



Школа Web-мастерства

Как создать
классный сайт
с минимальны-
ми расходами

Что можно сде-
лать бесплатно,
а что — очень
дешево

Как сэкономить
не только де-
ньги, но и свое
драгоценное
время

Какие этапы
проекта могут
погубить еще
не доделанный
сайт

Не надо делать
много! Лучше
меньше, но хо-
рошо

КЭРРИ БИКНЕР

ЭКОНОМИЧНЫЙ WEB-ДИЗАЙН

NT
PRESS

New
Riders

Кэрри Бикнер

**Экономичный
Web-дизайн**

Кэрри Бикнер

Экономичный Web-дизайн

Школа Web-мастерства

NT Press
Москва, 2005

УДК 004.738.5

ББК 32.973.26-018.2

Б60

Подписано в печать 16.03.2005. Формат 70х90 мм. Гарнитура «Баскервиль». Печать офсетная. Усл. печ. л. 17,98. Тираж 5000 экз. Зак. № 536)

Бикнер К.

Б60 Экономичный Web-дизайн / Кэрри Бикнер ; Пер. с англ. Д. С. Ремизова. - М. : НТ Пресс, 2005. - 248 с. : ил. - (Школа Web-мастерства).

ISBN 5-477-00076-7

Эта книга о том, как создать профессионально выглядящий Web-сайт при ограниченных финансовых возможностях и как применить принципы малобюджетного проекта к каждому аспекту Web-производства: управлению сайтом, созданию интерфейса, дизайну, написанию текста, хостингу и поддержке сайта после запуска. Книга предназначена для тех, кто экономит не только свои средства, но также и свое время, кто стремится разработать качественный сайт за короткие сроки.

Authorized translation from the English language edition, entitled WEB DESIGN ON A SHOESTRING, 1st Edition, ISBN 0735713286, by BICKNER, CARRIE, published by Pearson Education, Inc, publishing as New Riders, Copyright © 2004.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc. RUSSIAN language edition published by NT PUBLISHING HOUSE, Copyright © 2005.

УДК 004.738.5

ББК 32.973.26-018.2

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельца авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но, поскольку вероятность технических ошибок все равно остается, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможный ущерб любого вида, связанный с применением содержащихся здесь сведений.

ISBN 0-735-71328-6 (англ.)

ISBN 5-477-00076-7 (рус.)

Copyright © New Riders, 2004
© Издание на русском языке,
перевод на русский язык,
оформление НТ Пресс, 2005

Оглавление

Введение	8
ЧАСТЫ. Производство	19
Глава 1. Секреты успешного малобюджетного проекта	21
Журнал Astronomy Education Review.....	22
Лэй Бейкер-Фoley.....	24
Из чего состоит успешный малобюджетный сайт.....	27
Используйте подручные средства.....	28
Количество людей, принимающих решения, не должно быть большим.....	28
У сайта должна быть общая идея.....	29
Лучше сделать меньше.....	30
Глава 2. Экономичный план проекта	31
Польза от хорошо продуманного плана.....	32
Ценные советы.....	34
Задачи малобюджетного проекта.....	34
Задачи проекта для сайта Something Blue.....	34
Определите функциональные требования заблаговременно.....	36
Утвердите технические требования.....	40
Преимущества тщательного планирования.....	41
Глава 3. Разработка интерфейса	43
Экономичное тестирование.....	45
Проводите тестирование с самого начала работы.....	45
Малобюджетный инструментарий для создания интерфейса.....	50
Советы по меню и навигации.....	51
Пусть в вашем окне всегда горит свет.....	51
Правило трех щелчков.....	51
Живой и понятный язык.....	51
Правило пяти пунктов меню.....	52
Многоразовая HTML-библиотека.....	52
Разметка: простота использования и доступность Web-сайта.....	53
Глава 4. Почему хороший текст имеет большое значение	61
Составляющие хорошего текста.....	63
Пишите проще и понятнее.....	64
Разделите текст на маленькие части и чаще используйте заголовки.....	68

6 Экономичный Web-дизайн

Пишите о пользователе, а не о себе.....	71
Текст должен быть простым, но не примитивным.....	73
Избегайте грамматических ошибок.....	75
Бесплатные и недорогие элементы страницы.....	76
Тег title.....	77
Ссылки.....	80
Атрибут title.....	80
Подписи в электронных письмах.....	81
«Ошибка 404, страница не найдена» и другие сообщения об ошибках.....	82
Ресурсы.....	85
Глава 5. Как хорошо выглядеть с минимальными затратами ...	87
Принципы оформления текста.....	88
Количество шрифтов.....	91
Последовательное использование шрифтов.....	93
Размещение текста.....	95
Отображение шрифтов в браузерах.....	96
Экономьте время и деньги при помощи CSS.....	99
Экономия сCSS: пример из жизни.....	102
Цвет: меньше значит больше.....	104
Цвет: проще при помощи CSS.....	106
Рисунки и фотографии.....	107
Будьте осторожны с использованием клипартов.....	108
Как найти недорогой клипарт.....	109
Как найти подходящие фотографии и рисунки.....	115
Как из плохой графики получить приемлемый вариант.....	120
ЧАСТЬ II. Инструменты.....	125
Глава 6. Управление сайтом при небольшом бюджете.....	127
Системы управления сайтом.....	128
Не платите за то, что вам не нужно.....	131
Нужно ли вам контролировать план страницы при помощи шаблонов?.....	131
Возможности для написания и редактирования содержания.....	136
Нужна ли вам поддержка XHTML и CSS?.....	138
Нужны ли вам проверка разных версий содержания и управление рабочим процессом?.....	139
Нужна ли вам возможность работы с базой данных?.....	140
Нужны ли вам области для подготовки и для производства?.....	140
Нужен ли вам механизм RSS?.....	142
Лучшие CMS для вашего бюджета.....	145
CMS с открытым кодом.....	148
CMS-решения с открытым кодом на уровне предприятия.....	150
Открытые CMS меньших размеров.....	156

Коммерческие CMS.....\.	165
Не совсем CMS.....	168
Последний по порядку, но не по важности совет.....	171
Глава 7. Сэкономьте время и деньги благодаря	
Web-стандартам.....	173
Каким образом Web-стандарты помогут вам сэкономить.....	174
Часть I: Снижение расходов на производство.....	175
Хорошая разметка: разумные расходы.....	178
Почему презентационная разметка усложняет работу в Сети.....	179
Хорошая разметка облагораживает Сеть.....	180
Урок по разделению структуры и внешнего вида.....	181
Пользовательские агенты.....	185
Доступность Web-сайта достижима.....	188
Создайте сайт с хорошей разметкой.....	189
Механика хорошего XHTML-кода.....	189
Часть II: Выгода.....	194
Шаг 1: Чистка разметки.....	194
Шаг 2: Создание таблиц стилей.....	203
Шаг 3: Быстрое и простое изменение дизайна.....	213
Шаг 4: Создание таблицы стилей для печати.....*	215
XHTML- и CSS-ресурсы.....!	218
Глава 8. Экономичные домены и хостинг.....	219
Цена вашего доменного имени.....	223
Это звучит слишком громко, чтобы быть правдой.....	223
Надо быть организованным.....	223
Решение: выбор регистратора доменного имени.....	224
Экономный хост.....	226
Стоимость трафика и места на диске.....	229
Просто дайте мне дешевый хостинг.....	237
Предметный указатель.....	241

*Эта книга посвящается моим родителям
(они принимают активное участие в борьбе с раком),
моей сестре и, конечно же, моему любимому Джеффри.*

Введение

Для того чтобы создать хороший сайт, большой бюджет не нужен.

Сразу следует отметить, что в работе с небольшим бюджетом есть много преимуществ. Хорошо отлаженное малобюджетное производство позволит вам сфокусироваться на том, что действительно необходимо. В результате получится простой и более действенный сайт.

Эта книга о том, как определить правильный подход для создания отличных сайтов при небольшом бюджете. Я называю этот подход *малобюджетным дизайном*. В главах этой книги я покажу вам, как применить принципы малобюджетного проекта к каждому аспекту Web-производства: управлению проектом, созданию интерфейса, дизайну, написанию текста, хостингу и поддержке сайта после запуска.

В этой книге я поделюсь с вами советами, которые не только помогут вам пережить экономически трудные времена, но и улучшат качество и легкость вашей работы, а также помогут лучше понять потребности людей, читающих и использующих ваш сайт. Когда ваши материальные условия улучшатся, вы уже будете обладать качествами, необходимыми для эффективной работы, и сможете успешно работать с самым большим бюджетом.

Зачем нужен малобюджетный дизайн

Я хотела написать эту книгу, и у меня на это была личная причина. В дни расцвета эры Интернета я работала в качестве Web-разработчика в большой нью-йоркской некоммерческой организации. Время от времени нам поступало финансирование в очень скромных количествах. В 1999 году эти скудные средства превратились в постоянно поступающий денежный поток. Мне казалось, что сбылась мечта любого Web-дизайнера, работающего при небольшом бюджете. Мы потратили эти деньги на расширение штата работников, который увеличился с 2 до 10; пригласили проектировщиков структуры сайта, графических

дизайнеров, редакторов, писателей (какая роскошь!) и нескольких помощников. За два года мы собирались выполнить серию крупномасштабных проектов и создать студию тестирования интерфейса сайтов. Я работала с большими системами управления сайтом (CMS ~ content management systems) и надеялась, что у меня появятся деньги, чтобы приобрести все необходимое для выполнения работ на новом уровне. Мы все хорошо спланировали и думали, что наш бюджет готов к любым, даже неожиданным, затратам. Нам казалось, что мы сможем грамотно потратить деньги.

Однако из-за непредвиденных обстоятельств на фондовой бирже деньги, предназначенные для оплаты работы нашей новой великолепной команды, пришлось потратить на то, чтобы спасти фирму. Несколько месяцев спустя на Нью-Йорк обрушился очередной финансовый кризис и денег на создание сайта не стало. Было ощущение, что все мои надежды на великолепный Web-сайт разрушены.

