

## Анализ на често използвани методи за оптимизация на производителността в браузъра

Георги Димитров, Галина Панайотова, Светлозар Цанков

**Abstract:** *The article discusses the various stages of optimization of web page loading redirection, caching, DNS search, TCP connection, TCP query, TCP response, processing, and content loading. Two of the most popular sites are considered - youtube.com and facebook.com.*

**Key words:** *Web optimization, programming, caching.*

### ВЪВЕДЕНИЕ

В статията се разглежда влиянието на някои методи за оптимизация, влияещи на скоростта на зареждане на WEB страниците (пренасочване, кеширане, DNS търсене, свързване по TCP, TCP заявка, TCP отговор, обработка, зареждане на съдържание) [1]. Сред поредица изследвания Саудърс [2] дефинира “Златното правило за производителността” на уеб страниците. “80-90% of the end-user response time is spent on the frontend”. Като доказателство могат да се разгледат два от най-популярните сайтове в Интернет в момента – youtube.com и facebook.com.

### СКОРОСТ НА ЗАРЕЖДАНЕ НА СТРАНИЦИТЕ

Използвайки приставката на АпТелеметри за наблюдение на скоростта на зареждане на страниците (AppTelemetryPageSpeedMonitor) [3] се определя, че времето за пълно зареждане на началната страница на youtube.com е 1153 ms. и времето за зареждане на началната страница на социалната мрежа Фейсбук (facebook.com) е 2108 ms. Пълните данни за процеса на зареждането са представени в табл. 1.

**Табл.1. Време за зареждане на youtube.com и facebook.com**

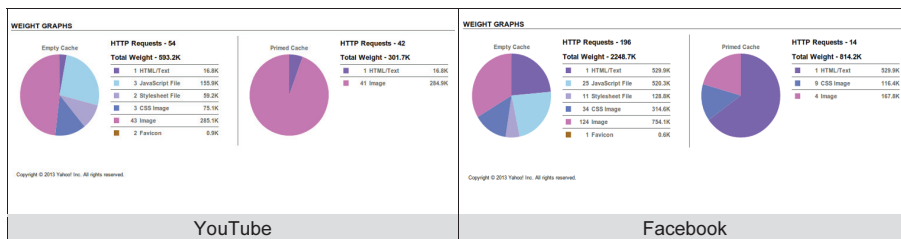
| Действие |                      | Отместване (ms) youtube | Продължителност (ms) youtube | Отместване (ms) facebook | Продължителност (ms) facebook |
|----------|----------------------|-------------------------|------------------------------|--------------------------|-------------------------------|
| 1        | Пренасочване         | 0                       | 0                            | 0                        | 0                             |
| 2        | Кеш на приложението  | 0                       | 0                            | 0                        | 0                             |
| 3        | DNS търсене          | 0                       | 0                            | 0                        | 0                             |
| 4        | Свързване по TCP     | 0                       | 0                            | 2                        | 112                           |
| 5        | TCP заявка           | 2                       | 395                          | 114                      | 170                           |
| 6        | TCP отговор          | 397                     | 86                           | 284                      | 1053                          |
| 7        | Обработка            | 408                     | 745                          | 296                      | 1812                          |
| 8        | Събитие за зареждане | 1153                    | 1                            | 2108                     | 39                            |

От получените данни личи, че липсва пренасочване, а времето за DNS търсене е сведено до нула благодарение на кеша на системата или на други DNS сървъри, намиращи се в близост. От изпращането на заявката през TCP протокола до

получаването на първия байт от отговора изминават 395 милисекунди. Това включва времето, за което заявката достига до сървъра, генерира се отговор и отговорът се изпраща обратно. Времетраенето до получаване на пълния отговор е 86 милисекунди, а времето за обработка на получения отговор е 745 милисекунди или 64,6% от времето за пълно зареждане на страницата.

### РАЗПРЕДЕЛЕНИЕ НА ЗАРЕДЕНИТЕ РЕСУРСИ ПО ТИПОВЕ

Следваща стъпка е генериране на графика в Yslow на зареденото съдържание, разделено по типове (фиг. 1) в YouTube.



