

Мацевский Н.С.

Реактивные веб-сайты. Клиентская оптимизация в алгоритмах и примерах : учебное пособие / Н.С. Мацевский, Е.В. Степанищев, Г.И. Кондратенко. — М.: Интернет-Университет Информационных Технологий : БИНОМ. Лаборатория знаний, 2010. — 335 с: ил., табл. — (Архитектор информационных систем).

# ОГЛАВЛЕНИЕ

5	<b>ВВЕДЕНИЕ</b>
7	Об авторах
7	Благодарности
6	Как ускорить работу сайта для посетителя
I 1	<b>Глава 1. ОБЗОР МЕТОДОВ КЛИЕНТСКОЙ ОПТИМИЗАЦИИ</b>
II	1.1. Клиентская оптимизация
1 2	1.2. Анализ веб-страниц
20	1.3. Уменьшение размера загружаемых объектов
28	1.4. Уменьшение количества запросов
40	1.5. Увеличение скорости отображения веб-страниц
43	1.6. Оптимизация структуры веб-страниц
49	<b>Глава 2. АЛГОРИТМИЗАЦИЯ СЖАТИЯ ТЕКСТОВЫХ ФАЙЛОВ</b>
49	2.1. Методы сжатия, поддерживаемые браузерами
52	2.2. Проблемы в браузерах, прокси-серверах и firewall
55	2.3. Настройка веб-серверов Apache, nginx и lighttpd
65	2.4. Собственная реализация сжатия со стороны сервера
71	2.5. Альтернативные методы сжатия
75	<b>Глава 3. АЛГОРИТМЫ УМЕНЬШЕНИЯ ИЗОБРАЖЕНИЙ</b>
76	3.1. Уменьшаем GIF (Graphics Interchange Format)
82	3.2. Оптимизируем JPEG (Joint Photographic Experts Group)
92	3.3. Оптимизируем PNG (Portable Network Graphics)
107	3.4. Оптимизируем SVG (Scalable Vector Graphics)
1 14	3.5. Средства онлайн-оптимизации
I 1 9	<b>Глава 4. УМЕНЬШЕНИЕ КОЛИЧЕСТВА ЗАПРОСОВ</b>
II 9	4.1. Автоматическое объединение текстовых файлов
1 27	4.2. Алгоритм разбора и сбора CSS Sprites
135	4.3. CSS Sprites и data:URI, или Microsoft и весь остальной мир
1 38	4.4. Автоматизация кроссбраузерного решения для data:URI
142	4.5. Автоматизация кэширования

1 56	<b>Глава 5. ОПТИМИЗАЦИЯ СТРУКТУРЫ ВЕБ-СТРАНИЦ</b>
1 56	5.1. Динамические стили: быстро и просто
160	5.2. Оптимизация CSS-структуры
167	5.3. Пишем эффективный CSS
171	5.4. Стыкуем асинхронные скрипты
177	5.5. Стыкуем компоненты в JavaScript
181	5.6. Что такое CDN и с чем его едят
1 85	5.7. Практическое использование CDN на примере Google Apps
1 92	<b>Глава 6. ТЕХНОЛОГИИ БУДУЩЕГО</b>
1 93	6.1. Профилируем JavaScript
1 97	6.2. Проблемы при оценке производительности браузеров
208	6.3. Перспективы «быстрого» JavaScript
214	6.4. Реализация логики C553-селекторов
220	6.5. API для CSS-селекторов в браузерах
224	6.6. Canvas: один шаг назад два шага вперед
234	6.7. Вычисляем при помощи Web Workers
239	6.8. Клиентские хранилища
251	<b>Глава 7. АВТОМАТИЗАЦИЯ КЛИЕНТСКОЙ ОПТИМИЗАЦИИ</b>
251	7.1. Обзор технологий
256	7.2. Установка Web Optimizer
263	7.3. Настройка Web Optimizer
269	7.4. Примеры использования Web Optimizer
272	7.5. Решаем проблемы с установкой Web Optimizer
279	<b>Глава 8. ПРАКТИЧЕСКОЕ ПРИЛОЖЕНИЕ</b>
279	8.1. Разгоняем ASP .NET: 100 баллов и оценка «А» в YSlow
285	8.2. Разгоняем Drupal
300	8.3. Разгоняем Wordpress
304	8.4. Разгоняем Joomla! 1.5
308	8.5. Разгоняем Joostina
315	8.6. Пара советов для Ruby on Rails
318	8.7. Разгоняем jQuery
328	8.8. Клиентская оптимизация для произвольного сайта
335	<b>ЗАКЛЮЧЕНИЕ</b>
335	В качестве послесловия