные люди великие события истории, или мы всё же плывём по непреодолимому течению исторической неизбежности.

Вот статья с Yorkshire Evening Post, типичного представителя тысяч местечковых новостных сайтов. Она никак не исследует отношение между временем и личной волей:

«Руководство больницы в Лидсе извиняется за перемешанные карри и крамбл»<sup>3</sup>.

Эта мучительная история о смешивании двух блюд на тарелке в больнице почти могла выйти из под пера Марселя Пруста, для которого размачивание кусочка пирожного в чае стало отправной точкой для расширяющейся спирали ярких воспоминаний, 9 томами и 3 мегабайтами рукописной прозы позже увенчавшихся в понимании того, что время и память — это лишь обман чувств.

Одни лишь скрипты в статье про карри и крамбл длиннее, чем «В поисках утраченного времени». Размер всей статьи — почти 40 МБ.





У меня есть предложение, чтобы браузеры как-то показывали этот вес. Может, они бы падали, а потом их нужно было бы поднимать из-за веса страниц. Так получилось бы какое-то понимание огромности. Особенно хорошо такое работало бы на мобильных устройствах: стороны могли бы взрываться.

Я могу продолжить в этом духе. И так и сделаю, поскольку мне нравится!

Вот статья о «Лучших методах улучшения онлайнпроизводительности». Одним из простых методов является не делать страницы размером в 3,1 МБ.

Статья упоминает, что Google смогла увеличить вовлечённость пользователей в Google Maps путём уменьшения страницы с 100 КБ до 80 КБ. А вы помните времена, когда сервис Google Maps, самое

продвинутое веб-приложение в ту эпоху, был в тридцать пять раз меньше, чем современная новостная статья?

Ожирение в вебе может принимать странные формы. К примеру, Тим Кадлек отлично пишет и выступает на тему производительности. Его личный сайт — это маленькая модель бережливости. Тим много знает о том, как уменьшить раздутие.

Но слайды его недавней речи о производительности доступны только на 9-

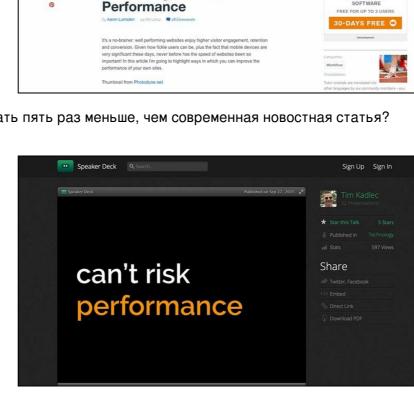
мегабайтовой странице или в 14-мегабайтовом PDF.

Давайте я закончу прекрасной статьёй TechTimes, которая предупреждает, что Google собирается помечать большие страницы специальным «медленным» значком в выдаче мобильного поиска.

Статья как-то умудряется достичь размера в 18 мегабайтов, включая (в том просмотре, который я замерил) 3-мегабайтный видеоролик с K-Y Jelly, «интимным лубрикантом».

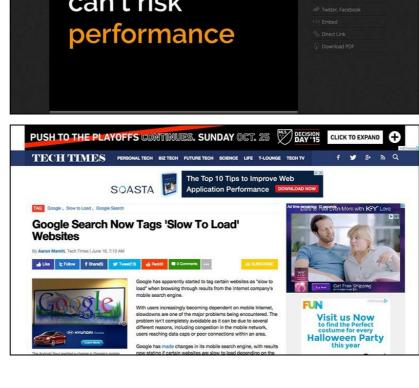
Сегодня нужно много интимного лубриканта, чтобы бродить по неотфильтрованному вебу.

Что за ерунда происходит?



**Best Practices for** 

**Increasing Website** 



## Ложные решения

Проблему признают все. Подобные страницы ужасны и на ноутбуке (у моего вентилятор гнал все три недели, что я готовил речь), а на мобильных устройствах это вообще ад. И издатели принимают меры.

В мае 2015 года Facebook анонсировал Instant Articles, специальный формат для новостных статей, созданных для практически мгновенной загрузки в ленту Facebook.



Facebook сделал это объявление на 6,8-мегабайтной странице, занятой портретной фотографией какого-то мужика. Он в Facebook даже не работает, это лишь фоторедактор National Geographic.

Ниже на странице можно увидеть 41-мегабайтный видеоролик, единственный способ узнать о проекте. В видеоролике редактор восхваляет интереснейшие ненужности нового формата вроде изменения размеров фотографий по повороту устройства. Это означает, что если телефон не держать ровно, фотографии поплывут

Также компания Facebook запустила internet.org, попытку расширить доступ в Интернет. Эмоциональная страница включает рассказы людей из различных развивающихся стран о том, что для них значит доступ в Интернет.

как в документалке Кена Бёрнса<sup>4</sup>.

гигабайта данных.

Ну вы знаете, что будет дальше. Я оставил вкладку с Internet.org и ушёл на обед, а когда вернулся, обнаружил, что она передала больше четверти

Но Мачей, скажете вы. Не может же быть, что эта Земля — это гигантский видеофайл в 16 МБ, на котором вращается планета? Но так оно и есть.

Это посыл Facebook миру: «Интернет медленный. Сидите и вращайтесь.»

И плохой канал — это не специфичная для стран третьего мира проблема! Я уже достаточно наколесил по Австралии и знаю, что в сельской местности в Тасмании и Квинсленде продавцы относятся к WiFi как к бренди со столетней выдержкой.





Покупать можно сколько угодно, но стоит целое состояние и даётся крошечными порциями. И после третьей или четвёртой порции на тебя смотрят с удивлением.

Даже в местах с хорошим подключением типа Сиднея мы сталкивались с проблемами, когда сигнал сети слабый, батарейка садится, а нужно загрузить какой-то огромный кусок сайта с видео и скриптами и выудить оттуда важную тютельку информации вроде адреса ресторана.

У меня есть простое решение: дизайнеры, которые вытворяют подобную пуергу уровня той страницы Facebook, заслуживают высшей меры наказания.

Их нужно заставлять пользоваться мышкой-шайбой от Apple до конца их карьеры. [Слушатели взвыли в ужасе.] Если годы не дают вспомнить, то это идеально круглая мышка. Выяснить, где кнопка, было невозможно, не посмотрев на мышь.

Google запустила конкурента для Instant Articles, который называется Accelerated Mobile Pages. AMP — это специальное подмножество HTML, созданное для быстроты работы на мобильных устройствах.

Почему бы не давать обычный HTML без ненужной фигни? Вопрос остался без ответа.

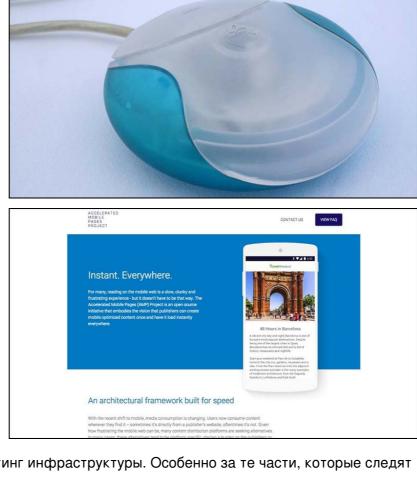
Проект АМР напоказ опенсурсен, в нём участвуют всякие разные издатели. Из-за переизбытка любви

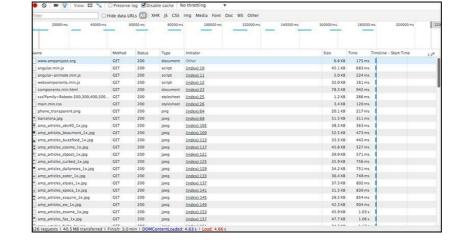
к мобильному вебу Google добровольно взялась за хостинг инфраструктуры. Особенно за те части, которые следят за пользователем.

Джереми Кейт показал мне, что страница с описанием АМР технически бесконечна в размере. Если открыть её в Chrome, то она будет продолжать бесконечно скачивать одну и ту же карусель с видео. На изображении это можно увидеть в морзянкоподобной серии линий загрузки,

показывающих безжалостное скачивание 3,4 мегабайта видео вновь и вновь. Даже если открыть страницу в Safari, где каруселька сломана, страница всё равно успевает

выесть 4 мегабайта.





Эти комично крупные заглавные страницы проектов по ускорению веба — эквивалент просмотра видеоролика про упражнения, где выступающий просто стоит и ест печенье.

Самые крупные в мире технологические компании не могут сделать для мобильных устройств быстрые и легковесные крошечные текстовые сайты, описывающие их будущие флагманские продукты по уменьшению раздутия страниц.

Я не могу представить более сильного признания поражения.

Технический руководитель Google AMP ответил нам в Twitter. Он признал наличие раздутия. Он объяснил, что в Google были ограничения по ресурсам, и проект пришлось аутсорсить.

Это произвело на меня глубокое эмоциональное впечатление. Я не считал, что Google находится в затруднительном положении. Мы так много говорим о том, что Google портит веб, а они не могут позволить себе сделать одностраничный сайт о флагманском продукте. Поэтому я наскрёб в



глубине сердца сострадания и потратил несколько часов личного времени, делая мою собственную статичную версию сайта АМР.

Начал я с того, что заменил карусельки с изображениями на фотографии Уильяма Говарда Тафта<sup>5</sup>, который был самым крупным президентом Америки по массе и жутко солидным малым.

Мне кажется, стало заметно лучше без неуместных анимаций оригинальной страницы.

Вырезанием мусора<sup>6</sup> за остаток дня после обеда мне удалось урезать размер страницы с оригинальных 4 МБ + ∞ до полумегабайта. Это в восемь раз меньше, чем в оригинале. Это не особо сильное достижение, уверен, у вас может получиться лучше.

Я бесплатно предложил мои изменения Google, но, видимо, ограничения ресурсов настолько сильны, что нет времени даже на копирование.

Этот проект вдохновил на тест Тафта:

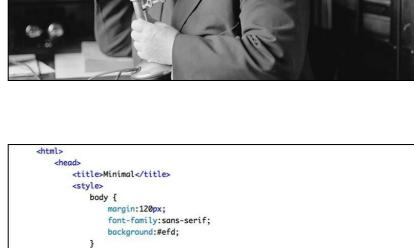
Улучшается ли дизайн страниц, если заменить каждое изображение фотографиями Уильяма Говарда Тафта?

Если да, то, наверное, эти картинки не особо-то и что-то значат для дизайна. По крайней мере, оставьте там Тафта! Вы же только что признались, что он там выглядит лучше. На него приятно смотреть: то ли это усы, то ли шубы, но выглядит он дружелюбно.

Меня поражает убеждение в том, что размер страницы компактного дизайна должен быть выше, чем у обычной замусоренной страницы. Хочу поделиться простым секретом из двух шагов по улучшению производительности любого сайта.

- 1. Отправьте пользователю самые важные элементы страниц. Убедитесь, что они рендерятся первыми.
- 2. Остановитесь.

Instant. Everywhere. An architectural framework built for speed Seamless, simplified publishing





Вы же выдали то, что нужно пользователям. Зачем что-то ещё, что не нужно по определению? Проявляйте смелость в минимализме.

Чтобы провести параллель со знаменитой мотивирующей речью , я бы мог сегодня вечером с материалами, которые есть у вас, переписать сайты, которые я показал в начале выступления, и заставить их грузиться быстрее секунды. За два часа.

А вы можете? Вы можете?

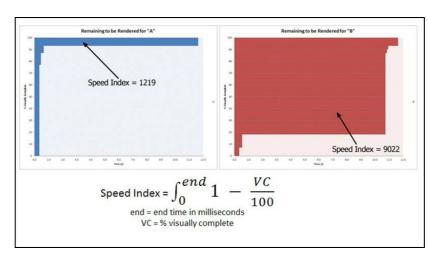
Конечно, можете! Это не тяжело! Мы знали, как делать простые сайты в 2002 году. Это не греческий огонь, дамасская сталь или ещё какой-то утерянный секрет, которым бредят историки.

Но мы сталкиваемся с давлением, заставляющим делать сайты больше.

Ставлю, что если бы вы пришли к клиенту и показали «рыбу» на 200 килобайтов, вас бы уволили. Даже если выглядело бы неплохо и чудом умудрялось содержать все нужные счётчики и мусор социальных медиа, который потребовали поставить. Просто сейчас в голове такой размер не укладывается.

Если вы пытались сбросить вес, то вам знакомы эти ухищрения, которыми люди убеждают себя, что они тоньше, чем на самом деле. Можно втянуть брюхо, носить узкую рубашку, стоять на нужной части весов.

Подобное случается с тестами на производительность. Чтобы убедить себя, что наши тугие сайты загружаются быстро, мы проявили творческую жилку и изобрели новые метрики.



У Google есть одна из популярных, называется SpeedIndex. (Понять, что это Google легко, поскольку только они могли без особых раздумий высокомерно сунуть знак интеграла в определение.)

SpeedIndex говорит: важно то, как быстро рендерится видимая часть сайта. Неважно, что происходит ещё где-то на странице. Неважно, что канал соединения забит, а телефон раскалён. Неважно, что батарейка убывает на глазах. Всё в порядке до той поры, пока быстро рендерится та часть страницы, которую мы видим в окне браузера первой.