Ирония состоит в том, что за следующий трудный с материальной точки зрения год мы вдвоем, закрывшись в офисе, смогли сделать гораздо больше, чем я ожидала вначале. При скудном бюджете нам пришлось выработать соответствующее отношение к работе.

Как только мы сфокусировались на добросовестном выполнении небольших задач и на четкой организации работы, вопросы, связанные с нашим сайтом, постепенно стали разъясняться. Мы научились обходиться без подробных исследований, извлекая максимум информации из самого малого. Более того, мы очень внимательно прочитывали каждый комментарий относительно нашего сайта* Мы пересматривали старые страницы, от которых отказались ранее, пытались улучшить язык сайта, делая его более живым.

Мы решили не покупать систему управления сайтом и создали маленькую систему, которая поддерживала несколько разделов нашего сайта. Мы разработали более эффективный метод работы. Мы перестали устраивать бесполезные встречи с персоналом и прекратили работу над долговременными проектами.

Мы принялись за новые проекты, но теперь были более избирательны: то, что мы делали, мы делали хорошо. Мы составляли документацию, изучали проект, пользовались распространенными стандартами, овладевали новыми навыками. После года работы подобным образом наш сайт стал более совершенным и полезным для клиентов, чем в период нашего финансового благополучия.

Именно в этот момент я поняла, что нашла необходимый подход к работе с небольшим бюджетом, и тогда я начала писать эту книгу. Что-то в этой книге вы найдете из моего собственного опыта, а что-то я почерпнула из бесед со

специалистами по разработке малобюджетных сайтов. Надеюсь, что все собранные здесь советы помогут Web-мастерам справиться со многими трудностями.

Читая лекции по Web-дизайну, я поняла, что многие разделяют мои взгляды на технику создания малобюджетных сайтов и сами работают над их созданием. Некоторые из нас работают на агентства или некоммерческие организации дома. Другие работают с теми же самыми агентствами или организациями, но уже в качестве клиентов. Чем больше я встречала таких людей, тем сильнее мне хотелось написать книгу, способную послужить гидом по созданию великолепного сайта при небольшом бюджете.

Для кого предназначена эта книга

Некоторые Web-разработчики работают, используя такие же средства, какие были у нас три-четыре года назад, однако наши финансовые возможности до сих пор не изменились. Эта книга предназначена для тех из нас, кто видел, как исчезает бюджет по мере роста требований заказчиков. В сущности, эта книга для всех, кому приходится создавать отличные сайты за копейки. Таких людей много: некоторые из нас - фрилансеры (специалисты без постоянного места работы), кто-то работает в маленьких агентствах, а кто-то работает дома на крупные компании. Когда я писала эту книгу, я думала обо всех профессионалах, принадлежащих к вышеперечисленным категориям. Вот четыре представителя низкооплачиваемых Web-профессионалов.

Эта книга адресована Джону, для которого создание Web-сайтов не является основным занятием, но который вынужден делать сайты на высшем уровне. Джон не дизайнер, однако ему было поручено сделать красивый и надежный сайт своей компании. В течение ближайших шести месяцев ему необходимо основательно обновить сайт так, чтобы его библиотека выглядела более современно. На сайте есть календарь событий, однако на данный момент он обновляется вручную. Это отнимает у него слишком много времени. Джон хочет купить, найти или создать автоматический инструмент, благодаря которому его коллеги смогли бы вводить данные в Web-форму, а новые события в календаре автоматически появлялись бы на сайте. Его начальник хочет обсуждать с другими пользователями разные книги на этом сайте, для этих целей Джон должен сделать форум. Джон работает Web-мастером только 17 часов в неделю, а в остальное время - в детской библиотеке.

Основная проблема Джона: как уложиться в срок и выполнять максимальное возможное количество работы в отведенные для Web-дизайна часы?

Эта книга для Клинта - владельца собственной фирмы, занимающейся Web-дизайном в Онтарио. Он работает с клиентами, которым интересен сложный дизайн, а не разработка Интернет-приложений. В течение нескольких лет дела Клинта шли хорошо, однако за последние два года количество клиентов уменьшилось. Некоторые из них занимались бизнесом, но оставили свое занятие. Клиенты, которых Клинту удалось удержать, платят немного; поэтому пришлось заказы стоимостью в \$20000 сменить на заказы в \$5000, а десяти тысячные проекты - на проекты стоимостью в \$500. Клинт очень любит свою работу и хочет создавать прекрасные сайты для своих клиентов, поэтому ему надо изменить подход к делу. Он хотел бы узнать, как можно делать красивые сайты за небольшие деньги. Он боится, что, если он и его клиенты потратятся не на те вещи, которые действительно нужны, слишком много денег будет потрачено зря и сайт может быть не окончен.

Основная проблема Клинта: как можно помочь своим клиентам грамотно израсходовать их небольшой бюджет, чтобы при этом еще и самому суметь заработать?

Эта книга для Джанет, которая работает в Интернет-отделе страхового агентства в Фениксе, штат Аризона. Она окончила школу искусств и когда-то была ведущим дизайнером в команде, состоявшей из нескольких разработчиков, специалиста по Flash и менеджера проекта. После сокращения в офисе остались лишь Джанет, которая теперь отвечает за все аспекты Web-производства, а также ассистент, который не имеет опыта Web-разработки.

Основная проблема Джанет: как в одиночку можно выполнять работу за пятерых?

Эта книга для Стива, владельца компьютерного магазина на окраине Толедо, штат Огайо. Он пообещал своей тринадцатилетней дочери создать, разместить и поддерживать сайт по бейсболу. Он не беден и очень хочет добиться своей цели. Если сайт выйдет неудачным, он уверен: все будут говорить об этом. К несчастью для Стива, у него нет ни плана, ни бюджета, ни времени, чтобы выполнить эту работу.

Основная проблема Стива: как сделать потрясающий сайт в свободное время?

Также эта книга для тех, кто хочет на добровольных началах делать сайты различных социальных и религиозных организаций. Она для всех, кто занимается созданием Web-сайтов, но ограничен во времени и в средствах. Книга предназначена также для тех, кто обладает достаточным бюджетом и хочет выжать по максимуму из каждого цента.

Предполагается, что у вас есть какой-то, пусть даже небольшой, опыт создания и размещения Web-сайтов. Для понимания некоторых глав требуются базовые знания по (X)HTML. Если ваши знания по HTML ограничены (потому что вы никогда не делали Web-сайты или работали исключительно с графическими редакторами, такими как Dreamweaver или GoLive), по мере чтения книги вам придется дополнительно изучить основы HTML. Возможно, при ознакомлении с этим изданием вам понадобится книга по языку HTML, например «DHTML и CSS». — 3-е изд., испр. и доп., ИТ Пресс.

Структура издания

Это издание охватывает все аспекты Web-производства с точки зрения профессионала, которому необходимо сделать великолепный сайт, но который располагает скудными финансовыми ресурсами. В каждой главе рассматривается один из аспектов создания сайта (планирование проекта, тестирование, написание текстов, дизайн, система управления сайтом, HTML-разметка и Web-хостинг) и предлагаются различные стратегии, как выжать максимум из маленького бюджета.

Я постаралась сфокусироваться на одной-двух экономичных стратегиях, а затем разбила их на несколько шагов, которыми вы можете незамедлительно воспользоваться.

В главе 1 максимально четко очерчивается круг задач. Это поможет вам не сбиться с пути в процессе работы.

В главе 2 дается совет не бояться делать меньше. При подготовке проекта слова «меньше значит больше» должны стать мантрой Web-профессионала, работающего при небольшом бюджете. Я рассматриваю способы, позволяющие упростить план вашего проекта и избежать ситуаций, которые, будучи неучтенными, могут вас разорить.

Глава 3 помогает сосредоточиться на том, как извлечь выгоду из обычных тестирований сайта, постоянно проводимых с самого начала работы над ним, а также о том, как создать инструментарий для создания интерфейса, который очень вам пригодится в дальнейшей профессиональной деятельности. Для того чтобы сайт был прост в использовании, многого не надо.

Глава 4 должна убедить вас в том, что экономные профессионалы, работающие с небольшим бюджетом, должны извлекать выгоду из хорошего текста. Слова - единственная составляющая сайта, которую вы можете полностью контролировать, и они могут положительно либо отрицательно сказаться на развитии вашего сайта. Хороший текст также может пригодиться для рассылок и

прочих бесплатных или недорогих инструментов маркетинга. Я рассматриваю различные способы написания хорошего текста для сайта, которые не потребуют больших финансовых затрат и научат вас, как извлечь максимальную пользу из каждого слова.

Глава 5 повествует о качественном дизайне при небольшом бюджете. Возможно, у вас нет средств, чтобы нанять первоклассных дизайнеров, однако, если вы продумаете стиль и будете ему следовать, четко определите цвета, линии и разметку текста, вы сможете создать недорогой сайт, который будет выглядеть на миллион долларов.

Даже работающим при небольшом бюджете Web-профессионалам придется заниматься управлением сайта. Управление содержимым вручную может дорого обойтись, однако система управления сайтом тоже может стоить дорого. В главе 6 я помогу вам найти наилучшую CM-систему за копейки или вообще бесплатно.

Профессионал, работающий при низком бюджете, должен экономить время и деньги на поддержке сайта, его хранении, расходах на хостинг и даже на доступности. В главе 7 я рассматриваю основы хорошо продуманного, надежного Web-сайта, который в дальнейшем сэкономит вам много денег.

Итак, вы закончили работу над сайтом. Теперь необходимо его разместить. В главе 8 я научу вас, как найти недорогой хостинг и завести доменное имя, а также как защититься от «скрытой оплаты», которая часто встречается у компаний, занимающихся предоставлением доменов и хостингом.

Как пользоваться этой книгой

Надеюсь, что в этой книге появятся ваши пометки, загнутся уголки некоторых страниц, обложка будет забрызгана кофе. Не относитесь к ней с излишним трепетом. Пусть она станет вашим добрым другом, к которому вы заходите время от времени. Забудьте ненужные места и наслаждайтесь теми моментами книги, которые помогают.