**Фиг. 1. Разпределение на заредените ресурси по типове при YouTube и Facebook**

В лявата част на фиг. 1 е показано разпределението на ресурсите при първоначално зареждане, когато кешът на браузъра е празен, а отдясно зареждане при вече кеширани ресурси. При първоначалното зареждане броят на направените заявки е 54, а общото количество на трансферираните данни е 593.2 KB. Изтеглените ресурси са разделени в 6 категории: един HTML файл, 3 скрипта, 2 стилкови ресурса, 3 изображения, дефинирани като част от стиловите, 43 изображения и 2 икони. При повторното зареждане броят на заявките е 42, а реализираният трафик 301.7KB или 50.75% от първоначалния трафик. При разпределението на ресурсите по типове се вижда, че са изтеглени единствено един HTML файл и 41 изображения, а останалите ресурси са заредени от кеша на браузъра.

От диаграмата на процеса на зареждане на отделните ресурси, от сайта HTTP Archive, се вижда, че събитието, указващо зареждането на страницата настъпва 3,06 секунди след първоначалната заявка, а зареждането на допълнителните ресурси и изпълнението на скриптовете продължава още 3,98 секунди, като по този начин до пълното зареждане на страницата минават 7,02 секунди. Причината за разликата между първоначалното време за зареждане и това, показано от HTTP Archive, че зареждането на страницата се извършва от различни компютърни системи и при симулиране с различни потребители. Въпреки това ясно се вижда, че времето за отговор на сървъра представлява малка част от времето за пълно зареждане на страницата.

Тук, както и при началната страница на видео портала ЮТуб, не се използва пренасочване и времето за DNS търсене отново е сведено до нула. Времето за изграждане на връзка през TCP протокола е 110 милисекунди, което се дължи на особености на мрежата или повишено натоварване в момента на изследването. Изпращането на самата заявка започва 114 ms след началото на процеса, като отговора започва да пристига 170 ms по-късно или 284 ms след началото. Времето за пълното зареждане на отговора е 1053 милисекунди, като обработката и зареждането му се случват паралелно. Събитието за зареждане на страницата се инициира 2,108 секунди след началото на запитването и продължава 39 милисекунди. Времето за обработка в този случай представлява 85,6% от времето

за пълно зареждане. Разпределението на заредените ресурси по типове е показано на фиг.1.

При разглеждане на типовете на заредените ресурси ясно се вижда ефекта от използването на методите за кеширане. При повторно зареждане броят на заявките към сървъра е 14 или само 7,1% от първоначалните 196. Освен броят на заявките е повлиян и генерирания трафик за зареждане на страницата. При първоначалното отваряне той е с размер 2248,7 KB, а при повторно 814.2 KB или с 63.8% по-малко. Ясно се вижда, че при използването на кеша, скриптовите ресурси, стиловете и основната част от изображенията биват изтеглени от кеша на браузъра, а не от сървъра.

### ВРЕМЕ ЗА ЗАРЕЖДАНЕ, ПРОЦЕС НА ЗАРЕЖДАНЕ В КЛИЕНТСКАТА ЧАСТ

И при двата сайта в диаграмата, показваща последователността на ресурсите се наблюдава, че зареждането на допълнителните ресурси започва след пълното зареждане на HTML страницата. Също така ясно се вижда паралелното изтегляне на допълнителните ресурси. Използвайки диаграмите на последователността може да се определи каква част от времето за зареждане представлява процесът на зареждане в клиентската част. За целта се разглежда процентното отношение между времето за зареждане на HTML страницата, спрямо времето за пълно зареждане. Получените проценти показват времето, прекарано в изчакване на сървъра да генерира отговора за съответната заявка, а останалото време е прекарано при потребителя. Изчислението на това отношение за YouTube.com е показано както следва:

$$\frac{491}{7040} * 100 \approx 7\%$$

Т.е. 7% от времето за зареждане са прекарани на сървъра, а останалите 93% при потребителя. По аналогичен начин се пресмята и процентното време за зареждане на facebook.com:

$$\frac{508}{2830} * 100 \approx 18\%$$

Резултатът показва, че 18% от времето са прекарани на сървъра и съответно 82% при потребителя. Тези данни потвърждават правилото на Саудърс [2], че основната част от времето за зареждане бива прекарана при потребителя, а не на сървъра.