Конечно, какая часть страницы появится первой, неважно, если страница по окончании загрузки начнёт показывать промежуточную закрывающую экран рекламу<sup>8</sup>. Или если вы, как и многие другие мобильные пользователи, начинаете немедленно мотать страницу, ловя неоптимизированные куски со спущенными штанами.

Есть лишь одна метрика веб-производительности: время, которое прошло между кликом на ссылку и пропуском последнего баннера.

Всё остальное — это голимая ерунда.

В обсуждениях с защитниками вебпроизводительности я иногда чувствую себя хиппи в разговоре с владельцем внедорожника о расходе топлива.

У них есть море разнообразных приёмов, чтобы улучшить расход. Немного сдуть переднее левое колесо, чтобы быстрее поворачивать. Поставить магнитик на крышку топливного бака. Загнуть боковые зеркала.



Большая часть разговоров про веб-производительность похожим образом пестрит техническими деталями: сжатие, асинхронная загрузка, планирование ассетов, группирование НТТР-запросов, пайплайнинг и минификация.

Всё затмевает простое решение. Если нужно заехать только в магазинчик на углу, сядьте на велосипед.

Если нужно показать всего пять предложений текста, используйте обычный HTML. Да блин, просто дайте текстовый файл. Тогда не понадобятся хаки сжатия, знаки интеграла и сложные диаграммы Ганта, показывающие, что грузится в каком порядке.



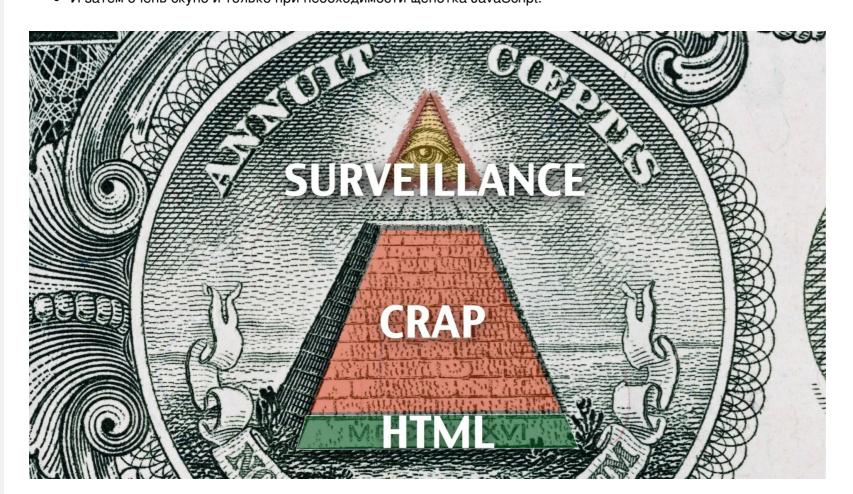
Браузеры очень, очень хорошо показывают обычный HTML.

Технология есть.



Раньше диетологи обожали концепт пищевой пирамиды. Думаю, нам нужна такая для веба, чтобы напомнить самим себе, как должен выглядеть здоровый сайт. Вот что я рекомендую для сбалансированного сайта в 2015 году:

- Твёрдая основа из текста, достойного чтения, отформатированного здоровой порцией разметки.
- Умеренное количество изображений, которые иллюстрируют и задают тон дизайну. Не надо злоупотреблять — хорошенько их сожмите.
- Ложечка CSS.
- И затем очень скупо и только при необходимости щепотка JavaScript.



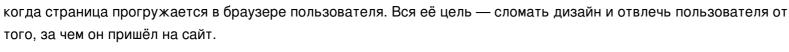
Вот как вместо этого выглядит веб-пирамида в реальности:

- Базовый слой HTML.
- Огромная куча ненужной требухи.
- Наверху целое море скриптов слежки.

## Жирная реклама

Веб-дизайнеры! Это не полностью ваша вина. Вы вкладываете душу в создание отличного сайта, заточенного на быстроту. Вы тратите время процесса разработки на попытки предугадать нужды пользователя и устилаете его пути лепестками роз.

Затем, когда работа закончена, клиент заставляет изгадить результат тяжёлого труда скриптами слежки и рекламой. Контроля над ней у вас нет. Её происхождение и содержимое решаются в момент,



FAT ADS

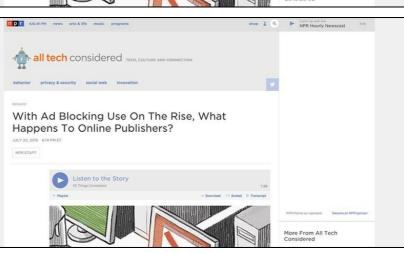
Опыт взаимодействия пользователя вашего сайта захвачен враждебными элементами, неподконтрольными вашей воле.

Вот скриншот статьи National Public Radio, обсуждающей растущую популярность блокировщиков рекламы. В обычном браузере без дополнений страница весит 12,4 мегабайта.

Та же статья с включённой базовой резкой рекламы — это 1,1 МБ. В двенадцать раз меньше. Это не образец бережливости, но всё же разница, которую создаёт всего лишь плагин.

Если посмотреть на то, что торчит в неурезанной версии, то это не просто видеоролики и баннерная реклама, но файлы и файлы скриптов. Каждый маячок-beacon, трекер активности и виджет соцсетей для расшарки несёт свой набор скриптов, которые нужно получить со стороннего сервера. Каждый запрос возвращается с горсткой куки.

all tech considered With Ad Blocking Use On The Rise, What Happens To Online Publishers?



Больше печеньев-куки — это последнее, что нужно ожиревшему сайту.

Эти скрипты идут бог пойми откуда. Они — идеальный вектор атак для доставки вредоносных программ.

Рекламщики скажут, что так и должно быть. Но при разговоре с ними нужно помнить, что они профессиональные лжецы. Не хочу оскорбить. Я говорю про должностные обязанности. Работой рекламщика является убеждение сделать что-то, что иначе вы бы делать не стали. В разговорах с веб-дизайнерами задачей рекламщиков является склонение к мысли, что единственный правильный способ показывать рекламу — это включать гору сторонних мусора и слежки.

Раздутие, ужасы производительности и безопасности, убеждают они, — это цена, за которую читателям обходится за бесплатный контент.

Я наткнулся на схемы «экосистемы рекламных компаний». Схемы мне нравятся. Они передают алчность рекламной индустрии так, как простые числа не смогли бы.

Вот обзор экосистемы в 2011 году, когда было 100

подобных компаний.

Вот так дела обстояли в 2012 году, когда их было 350.

nto LYRIS

К 2014 году Господь ниспослал ещё, и их стало 947.

И в текущем 2015 году этих штук 1876. Они все борются за маленькую часть ваших онлайн-трат.

Эта растущая индустрия очень сложна — мне

кажется, намеренно. Когда нужно понять сложную систему, полезно посмотреть на ситуацию с более крупного масштаба.

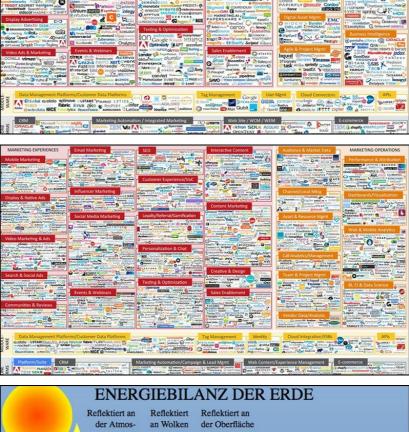
показывает энергетический бюджет Земли. Когда солнечный свет падает на воду или листья, с ним происходят всякие разные сложные вещи, но их можно полностью проигнорировать и просто измерить всю энергию, которая входит и выходит.

К примеру, вот схема на немецком, которая

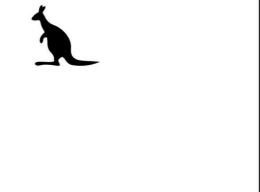
Позвольте мне сделать подобную зарисовку для пузыря рекламы.

попытке добавить восприятия чужой культуры я представил клиента в виде кенгуру.

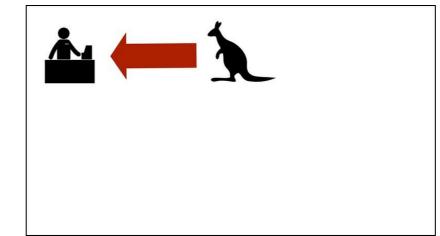
Вначале есть клиент. В жутко неправильной



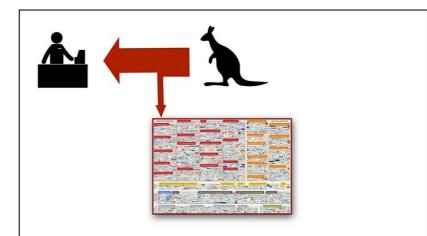




Клиенты дают продавцам деньги в обмен на товары и услуги. Красная стрелочка здесь представляет поток к продавцу денег или, как вы говорите в Австралии, долларов.

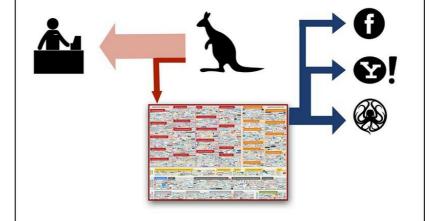


Кусочек этих денег отклоняется для оплаты рекламы. Представим это как маленький налог с продаж на всё, что вы покупаете.



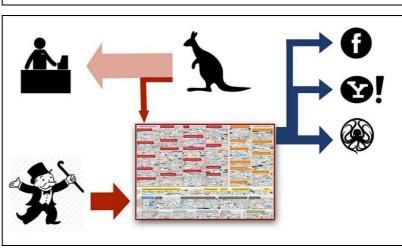
Деньги крутятся по миру рекламных посредников до тех пор, пока поток не оседает в чьём-то кармане.

Сейчас деньги залетают в карманы таких успешных операторов рекламных сетей, как Facebook, Yahoo и Google.



Заметьте, что из системы уходит больше денег, чем входит.

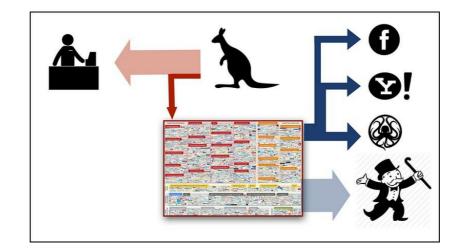
Есть ограничение тому, сколько денег может достаться рекламным компаниям от просто клиентов. Задумайтесь над тем, сколько рекламы вам показывают, и сравните с числом покупок, которые вы делаете за тот же временной



Так что благодарим удачу, что есть инвесторы!

Сейчас они заполняют разрыв заливанием инвестиций в этот раскалённый добела рынок. Инвесторы надеются, что они попадут на ту компанию, которая окажется победителем.

Однако, на какой-то стадии инвесторы, которые добровольно вливают деньги, захотят оказаться на правой части схемы. И получить они пожелают денег больше, чем вложили.



Когда это случится (и как мне кажется, это происходит прямо сейчас), чем-то придётся пожертвовать.

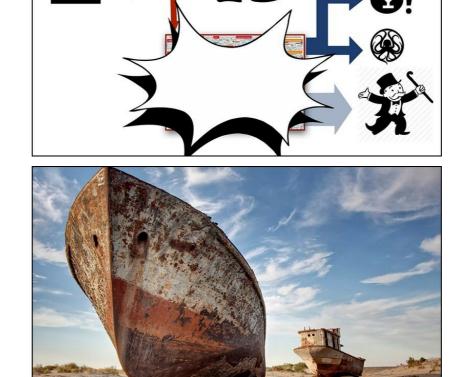
Либо покупать больше, или куда большая часть наших трат уйдёт на рекламные бюджеты...

Или пузырь лопнет.

промежуток.

После взрыва пузыря оставшимся рекламным стартапам станет туго. Начнутся поиски путей выделить себя из стаи с помощью инновационных форм наблюдения.

Мы увидим волну объединений, слияний, новых агрессивных форм слежки и полное уничтожение остатков онлайн-приватности. Это как стайка живородящихся акул в животе у мамы-акулы. Все они сражаются друг с другом за жизнь, чтобы стать одной-единственной, которая родится.



Именно поэтому я предлагал вырегулировать эту отрасль ко всем чертям. Я считаю, что нужно запретить сторонний трекинг и сторонний таргетинг рекламы. Реклама вновь станет тупой. Она будет выдаваться сайтом, на котором появляется.

Сегодня общепринятой практикой является аукцион рекламного пространства во время загрузки страницы. Сама реклама (включая все её скрипты и структуры слежки) появляется в браузере после того, как элементы контента оказываются на своих местах.

Если переводить пользовательский опыт на язык аналогий, это как торговый агент, который появляется на вечеринке после её начала, требует выключить музыку и занимает стол продукцией

Tupperware<sup>9</sup>, чтобы мучать гостей. Атмосфера убита на корню.