Вы обнаружите, что каждая глава начинается с изложения основных идей. Они помогут вам ориентироваться в содержании книги.

Небольшие врезки под названием «Как получить золото из песка» подчеркивают особо важные моменты; в этих врезках выделены основные экономящие средства аспекты главы. Также я дала определения терминам и выражениям, которые встречаются в книге; так вам будет проще ознакомиться с новыми понятиями и концепциями.

Врезка «Угроза бюджету» в каждой главе дает представление о существующем риске для ваших ограниченных ресурсов. Как и в жизни, форс-мажорные обстоятельства могут оставить вас без гроша в кармане. К счастью, угрозы бюджету Web-сайта немного проще предсказать, нежели затопление подвала или повышение цен у ветеринара. Надеюсь, что рубрика «Угроза бюджету» позволит вам избежать лишних затрат.

Главы книги не представляют собой единства: вам не надо читать главу 1, чтобы понять главу 2, поэтому вы можете перескакивать из главы в главу в любом порядке. Можете начать читать с вопросов, интересующих вас в данный момент, либо прочитать все от начала до конца. Если у вас нет времени, можете просто бегло ознакомиться с содержанием той или иной главы. Главное - прочитать содержание и ознакомиться с врезками «Как получить золото из песка» и «Угроза бюджету». Затем, когда будет время, можете вернуться и прочитать главу целиком. Я составила эту книгу специально для занятых людей, поэтому даже поверхностное ознакомление с ней поможет вам продвинуться в работе.

Делайте меньше

Если вы читаете эту книгу, то, вероятно, ограничены во времени так же, как и в финансах, и вам не терпится поскорей перейти к основному ее содержанию. Главный секрет выполнения малобюджетной работы состоит в том, что нужно делать мало, но хорошо. Если у вас не хватает средств на полнофункциональную CMS, выберите такую, которая будет выполнять 80% того, что должна выполнять CMS, но пусть эти 80% функций выполняются как следует. Если у вас недостаточно средств для большой галереи, используйте меньше фотографий, но пусть они будут самого лучшего качества.

Двигаясь вперед, старайтесь делать меньше.

Об авторе

Кэрри Бикнер (Carrie Bickner) - помощник директора по информации и системному проектированию в Нью-Йоркской публичной библиотеке. Бикнер пишет для журналов "List Apart, LibraryJournal" и "Technology Electronic Reviews", а также для своего персонального сайта www.roguelibrarian.com. Также она является соавтором "The NYPL Online Style Guide", учебника по CSS и XHTML. Она

постоянно выступает на различных конференциях, таких как Web Design World, ежегодной встрече The Public Library Association, а также SXSW.

Она работала над большими и маленькими проектами, создавая и управляя созданием сайтов с бюджетом от \$3000 до \$3 млн. В данный момент Бикнер работает над созданием цифровой галереи Нью-Йоркской публичной библиотеки, содержащей более 100000 изображений из собраний этой библиотеки, а также над созданием «Истории афро-американского переселения» (The African-American Migration Experience) - мультимедийной online-энциклопедии, включающей информацию о 13 этапах развития африканской диаспоры.

Кэрри живет в Нью-Йорке. В свободное время она занимается любительской фотографией, смотрит классические фильмы и гуляет по многочисленным кварталам Манхэттена.

Рецензенты

Рецензенты приложили немалые усилия при проверке описываемых в книге процессов Web-разработки. Когда книга была написана, эти профессионалы проверяли все материалы, касающиеся технического содержания, организации и процесса разработки. После долгих проверок они подтвердили, что данная книга отвечает потребностям читателей и содержит высококачественную техническую информацию.

Чед Брандт

Чед Брандт (Chad Brandt) - менеджер по Web-проектам в компании Best Software's Small Business Division в Скоттсдейле, Аризона. Одновременно являясь Web-дизайнером и разработчиком, совмещая в себе таланты программиста и великолепного дизайнера, Чед специализируется на планировании интерфейса сайтов, работе с незапланированными задачами и с недавнего времени имеет дело с Web-стандартами. Помимо своей непосредственной работы в Best Software, Чед занимается созданием дружественных Web-сайтов и Интернет-приложений в своей собственной компании Active Internet Solutions (www.activei.com).

Чед создал фирму Active Internet Solutions в 1998 году, чтобы предлагать экономичные Web-решения на уровне предприятий для бизнеса любого размера, типа и индустрии. Экономичный Web-дизайн - понятие, близкое Чеду, создавшему массу сайтов при небольшом бюджете, а иногда и при полном его отсутствии.

Элисон Сесил

Поддельвала ли она отметки в своем дневнике или рисовала рекламные плакаты для лимонада, дизайн всегда был частью жизни Элисон Сесил (Allison Cecil) начиная с ранних лет. Несмотря на то что в колледже она увлекалась искусством, рисование вскоре ей надоело и она перестала им заниматься. Пять лет назад переехав в Нью-Йорк, она увлеклась дизайном и поступила в школу дизайна Parsons, где она узнала, что дизайн - это не просто красивые картинки.

После трех лет сотрудничества с мастером журнального и Web-дизайна Роджером Блэком (Roger Black) Элисон начала работать самостоятельно. Она работала над проектами как в печати, так и в Сети. Среди ее клиентов такие известные холдинги, как New Riders Publishing, Market News International, Watson Adventures, J.Crew и The National Enquirer.

Среди увлечений Элисон: чтение античной и средневековой литературы, катание на пони в окрестностях Бруклина и вольная борьба со своим семикилограммовым котом по имени Вестон Сесил. Она окончила курсы Американского Красного Креста по безопасности на воде, а также Калифорнийский университет в Сан-Диего.

Стив Чемпеон

Стив Чемпеон (Steve Champeon) - руководитель технического отдела в фирме hesketh.com/inc, в Релейе, Северная Каролина. Будучи основателем популярного Интернет-сообщества Web-дизайнеров, Стив также является автором книги "Building Dynamic HTML GUIs", издательство John Wiley & Sons, ©1999 и соавтором книги "Cascading Style Sheets: Separating Content from Presentation", издательство glasshaus, ©2002. Кроме того, он - автор десятков статей по Интернет-технологиям и культуре. Он - научный технический редактор, принимавший участие в создании более 20 различных книг. Стив - один из основателей Проекта по Web-стандартам (The Web Standards Project), который внес большую лепту в развитие поддержки и обучения стандартам. Его страсть - кресла-качалки, холодный чай, компьютеры Macintosh и сальса, которую он танцует дома так, что многие бы удивились.

Стив Санчес

Стив Санчес (Steve Sanchez) - основатель и президент iNexus.com - фирмы в Лос-Анджелесе, которая консультирует, создает и «раскручивает» бизнес-сайты. В течение последних 10 лет он работал с ведущими компаниями в различных

областях, таких как туризм, медицина, издательство, создавая сайты, которые радовали и владельцев, и посетителей. Стив, изучавший в течение долгого времени пользовательские потребности, является поклонником Web-сайтов, основанных на базе данных, деловых Web-сообществ, а также таких технологий, как динамический Flash, VR-Tours, Active Server Pages и Microsoft .NET. Он увлекается фотографией, путешествиями, плаванием и дайвингом. Он живет в Лос-Анджелесе вместе с женой и четырьмя детьми.

Благодарности

Спасибо Дженнифер Эберхардт (Jennifer Eberhardt), моему старшему редактору по разработке, за то, что эта книга получилась такой, как я хотела. Спасибо Майклу Нолану (Michael Nolan), моему исполнительному редактору, за то, что он помог мне реализовать мои идеи и принести их в издательство New Riders. Спасибо Стиву Чемпеону (Steve Champeon), Элисон Сесил (Allison Cecil), Чеду Брандту (Chad Brandt) и Стиву Санчесу (Steve Sanchez), моим техническим редакторам.

Я благодарю моего продюсера Стефанию Волл (Stephanie Wall), менеджера по проектам Джину Канаус (Gina Kanouse), старшего менеджера по маркетингу Тэмми Детриха (Tammy Detrich), менеджера по рекламе Сюзан Никсон (Susan Nixon), старшего редактора проекта Сару Керне (Sarah Kearns), редактора моей рукописи Кристу Хенсинг (Krista Hansing), а также составителя указателя Лизу Стампф (Lisa Stumpf).

Выражаю признательность всем моим коллегам из Нью-Йоркской публичной библиотеки, особенно Кэтрин Джонс (Catherine Jones).

Отдельное спасибо Сюзан Кауп (Susan Kaup), Дэвиду Гэссону (David Gasson), Дэйву Бэllu (Dave Bell), Марку Ньюхаусу (Mark Newhouse), Эрику Мейеру (Eric Meyer), Лэй Бэйкер-Фoley (Leigh Baker-Foley), Нику Асборну (Nick Osborne), Келли Эбботт (Kell Abbott), Люку Тимовски (Luke Tymowski) и Сэнди Коэн (Sandee Cohen).

Спасибо Элизабет Хепола (Elizabeth Hepola) за то, что она показала мне, как сделать Web-страницу, и Тане Рабаурн (Tanya Rabourn) за то, что она показала мне, как сделать Web-сайт.

Сюзан Хой (Susan Hoy), Надалин Темпелман-Клюит (Nadaleen Tempelman-Kluit), Лэй Бэйкер-Фoley (Leigh Baker-Foley), Мира Лажуа (Mira Lajoie), Астрид Бэк (Astrid Beck) и Элен Бертон (Ellen Burton): спасибо вам за то, что освещали мой путь.

Моим бабушке и дедушке, Гарольду и Марриел Бикнер (Harold and Muriel Bickner), спасибо за то, что заставили меня по-настоящему поверить, пусть ненадолго, в то, что я ирландская принцесса.



Производство

ГЛАВА 1. Секреты успешного малобюджетного проекта

ГЛАВА 2. Экономичный план проекта

ГЛАВА 3. Разработка интерфейса

ГЛАВА 4. Почему хороший текст имеет большое значение

ГЛАВА 5. Как хорошо выглядеть с минимальными затратами

Г т м 1 . Секреты успешного малобюджетного проекта

Нет. Я из Айовы; я работаю во внешнем космосе.
*Уильям Шатнер в роли Джеймса Т. Кирка
в фильме «Звездный путь 4: Путешествие домой» (1986)*

Основные идеи главы

1. Используйте подручные средства.