### РАЗПОЛОЖЕНИЕ НА СТИЛОВЕ И СКРИПТОВЕ

За да се докаже важноста на разположението на стиловете и скриптовете, е разглеждано зареждането на примерна страница, която съдържа връзки към 7 ресурса – два скрипта, два стилни ресурса и 3 изображения. На фигура 3 е показана диаграма на последователното им зареждане.

| URL               | Status | Domain                | Size          | Remote IP                    | Timeline |
|-------------------|--------|-----------------------|---------------|------------------------------|----------|
| ⊕ GET /tests/     | 200 OK | folder                | 421 B         | 127.0.0.1:80                 | 1ms      |
| ⊕ GET jquery.     | 200 OK | ajax.googleapis.com   | 77.7 KB       | 173.194.70.95:80             | 551ms    |
| ⊕ GET jquery.     | 200 OK | ajax.googleapis.com   | 105.1 KB      | 173.194.70.95:80             | 628ms    |
| ⊕ GET screen      | 200 OK | developers.google.com | 41.2 KB       | 173.194.39.198:443           | 86ms     |
| ⊕ GET table.c     | 200 OK | ajax.googleapis.com   | 923 B         | 173.194.70.95:443            | 608ms    |
| ⊕ GET screen      | 200 OK | cdn.seomoz.org        | 22.9 KB       | 54.240.162.18:80             | 172ms    |
| ⊕ GET google      | 200 OK | cdn.seomoz.org        | 9.7 KB        | 54.240.162.18:80             | 174ms    |
| ⊕ GET link-an     | 200 OK | cdn.seomoz.org        | 11.2 KB       | 54.240.162.18:80             | 227ms    |
| <b>8 requests</b> |        |                       | <b>269 KB</b> | <b>1.64s (onload: 1.68s)</b> |          |

Фиг. 3. Диаграма на зареждане на ресурсите в YouTube

Синята линия показва момента, в който браузърът вече е генерирал DOM дървото, а червената крайния момент в който всички елементи от страницата са изрисувани на екрана на браузъра. Както се вижда изтеглянето на стиловете и скриптовете става едновременно, но след това има период, в който изтеглянето на нови ресурси е блокирано от изпълнението. Едва след изпълнение на скриптовете започва изтеглянето и зареждането на изображенията.

Тук се вижда, че зареждането на всички ресурси – стилове, скриптовете и изображенията започва почти едновременно т.е. блокирането на зареждането е избегнато. Забелязва се също така и промяна във времето за зареждане на ресурсите и страницата. В първия случай изтеглянето на ресурсите продължава 1,64 секунда, а във втория 0,935 или с близо 43% по-малко. Времето за пълното зареждане намалява от 1,68 секунди до 1,53 или с 9%. Важно е тук да се има предвид факта, че макар и пълната скорост за зареждане да е намалена само с 9%, у потребителя се създава усещането за значително по-бърза работа поради факта, че стиловете и изображенията биват заредени и визуализирани преди края на обработката и изпълнението на скриптовете.

Лесно могат да бъдат определени ползите от използването на минимизацията на скриптове и стилове. За целта са разгледани размерите на оригиналната и минимизираната версия на две от най-използваните скриптов библиотеките (табл.3).

**Таблица 3. Сравнение на размера на скриптове преди и след минимизиране**

| Име/версия       | Размер на оригинала (KB) | Размер след минимизиране (KB) |
|------------------|--------------------------|-------------------------------|
| jQuery/1.8.3     | 77.7                     | 32.7                          |
| jQuery UI/1.10.2 | 105.1                    | 59.3                          |

Процентното намаление на двете библиотеки е съответно 57,9% и 43,6%, а в случай на комбинирана употреба общият процент на редуциране е 49,7. Това показва, че размерът на използваните скриптове може да бъде намален наполовина, без това да доведе до промяна на функционалността или загуба на възможности.

### **ПРАВИЛА, ДЕФИНИРАНИ ОТ САУДЪРС ЗА МИНИМИЗАЦИЯ**

Като пример се изследва HTML темплейт, прилагат се последователно правилата, дефинирани от Саудърс [9] с цел установяване на точния им ефект върху страницата.

Данните за изменението на размера на страницата и времето за зареждане след последователно прилагане разглежданите методи са представени по-долу в таблица 4.

\* Добавянето на “Expires” тага не променя времето за презареждане на страницата, но води до промяна в скоростта при навигация между отделните страници в рамките на домейна.

\*\* Едва след добавянето на Last-Modified се изравнява времето за презареждане на страница и това при навигация обратно към нея.

В нулевата стъпка от изследването, когато всички механизми за кеширане и компресия са изключени, размерът на страницата и всички използвани ресурси е 790.5 KB, а времето за зареждане 1.82 секунди, като поради липсата на кеширане не се наблюдава разлика между скоростта и размера на страницата при първоначално и следващо зареждане. В края на експеримента, след прилагане на методите, които типът на страницата позволява отчитаме, че времето за първоначално зареждане е 1,28 секунди, а размерът на страницата и всички прилежащи ресурси е 683,5 KB. При последващо зареждане на страницата размерът на страницата е намален до

1,1 KB, което представлява размерът на HTML страницата, а всички останали ресурси са запазени в кеша на браузъра и се изтеглят от там. Времето за повторно зареждане е намалено до 1,24 секунди. Сравнение на стойностите преди и след оптимизация е представено в табл. 5.