Представьте, что загружаемая на стороне сервера реклама означала бы для дизайнеров. Было бы ясно, как на самом деле выглядели бы страницы. Можно было бы грузить ассеты в адекватном порядке. Можно было бы не распространять вирусы. Огромные анимации больше бы не влетали на страницу после загрузки, уничтожая дизайн и заставляя пользователей ненавидеть ресурс.

Действительно, давайте мыслить шире. Не уверен,

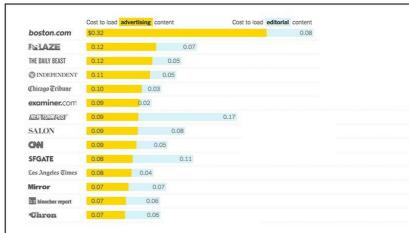
что издателям вообще нужна поддержка рекламой.

Люди забывают про микроплатежи, игнорируют факт того, что у нас уже де-факто есть хорошо работающая система микроплатежей.

Вот график из New York Times, который показывает, сколько денег тратится на загрузку страниц с помощью американской сети сотовой связи на основе используемых данных. К примеру,

связи на основе используемых данных. К примеру, загрузить рекламу страницы Boston.com на типичном тарифе стоит тридцать центов.

CONTENT MAY
SHIFT DURING
LOAD



Это ничто иное как микроплатёж телекоммуникационной компании. И я уверен, что это больше, чем Boston.com получает с показов рекламы на страницах. Представьте, что сайты получали бы эти деньги, которые тратятся на прогрузку рекламы. Они бы могли полностью убрать баннеры, но всё равно бы получали больше, чем сейчас.

Мы попали в глупую ситуацию, когда реклама приносит огромную прибыль операторам связи и рекламным сетям за счёт всех остальных.

Рекламщики будут изо всех сил брыкаться против попыток заставить использовать старую модель тупой рекламы. Но нет фактов, что тупая реклама хуже умной.

Годами и годами слабо таргетированная реклама приносила достаточно денег для целых телестудий, радиопередач и всяких других форм развлечений. Тупая реклама купила Бэтмобиль 10. Денег от ужасно таргетированной рекламы было достаточно, чтобы снять телепередачу с огромными



площадками, стильными костюмами и Бэтмобилем, а затем бесплатно показывать её по телевизору.

Нанять пару журналистов-фрилансеров и веб-дизайнера стоит куда меньше, чем снять ситком. Так почему же заставлять возвращаться к успешной модели финансирования, которая не нарушает частную жизнь — это немыслимо?

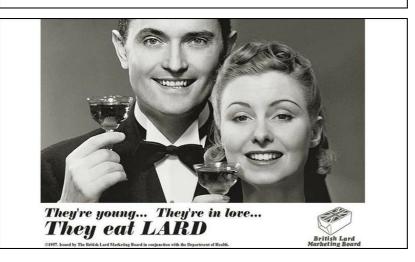
Конечно, рекламщики расскажут нам, как старое телевидение могло бы быть лучше с камерой на каждом телевизоре.

Но мы их уже и так наслушались.



Тупая реклама означает меньше прибыли от рекламы, поскольку большая доля трат на рекламу подстёгивается нелепыми обещаниями о возможностях технологий наблюдения.

Но рынок рекламы всё равно взорвётся, когда лопнет существующий сейчас пузырь. Вопрос лишь в том, захотят ли издатели уйти вперёд и пожинать плоды или исчезнуть в никуда, как и все остальные.



## Жирные ассеты

Давайте поговорим о другом случае веб-ожирения.

Жирные ассеты!

Проблема существовала всегда, но с ростом быстроты рекламных сетей и сложности автоматизации технологий издателей стало проще случайно опубликовать здоровенные файлы на сайте.

## Примеры!

Вот лицемерный блогер, который обожает распекать других за раздутые сайты. [Здесь Мачей говорит про собственный блог.] Но при этом у него есть абсолютно неуместная 3-мегабайтная картинка в начале его самого последнего поста.

Можно предположить, что это типичный случай, когда картинку забыли сжать. Ошибку трудно заметить, если не работать на медленном соединении.

Рост быстроты сетей увеличивает эту проблему.

Вот недавняя фотография пробки в Китае. На дороге 50 полос движения. Добавление 51-ой делу не поможет.

Похожим образом добавление пропускной способности каналов не склонит никого к мысли пихать на страницы меньше контента.

Рассмотрим вот эту недавнюю статью Vice о ботнетах.

В начале статьи стоит бессмысленная 3-мегабайтная фотография наушников. Страница проваливает тест Тафта.

Это часть прискорбного тренда заслуги быстрых каналов связи, при котором в статьях размещаются «изображения-герои» (hero image), вся цель которых — создавать пространство для прокрутки.

Green Arabia

Frequent Tiguis

Angentini is at sease to bidge canada
chinit sed facilities as the Discontinue
chinit sed facilities as the Discontinue
chinit sed facilities as the Discontinue
chinities and facilities as the Discontinue
chinities and facilities as the Discontinue
chinities and facilities as the Discontinue
continue process

Consider Mills

Facilities and Showed
Appetition on Time Blasse A Discontinue
and Continue
Appetition on Time Blasse A Discontinue
control of continue
Appetition on Time Blasse
A Discontinue
Continue
And Conti

FAT ASSETS





В этом случае обвинять автора бесполезно. В инструментах публикации не сработало что-то, ответственное за уменьшение этой огромной картинки. Есть проблема поважнее: быстрые каналы связи провоцируют на включение в страницы подобного визуального наполнителя.

Мы всё больше полагаемся на приёмы сжатия, минимизацию, кэширование и конфигурирование серверов. Вместе с этим ошибки становится труднее отлавливать, они обходятся дороже.

Вот другой пример, интересный по двум причинам. Во-первых, качество оригинальной картинки ужасно. Изображение выглядит так, будто снимали на ботинок, потому что это скриншот телепередачи.

В любом случае картинка огромна. Если загрузить сайт в Safari, размер изображения составит несколько мегабайтов. Если загрузить в Chrome, получится 100 килобайтов, потому что Chrome поддерживает формат сжатия на лету, которого в Safari нет.

Со всеми этими усложнёнными пайплайнами оптимизации трудно понять, видим ли мы то же, что и наша аудитория.

Как бонус, если прокрутить вниз, то в части страницы, которую веб-дизайнеры зовут chum, мы увидим крошечную GIF-анимацию более мегабайта в размере.

Это бесполезный кусок кликбэйта, но он сильно влияет на общий вес страницы.

Ни у кого другого статика не жирна настолько, как у Apple. Сайт до смешного раздут. Думаю, это маркетинговый приём.

«Эти картинки на моём андроидофоне грузятся жутко медленно. Жду не дождусь возможности купить один из гаджетов Apple!»

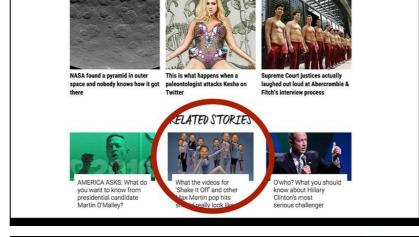
Посмотрим на страницу сайта Apple, которая объясняет iOS на iPad Pro. На ней сколько, 10 предложений?

Как вы считаете, как огромна эта страница?

Можете ли вы поверить, что она крупнее, чем вся память легендарного iMac (32 MБ)?

CONTINUE AND ROTH AND









iPhone 6 S
The only thing that's changed is everything.

Там остаётся место для Macintosh SE... (5 МБ).

iPhone 6 S
The only thing that's changed is everything.

Можно уместить содержимое всего главного компьютера корабля «Спейс шаттл».

iPhone 6 S

The only thing that's changed is everything.

И не одного, а всего флота из пяти космических кораблей (5 МБ).

iPhone 6 S
The only thing that's changed is everything.

Уместятся все работы Шекспира (5 МБ).

51 мегабайт.

И ещё много свободного места. Размер страницы —

## iPhone 6 S The only thing that's changed is everything.

Хреновый минимализм

Страницы Apple иллюстрируют то, что я называю хреновым минимализмом. Это доминирующий тренд

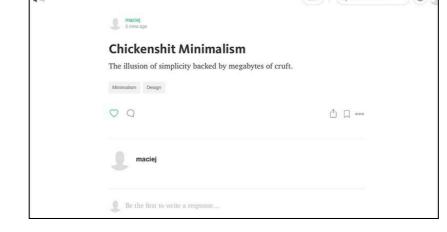
дизайна в современном вебе.

Я написал эссе на эту тему на Medium. Поскольку
это речь на пятьдесят минут, позвольте мне

«Хреновый минимализм: иллюзия простоты,

поддержанная мегабайтами мусора.»

зачитать его полностью:



Я уже упоминал, насколько раздуты статьи в Medium. Это эссе из одного предложения — 1,6 MБ.

Я начал разбираться, почему. Дело не только в бессмысленных скриптах на 400 КБ. Ещё есть эта большое изображение в футере.

Поскольку моя статья настолько мелкая, то до него докрутить физически невозможно. Но с помощью инструментов разработчика я смог увидеть, что

инструментов разработчика я смог увидеть, что там. Не понимаю, что на картинке — какие-то люди в космических скафандрах с планшетами и мобильными телефонами. Но она там. Почему, почему?

Google Contributor

Maciej III 🚇

Tatamagouche
BREWING CO

**Apple Watch Review** 

Её размер — 900 килобайтов.

Вот ещё один пример хренового минимализма: заглавная страница проекта программы Google Contributor.

Это огромная голубая пустыня на 2 мегабайта почему-то требует три щелчка мыши для чтения трёх предложений.

Последнее предложение расскажет, что здесь в Австралии программа недоступна.

Вот домашняя страница Tatamagouche Brewing Company, какой-то несчастной пивоварни, которую я публично высмеиваю. Единственное на странице — это вкуснейшее пиво. Чтобы открыть навигацию, приходится налегать на иконкугамбургер.

Проблема обрюзгшего интерфейса не решится, если налегать на гамбургеры.

Дизайнерские студии обожают этот антипаттерн невидимого гамбургера. Вот 3-мегабайтная домашняя страница компании POLLEN. Не знаю, видно ли с задних рядов иконку-гамбургер.

Самый сильный пример эстетики хренового минимализма, который поставят в музей — это обзор часов Apple Watch от The Verge.

Сколько из вас... даже не знаю, можно ли употреблять глагол «читать». Сколько из вас пытались загрузить эту статью в браузере? Пожалуйста, не надо это делать прямо сейчас, а то вы обрушите Wi-Fi сеть конференции.

Обзор The Verge — это полная мерзость пользовательского интерфейса, которая полностью захватывает механизм прокрутки в браузере. Если попытаться прокрутить статью, начинает происходить странное.

Элементы интерфейса вылетают слева.

Элементы интерфейса вылетают справа.

Элементы интерфейса, с которыми вы не виделись с восьмого класса, внезапно начинают названивать посреди ночи и спрашивать, как дела.

Крайне изредка страница на самом деле прокручивается.

Постоянно и изо всех сил ваш ноутбук крутит вентилятором, пытаясь не умереть от перегрева.

Я попытался сделать видеоролик, как я прокручиваю обзор Apple Watch от The Verge, но мне это не удалось. Видеокарта не так давно выпущенной модели ноутбука Apple в буквальном смысле не могла справиться с нагрузкой.

## Растяжение интерфейса

Когда вышел iPad, дизайнеров поразил какой-то мозговой паразит. Некоторые до сих пор не вылечились. Теперь всё должно выглядеть как тачскрин.

Это британская версия журнала Wired, ещё одного сайта, который объявил войну механике прокрутки. У дизайнеров внезапно прорезалась мысль «горизонтально — это теперь вертикально». Попытаться прокрутить можно, но вместо этого сайт будет двигаться вправо. Плитки статей появляются как огромные пожирающие пространств

сайт будет двигаться вправо. Плитки статей появляются как огромные пожирающие пространство экрана плитки мусора. Всё в этих больших плитках. Кроме, конечно, крошечной навигации слева.

Другой отличительной чертой шика iPad стала элегантная инфографика нечитабельным тощим белым шрифтом на ярком фоне. Эта картинка местной газеты Балтимора была получена с экрана полноценного размера.

Попробуйте забронировать полёт на сайте авиакомпании Virgin America, и вас встретит эта колонна кнопок, парящих в море красного.

Может, на телефоне интерфейс смотрится неплохо, но на большом экране такое выглядит пугающе.

Кнопка «забронировать» отправляет в огромные просторы полей ввода.

Обратите внимание на отличительные признаки в виде огромных шрифтов, крошечных шрифтов и крайне бледных шрифтов.

После выбора пункта назначения сайт направляет на вот этот виджет календаря.

У него одинаково огромные кнопки, но единственная информация, которая мне нужна — цена на билет в каждый из дней — появляется в виде микроскопического текста под датой.

В этой дизайнерской эстетике меня раздражает утеря плотности информации. Я взрослый человек, который сидит у большого экрана, с мышкой и клавиатурой. Я заслуживаю большего.