Возможно, в вашем распоряжении имеются средства, которые вы до сих пор не использовали. Я помогу вам определить, чем вы располагаете - «железом», программным обеспечением, инструментами дизайна или талантливыми специалистами, - и эффективно этим воспользоваться.

2. Работайте с небольшой группой специалистов, принимающих решения.

Начальники или кураторы иногда бывают полезны, например при мозговом штурме для получения новых идей. Однако вам необходимо отслеживать, чтобы решения принимались маленькой, постоянной группой специалистов. Это поможет придерживаться одной концепции и экономить деньги за счет меньшего количества времени, потраченного на принятие решений.

3. Придерживайтесь четких задач.

Точно сформулируйте, какой сайт вы хотите получить. Если это всего лишь малая часть большой задачи, которая впоследствии будет расширяться, объясните своей команде цели на данном этапе и сообщите, что будет реализовано в дальнейшем.

4. Делайте меньше.

Не надо делать все и сразу! Если у вас маленький сайт, пусть и задачи у него будут небольшие. Лучше сделайте мало, но хорошо.

Единственно верного способа успешного производства малобюджетных сайтов не существует. Я и сама была бы рада сказать что-нибудь вроде: «Никогда не платите больше \$200 за систему управления сайтом» или «Всегда выделяйте как минимум треть бюджета на дизайн интерфейса». На самом деле способов создать сайт очень много, а способов потратить деньги еще больше.

Но все-таки есть четыре составляющие успешного создания малобюджетного сайта, одинаковые во всех случаях:

1. Используются имеющиеся под рукой ресурсы.
2. Работу выполняет относительно небольшая группа людей, принимающих решения.
3. Четко определяются задачи.
4. Ставится цель делать меньше, но качественно.

В следующих двух разделах мы рассмотрим несколько сайтов, созданных с учетом перечисленных составляющих.

Журнал **Astronomy Education Review**

В 2001 году три Web-разработчика из Национальной оптической астрономической обсерватории - Дэвид Гэссон (David Gasson), Дэйв Бэлл (Dave Bell) и Марк Ньюхаус (Mark Newhouse) - приступили к созданию первого и единственного в своем роде профессионального образовательного журнала, посвященного астрономии. Этот сайт, получивший название "Astronomy Education Review" - <http://aer.noao.edu>, был создан в домашних условиях, без какого-либо дополнительного финансирования (рис. 1.1 и 1.2). Вот как сформулировал основную задачу журнала Дэвид Гэссон: «Мы крайне внимательно отслеживали первоначальные объемы проекта и создали базу данных таким образом, чтобы в дальнейшем ее можно было расширить, если появятся деньги, время и желание».

Эта команда использовала стратегию, именуемую MAP, - название состоит из первых букв MySQL, Apache, а также Perl или PHP. Все это бесплатное, доступное для скачивания программное обеспечение:

- MySQL - разработчики использовали MySQL (www.mysql.com) в качестве базы данных, в которой хранится все содержимое сайта;

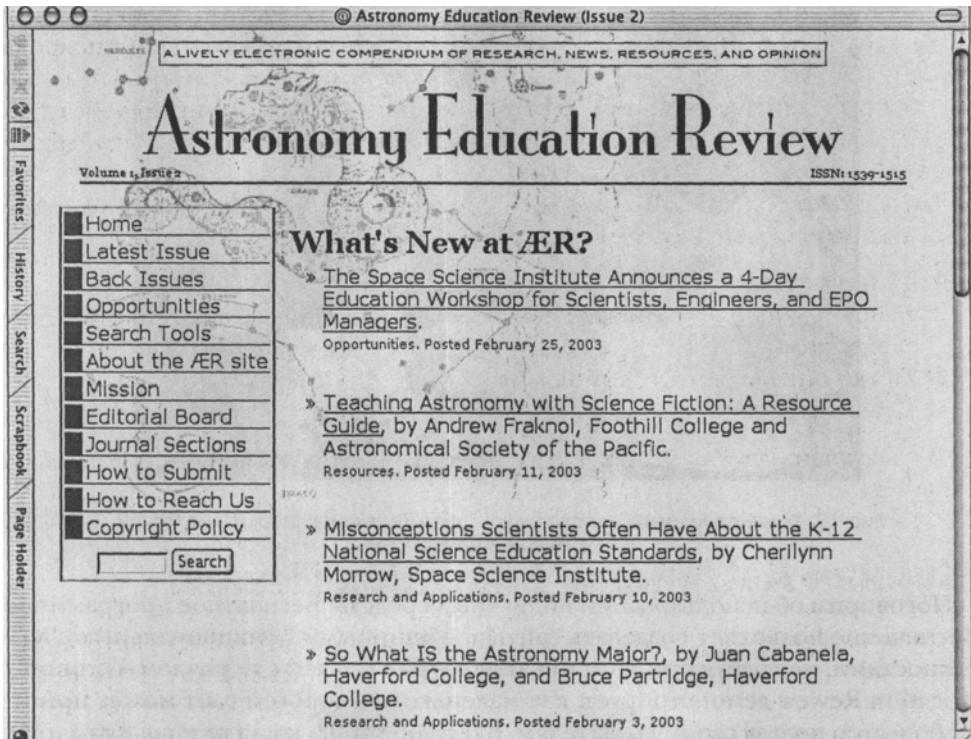


Рис. 1.1. Журнал Astronomy Education Review (<http://aer.naoa.edu>) - это первое и единственное в своем роде профессиональное образовательное издание, посвященное астрономии. Сайт стартовал при полном отсутствии бюджета, но был настолько успешным, что команду разработчиков наградили премией от NASA, позволившей развить проект

- Apache - сайт журнала "Astronomy Education Review" работает на сервере Apache (www.apache.org) под операционной системой FreeBSD (www.freebsd.org), служащей в качестве серверного программного обеспечения;
- Perl и PHP - все серверные сценарии написаны с использованием языков Perl и PHP (www.php.net). Это связующая среда между содержимым базы данных и Web-страницами, которые видит посетитель сайта.

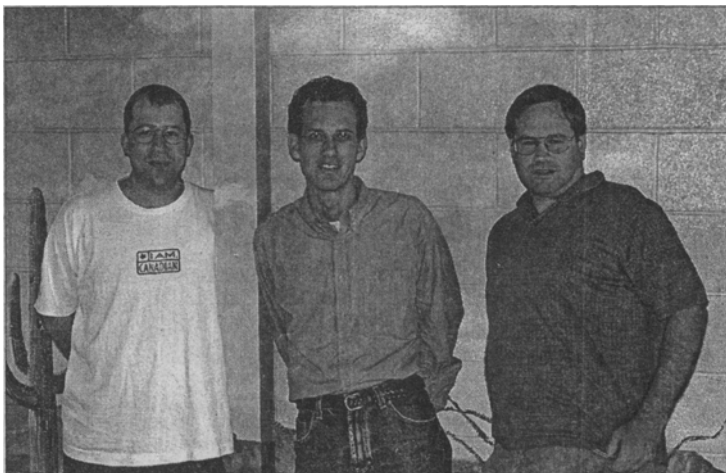


Рис. 1.2. Команда журнала Astronomy Education Review. На фото: Дэвид Гэссон, Марк Ньюхаус и Дэйв Бэлл

Поговорим об использовании подручных средств. Бесплатное программное обеспечение позволяет создавать сайты с обширными функциональными возможностями, не требуя при этом никаких затрат. Случай с журналом Astronomy Education Review демонстрирует, как маленький недорогой сайт может принести большую пользу своим создателям. В октябре 2002, когда журнал Astronomy Education Review вышел в свет, его первыми посетителями были наши друзья и коллеги, а редакция состояла лишь из группы добровольцев. Сразу же после первых публикаций в редакцию посыпались статьи от самых разных ученых. Сегодня издание постоянно получает массу новых работ.

Недавно НАСА выделил журналу Astronomy Education Review около \$150000 на два года в качестве поддержки. Сайт постоянно получает положительные отзывы, благодаря тому что там размещается новейшая информация в области астрономического образования.

Команда из Национальной оптической астрономической обсерватории создала успешный сайт, воспользовавшись бесплатными подручными средствами.

Лэй Бейкер-Фoley

Лэй Бейкер-Фoley (Leigh Baker-Foley) из компании NotLimited NYC, LLC (рис. 1.3) недавно оформила сайт для деревообрабатывающей компании

Ричарда Ронко (Richard Ronco), www.nodtld.com/ronco - рис. 1.4. Выполнение этого проекта обошлось ей примерно в \$2500.