**Таблица 4. Изменение на размера и времето за зареждане на страница след оптимизация**

|                                                   | Първоначално зареждане    |                        | Последващи зареждания     |                        |
|---------------------------------------------------|---------------------------|------------------------|---------------------------|------------------------|
|                                                   | Размер на страницата (KB) | Време за зареждане (s) | Размер на страницата (KB) | Време за зареждане (s) |
| Намаляване на броя на HTTP заявките               | 790.5                     | 1.82                   | 790.5                     | 1.82                   |
| Използване на Content Delivery Network (CDN)      | 758.7                     | 1.41                   | 758.7                     | 1.41                   |
| Използване на GZip компресия за ресурсите         | 758.7                     | 1.41                   | 1.1                       | 1.41 / 1.24 *          |
| Разполагане на стиловете в началото на страницата | 746.4                     | 1.36                   | 1.1                       | 1.41 / 1.24            |
| Избягване на скриптови изрази в стиловите ресурси | 746.4                     | 1.36                   | 1.1                       | 1.41 / 1.24            |
| Минимизиране на скриптовете                       | 746.4                     | 1.32                   | 1.1                       | 1.41 / 1.24            |
| Избягване на използването на пренасочвания        | 683.5                     | 1.28                   | 1.1                       | 1.41 / 1.24            |
| Кеширане при AJAX заявките                        | 683.5                     | 1.28                   | 1.1                       | 1.24 / 1.24**          |

**Таблица 5. Сравнение на размера и времето за зареждане на страница преди и след оптимизация**

|                   | Първоначално зареждане    |                       | Последващи зареждания    |                       |
|-------------------|---------------------------|-----------------------|--------------------------|-----------------------|
|                   | Размер на страницата (KB) | Време за зареждане(s) | Размер на страницата(KB) | Време за зареждане(s) |
| Преди оптимизация | 790.5                     | 1.82                  | 790.5                    | 1.82                  |
| След оптимизация  | 683.5                     | 1.28                  | 1.1                      | 1.24 / 1.24           |
| Разлика(%)        | 13,5                      | 29,7                  | 99,7                     | 31,9                  |

След оптимизацията се наблюдава 29,7% разлика в скоростта на зареждане спрямо началното състояние. Причината за малката разлика от 2,2% между скоростта за първо (29,7%) и последващо зареждане (31,9%) се дължи на множеството използвани JavaScript библиотеки в страницата, а времето за тяхната обработка и изпълнение не може да бъде намалено без промяна на програмния код на самите библиотеки. По отношение на размера на страницата, при първо зареждане той е намален с 13,5% процента, но при последващи посещения разликата спрямо началното състояние е 99,7%. Броят на заявките, необходими за

пълното зареждане на страницата преди оптимизацията е 70, а след оптимизацията 36 или с 48.6% по-малко.

### **ЗАКЛЮЧЕНИЕ**

Получените данни недвусмислено показват ефекта от оптимизацията, както върху скоростта на зареждане за потребителите, така и върху работата на сървъра – намаленият брой заявки и генериран трафик позволяват на сървъра да обслужва по-голям брой потребители едновременно, без това да се отрази на общата скорост.

### **ЛИТЕРАТУРА**

- [1]. King, Andrew B. Website Optimization, O'Reilly Media; July 2008
- [2]. Souders, Steve Even Faster Web Sites: Performance Best Practices for Web Developers, O'Reilly Media; June 2009
- [3]. Младенов, Огнян, Юли 2012, <http://www.slideshare.net/oggin/ss-13559485>

### **За контакти:**

Проф. д-р Георги Димитров, Университет по библиотекарство и информационни технологии, София, e-mail: [geo.p.dimitrov@gmail.com](mailto:geo.p.dimitrov@gmail.com)

Проф. д-р Галина Панайотова, Университет по библиотекарство и информационни технологии, София, e-mail: [panayotovag@gmail.com](mailto:panayotovag@gmail.com)

Гл. ас. д-р Светлозар Цанков, Катедра *Информатика и информационни технологии*, Русенски университет *Ангел Кънчев*, тел.: 082-888 645, e-mail: [stzancov@ami.uni-ruse.bg](mailto:stzancov@ami.uni-ruse.bg)