WIRE D.CO.UK

G. SECRETAL.

TOP STORMS

At WeiWei parties with cyber games and emoji art

CANA MCGOOGAN/ AT WEIWII

WHITED BOIS

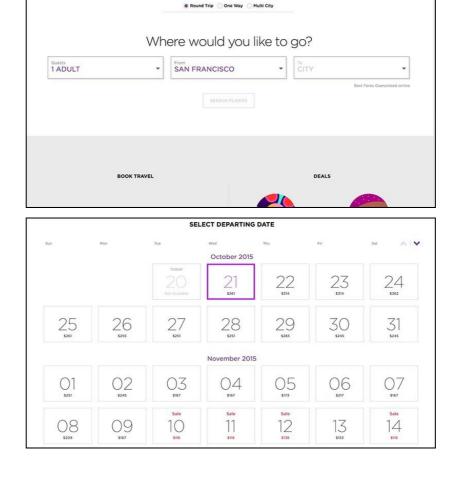
WHITED



MR. TRASH WHEEL GARBAGE & DEBRIS

REMOVAL STATS:

331 TONS (1,543 CUBIC YARDS) SINCE MAY 9, 2014



Не каждый интерфейс должен создаваться под сидящего в туалете с телефона.

Вот так интерфейс PayPal выглядел раньше.

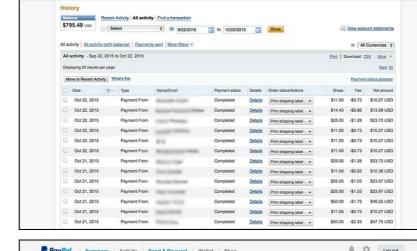
Я никогда не падал ниц по утрам, чтобы поблагодарить бога за дар зрения, что дал мне возможность узреть красоту старого интерфейса PayPal.

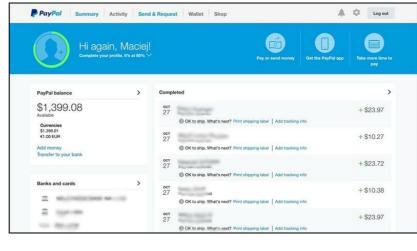
Но задачу он выполнял.

Вот так сайт PayPal выглядит сегодня.

Самый большой элемент — это иконка, которая мучает напоминанием о том, что я не показал PayPal, как я выгляжу. Рядом с ней расположено бесполезное напоминание «скачать приложение», а дальше предложение кредитной карты.

Порядком сортировки управлять нельзя, инструментов фильтрации нет, и без прокрутки видно куда меньше записей. Конечно, самый большой минус — это размытость текста, я едва могу его прочесть.





Вот «контрольная панель» Google, которая позволяет настроить «рекламные предпочтения». Я сходил туда, чтобы сказать, что меня не интересует ничего, и, пожалуйста, отстаньте от меня. После пятнадцати лет наблюдения за мной они смогли определить возраст, но никак не могут понять, что я мужского пола.

Она аналогично игрушечна и раздута. Такое впечатление, что мы проснулись в 2008 году и разом обнаружили, что наши фигурки Lego вдруг

превратились в Duplo 11. Сайты, которые раньше показывали полезные данные, напоминают мультики. Элементы интерфейса большие и толстые. Любые намёки на сложность убраны в какие-то подгамбургерные глубины.

**docker** 

Docs Support Training Tech Blog Blog Docker Hub

FEATURED OPINIONS

The Ad Blocker's

**Docker Solutions** 

SEARCH RESULTS FOR "WEB"

Lazy Loading Images on the Web

Linzer Tarts

Images on the web offer a bit of a conundrum. They are often what makes a web page feel vibrant and interesting, but they can also dramatically hurt web page performance by adding a significant amount of weight to a page. On this site, we use a lot of images within our content. As the  $[\ldots]$ 

Сайты заточены на пользователей-новичков на тач-скринах. Все остальные вынуждены страдать.

Другой пример раздутия интерфейса: домашняя страница Docker. Она состоит из блёклого текста, разделённого огромными порциями пустоты.

Чтобы перемещаться по вашему графическому дизайну, я не должен искать упряжку ездовых собак и сушёную оленину.

Сильнее всего удар пришёлся на поисковые страницы. Огромные буквы и жирные кнопки заменили единственное, что было нужно — поисковую выдачу.

Вот дизайн, в котором места есть только на один результат даже на мониторе высокого разрешения.

Не хотелось, но придётся сделать нагоняй отзывчивому дизайну. Я не знаю, ожирение ли это, и решения у меня тоже нет.

Все понимают, что трудно сделать сайт, который будет хорошо выглядеть на экранах всех размеров. Но акцент на размере заставил забыть о важном различии в том, как пользователи работают с элементами интерфейса.

На телефоне человек тыкает в маленький экран стилусом из мяса, который у него приделан к руке. В этом сценарии логично иметь большие кнопки. На большом экране, где есть гектары пространства и изысканно точное устройство для перемещения курсора, тот же интерфейс сводит с ума. Проблема ухудшается выходом iPad Pro и распространением стилуса Pencil. Заметить разницу может быть невозможно. У меня впечатление, что дизайнеры просто ждут, когда мы все перестанем пользоваться ноутбуками.

allrecipesi

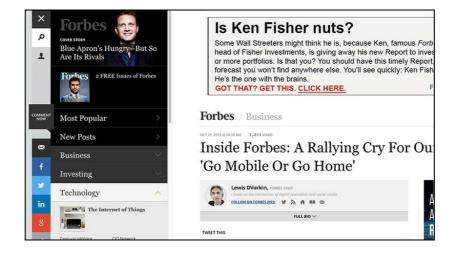
Ingredients

Вот типичный сайт рецептов, пойманный меж двух огней. Не хочу начинать взбучку, поскольку он хорошо старается. Но заметьте, что некоторые элементы крошечны, некоторые огромны. Половина страницы говорит на диалекте тачинтерфейса, и всё это трудно понять и прочесть.

Интерфейс находится в состоянии войны с самим собой, поскольку не знает своей ширины. Маленький шрифт, большой текст, огромная рыжая кнопка «включено», сердечки большого размера, чтобы их можно было тапать. И, разумеется, иконка-гамбургер, обязательность которой, кажется, уже закреплена законодательно.

Вот домашняя страница Forbes с раскрытым левым меню-гамбургером. Похоже на случайный чанк видеопамяти, который почему-то отрендерился в видеовывод. Понятия не имею, что там происходит, какое-то время я просто пялился.

Есть стрелки вверх, стрелки вниз, целый шведский стол из шрифтов, несколько наборов иконок для расшарки этого важного фрагмента информации. Есть раскрываемое меню, у которого есть своя прокрутка, и я даже не знаю, что будет, если прокрутить.



А наверху самодовольно сидит большой смачный кусок рекламы, у которого есть свои представления о типографике и разметке. Он сидит и ворует внимание от всего остального на странице.

Так жить нельзя. Мы же не животные!

## Тяжёлые облака

Наконец, я хотел бы затронуть наши гигантские бэкэнды.

Как мы можем ожидать стройности от вебинтерфейсов, если на стороне сервера устанавливается такой плохой пример?

У меня есть знакомая, которая зарабатывает на жизнь выпечкой печенья. Как и многие другие домашние пекари, она начинала с собственной кухни. Её она нагружала по полной, и вскоре всё дома было в муке, а в квартире стало жарко как в тропиках.

В определённый момент знакомая поняла, что пора покупать коммерческое оборудование для выпечки.

Хороший навык выпечки печенья не означает хорошие навыки выбора профессионального ресторанного оборудования, обсуждения цены и

установки.

Домашнего повара пугает необходимость покупать промышленную плиту, полки для охлаждения и производственный миксер. Пугает то, что теперь ингредиенты нужно покупать в двадцатипятикилограммовых мешках.



Ещё больше пугает найм работников, аренда площади для кухни, получение сертификатов и лицензий. Одна ошибка — и бизнесу конец.

Годами Интернет работал схожим образом. Для хостинга мелкого сайта можно было обойтись потребительскими серверами. Но вот проект стал популярней, и внезапно чарующий голос продавца обсуждает с вами по телефону контракты на оборудование, каналы связи и колокацию стоек.

Легко потерять почву под ногами и наделать дорогих ошибок.

Всё изменил проект Amazon Web Services. Там предложили профессиональные инструменты по запросу, почасово, масштабируемо. Нужно хранилище? Вот сколько угодно. Крутое оборудование? Пожалуйста, сколько захотите.

Не нужно было покупать «железо», не было зависимости в факте его обладания. За услуги приходилось платить наценку, но так удалось убрать много рисков. Конечно, всё ещё нужно было учиться пользоваться подобным. Но процесс на деле оказался интересным.

Но всегда был подвох. Горелки на плитах были мелковаты. Ручки сковород иногда выламывались в неожиданные моменты.

Стоит отдать Amazon должное. В компании с самого начала предупредили об этом и заявили, что нужно создавать свои процедуры с учётом отказов.

Некоторое гарантированно никогда не могло отказать — скажем, холодильники никогда не могли разморозиться. Но были возможны случаи,

вам ни к чему.

когда дверцы холодильников нельзя было открыть по нескольку часов подряд. Так что когда планируете готовку, учтите подобное. Это не капризы Amazon, это просто природа распределённых систем.

Люди начали переезжать в облако, и этот факт заставил думать в крупном масштабе. Думать нужно было

При масштабе всё это хорошо, но для мелких сайтов — перебор. Чтобы пожарить одно яйцо, вся кухня ресторана

несколькими машинами и зонами доступности, и это означало мысли о избыточности и отказоустойчивости.







А потом системы стали крупнее, и Amazon начала предлагать больше автоматизации. Они не только предлагали арендовать огромные печи, но и целый взвод кухонных роботов, которых можно запрограммировать на всякие однообразные задачи.

И опять же, программировать роботов оказалось куда веселее, чем выполнять кухонные операции самому.

Во множестве технологических компаний найти хороших программистов тяжелее, чем найти деньги. Поэтому для них переключение на высокоавтоматизированные сервисы было логичным.

Для программистов облако предложило возможность запроектировать распределённые системы на десятках и сотнях серверов. И это в начале карьеры. Это как получить ключи от 747-ого «Боинга» прямо по окончании лётной школы.

Большая часть работы по сайтам рутинна. Подключаем базу данных к шаблону и смотрим, чтобы никто не поскользнулся на шнуре питания. Работа важная, но вряд ли вдохновляющая.

Но автоматизация на большом масштабе? Это же классно и очень сложно!

Это как взять несколько бухгалтеров мелкого бизнеса и сказать, что теперь они будут создавать офшор на Сейшеллах для ухода от налогов на мультимиллиардную корпорацию.

Внезапно, они оживляются, они чувствуют свободу. Они прямо на верху иерархии потребностей по Маслоу, самоактуализируются на всех парах. Они не захотят вернуться.

Вот так себя чувствуют программисты, запутавшиеся в облаках. Они любят играть с подобным. И я не исключение.

Сложность для умных людей — это как привлекающая насекомых горящая лампочка. Мы не можем с ней бороться, даже если знаем, что она пагубна. Работать с подобным просто круто.

В результате большая часть веба ужасно перезастроена. Некоторым компаниям нужны технологии для работы в большом масштабе, и они их разрабатывают. Такие технологии оказываются в руках людей, которые мечтают работать в этих масштабах.

И сказать «нет» некому.

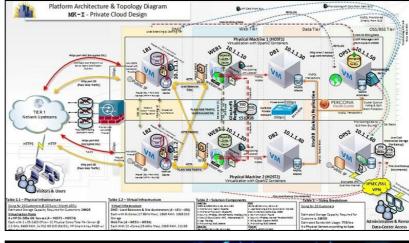
Адам Дрейк написал интересный пост про анализ 2 миллионов партий в шахматы. Вместо того, чтобы использовать кластер Hadoop, он просто навинтил несколько Unix-утилит на ноутбуке и получил 235-кратный прирост производительности над подходом big data.

Суть не в том, что использующие кластеры Наdoop — дураки. И не в том, что всё можно сделать на ноутбуке. Суть в том, что интуиция многих о том, что является большой системой, не соответствует аппаратному обеспечению 2015 года.













find . -type f -name '\*.pgn' -print0 | xargs -0 -n4 -P4 mawk
'/Result/ { split(\$0, a, "-"); res = substr(a[1], lengt
h(a[1]), 1); if (res == 1) white++; if (res == 0) black++; i
f (res == 2) draw++ } END { print white+black+draw, white, b
lack, draw }' | mawk '{games += \$1; white += \$2; black +=
\$3; draw += \$4; } END { print games, white, black, draw }'

This find | xargs mawk | mawk | pipeline gets us down to a runtime of about 12 seconds, or about 270MB/sec, which is around 235 times faster than the Hadoop implementation.

Можно сделать жутко много вещей на ноутбуке или крошечном блейде формата коробки пиццы, если пропустить пятьдесят слоёв оверхеда абстракций.

Давайте я дам собственный конкретный пример. Недавно со мной контактировал конкурент (назовём их ACME Bookmarking Co.), который собирался покинуть сектор обслуживания закладок и продать сайт.

Хотя у АСМЕ было куда больше трафика, чем у меня, оказалось, что ежедневных пользователей у них в два раза меньше. Меня это успокоило: тяжёлой задачей при росте сайта закладок является работа с пользователями, сохраняющими контент.