Рис 1.3. NotLimited NYC, LLC - одна из многочисленных маленьких фирм, занимающихся разработкой малобюджетных проектов. Глава компании, Бейкер-Фoley, обслуживает клиентов, которые обнаружили для себя, что малое может (и должно) равняться многому, особенно в отношении финансовой и клиентской отдачи/Дизайнеры, способные создавать то, что Бейкер-Фoley называет «прекрасными функциональными сайтами без излишеств», лучше других знают, как выполнять великолепные работы и при этом практически ничего не тратить. Искусству создавать отличные сайты, символизирующие успешные компании, Бейкер-Фoley научилась сама

Успех этого проекта зависел от двух вещей:

1. Задачи клиента и производителя совпадали - необходимо было создать сайт при небольшом бюджете.
2. Была ясна концепция сайта. Когда я спросила Бейкер-Фoley, как ей удалось удержаться в рамках экономичного бюджета сайта Ричарда Ронко, она выделила несколько ключевых моментов:
 - компоновка и структура сайта должны быть простыми и открытыми. Компоновка и графический пользовательский интерфейс (Graphical User Interface - GUI) были продиктованы элегантной простотой продукции деревообрабатывающей компании Ричарда Ронко. Впоследствии, когда у клиента будет возможность, на сайте появятся новые галереи и дополнительные страницы с информацией. Эти обновления уже не потребуют специальных затрат на дизайн;

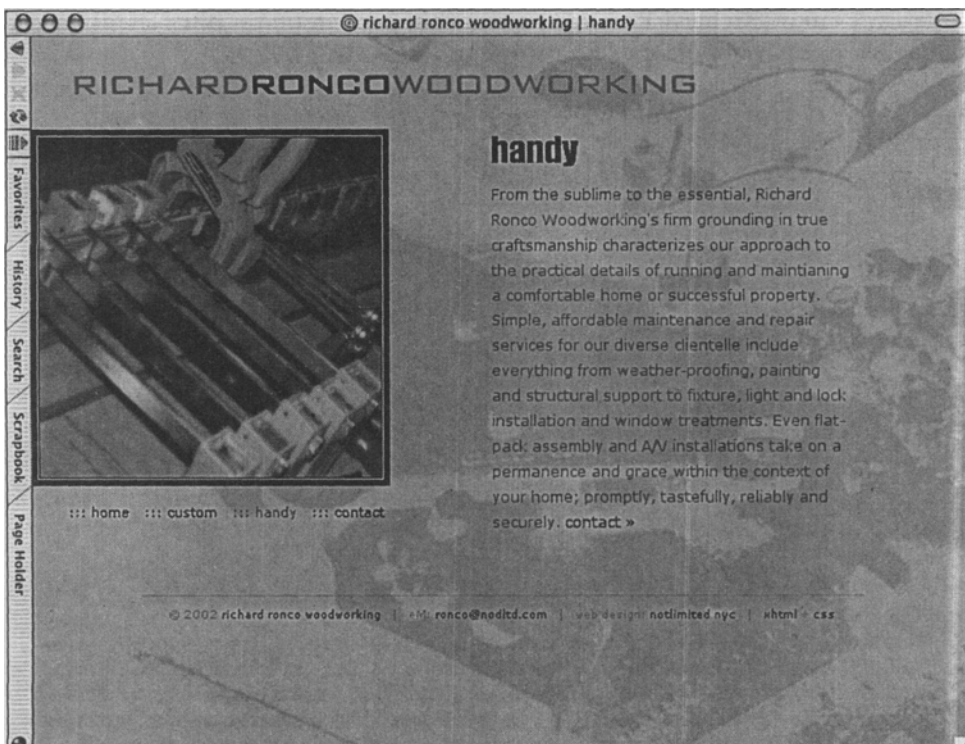


Рис. 1.4. Один из последних проектов Бейкер-Фолей - сайт для небольшого магазина деревообрабатывающей продукции в Бангоре, штат Мэн (<http://www.nodtld.com/ronco>). Бейкер-Фолей сэкономила деньги клиента за счет простого оформления сайта, самостоятельной подготовки фотографий, хостинга на собственном сайте и использования стандартной разметки

- содержание сайта должно обсуждаться вместе с клиентом. Содержание сайта основывалось на разговорах с клиентом, а также на примерах подобных сайтов конкурирующих компаний. Поэтому не пришлось платить писателям, редакторам и т.п.;
- самостоятельная разработка иллюстративного материала. Все фотографии, картинки и даже логотип клиента создавались в домашних условиях, тем самым были сэкономлены расходы на зарплату дизайнеров, фотографов и других специалистов;
- сайт должен поддерживать стандарты XHTML и CSS. Использование стандартных кодов для отображения страницы и CSS для компоновки

обеспечит не только читабельность, но и совместимость со всеми браузерами и компьютерами, а также поможет серьезно сэкономить при последующем редизайне и обновлениях сайта за счет доступности и простоты в использовании (см. главу 7);

- размещение готового сайта на уже существующем сайте. Имея свободное место на сайте своей компании, Бейкер-Фолей предложила клиенту недорогой хостинг, который позволил сэкономить деньги на регистрации домена. Сайт выглядит великолепно, а за счет ориентированной на стандарты структуры его размер в килобайтах весьма мал. Выгода очевидна.

Пример сайта деревообрабатывающей компании очень хорош по многим причинам.

Во-первых, здесь соблюден первый принцип успешного создания малобюджетного сайта - проект выполнен подручными средствами. У дизайнера был приличный цифровой фотоаппарат, с помощью которого отсняли продукцию клиента (подробнее о том, как делать фотографии, объясняется в главе 5).

Во-вторых, у проекта есть перспективы, когда средств будет больше. Деревообрабатывающий бизнес развивается, и на сайт без труда можно будет добавить новые галереи с продукцией Ронко. В главе 7 говорится о том, что сайт становится более экономичным при использовании технологии CSS для структурной разметки. Таким образом, экономятся время и деньги на производстве, редизайне, хостинге и поддержке.

И наконец, еще одна деталь, которая мне понравилась в этом сайте, - он расположен непосредственно на домене NotLimited. Это значительно уменьшает расходы клиента на хостинг. Позднее, когда увеличится бюджет, у сайта деревообрабатывающей компании Ричарда Ронко будет свой собственный домен.

Успех любого малобюджетного проекта зависит от того, на что вы хотите потратить средства сейчас и что вы сможете сделать потом, когда бюджет вырастет. В случае с деревообрабатывающей компанией клиент и дизайнер не сомневались в том, что необходимо для успешного создания сайта.

Из чего состоит успешный малобюджетный сайт

Теперь, когда на примере двух реально существующих сайтов вы увидели, как можно поступить с небольшим бюджетом, давайте обратимся к деталям.

Используйте подручные средства

В одной из апокрифических книг рассказывается история о Марте Вашингтон (Martha Washington) и носках. История гласит, что она сделала платье из ниток от старых носков Джорджа. Носки она распустила, соткала из нитей ткань, а из этой ткани сшила платье.

Не думаю, что из носков у нас получится отличный сайт, зато я уверена, что у каждого из нас найдутся вещи, которые можно грамотно применить. То, как Бейкер-Фoley использовала свой любительский фотоаппарат, чтобы сделать Ронко фотографии, - очень удачный пример. Я знаю еще несколько способов, как использовать то, что у вас есть под рукой, для создания недорогого сайта.

- Как получить золото из песка

Работа не приносит Web-профессионалам особую прибыль, поскольку на создании Web-страниц не разбогатеешь. Однако люди, занимающиеся нетривиальными делами и добивающиеся успеха, стараются извлечь из своего небогатого жизненного опыта максимум пользы. Те, кто специализируется на создании маленьких недорогих сайтов, играют особую роль в деле своей компании и клиентов. Одна из задач, которую я надеюсь реализовать при написании этой книги, состоит в том, чтобы поделиться с вами техникой и различными тонкостями, которые помогут вам стать одним из таких ценных работников.

Количество людей, принимающих решения, не должно быть большим

В 1939 году, буквально перед выходом «Волшебника страны Оз», руководители MGM studio чуть было не вырезали песню "Somewhere Over the Rainbow" из фильма. Им показалось, что она слишком медленная и делает картину скучной. В конце концов песню оставили, но каждый раз при мысли о том, что это чуть было не случилось, меня бросает в дрожь. С подобными проблемами сталкивается любой творческий проект: слишком большое количество людей, принимающих решения, могут погубить его.

Дэвид Оджилви (David Ogilvy), один из создателей современной рекламы и автор самых восхитительных высказываний 20 века (попробуйте набрать в поисковике Google «Ogilvy quotations»), однажды написал: «Большая часть отвратительной рекламы, которую вы видите сегодня по телевизору, - продукт

коллективной деятельности. Коллектив может критиковать рекламу, но ему ни в коем случае нельзя позволять ее делать».

То же самое можно сказать и о Web-сайтах. У менее успешных сайтов, в разработке которых мне приходилось участвовать, были навигационные схемы, названия или цветовые гаммы, созданные коллективно. Творческая деятельность сводится к нулю, когда из разных идей пытаются получить нечто среднее. Я бы предпочла видеть одного плохо работающего человека или маленькую плохо работающую команду, чем большой и слаженный коллектив. У маленькой команды, даже если она плохо работает, гораздо больше шансов подготовить качественный проект в отведенные сроки.

У сайта должна быть общая идея

В течение долгого времени я преподаю академическое письмо студентам колледжа, и лучшие работы - не всегда самые большие по объему, лучшими я считаю те, где меньше всего неопределенности. Некоторые студенты строчат, как из пулемета, пытаясь затронуть каждую тему, изученную в течение семестра, и полагают, что если изложить как можно больше фактов, то один из них обязательно попадет в цель. Работы таких студентов всегда плохие. В лучших работах, как правило, прослеживается одна основная мысль, пусть даже нечетко оформленная, зато основанная на четких и твердых убеждениях.

Что отличает хорошие работы от плохих? То же, что и успешные сайты от неуспешных: общая *идея*.

Идея особенно важна при малом количестве средств. Однажды я принимала участие в проекте, где преобладали две соревнующиеся концепции. Сначала работа шла в одном направлении, затем - в диаметрально противоположном. Разработчики, дизайнеры и писатели были рассредоточены по группам и точно не знали, чего именно они добиваются. Месяцы работы пропали зря просто потому, что идея проекта стала нечеткой.

В конце концов концепция проекта прояснилась. За месяц мы смогли сделать то, что не удавалось выполнить в течение года. Боевой дух укрепился, и это помогло нам работать гораздо продуктивнее.

Проект был относительно большой, поэтому многие затраты растворились в бюджете. Но отсутствие идеи смертельно для малобюджетного проекта, в котором нет места для неэффективной работы.

Если у вас есть определенная идея, это поможет успешно выполнять краткосрочные и долгосрочные задачи. Лэй Бейкер-Фoley помогла своему клиенту из деревообрабатывающей компании принять краткосрочные и долгосрочные

решения. Помните, что для сайта Ричарда Ронко отдельное доменное имя не регистрировалось, сайт был размещен в поддиректории официального сайта компании Лэй. Так, экономя понемногу, за год можно скопить сумму, достаточную для того, чтобы нанять фотографа или разместить на сайте новую галерею. Неважно, оформляете ли вы отношения клиент/консультант или создаете сайт для какой-нибудь компании в домашних условиях, забота о долгосрочных задачах может окупиться благодаря вашим усилиям, потраченным на выполнение краткосрочных задач, и принесет вам удовлетворение от работы уже на начальном этапе.

Лучше сделать меньше

Я обожаю фильмы Джима Джармуша (Jim Jarmusch). Отчасти мое восхищение обусловлено скромным оформлением, которого я обычно придерживаюсь при создании сайтов. Однажды я слышала интервью, в котором Джима спрашивали о фильме "Down By Law". Он рассказывал о том, каким образом ему удается передать атмосферу того или иного места. Вместо того чтобы использовать дорогостоящие панорамные съемки либо фразы наподобие: «Посмотри, это Новый Орлеан», он заставляет зрителей проникнуться настроением города, показывая интерьеры безмянных домов в ничем не приметных местах. Благодаря такой сдержанности его малобюджетные фильмы отличаются спокойствием и утонченностью.

«Лучше сделать меньше» - эту фразу я услышала от Тима Брея (Tim Bray), одного из авторов XML, когда он говорил об основах успешной технологии. Когда мы просматриваем список новых технологий, заполонивших рынок, нам трудно определить, какая из них лучше. Брей понял, что одна из составляющих успеха заключается в том, что технология должна помогать выполнять хорошо одну-две задачи. В противном случае проект обречен на провал.

Данная книга не рассказывает о развитии новых технологий, однако Web-дизайнеры могут многое почерпнуть из идеи Брея: создавайте сайт, хорошо выполняющий одну-две задачи, и не пытайтесь напичкать его всеми возможными функциями.

В этой книге обсуждается тема создания сайтов с небольшими возможностями, которые позволят вам, вашим клиентам, коллегам и посетителям вашего сайта извлечь максимум пользы.

ГЛАВА I. ЭКОНОМИЧНЫЙ ПЛАН проекта

О! Это невыносимо: пенс туда, пенс сюда. Посмотрите на меня: благодаря работе я из никого превратился в нищего.

*Гручо Маркс
в "Monkey business" (1931)*

Основные идеи главы

1. **Делайте меньше.**

Трудно отказаться от реализации в проекте всех идей, которые придут в голову вам или вашим коллегам. Помните, что бюджету вас маленький; поэтому лучше сделать немного, но качественно. Приберегите дорогие идеи для следующего этапа разработки сайта.

2. **Напишите краткий список задач проекта.**

Если у вас нет четкого плана проекта, ваш небольшой бюджет будет исчерпан из-за нерешительности, долгого обдумывания и исправления ошибок. Необходимо удостовериться, что вы начинаете работу, имея четкий список целей проекта.

3. **Составьте список функциональных требований.**

Нужно определиться с функциональными требованиями сайта. Отсутствие необходимых функциональных требований может вылиться в круглую сумму, а значит, придется тратить время и деньги на производство того, что вам совсем не нужно.

4. **Определите технические требования.**

Не жалейте времени на составление списка технических требований (браузеры, операционные системы пользователя и т.д.), пользуйтесь этим списком при тестировании вашего сайта и в процессе его разработки. Если ждать готовности сайта для того, чтобы протестировать его на предмет соответствия техническим требованиям, то изменения и исправления, которые придется внести, обойдутся гораздо дороже, чем запланировано. Старайтесь избегать ситуаций, когда приходится платить за переделку сайта.

5. **Храните документацию под рукой.**

Важно не только составить перечень целей проекта, функциональных и технических требований, но и прочитать их, а также удостовериться, что ваши коллеги тоже с ними ознакомились. Лично я предпочитаю хранить эти бумаги поблизости, чтобы всегда знать о том, что необходимо сделать.

Эта книга обещает научить созданию великолепных сайтов с маленьким бюджетом. И, хотя на страницах книги вы найдете полезные советы по каждому этапу и аспекту создания малобюджетного сайта, успех проекта зависит от того, как вы его изначально спланируете. В конце концов, успех любого дела является результатом хорошей организации и четкой концепции. Это особенно верно при создании малобюджетного сайта. Если у вас немного денег, вы не можете себе позволить попусту тратить время при плохо организованной работе.

При создании высокобюджетных сайтов на бесполезные затраты часто не обращают внимания. Когда вы имеете большой бюджет, нерешительность и недостатки управления не так страшны. Нечеткий план может повлечь за собой многочасовую проверку; масса времени уйдет на исправление ошибок, которых можно было бы избежать при хорошо продуманном плане.

Одно из преимуществ работы за незначительную зарплату в день состоит в том, что вам приходится довольствоваться малым. Web-профессионалы, ограниченные в деньгах и вынужденные сосредоточенно работать над проектом, в чем-то напоминают путешественников, которые не могут себе позволить набить рюкзак бесполезными вещами. Стадия планирования в Web-производстве напоминает укладывание вещей перед путешествием. Составляя план малобюджетного сайта, вы должны быть избирательны в том, какие задачи и требования нужно «взять с собой».

Польза от хорошо продуманного плана

В 1996 году экспедиция, возглавляемая альпинистом Робом Холлом (Rob Hall), состоящая из обычных туристов, обернулась трагедией. Коммерческий туризм был прибыльным делом, и в горы часто шли люди, не имевшие никаких альпинистских навыков, но способные заплатить за подобное путешествие большие деньги. Одна «альпинистка» взяла с собой кофеварку. К счастью, женщина выжила, но вес ее рюкзака на такой высоте и при такой температуре вызывал серьезный и неоправданный риск. Такой «кофеваркой» являются ненужные вещи, от которых Web-профессионалы должны избавляться.

«Переполненные рюкзаки» погубили многие сайты, независимо от бюджета. Собираясь в любое путешествие, трудно отказать себе и не захватить вещи, которые могли бы пригодиться. Но, если у путешественника маленький

бюджет, собираться надо целенаправленно. Представьте себе, что работа над малобюджетным сайтом - это восхождение на высокую гору, где любой лишний груз станет обузой. Взяв с собой слишком много вещей, вы поймете, что за них придется заплатить запредельную цену.

Я никогда не занималась альпинизмом, но привыкла к скалолазанию. Правда, большинство моих скал находилось в спортивном зале. Мы с друзьями взбирались по 70-футовым плитам, сделанным из фанеры и искусственных горных пород. Занятия в зале стоили недорого, и мне кажется, что основную прибыль владельцы получали от продажи напитков. Вода стоила примерно \$3 за бутылку. В те годы я была студенткой и лишних денег у меня не было, поэтому я приносила с собой бутылку с водой из-под крана. Вроде ничего особенного, но вполне возможно, что за годы моих занятий мною была сэкономлена не одна сотня долларов.

г- Угроза бюджету

Небрежное отношение к малобюджетному сайту может дорого обойтись. Я любила получать небольшие дополнительные заработки от создания малобюджетных сайтов. На собственном опыте я убедилась, что если не заниматься планированием такого рода сайтов, то в дальнейшем это обернется серьезными затратами. Пренебрегая планированием и подсчетом расходов времени и денег на элементарные вещи (например, новые шрифты, время на подготовку изображений, переработку авторских текстов), я проработала массу лишнего времени, не получив за это дополнительно ни гроша. Работая фрилансером, в агентстве или дома, не допускайте, чтобы ваш бюджет пострадал из-за плохой организации труда.

Очень часто Web-профессионалы, работая над маленьким сайтом, не тратят время на составление плана. Если у сайта маленький бюджет, люди думают: «Это всего лишь мелкий проект, над которым я тружусь в свободное время. Это временная работа, она получится сама собой». Не позволяйте себе так думать! За такое отношение придется дорого заплатить. Когда вам обещана фиксированная сумма, не зависящая от количества выполненной работы, вы не сможете оправдать даже расходы. Если при почасовой оплате вы тратите деньги клиента, покрывая ваши расходы на создание сайта, то вам должно быть стыдно. Если в течение полного рабочего дня вы находитесь дома, то заплатите своим свободным временем: придется работать допоздна, в выходные, чтобы восполнить часы, потраченные зря из-за нежелания спланировать все заранее. В любом случае без плана на работу уйдет гораздо больше времени.

Ценные советы

Имея хороший план, гораздо проще оставаться в рамках данного бюджета. Прежде всего необходимо добросовестно выполнить следующие шаги:

1. Определить задачи сайта.
2. Сформулировать функциональные требования.
3. Определить технические требования.

Чересчур увлекаться каким-то одним пунктом не стоит, но и пренебрегать ни одним из них нельзя. Это поможет вам оставаться в рамках бюджета.

-Угроза бюджету _____:

Изменение объема. Изменение объема - это то, чего не избежать ни одному сайту. Развитие Web-сайта трудно остановить. Появляются новые идеи - увеличиваются потребности. Объем вашего проекта с появлением новых идей и задач существенно увеличивается. Изменения объема могут быть благоприятными и неблагоприятными, в зависимости от того, как их произвести. Главное - уметь управлять изменениями, сознательно применяя их к вашему сайту и аккуратно приспособив бюджет к новым нуждам. Четко сформулированные задачи проекта и правильно определенные функциональные и технические требования помогут вам сохранить душевное и материальное равновесие, когда придет пора менять объем.

Задачи малобюджетного проекта

Прежде чем приступить к работе, сядьте и напишите, чего вы ждете от сайта. Обсудим вымышленный пример. Это пример внештатной работы, но основы для любого случая Web-разработки одни и те же.

Задачи проекта для сайта Something Blue

Предположим, что некая художница, Мэри Хой (Mary Hoy), попросила вас сделать сайт, посвященный ее последней выставке. Заплатить она может совсем немного. Вам нравятся ее картины, и вы надеетесь, что этот гонорар покроет ваши расходы, поэтому соглашаетесь взяться за работу. Мэри никогда не занималась созданием сайтов и не имеет ни малейшего представления о том, каким должен быть его объем. Вы вполне обоснованно ожидаете, что из-за недостаточного опыта ей в скором времени может прийти в голову изменить задачи и

требования. Когда она увидит, как выглядят ее работы в Сети, ей захочется попробовать реализовать новые идеи или дополнить сайт новыми функциями. Чтобы сэкономить время и деньги, вы составляете список задач проекта:

г-Примечание

Задачи сайта Something Blue. Something Blue будет представлять собой виртуальную выставку работ художницы Мэри Хой. На сайте будут находиться 15 из 40 картин с ее выставки под названием Something Blue, которая проходила в центре Good Girl Gallery (Студио Сити, Калифорния, июль 2003 года). Помимо 15 картин на сайте будут размещены два интервью с художницей, ее биография и контактная информация. Назначение сайта - рассказать о работах Мэри и, возможно, продать одну-другую картину, хотя непосредственной продажи через сайт не будет.