	PINBOARD	ACME
DAILY USERS	8 K	3 K
EMPLOYEES	1/2	1/2
REVENUE	12 K	5 K
HOSTING	1 K	9 K
INCOME	11 K	(4K)

У нас было одинаковое количество сотрудников. У них был стажёр, который часть дня работал над проектом. Я маялся дурью, ездил по миру и выступал на конференциях. Скажем, по половине работающего на полный рабочий день сотрудника.

У нас был похожий доход на пользователя. Мне приходит 12 000 \$ в месяц, им — 5 000 \$.

Но проекты глубоко отличаются в цене. ACME хостит сайт на AWS, где в определённый момент они платили 23 000 \$ (!!) в месяц. Путём титанических усилий эту цифру удалось опустить до 9 000 \$ в месяц.

Я плачу лишь чуть за тысячу долларов в месяц за хостинг на основе моего собственного оборудования. Это число включает и амортизированную стоимость «железа», и газировку из торгового автомата в дата-центре, где я размещаюсь.

Для меня закладки— это прибыльный бизнес, который позволяет быть самозанятым. Для них— это дыра на 4 000 \$/месяц. Я неплохо живу на том же потоке прибыли, который их заставляет продавать пользовательские данные маркетинговым компаниям и уходить из сегмента.

Дело в том, что предположения о сложности закрепят на месте ваши ожидания и ограничат пространство ваших попыток. Если вы считаете, что «настоящий» сайт должен жить в облаке и запускаться на десятке машин, станет убыточным целый ряд продуктов, которые иначе могли бы приносить деньги.

Аналогичным образом, если вы считаете, что вам нужна многослойная CMS и мощный кастомный JavaScript для ведения бизнеса по онлайн-публикации, пространство возможных решений, которые вы попробуете, значительно сужается.

Вместо того, чтобы пытаться заставить перезастроенные проекты выглядеть простыми, спросите себя, почему они не могут быть простыми.

Не хочу быть слишком жёстким по отношению к облакам. Несколько моих лучших друзей в облаках.

Скорее, я хотел бы всем напомнить, что внизу есть много места. Сегодня разработчики работают на слишком большом количестве слоёв, чтобы заметить, какой сильной стала технология. Проблема обратна сказке про принцессу на горошине.



Похожее случается с браузерами. Основная технология настолько хороша и быстра, что мы смогли нагрузить её сверху кучей дряни, и она всё ещё хорошо себя ведёт. Есть способ заставить сайт выделяться — найти смелость дать браузеру делать то, на что он оптимизирован. Не считайте, что все ваши фреймворки и инструменты сохраняют время и деньги.

К сожалению, сложность достигла эдакого статуса гордости. Разработчики хвастаются друг другу, что у них в «стэке» технологий и делятся советами, как этим управлять.

«Стэк» — это аналог слову «полифилл» <sup>12</sup>. Оба термина являются признаками того, что у вас получается жутко переусложнённая структура.

## Вдохновляющее заключение

Есть причины беспокоиться о подобном не только из-за эстетики или эффективности.

Давайте я проведу аналогию, которая покажет будущее веба с помощью компьютерных игр.

Первая версия веба — Minecraft. Это открытый мир с простыми объектами, которые подчиняются простым правилам. Графика угловата, но суть не в этом, да и никому нет дела.



В этой версии от вас предполагается быть активным участником, ожидается, что вы создаёте новое, а веселее всего играть, если сотрудничать с другими. Правила игры просты, они накладывают мало ограничений.

Игроки создают в Minecraft поразительное.

Вот с любовью созданный целый город из небоскрёбов.

Вот какие-то маньяки сделали полный работающий процессор из красного камня. Если его увеличить до нужной степени, он тоже сможет запустить Minecraft — в голове такое укладывается с трудом.



Другая версия веба — это Call of Duty. Это отполированная до блеска игра, в которой как бы нужно участвовать, но не особо много, с множеством захватывающих дух эффектов и разнообразными возможностями для покупки контента.

Создание подобной версии веба требует большую команду специалистов. Никто не понимает всю картину работы, да ни от кого и не требуется. Даже если бы кто-то и мог отточить все технологии, стоимость разработки непомерно выросла бы.



Пользовательский опыт в подобной версии веба — это сопровождение с иллюзией помощи по довольно жёстким ограничениям. Есть очевидный путь, по которому нужно идти, и препятствия не дают отклониться. В дополнение к этому игра несёт слегка пугающий империалистический взгляд на мир. Единственный способ отказаться от него не играть.

Несмотря на чрезмерную стоимость разработки, во всём чувствуется однообразность. Вы всегда в той же коричневой зоне боевых действий. Ценой больших усилий и навыков вы можете внести мелкие изменения в такой игровой мир. Но большинство будет играть ровно так же, как и задумали издатели. Это пассивная форма развлечений с редкими вспышками долбления по клавишам.

сайта или редактирование веб-страницы, делает сложнее изучение написания кода путём просмотра исходников, продвигает это потребительское отношение к вебу.

Любой шаг, который делает сложнее создание

Я сам в 1999 году изучил HTML путём просмотра исходников. Я считаю, что нужно вовлечь как можно больше людей в веб, сделать инструменты, которые делают участие проще. Обязательности становиться полноценным программистом нет. Но



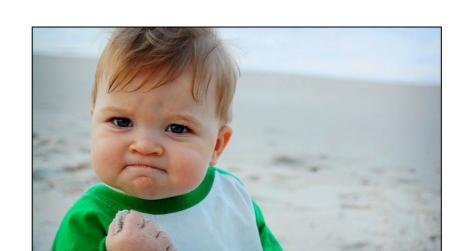
нужно иметь возможность учиться из кода друг друга без специальных уроков, чтобы при необходимости заглянуть и понять, что под капотом.

Если считать, что для публикации статей нужна команда профессионалов, то подобное станет самоисполняющимся пророчеством. Переусложнение веба означает устранение тех возможностей, которые раньше давали людям шанс научиться самостоятельно и удивить окружающих новыми неожиданными идеями.

Вот она, поучительная часть выступления:

Давайте сохраним веб как среду гипертекста, которой он является, единственным в своём роде в мире. Давайте не станем превращать веб в ещё одну среду для потребления, как это уже случалось в других примерах.

Давайте пообещаем, что с ростом мощностей и быстроты компьютеров и сетей веб тоже должен стать быстрее.

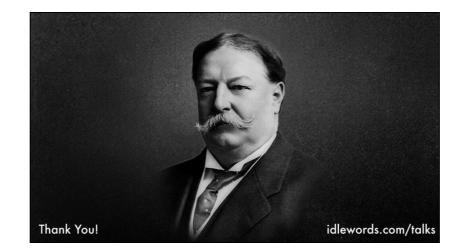


Давайте не дадим запаниковавшим динозаврам онлайн-издательств растоптать нас, когда они в панике убегут от метеора. Вместо этого давайте скроемся в норах и дадим природе сделать должное.

И самое главное, давайте сломаем хребет мировым кругам онлайн-слежки, которые угрожают не просто нашим средствам к существованию, но и свободе. Не только здесь, в Австралии, но и в Америке, Европе, Великобритании, в любой свободной стране, где лишь десять лет назад идея постоянной полной слежки звучала как глупая научная фантастика.

Как остановить гигантские компании от стерилизации Интернета? Сделать их сайты ненужными. Если интересное происходит где-то ещё, за нами последуют. Мы уже сделали подобное с AOL и Prodigy, мы можем сделать это ещё раз.

Для возможности подобного жизненно важно, чтобы в вебе могли принимать участие многие. Это означает создание сайтов, не только маленьких настолько, чтобы их мог посетить весь мир, но и маленьких настолько, чтобы любой мог научиться создать собственный по подобию.



Раздутие волнует меня не из-за неэфективности. Раздутие волнует меня, поскольку оно делает веб труднодоступным.

Сохранение простоты веба означает сохранение его шарма.

Спасибо!

[Тяжёлый взволнованный поток аплодисментов.]

## Примечания переводчика:

В прошлом Мачей Цегловски создал и построил хранилище данных для Twitter, работал во внутреннем инкубаторе стартапов Yahoo и ряде других интересных компаний. Цегловски решал разнообразные задачи по работе с рекламой и созданию вебприложений с помощью LAMP-серверов, АЈАХ, иногда прибегая к Java, C++ и другим языкам. Он ведёт блог Idle Words и проявляет интерес к сфере венчурного капитала. Сегодня Цегловски занимается рядом других личных проектов и сайтом социальных закладок Pinboard. Последний получил популярность как прибежище пользователей Delicious, проданного Yahoo компании AVOS Systems.



Перевод очень волен по своей натуре: это комбинация самой речи и её более короткого варианта в виде текста.

- 1. В оригинале используется выражение Defining Deviancy Down, которым сенатор Дэниэл Патрик Мойнихэн
- озаглавил свою статью с критикой нормализации общественного отношения к преступности. 2. Здесь и далее ссылки заменены на оригиналы или переводы на русском.
- 3. Карри и крамбл блюда индийской и британской кухонь, соответственно. Суть конфликта сводится к тому, что это основное блюдо и десерт, и их смешивание портит удовольствие от трапезы. Мачей нарочно выбрал такую мелкую бытовую проблему для контраста сравнения с возвышенной литературой.
- 4. Кен Бёрнс американский кинорежиссёр и продюсер документальных фильмов. Здесь идёт речь о его характерном стиле работы, который даже получил название «эффект Кена Бёрнса»: при повествовании кадр идёт по фотографии, выхватывая нужные детали, отдаляясь или приближаясь.
- 5. Уильям Говард Тафт 27-й президент США (1909—1913). В выступлении используются две характеристики этой политической фигуры: огромная масса тела (менялась от 115 до 160 кг) и характерная внешность на фотографиях. 6. В оригинале используется слово cruft. Это слово из технического жаргона описывает ненужные аппаратные или программные компоненты. В переводе здесь и далее ему соответствует слово «мусор».
- 7. Отсылка к речи героя фильма «Американцы»/«Гленгарри Глен Росс», в которой матёрый брокер заявляет своим не самым успешным коллегам, что он мог бы добиться большего даже с теми же ресурсами.
- 8. В оригинале используется конкретный термин интерстициальная реклама (interstitial ads).
- 9. Tupperware американская компания, поставляющая предметы для кухни (пластиковая посуда, контейнеры и так далее). Известна практикой работы по схеме «прямых продаж», когда дистрибьютор контактирует с покупателями вне точек розничной продажи. В подобных целях могут организовываются «вечеринки Tupperware», но возможно появление торгового представителя на небольших социальных мероприятиях другой тематики.
- 10. Речь идёт про реквизит середины шестидесятых, созданный на основе концепт-кара Lincoln Futura. Только сам оригинал без модификаций обошёлся в 250 тысяч долларов. С учётом инфляции это 2,2 миллиона долларов 2016 года. 11. DUPLO — фигурки от компании LEGO, предназначенные для детей до 5 лет. В связи с этим они по каждой из
- сторон в два раза больше классических кирпичиков LEGO, чтобы маленький ребёнок их не вдохнул и не проглотил. 12. Polyfill — фрагмент кода, призванный обеспечить в одном браузере функциональность, которая обычно считается стандартной. К примеру, некоторые функции HTML5 в старых версиях Internet Explorer приходится реализовывать через polyfill.

веб-разработка, Мачей Цегловски, Maciej Cegłowski, передача данных, облака, облачные вычисления, веб-дизайн, HTML, литература, веб-сайты, браузеры



```
Что почитать по веб-разработке
     ● 103k ★ 1004 ■ 137
     Облачные вычисления: работа над ошибками
+1
     ⊙ 1,6k ★ 2 ■ 13
     Облачные вычисления: основы применения в сфере Интернет-стартапов
     ● 11,5k ★ 66 ■ 32
```

## Комментарии (70)

8 PerlPower 9 марта 2016 в 04:29 #

Добавление всего этого мусора на сайт дает колоссальный прирост скорости разработки, а также позволяет экономить на качестве рабочей силы. Чтобы из ситуации был выход, нужно чтобы экономический выхлоп от оптимизированных сайтов был заметен их владельцам. А он не будет заметен, потому что в наше время каждый год что-то переписывается, и скорость разработки ставится во главу угла. Почти наверняка нет смысла возиться с оптимизацией фронтенда, который завтра перепишут заново(кстати зачем его все время переписывают?). А потому остается только кукарекать, чем и занимается автор и в чем я его всячески поддерживаю.