Это хороший пример задач экономичного проекта, потому что эти задачи конкретные. Во-первых, вы определили жанр сайта: это выставка, а не ретроспектива или виртуальный магазин. Во-вторых, вы подсчитали объем проекта. Есть небольшая разница в цене между сканированием 15 и 40 картин. Возможно, только вы будете заниматься сканированием и редактированием изображений. Но это занимает определенное время, поэтому если составить список 15 картин, то Мэри будет сложнее просить вас отсканировать еще несколько работ. То же самое с интервью и контактной информацией: можно подсчитать примерное количество Web-страниц, которые понадобятся для реализации этих задач. Поскольку стоимость проекта вы подсчитываете вместе, то можете назначить отдельную цену за каждый дополнительный элемент.

г- Угроза бюджету

Сократите число людей, принимающих решения. Как объяснялось в главе первой, один из секретов успешного создания малобюджетного Web-сайта состоит в том, что количество людей, принимающих решения в проекте, не должно быть большим. Чем больше людей вы привлечете к определению задач проекта, тем более пространными будут эти задачи. Правильное распределение ресурсов требует тщательно разработанного проекта. Если на начальном этапе работы решения принимает большая группа людей и это мешает вам определить задачи проекта, постарайтесь сократить число участников, оставив только самых необходимых. Если этого не сделать, очень много денег уйдет на непонятную работу, которая не принесет положительных результатов.

Список задач проекта не спасет ни вас, ни ваш бюджет, если с этими задачами не ознакомить людей, заинтересованных в создании сайта. В примере с сайтом Something Blue единственный человек, принимающий решения, - это ваш

клиент Мэри. Однако в большинстве случаев лиц, заинтересованных в создании сайта, намного больше, поэтому постарайтесь сделать так, чтобы они не имели права голоса. Чем больше одобрений вам придется получать, чтобы продолжать работу над сайтом, тем сложнее и дороже будет этот процесс. И самое главное: чем больше людей вовлечено в процесс создания, тем менее гармоничным получится сайт. Отличить людей, заинтересованных в проекте, от людей, принимающих решения, непросто; старайтесь не поспорить с людьми, от которых зависит проект: если вы начнете откровенно их игнорировать, проект может быть закрыт еще до его запуска. Поговорите с ними о временных и денежных затратах! Убедитесь, что людей, принимающих решения, мало, но будьте внимательны при выборе этих людей.

1~ Определение

Клиент. В этой книге я использую термин *клиент* по отношению к любому человеку, заказавшему создание Web-сайта. Это может быть обычный клиент, с которым вы заключаете договор, ваш непосредственный начальник или коллеги, которые доверяют вам создание сайта компании.

Когда вы официально определили людей, принимающих решения, удостоверьтесь в том, что каждому из них вы сообщили о задачах сайта. Неважно, клиенты это или коллеги. Важно, чтобы все согласилось с задачами сайта. Если в дальнейшем задачи изменятся, кому-то придется платить за дополнительное время и материалы. Вы можете избежать дополнительных проблем, если после ознакомления с задачами проекта попросите клиентов, коллег или начальников поставить подпись, означающую, что задачи одобрены.

Подписанный клиентом документ, подтверждающий первоначальный объем, дает вам преимущества, если придется просить о дополнительной оплате, когда, например, художница Мэри Хой решит, что нужно разместить 30, а не 15 отсканированных изображений картин. Никогда не платите из собственного кармана, если кто-то изменил свое решение.

Определите функциональные требования заблаговременно

Утвержденные задачи - лучший способ контролировать создание малобюджетного сайта, но помимо этого необходимо, чтобы работа была основана на четко определенных функциональных требованиях.

Давайте снова вернемся к сайту Something Blue и составим список таких требований. Как и раньше, вообразим, что вы Web-профессионал, работающий

на дому, - профессионалы могут выполнять свою работу в любых условиях. Можно начать с опроса клиента на предмет того, что он хочет видеть на своем сайте. В результате опроса вы составите список пожеланий, среди которых могут быть следующие:

- я хочу, чтобы посетители могли искать на сайте изображения, а также иметь возможность, щелкая по ним, увеличить картинку и рассмотреть любую деталь, вплоть до каждого мазка кисти;
- у моих посетителей должна быть возможность быстро просмотреть все изображения, а затем отобразить понравившуюся работу в увеличенном масштабе;
- у моей мамы медленное коммутируемое соединение, но я бы хотела, чтобы она могла просматривать мой сайт. Ненавижу сайты, которые долго загружаются;
- посетители должны иметь возможность связаться со мной без особых проблем. Я хочу, чтобы они могли позвонить мне или послать сообщение при желании купить мою картину или организовать выставку.

Сразу хочу отметить, что не все из этих пожеланий можно реализовать на сайте с маленьким бюджетом. Ваша задача как профессионала - дать клиенту понять, что вы прислушались ко всем его идеям и серьезно к ним отнеслись, а затем постараться сократить список пожеланий клиента, преобразовав его в список функциональных требований, выполнимых при имеющихся ресурсах:

-Примечание

Функциональные требования для сайта Something Blue (версия один). Сайт Something Blue обладает разнообразными возможностями, но некоторые из них слишком дороги для первоначального запуска. Реализация всех потенциальных требований выльется в сумму, превосходящую имеющиеся средства. Мы сократили список до нескольких основных требований, которые возможно осуществить в рамках текущего бюджета и времени. Самые дорогостоящие требования, которые не были реализованы на данном этапе, стоит сохранить для следующей версии сайта.

Запомните следующее: если вам не удастся убедить клиента в необходимости сократить требования, тогда лучше всего отказаться от выполнения проекта. На начальном этапе следует точно определить то, что должно быть и чего не должно быть на сайте. Если это сделать не удастся, недовольными в итоге останутся абсолютно все. Долгие часы вы будете работать, не получая за это ничего, а конечный результат не понравится вашему клиенту.

Таблица 2.1. Перечень требований и стоимость их реализации

Требование	Комментарии	Цена
Посетители смогут посмотреть увеличенное изображение, щелкнув по маленькой картинке. Размер большого изображения будет примерно 600 пикселей шириной, такой размер поддерживает большинство мониторов	Это выполнимая с экономической точки зрения альтернатива запрашиваемой ранее технологии увеличения и прокрутки, которая позволит посетителям увеличить каждое изображение. Это не позволит посетителям рассмотреть картину вплоть до таких мельчайших деталей, как мазки кисти, тем не менее размеры изображения будут достаточно большие. Дополнительный плюс такой альтернативы состоит в том, что пользователям с низкоскоростным подключением к Интернету не придется ждать загрузки увеличенных изображений, пока они сами не выберут эту опцию	\$
Посетители смогут быстро просмотреть все изображения благодаря картинкам в уменьшенном масштабе		\$
Сайт будет быстро загружаться, и посетители с низкоскоростным подключением смогут легко его просматривать		\$
Посетителям будет легко найти контактную информацию		\$
Возможные требования для следующей фазы Web-развития		
Сайт должен быть оснащен машиной поиска	Это может дорого стоить и, возможно, в данный момент это излишне, поскольку вы работаете с таким небольшим количеством изображений. Но вы можете сохранить эту идею для следующей стадии производства, если в дальнейшем вы добавите на сайт новые изображения	

Наиболее важный фактор в создании малобюджетного сайта - это сознательное уменьшение объема до ограниченного количества выполнимых функциональных требований. Помните: лучше меньше, да лучше.

- Как получить золото из песка —————

Даже при большом бюджете вам будет трудно выполнить все пожелания клиента. Знание того, какие из требований клиента можно выполнить за короткий срок, поможет вам сделать сайт с четко поставленными задачами, не выходя за рамки времени и бюджета. Задачи, отложенные на следующий этап, принесут вам прибыль впоследствии. Довольный клиент может позвать вас позже, когда у него появятся деньги на развитие сайта. Многие Web-профессионалы, работающие по договорам, сейчас держатся на плаву благодаря старым клиентам, предлагающим пусть небольшую, но регулярную работу. Постарайтесь завести постоянных клиентов - это обеспечит вам некоторую стабильность.

«

Каждый раз, когда вас просят изменить объем сайта, напоминайте о списке функциональных требований, одобренных вашим клиентом, и обсуждайте, как лучше этот список изменить в соответствии с новым требованием. Возможно, вам придется заменить одно требование другим. Сообщите клиенту о таких заменах и попросите его помочь вам оценить их стоимость. Кстати, это хорошая возможность поговорить о выделении дополнительных средств.

Как отмечалось ранее, в процессе создания сайта его объем может меняться. Если во время вашей работы художница Мэри скажет, что ей хочется еще и продавать картины на сайте *Something Blue*, вы просто обязаны сообщить ей о том, сколько подобные изменения будут стоить. Каждый раз, когда клиент заставляет вас менять список требований к проекту, создавайте новую версию этого списка. Храните старые версии списка требований - так вы сможете проследить за развитием (или изменением) задач проекта. Если четвертая версия списка значительно отличается от первой, проверьте, хватает ли первоначально выделенных средств на реализацию проекта.

Обычно новые идеи, с которыми приходят клиенты и начальники, достаточно просты для выполнения, поэтому вам не придется просить дополнительных средств. В любом случае составляйте новые версии перечня требований и не забывайте показывать их клиенту или начальнику. Если делать это согласованно с клиентом, он, по крайней мере, поймет, что любые изменения объемов должны быть официально подтверждены. Также это помешает внесению мелких изменений, и вам не придется платить за изменения объема из собственного кармана.