💇 DIHALT 9 марта 2016 в 06:54 # 🔓 🚶

+12 ↑ ↓

+8 1 1

Что обсуждают

Вчера

Как Облако@mail.ru спасло все\* мои

файлы и что из этого вышло \_\_\_\_ 54

Центр сертификации Let's Encrypt

Релиз ownCloud 9.0 — opensource

Проблемы после миграции с openvz на

Долгая история про локализацию даты

альтернативы Dropbox и другим

облачным хранилищам \_\_\_\_ 11

Ixc в Proxmox 4.x 26

без года в РНР 📉 26

выдал миллион бесплатных

сертификатов 6

Неделя

Сейчас

Так о чем и речь, что все стараются зачем то навертеть себе всех возможных новых технологий, но сами не понимают зачем им они. Ну чтобы было как у всех. Хотя в 90% останься они на технологиях начала 10 года или даже 2005, то ничего бы и не изменилось в целом.

hssergey 9 марта 2016 в 10:35 # h

+9 **1** 

Тут зачастую может выйти так, что уменьшая время на разработку, мы воруем это время у тех, кто будет вынужден в дальнейшем такую систему поддерживать и развивать. Пример (надуманный но на основе реальных событий): пусть есть сайт на джанге, на которую для примера наверчено еще 3 библиотеки. Понадобилось обновить джангу (в старой версии нашли дыру, или нет нужной функциональности, которая понадобилась, или переносим сайт на другой хостинг, где уже предустановлена джанга но не той версии). И выясняется, что библиотека 1 не работает под новой версией. Обновляем библиотеку. А у нее изменился АРІ и существующий код сайта не работает. Переписываем код. И тут выясняется, что библиотека 2 конфликтует с новой версией библиотеки 1. Обновляем ее. И так далее... В результате сопровождение и развитие такого продукта, в которого напихана тьма современных технологий, превращается в нетрадиционный секс, где большую часть времени не занимаешься непосредственно самим функционалом системы, а прилагаешь титанические усилия,

чтобы все это нагромождение не развалилось, ваяя "стройную систему костылей и подпорок"...

+4 ↑ ↓

А вот второй пример, тоже надуманный, но на основе реальных событий: продукт написали быстро, и использованием кучи библиотек и костылей, некоторое время развивали, получили фидбэек, вторую версию написали с нуля, выкинув часть библиотек, заменив меньшими или собственными решениями, а что-то выкинув совсем. Писали бы с нуля всё первый раз — получили бы в 3 раза бОльший срок на "боевой прототип". Но зачем?

dom1n1k 9 марта 2016 в 15:03 # h

 $0 \uparrow I$ 

father\_gorry 9 марта 2016 в 11:47 # h 1

При чем тут прототип? Автор (оригинала) говорит исключительно о реальных сайтах на продакшене.

А дает ли? Насколько мне известно, никто не делал серьезного исследования на эту тему, а самим

разработчикам доверять нельзя из-за предвзятости.

+1 ↑ ↓ Чтобы сделать первый шаг (и причем неплохой такой шаг!), достаточно убрать со страницы какую-нибудь

лайкалку. Потому что за крохотной кнопкой часто спрятаны огромные скрипты. И я видел немало сайтов, где

количество тех лайков измерялось единицами — то есть их можно было выбросить вообще.

alex\_blank 9 марта 2016 в 07:18 #

+3 ↑ ↓

Это не только с веб-страницами, а с технологиями в целом происходит. Портировал недавно одну старую MS-DOS игру для веба. Были использованы DOSbox и emscripten. То что раньше весило пару килобайт, стало весить 50 мегабайт — но зато запускается в браузере.



## Процитирую себя:

Кому-то кажется, что игра выглядит несовременно? Вы сами отстали от времени: несовременной она была в 2003 году. Сегодня, чтобы просто запустить её, понадобился технологический стек настолько умопомрачительной сложности, что остается только гадать, как же это мы до такой жизни дошли. В целом, вышла неплохая иллюстрация того, как некоторые изначально простые вещи со временем становятся не проще, а сложнее.

сегодня вам понадобился стек чтобы запустить эту игру где она не была предназначена запускаться. Игра запускалась только на определенной архитектуре под определенной ос. Игра была просто заточена на очень конкретное окружение.

Теперь, вы запустили ее в браузере, технически позволив запустить игру на целой куче платформ, используя технологический стек с открытыми стандартами, а значит завтра разработчик собственного браузера тоже сможет запустить игру у себя.

Теперь вопрос, что быстрее, написать один раз эмулятор под DOS игры, или переписать тысячу игр? Сегодня эта игра должна работать не под DOS, а под windows 7, 8, 10, android, ios, при этом желательно не писать каждую версию с нуля. Потому и раздувается стэк, потому что это сокращает время разработки.

+6 ↑ ↓

Хорошо что я обновил комментарии перед тем как написать всё тоже самое. Не понимаю, как можно говорить о "простые вещи становятся сложнее" в контексте — стало хуже. Нет, хуже не стало. Да, технологии стали сложнее. Но и возможности которые они дают — несравнимы с тем что было.

@alex\_blank вы не портировал игру, а эмулировал ее окружение! Уверен, порт игры был бы простым, примитивным, весил бы десятки килобайт и не требовал бы мощную машину. Но нет, надо полностью сэмулировать окружение для игры и заявить, что все непомерно усложнилось... Так усложнилось, потому что решаете задачи "в лоб". Вторая фишка усложнившегося стэка технологий — это как раз возможность решать задачи просто. Вы не тратили месяцы чтобы разобрать игру и написать ее заново под современные технологии. Вы взяли готовый эмулятор и запустили на нем игру... Именно Вы. Лично ВЫ — виновник того что

главной ЦА этих "сложных вещей, которые раньше были проще"

alex\_blank 9 марта 2016 в 09:14 (комментарий был изменён) 🙏 🚶 ↑

эта игра требует кучу современных технологий и весит 50 мегабайт. Лично Вы являетесь потребителем и

Уверен, порт игры был бы простым, примитивным Это не так. Если под портом вы понимаете переписывание исходника вручную на JS — то это *архисложная* 

задача в сравнении с тем что было проделано. Хотя бы потому что исходника попросту нет, а если бы и был — никто бы не захотел возиться с его переписыванием. Ведь возни много, а результат тот же самый для пользователя. Конечно, за счет усложнения run-time стека.

То есть мы наблюдаем экспорт сложности из одного domain'а в другой.

+1 ↑ ↓

Программа от апполона писалась специально под апполон и специально под одну конкретную задачу. Эта игра писалась специально под конкретное окружение и больше ничего не поддерживает. Написать сегодня 2D игру под windows проще чем раньше под msdos, достаточно взять например SDL и вы приблизительно за тоже время что и автор игры напишете порт, это в случае если бы у вас было полное ТЗ на игру, при этом вы с нулевым оверхедом сможете себе позволить графику получше. Стек у вас получился бы больше, но он стал бы более "общим", арі стаал бы общим, вы не затачиваете свою игру под узкое, конкретное окружение. И больше авторов смогло бы работать над игрой, потому что использовалась популярная известная библиотека, а не какой то узкий самописный код с собственными решениями.

Раздутие стэка? да. Зато проще портировать на те платформы где этот стэк присутствует (в моем примере C + SDL)

Однако портирование игры с одной платформы на другую ВСЕГДА было сложной задачей. Взять хотя бы какой нибудь допустим apple ||, как просто было бы вам портировать DOS игру на подобный компьютер? думаю это было бы архисложной задачей, ничуть не проще чем написание эмулятора. Помимо этого у JS совершенно другая парадигма программирования (асинхронные события) которой в MSDOS нет, там код однопоточен и синхронен (образно).

Выводы: раздутие стэка увеличивает скорость разработки, позволяет запустить нечто на нескольких платформах обхватив большую аудиторию, зачастую позволяет привлекать новых программистов в проект многократно быстрее чем если писать собственное решение.

Портирование с одного "domain" на другой, как было сложным так таким и осталось. Если не стало проще.

Sadler 9 марта 2016 в 08:31 # Лично для меня всегда при разработке проблемой становятся нестандартные шрифты: они должны включать все

+3 **1 J** 

необходимые символы, все необходимые варианты начертания и при этом не выходить за рамки допустимого размера, прилично выглядеть и быть легальными. Получить всё это сразу иногда бывает столь сложно, что проще отказаться от нестандартных шрифтов вообще.

пользуй гугл-шрифты

+2 ↑ ↓

titulusdesiderio 9 марта 2016 в 09:46 #

+3 ↑ ↓

Подискутировали с @Рu4u на тему этой статьи. Но он ридонли, так что от его имени:

а еще например, я сохранил эту страницу на диск и она весит 243 КБ

250 КБ для 2016 — мелочь.

высказывание спорное

а картинки + јѕ (т.е. всё прочее) еще 3,2МБ интересно как автор замерил размер той страницы

но это бред, из 900 кб(html+css) сам html только 25% css нормально кэшируется, даже если бы его было суммарно на несколько мегабайт — ничего в этом страшного

не было бы поэтому тема спорная, кэш никто не отменял

Если поднапрячь память, то еще когда я учился в школе началось нытьё, что некоторые веб-страницы весят

около мегабайта но если мегабайт в 2005 было овердофига, то 3 мегабайта в 2016 — ерунда.

скорости выросли в десятки раз специально порылся, нашел мастера и маргариту ansi — размер 850КБ utf-8 — размер **1,4МБ** сайты-то сейчас на utf-8 делают, опять же сравнение с другйо кодировкой некорректно **dbelka** 9 марта 2016 в 11:36 # h ↑ 1. Первую загрузку тоже никто не отменял. Вот вы ходите с поисковика на сайты, каждый из них открываете первый раз, в итоге все работает медленно. Оправдания типа второй раз быстрее тут не работают. 2. Картинки и js+css это совершенно разные вещи. Картинки могут грузится после отображения страницы параллельно, загрузка css и некоторого јs блокирует отображение страницы вообще. Так что не надо сравнивать мегабайт картинки с мегабайтом css, пожалуйста. К тому же, css и јs нужно ещё проинтерпритировать, это особенно сказывается при открытии сайтов на мобильном. forgotten 9 марта 2016 в 09:55 # Это почти на 100 КБ больше, чем полный текст «Мастера и Маргариты», наполненного мистикой и комедией романа Булгакова о Дьяволе, который наведывается в Москву со своей свитой (включая огромного кота!) во время Большого террора 1937 года. Действие «Мастера и Маргариты» разворачивается в 1936 г. https://otvet.mail.ru/question/7584044 Развелось, блин, специалистов во всех областях знания. **atomlib** 9 марта 2016 в 11:20 # h 1 Тут говорят иначе. годов – он мог там появиться только после смерти в 1937 году. Булгаков (пьеса «Последние дни»). http://www.e-reading.club/chapter.php/82995/6/Pozdnyaeva\_-\_Voland\_i\_Margarita.html Russian and Studio Art), хотя бы приблизительно знает, что он говорит. forgotten 9 марта 2016 в 11:55 # h 1 «Тут» элементарно историю не знают. Ягода был расстрелян 15 марта 1938 года. художественно уточнил и спрессовал «реалии» целого десятилетия» **Anisotropic** 9 марта 2016 в 09:57 # Интересно, про Prodigy — это он на "The fat of the land" намекает? **atomlib** 9 марта 2016 в 10:00 # h ↑ Нет, это американский провайдер услуг связи. Пик пришёлся на девяностые. dim\_s 9 марта 2016 в 10:03 # сайтах.

+2 **1** Особо стоит отметить и появление двух «новеньких» гостей на балу у сатаны, имен которых Коровьев якобы не знает. Один из них – автор плана убийства человека, «разоблачений которого он чрезвычайно опасался» (с. 686), с помощью яда, которым обрызгали стены кабинета. Второй – исполнитель. Этот изысканный способ убийства и анонимность гостей на балу ассоциируются с попыткой Ягоды отравить Ежова, стены и шторы кабинета которого были, по его приказанию, обрызганы ртутью (ртуть добавили и в побелку). Правда, желаемого результата Ягода не добился. Но присутствие Ягоды на балу у Воланда связано уже с концом 30-х Из реалий 1930-х годов Б. Гаспаров отмечает неоднократное упоминание в романе имени А. С. Пушкина в связи со столетием гибели поэта в 1937 году. В торжествах по этому поводу активное участие принял

+7 ↑ ↓

+1 ↑ ↓

+1 ↑ ↓

Вообще, думаю, человек, который выбрал в университете специализацию по русской культуре (double major in

А вообще, основной вывод вашей статьи выглядит так, если что: «В «Мастере и Маргарите» Булгаков

Я думаю 90% не осилили прочитать эту статью полностью, такой же процент мусора находится на современных

P.S. А теперь выдохнули) Эта аналогия не говорит о том, что кто-то из нас мусор, просто ее реально стоит прочитать полностью, от и до. В ней описывается куча проблем современного веба, не только на стороне клиентского кода, но и на стороне дизайна, рекламы и многого другого.

0 1 1 Так ведь у автора на странице девиз "brevity is for the weak" :) sim-dev 9 марта 2016 в 10:34 #

Нужно быть специалистом очень высокого уровня, обладающим очень весомым (жирным?) авторитетом, чтобы рискнуть высказать такое мнение. И автор перевода должен обладать всеми этими качествами, чтобы рискнуть запостить это на хабре. Потому что в статье покушение на святое — смысл развития технологий.

дизайна, рекламы и многого другого. Нет, в ней на примере веба описывается проблема всего современного мира технологий — мощность без смысла,

В ней описывается куча проблем современного веба, не только на стороне клиентского кода, но и на стороне

сложность без нужды, дизайн без идеи, существование без цели. Все, рассчитанное на обычного "потребителя" таково, без исключений (разве что скотч не такой).

**RubaXa** 9 марта 2016 в 11:07 (комментарий был изменён) #

А проблема кстати на поверхности, мы все хотим быстрый веб, который «всё умеет», но барузеры не дают нам нужного и/или быстрого АРІ. Началось с мелочей, всякие анимашки, балуны, календарики, алерты, лаеры, Drag'n'Drop и т.п., а браузеры все разные и пришлось изобрести сначала Prototype + script.aculo.us, а потом jQuery + jQuery UI (40KB + 78KB min + gzip).

И вот прошли года, в этом jQuery будет праздновать юбилей, как ни как, 10 лет уже и за все эти года даже в самых современных браузерах не появилось арі, которое позволило бы избавиться от него (тут можно возразить, а как же http://youmightnotneedjquery.com/ и т.п., это не альтернатива, даже для современных браузеров, слишком упрощено).

Скажу больше, за эти 10 лет, всё стало только веселей, кроме абстракций над DOM/Ajax/Animation/Events, появились над массивами и объектами в виде Lodash/Underscore и не потому, что они дают новые возможности, хотя и это тоже, а потому что Lodash просто быстрей чем Native код 0\_o (ситуация конечно меняется, но флагман v8, до сих пор позади).

Несколько лет назад появился React, который реализовал Virtual DOM, чтобы полностью избавиться от взаимодействия с реальным DOM, и тут дело не в том, что он по умному обновляет только нужные узлы, как многие думают, проблемы глубже, любое взаимодействие с реальной нодой (вызов геттеров firstChild/nextSibling/children/childNodes и т.п.) приводи к неминуемым провалам в производительности, даже элементарное element.myProp = 123 — проблема. Именно поэтому, самые быстрые реализации VDOM использую максимум createElement/insertBefore/replaceChild/removeChild и только в случае крайней необходимости, некоторые даже пытаются переиспользовать ранее удаленные элементы, чтобы лишний раз не вызывать createElement.

Итого мы и имеем около мегабайта кода (несжатого ;]), который нужен только для того, чтобы быстро работать с DOM, массивами/объектами и т.п. и т.д. А ведь ещё нужно само приложение написать, да побороть кроссбраузерность и это не ІЕ7/8/9.

Так и живем.

tangro 9 марта 2016 в 11:57 # h ↑

Всё станет лучше с приходом WebAssembley. Все эти jQuery\jQuery UI\React станут нативным кодом, который будет компилироваться/кэшироваться/векторизироваться/параллелиться (и все прочие умные слова из мира C++), а сайты смогут этим пользоваться уже воспринимая их как часть API браузера, а не JS-библиотеки.

**RubaXa** 9 марта 2016 в 12:10 (комментарий был изменён) # 🗎 🚶

Ох, сколько этих разговоров было, тот же NaCL, но вериться в это с трудом, да и сколько ждать? По самым оптимистичным заявлениям, как минимум год, потом обкатка, появление первых инструментов, вопрос кроссбраузерности, поддержка мобильными браузерами, вот этих но-но-но слишком много, слишком.

tangro 9 марта 2016 в 12:19 # h ↑ +1 ↑ ↓ тот же NaCL

Совершенно не "тот же NaCL". Никому в здравом уме не могло прийти в голову делать общедоступный сайт широкого профиля на NaCL, поскольку у половины домохозяек в их левых и старых браузерах он бы не запустился. В то же время делать сайт на привычных jQuery\jQuery UI\React, который будет работать чуть медленнее на старой JS-библиотеке и чуть быстрее на современном браузере с WebAssembley — вполне себе нормальный вариант.

На счёт "минимум год" — это верно, даже скорее минимум полтора-два года. Но это единственный путь, по которому вообще можно эволюционировать, потому что альтернативы вроде "выбрасываем вообще весь HTML\DOM\JS и городим что-то кардинально новое" обречены на провал сразу.

+2 ↑ ↓ Совершенно не "тот же NaCL".

Вы меня неправильно поняли, это в качестве примера «Всё станет лучше с приходом XXX», тогда писали ровно тоже самое и вы можете его уже много лет использовать в приложениях, но больше никто это не подержал.

что говорить, что в ближайшие 3-4 года что-то измениться я бы не стал. А в промежутке компания Google хотела втюхать Dart, как решение всех проблем с

производительностью JavaScript/DOM/CSS и т.п.

Но кроме NaCL/PNaCl, ещё есть asm.js, заходы были, но не нашли хоть какого-то распространения. Так

Свой комментарий я писал к тому, что в рамках существующий браузеров, за эти 10 лет jQuery/GWT/knockout/React/Angular/XXX, можно было решить кучу проблем с производительностью, но этого не происходит.

tangro 9 марта 2016 в 17:27 # h ↑ 0 1 1 Все предыдущие подходы (NaCL, asm.js, Dart) предлагали писать **другой** клиентский код, поверив на

слово, что технология будет жить\поддерживаться\станет популярной. Люди не верили и правильно делали. Подход WebAssembley предлагает конечному веб-разработчику писать то же самое, что он и до этого писал, и лишь разработчикам библиотек и фреймворков — возможность ускорить их код в N раз. При этом даже и разработчикам библиотек не обязательно верить в успех WebAssembley если не взлетит, то всегда есть привычная JS-версия библиотеки.

**splav\_asv** 9 марта 2016 в 18:42 # 🗎 ↑ +1 ↑ ↓

asm.js собственно это подмножество js же? В худшем случае ничего не теряется. Про него можно сказать всё то же самое, что и про WebAssembly.

>любое взаимодействие с реальной нодой (вызов геттеров firstChild/nextSibling/children/childNodes и т.п.) приводи

к неминуемым провалам в производительности, даже элементарное element.myProp = 123 — проблема.

не приводит, и присваивание кастомных дом свойств — не проблема на webkit/blink(тормозные андроиды).

>Именно поэтому, самые быстрые реализации VDOM использую максимум createElement/insertBefore/replaceChild/removeChild и только в случае крайней необходимости, некоторые даже пытаются переиспользовать ранее удаленные элементы, чтобы лишний раз не вызывать createElement.

Эти функции используются каждый раз когда нужно добавить/удалить/переставить ноду, что как бы очевидно:) Не понимаю что такое «используют максимум ...», используются только в тех случаях когда иначе никак не

сделать и делается почти всегда минимальное кол-во вызовов. +1 ↑ ↓ не приводит, и присваивание кастомных дом свойств — не проблема Уверены? https://dl.dropboxusercontent.com/s/d5jtfjpaho6plxx/Screenshot%202016-03-09%2018.05.41.png?dl=0 Даже если сравнивать childNodes vs. firstChild + nextSibling, второй будет быстрей. nuit 9 марта 2016 в 18:14 # 🔓 ↑ 0 1 1 >Уверены? И зачем замерять разницу между жс объектом и дом объектом? Если вам понадобится присваивать свойство дом объекту, то у вас не будет выбора, вот и всё, и это не приведёт к катастрофическим последствиям. 0 1 1

Мы про VDOM говорим и почему он быстрей или нет? Если об этом, то его сила как раз в том, что он сводит взаимодействие с DOM к минимум, тем самым достигая выигрыша. Проверти это ещё не самое плохое, геттеры — вот где основной профит. nuit 9 марта 2016 в 18:29 # h ↑ 0 1 1 В чём смысл замерять скорость присваивания оычному жс объекту и дом ноду, если в итоге всё равно нужно будет присвоить это свойство реальному дом ноду? >Если об этом, то его сила как раз в том, что он сводит взаимодействие с DOM к минимум, тем самым достигая выигрыша. можно делать минимум взаимодействий с DOM и без виртуального дома. **RubaXa** 9 марта 2016 в 18:33 (комментарий был изменён) # 🗎 🚶 0 1 1 можно делать минимум взаимодействий с DOM и без виртуального дома. Например параллельно хранить где-то актуальные состояния, на основе которых построен DOM или что вы предлагаете? nuit 9 марта 2016 в 18:54 # 👌 ↑ 0 1 1 Например kvo датабайндинг. **RubaXa** 9 марта 2016 в 19:00 # 📙 ↑ 0 1 1 Понятно, «те же яйца, только в профиль». Только я не видел быстрого kvo, а вот быстрый vdom использую. vintage 9 марта 2016 в 20:49 # 🔓 ↑ 0 1 1 А какие кво вы пробовали? nuit 9 марта 2016 в 18:19 # 🔓 🚶 0 1 1

Что будет быстрее чего? Они делают совершенно разные вещи, зачем их сравнивать, сравнивайте скорость того что вы пытаетесь сделать с помощью этих вызовов.

>Даже если сравнивать childNodes vs. firstChild + nextSibling, второй будет быстрей.

**RubaXa** 9 марта 2016 в 18:27 # h ↑

**RubaXa** 9 марта 2016 в 18:55 # h

Какую разную? Всё тужу, обход дерева элементов, но если мы продолжаем говорить про VDOM, хотя уже понял, что нет, но если всё же говорить про него, то он с этой операцией справиться ещё быстрей, потому что вместо HTMLCollection, он пробежится по обычному массиву из обычный объектов.

0 1 1

0 1 1

0 1 1

+2 **1** 

0 1 1

0 1 L nuit 9 марта 2016 в 18:45 # h ↑

Ну так зачем обходить дерево, используя childNodes, который возвращает динамический(лайв) массив, используйте для этого nextSibling, оно как раз для этого и сделано, но зачем сравнивать разные инструменты мне так и непонятно.

vnode'ax, кому-то ещё важны immutable vnode'ы.

А про vdom — зависит от того какую реализацию vdom возьмём, не все хранят ссылки на дом ноды в

Ок, тут ваша правда, сравнивать их и в правду некорректно.

A про vdom — зависит от того какую реализацию vdom возьмём

DOM. 

Я ещё раз говорю, вся это карусель с vdom, только ради уменьшения взаимодействия с реальным

>Я ещё раз говорю, вся это карусель с vdom, только ради уменьшения взаимодействия с реальным DOM.

Ну, являясь автором одной из самых быстрых вдом либ, и общаясь с большинством разработчиков остальных вдом либ, мне всё же кажется что я имею небольшое представление о том ради чего весь этот vdom :) И для меня это в первую очередь очень удобный арі для работы с домом.

Ну так и о чем тогда разговор? Где я был неправ, говоря, что за последние 10 лет так и не появилось быстрого и/или удобного арі?

P.S. Что именно за либа, если не секрет?

nuit 9 марта 2016 в 19:22 # h ↑ 0 1 1 >Ну так и о чем тогда разговор? >любое взаимодействие с реальной нодой (вызов геттеров

firstChild/nextSibling/children/childNodes и т.п.) приводи к неминуемым провалам в производительности, даже элементарное element.myProp = 123 — проблема. не приводит, не проблема:)

https://github.com/localvoid/kivi https://github.com/localvoid/uix

не приводит, не проблема:)

>P.S. Что именно за либа, если не секрет?

Я же говорю про полную абстракцию над DOM, а это значит, что «свои» или «data» свойство, двигло может держать в объекте vnode. firstChild/nextSibling/parentNode всё равно медленней, чем не вызывать их вовсе, а ходить по vtree, вы же сами это прекрасно знаете.

https://github.com/localvoid/kivi https://github.com/localvoid/uix

померли многие либы (не запускаются бенчмарки). nuit 9 марта 2016 в 20:34 # 🔓 ↑

Знакомые названия, как и vdom-benchmark, который ой как путает людей, да и

>firstChild/nextSibling/parentNode всё равно медленней, чем не вызывать их вовсе, а ходить по vtree, вы же сами это прекрасно знаете.

Да, бегать с использованием nextSibling будет медленнее, но относительно всего остального оверхэда в конечном приложении — это будет не так сильно заметно. virtual-dom либа вроде до сих пор использует childNodes во время patch'a и ничего, куча приложений нормально работают:)

>как и vdom-benchmark, который ой как путает людей Я бы давно прикрыл этот бэнчмарк, но некоторые разработчики до сих пор

используют его для тестирования своих либ :)

Я бы должен поддержать статью и ратовать за облегчение сайтов, так как живу в пригороде в частном доме,

sim31r 9 марта 2016 в 11:22 #

https://habrahabr.ru/post/268175/

беспроводная сеть 1-2 мегабита днем и пинг 100 мс. Но нет. Основная проблема сейчас, вообще, единственный момент, по которому заметно ограничение скорости канала — видео потоки. Делаешь качество фильма 360р идет хорошо, выше, уже надо покурить/попить чаю, пока загрузится предварительный кэш. Домашние всё смотрят через интернет, телевизор оказался не нужен. Даже торренты как-то ушли в прошлое, иногда проще найти фильм в

онлайн кинотеатре, чем скаченный на своем диске. На фоне мультимедиа контента, 5 мегабайт странички вообще исчезающе малая величина. Статья, возможно, актуальна для стран, где очень дорогой интернет, у нас как-бы один из самых быстрых и дешевых на планете (второе место), хотя бы по данным замечательной статьи на Хабре

Да и сложно представить, кто-то нашел нужную ему статью, и размер в 5 мегабайт стал препятствием для получения информации. Ну только, если это не проявление специфического перфекционизма, я его понимаю, так как когда-то был обладателем модема на 2400 бит/с в сети FIDO, там да, каждый байт был на счету :) По аналогии, как автор статьи приводил лаконичность классиков, отмечу, что сайты текстовые весом даже в 5 мегабайт, так же лаконичны, по сравнению с той же информацией в форме видеороликов от стримеров. Сайт — 5 мегабайт, ролик на youtube в HD качестве может весить 5 гигабайт, а смысловое содержание примерно такое же (а иногда просто нулевое). Вот там да, глядя как стримеры некоторые забивают место на youtube терабайтами, и

потом грузят этим хламом каналы беспроводного интернета, где частоты ограниченны весьма, и возникает вопрос о том, насколько эта информация нужна, целесообразна, оптимальна. Ну и еще один момент "ожирения", высокие требования к количеству памяти на компьютере. Только по этой причине невозможно пользоваться интернетом на старых ноутбуках, где менее 1 гигабайта памяти. Браузеры под современные сайты просят памяти больше, чем есть в системе. Вот тут, наблюдаю реальные потери денег, нужно

обновлять оборудование. +1 ↑ ↓

В стране есть ещё некоторое количество суровых мест, где Интернет (если он вообще есть) идёт, разве что, через спутник по космическим ценам. И там это весьма актуально, особенно если речь идёт о бизнесприложениях.

**sapper** 9 марта 2016 в 19:39 (комментарий был изменён) # 👌 \uparrow

Я пользуюсь интернетом в телефоне в основном в транспорте. С учетом плохого покрытия 3g во многих городах

миллионниках, больше половины сайтов, которые я обычно просматриваю не могут открыться — где-нибудь проскакивает 2g, подвисает загрузка какого-нибудь скрипта и все, привет, сайт не загружается. Так что

проблема во многом актуальна, на мой взгляд. sim31r 9 марта 2016 в 19:50 # h ↑ 0 1 1 Если проскакивает 2G, то это беда, может прерваться TCP сессия по таймауту и далее множество сложностей. Как браузер, так и сервер в противоречии, рвать сессию или подождать? Для беспроводной мобильной связи лучше было бы использовать UDP протокол и разбирать пакеты более тщательно, с использованием приоритетов с учетом контекста ситуации. +3 ↑ ↓ Fuzzyjammer 9 марта 2016 в 11:50 # Ох, это все настолько точно и актуально, хочется соглашаться и соглашаться с автором под каждым абзацем! Спасибо за перевод столь объемной (по нынешним меркам, когда принято вешать 2-3 мб картинок на строку текста) altima 9 марта 2016 в 13:18 # +5 ↑ ↓ 1 Версия статьи на Хабре заняла больше 10 Мб против 1 Мб в оригинале :) **Pingdom Website Speed Test** Enter a URL to test the load time of that page, analyze it and find bottlenecks **Test Now** https://habrahabr.ru/post/278655/ Settings \* https://habrahabr.ru/post/278655/ Tested from Amsterdam, Netherlands on March 9 at 20:57:12 Perf. grade Requests Load time Page size 352 3.24s 76/100 10.8<sub>MB</sub> Your website is faster than 51% of all tested websites Post to Timeline Tweet DOWNLOAD HAR +1 ↑ ↓ atomlib 9 марта 2016 в 13:38 # 🔓 🗋 Тут превьюшки как больше, так и оптимизированы под Ретину. **23rd** 9 марта 2016 в 14:25 # 🔓 ↑ +1 ↑ ↓ Хмм. ;) gxcreator 9 марта 2016 в 13:23 # +2 ↑ ↓ В последнее время "компьютер для интернета" имеет совсем другое значение, нежели 10 лет назад. **jurok04** 9 марта 2016 в 13:27 # +5 ↑ ↓ Целое счастье, что поисковики пока не "ожирели" до невозможности. С каким же удовольствием я пользуюсь ежедневно такими страничками, как ya.ru, gmail.com. Радует, что там нет ничего лишнего, только то, зачем я туда зашел. lockywolf 9 марта 2016 в 17:00 # h 0 1 1 На ya.ru кнопка поиска не нажимается с выключенным JavaScript. **zuborg** 9 марта 2016 в 13:30 # +7 ↑ ↓ Пользователь 100Мбит/с сети в мегаполисе: • Меня это не касается, лучше фейсбук полистаю. Пользователь мобильного интернета на 128кбит/с в поселке. • А куда деваться (. Может, в следующем году оборудование новое поставят, будет быстрее, вроде ж обещали... Директор хостинга: • Большие странички? Ну хоть как-то компенсирует падение цены на бендвич, а то скоро доплачивать придется за трафик, чтобы брали хостинг у нас, а не конкурентов. Менеджер по хостингу: • Да кого волнует размер страницы, вы цены на бендвич видели? И так полностью укладываемся в тарифный пакет. Менеджер разработки проекта: • Оптимизировать размер? Нам что, время больше некуда девать. А тот единственный специалист, у которого есть опыт и знания, чтобы вручную оптимизировать html — да Вы хоть представляете, сколько стоит его время? Менеджер по рекламе: • Рекламная вставка тормозит? Так другая дохода такого не дает... А у тех, кто кликает по рекламе, каналы быстрые, у них не тормозит, вот замеры даже есть. sim31r 9 марта 2016 в 18:39 # h ↑ Я пользователь беспроводного (безлимитного) интернета в поселке, 6 км от черты города. Скорость не 128 кбит, больше в 10 раз, но и не 100 мегабит. Вообще без разницы сколько весит страничка. Уже отписывался, что основной трафик это видео файлы, да и там со сжатием и предварительным кэшированием всё хорошо. Подождать 5 секунд на открытие странички, которую потом будешь изучать 10 минут, вообще не проблема (зачем странички кликать каждую секунду?). Тем более что загрузка в основном динамическая, пока знакомишься с текстом, подгружаются картинки. Канал в 128 кбит/с (почему не 2400 бит/с, будем под такой канал подстраиваться?) это из области предельных режимов доступа к интернету, для этого есть мобильные версии сайтов и разнообразные технические ухищрения. Именно мобильный интернет, больше напрягает высочайшим пингом и разрывом ТСР сессии, когда соединение вроде бы есть, но данные уже никогда не придут. Больше UDP подойдет для таких случаев. Сам думаю переписать специфическое ПО для GSM модемов, собирающее телеметрические данные, при пинге 2000 мс, TCP соединение ведет себя как-то странно, с UDP проще. Вторая проблема пользователя, то что тяжелые странички напрягают процессор телефона или планшета, по моим наблюдениям это больше раздражает, чем медленный канал связи (главное чтобы он стабильный был).

У моих родителей беспроводный интернет (не мобильный, используется антенна) такой, что скайпом можно

только аудио пользоваться. Так что не сказал бы, что это предельная редкость.

sim31r 9 марта 2016 в 19:44 # h ↑

Скайп не может кэшировать ведеопоток на час вперед. Когда смотришь фильм с youtube, не проблема даже пропадание интернета на 10 минут, скорость может замедляться, расти, дискомфорта это не вызывает. Тем более что нет цели сидеть у экрана от начала до конца трансляции, дома полно дел что могут отвлечь от фильма, в это время идет загрузка информации. У беспроводного интернета проблема не столько скорость, сколько высокий пинг и замирания потоков информации, к которым многие программы часто не готовы. Youtube тоже не восстанавливает связь после полного разрыва связи с получением нового динамического ІР (но это и редкость, при перезагрузке модема возможно такое).

0 1 1 **zuborg** 9 марта 2016 в 19:52 # h ↑

Страничку весом в 10МБ теоретически можно загрузить на dialup-e, отвлекшись на обед, тут не поспоришь.

Но смысл? Если скайп не может нормально играть видео, то можете себе представить, как выглядит браузинг на

таких скоростях — унылое зрелище. В то время как адекватный размер страниц решил бы эту проблему. dim s 9 марта 2016 в 13:33 ##

Нашел интересное расширение для FF по этой теме, https://addons.mozilla.org/ru/firefox/addon/tab-memory-usage/, отображает потребление памяти на каждой вкладке, удобно знать, какой сайт потребляет больше всего ресурсов, ведь весь мусор кроме того как нагружает интернет канал, еще и лежит в памяти.

R Охогоп 9 марта 2016 в 13:45 # Выкиньте из отличной статьи "Кризис ожирения сайтов" все про ожирение сайтов, и она не станет хуже! Никогда

бы не подумал, что напишу подобное без сарказма. Интересно, автор просто увлекся, или это осознанный прием?

**Alexey2005** 9 марта 2016 в 14:38 # Увы, простыми призывами "перестать так делать" проблему не решить. Потому что у тех, кто выделяют деньги на

создание сайтов, никакой проблемы как раз нет. Поэтому просто не остаётся иного выхода, кроме как решать проблему там, где она мешает — в браузере

пользователя. К счастью, пока ещё пользователь может контролировать как содержимое, так и работу веб-странички, а весь "жир" и все тормоза сосредоточены в картинках и скриптах, которые можно избирательно резать. Тяжёлая артиллерия Adblock + uMatrix + NoScript + imgLikeOpera посадит любой сайт на диету. Достаточно, в принципе,

даже AdBlock+uMatrix, остальное уже на любителя. А для тех ресурсов, которые вы посещаете регулярно, можно написать юзерскрипт (или даже найти готовый), который отформатирует страничку так, как надо вам, а не дизайнеру.

Конечно, рано или поздно под давлением рекламщиков разработают DRM для сайтов, разумеется исключительно

для безопасности пользователя (ну чтоб дополнения не занимались фишингом и не крали приватную информацию). Но пока такого нет, ещё лет 5-7 можно наслаждаться нормальным вебом, пусть и с костылями в виде пачки дополнений.

Отдельный железнодорожный костыль надо забить в голову тем дезигнерам которые вкрячивают видео на первую же страничку загрузки сайта.

hungry\_ewok 9 марта 2016 в 19:39

Это доставляет неимоверную массу ощущений когда работаешь удаленно.

От Артемия Лебедева страдают сайты этой болячкой:)

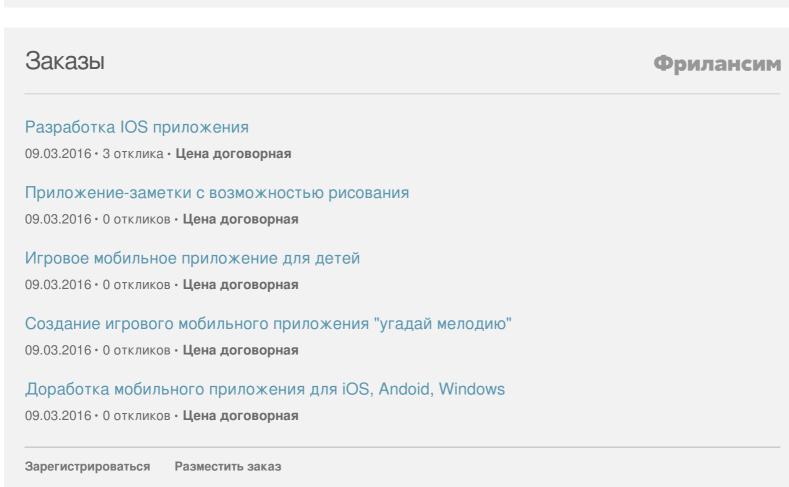
Только зарегистрированные пользователи могут оставлять комментарии. Войдите, пожалуйста.

0 1 1

+2 ↑ J

B SQL Server 2016 RC0 ■ 5
 P Релиз ownCloud 9.0 — opensource альтернативы Dropbox и другим облачным хранилищам ■ 11
 Microsoft исправила очередную Stuxnet-like уязвимость в Windows ■ 0
 M Во что превратится ваша работа за 10 лет ■ 3
 Зонтичный мониторинг ИТ-ресурсов ■ 0
 IBM и X Prize Foundation объявили конкурс по искусственному интеллекту с призовым фондом в \$5 млн ■ 0
 До Марса за три дня? ■ 60
 М Новый фаворит: научно-технологическая долина МГУ может стать российской «кремниевой долиной» вместо «Сколково» ■ 3
 М NTT Data может выкупить подразделение Perot Systems за \$3,5 млрд у Dell ■ 0
 Фреймворк Fortinet для защиты от продвинутых угроз ■ 1

# Вакансии мой круг iOS разработчик на Swift Санкт-Петербург • от 120 000 руб. Руководитель проектов / Project Manager Санкт-Петербург • Полный рабочий день • от 50 000 до 110 000 руб. Руководитель группы/Тимлид/Senior C# разработчик Санкт-Петербург • Полный рабочий день js-frontend Санкт-Петербург • Полный рабочий день Инженер-программист SharePoint Санкт-Петербург • Полный рабочий день Создать резюме Разместить вакансию



Регистрация

Войти

Разделы
Публикации
Хабы
Компании
Пользователи
Q&A

Песочница

Инфо О сайте Правила Помощь Соглашение Услуги Реклама Спецпроекты Тарифы Контент Вебинары

Тест-драйв Помощь стартапам Работа в IT

Разное

Приложения

Служба поддержки
Мобильная версия

© TM