Утвердите технические требования

Определение технических требований так же важно с точки зрения экономии средств, как и определение целей проекта или функциональных требований. В первые годы моей карьеры Web-специалиста я работала над несколькими сайтами, технические требования которых были определены нечетко. Это часто создавало нам проблемы. Как-то мы наняли консультанта, который сделал нам прекрасный сайт. Сайт был великолепен, когда его просматривали в последней версии Netscape Navigator, однако в Internet Explorer он не работал. Мы не упомянули, какие браузеры должен поддерживать сайт, и наш консультант решил, что сайт должен работать только в Netscape. (У вас, скорее всего, подобное тоже случалось.) После небольшого выговора наш консультант изменил разметку и код так, что потом сайт смог поддерживать большее количество браузеров. А нам был выставлен счет за выполнение доцелнительной работы. Если бы мы сразу же оговорили, какие браузеры должен поддерживать сайт, то смогли бы сэкономить и время, и деньги. Более того, нам бы не пришлось применять санкции, которые потребовались, чтобы ускорить внесение исправлений.

Четко определенный список технических требований защитит ваш бюджет от затрат на изменения, вносимые в почти готовый проект. С самого начала постарайтесь объяснить, что вам надо. После этого составьте документ по техническим стандартам и другим требованиям Web-разработки.

Если вы клиент, обязательно сообщите эту информацию разработчику сайта. Если вы разработчик, вам, возможно, придется помочь клиенту определить необходимые технические требования. Первый этап работы может показаться слишком длительным, однако в дальнейшем он экономит массу времени и средств. Приведу несколько технических требований, благодаря которым я успешно сэкономила время, деньги и собственные силы:

- поддерживаемые сайтом браузеры и операционные системы. Вы подумали о браузерах версии 4.0? Только о современных браузерах? Карманных компьютерах? Важно ли вам, будет сайт работать на PC, или он создан только для пользователей Macintosh? Удостоверьтесь, что вы составили подробный и точный список компьютеров, на которых ваш сайт будет правильно отображаться. Этот список вам пригодится при тестировании сайта. (Не ждите завершения проекта, чтобы протестировать сайт на предмет поддержки различных машин и браузеров. Намного дешевле выявить все ошибки в процессе разработки, чем исправлять их, когда сайт уже будет готов.);
- HTML и CSS. Есть ли у вас какие-нибудь особенные идеи насчет использования HTML и CSS? (Подсказка: Да. Отложу эту тему и вопрос экономии денег до главы 7);

- сценарии на стороне клиента. Уточните, как использовать JavaScript. Лично я верю, что любая функция должна работать при выключенном JavaScript. Это придется по вкусу многим пользователям, включая тех, у кого нет браузера, поддерживающего этот язык;
- серверы и базы данных. Уточните всю информацию насчет сервера, на котором будет запущен сайт, а также насчет поддержки программного обеспечения и базы данных;
- требования к пропускной способности. Не поленитесь узнать, какое подключение к Интернету у большинства посетителей вашего сайта: низкоскоростное или высокоскоростное (DSL или кабельное). В первом случае вам придется позаботиться об оптимизации скорости соединения, во втором - нет. В локальных сетях все подключены по высокоскоростному доступу. Что же касается общедоступных сайтов, о которых, в общем-то, здесь и говорится, то практически у половины их посетителей скорость dial-up-соединения низкая - 56 Кбит/с и менее. На сайте [Web Site Optimization](http://www.websiteoptimization.com/bw) вы сможете ознакомиться со статистикой скорости подключения к Интернету у пользователей на работе и дома (www.websiteoptimization.com/bw).

Проведя исследования на основе этого перечня, вы можете смело садиться за составление списка технических требований. Он должен быть кратким и легким для восприятия. Подготовив список технических требований, вы избежите ситуаций, когда приходится платить за установку элементов, которые невозможно использовать. Следующая глава содержит мало теории: в ней собраны примеры, предназначенные для того, чтобы вы могли научиться расходовать средства экономно.

Преимущества тщательного планирования

Теперь, когда определены задачи проекта, технические и функциональные требования, вы можете приступить к работе. Ваша жизнь станет намного проще, а работа эффективней, поскольку вы не пожалели времени на составление плана. В процессе работы вы будете вносить какие-то изменения, но не забывайте их фиксировать. Вам будет проще требовать дополнительные средства, если они понадобятся, а также распределять уже имеющиеся.

Эта глава - всего лишь вершина айсберга. О планировании сайта и процессе Web-производства можно рассказать очень много, многому еще можно научиться. Я дала всего лишь несколько советов, которые помогут вам. Не жалейте времени на обучение.

Глава 3- Разработка интерфейса

Щедрость означает не дать много, а дать прямо сейчас.
ЖандеЛабрюе(1645-1696)

Основные идеи главы

1. Разработайте интерфейс за небольшую цену.

Для того чтобы обеспечить простоту использования и просмотра сайта, необходимо побеспокоиться об этом на начальном этапе создания проекта. Тренируйтесь, чтобы умение создавать удобный интерфейс стало для вас обычным делом.

2. Бережно относитесь к удобству просмотра.

С самого начала работы над сайтом постоянно проводите небольшие тестирования. Вашим девизом должны стать следующие слова: «Намылить, смыть, повторить». После каждого шага тестируйте!

3. Пусть все будет проще.

Исследования, проводимые на опытном образце сайта, рассматриваются в этой главе в качестве примера: попросите нескольких потенциальных посетителей выполнить определенные задания. Не думайте, что это займет много времени: написание сценария, опрос нескольких товарищей или коллег и анализ полученных результатов потребуют всего нескольких часов.

4. Используйте инструментарий интерфейса.

Специалисты, ограниченные рамками небольшого бюджета, должны стать экспертами по многим вопросам Web-производства, включая разработку интерфейсов. Если в этой области вы не особо сильны, постарайтесь изучить хотя бы основы. Внимательно ознакомьтесь с советами по дизайну интерфейса, приводимыми в этой главе. Не жалейте времени на пополнение своих знаний. Вам придется постоянно использовать эти знания, и время, которое вы потратите сейчас, впоследствии обязательно окупится.

5. Пополняйте HTML-библиотеку.

То, как вы составляете разметку, имеет огромное значение с точки зрения удобства просмотра сайта и его доступности для начинающих пользователей. Иногда на создание хорошей разметки уходит больше времени, чем на создание плохой; однако если вы начнете создавать библиотеку грамотных образцов разметки, то в дальнейшем она может вам пригодиться для новых работ. Таким образом, вы сможете сэкономить массу средств по мере того, как будет совершенствоваться ваше мастерство создания разметки, а количество посетителей ваших сайтов будет увеличиваться.

В этой главе говорится о том, как добиться простоты использования и просмотра сайта при наличии скромного бюджета. Чтобы не растратить весь бюджет, Web-профессионалы должны подходить к этому важному аспекту создания сайтов, имея выгодную, четко продуманную стратегию. Такая стратегия должна состоять из трех частей:

1. Постоянно, с самого начала работы проводите небольшие тестирования сайта с привлечением пользователей.
2. Создайте инструментарий работы с интерфейсом, который вы будете постоянно использовать в процессе подготовки сайта.
3. Создайте библиотеку удобных и доступных HTML-разметок, которые можно использовать постоянно.

-Определение

Простота использования сайта. В учебниках по Web-дизайну определение простоты использования сайта куда более громоздкое, нежели в моей книге. Не исключаю возможности, что таким определением этого термина я могу обидеть знатоков искусства и науки простоты использования сайта. Но мне кажется, что привести простое определение гораздо удобнее: простота использования сайта - это качество сайта, заключающееся в интуитивном характере его навигации и использования.

Простота использования ставит вас перед выбором: заставить посетителей задуматься и изучать, как перемещаться по сайту, или сделать так, чтобы им было доступно все содержимое сайта. Эта книга об экономии - принципе, которому вы должны постоянно следовать в своей работе.

Термины «достижение простоты использования сайта» и «разработка правильного интерфейса сайта» во многом синонимичны.

г-Определение

Тестирование с привлечением пользователей. Такой вид тестирования используют при разработке не только Internet-продуктов, но и системы телефонной связи, чайника или сайта. Оно помогает наблюдать за функционированием модели раньше, чем она появится в продаже, увидеть то, как ей будут пользоваться в реальной жизни. Благодаря отзывам посетителей сайта вы сможете узнать о недочетах, упущенных в процессе работы.

Два или три года назад в команду по созданию Web-сайтов входили эксперты по разработке интерфейса и простоте использования, обязанности которых сильно отличались. Программисты создавали коды, дизайнеры разрабатывали

шаблоны, а инженеры проводили тестирования с привлечением пользователей и часто отвечали за общее впечатление посетителей от сайта.

Если у вас есть возможность нанять людей, в задачу которых будет входить обеспечение простоты использования, - великолепно! Воспользуйтесь преимуществом больших ресурсов! Но, как мы говорили в самом начале, большинство малобюджетных сайтов (впрочем, как и многих других) ограничено финансами. Web-профессионалы, занимающиеся созданием малобюджетных сайтов, вынуждены отвечать за многие аспекты производства. Если вы держите в руках эту книгу, велика вероятность, что вы как раз и есть тот самый счастливец, озадаченный вопросом простоты использования сайта, но у вас нет необходимых ресурсов, чтобы обеспечить эту простоту в полной мере. Вышеописанная стратегия, состоящая из трех пунктов, поможет вам достичь необходимого уровня, не затрачивая на это лишние часы и деньги.

Экономичное тестирование

Тестирование с привлечением пользователей может потребовать солидных денежных вложений, если они у вас имеются: попробуйте воспользоваться услугами фирмы, специализирующейся в этой области, либо установить лабораторию по исследованию интерфейса с видеокамерами и прочими устройствами, чтобы наблюдать за группой посетителей сайта-образца (как они просматривают сайт или пытаются выполнить какие-либо задания на нем). Но, если у вас маленький бюджет, постарайтесь прибегнуть к какому-либо другому способу, чтобы добиться простоты использования.

Мои родители познакомились в университете Детройта в 1965 году. Их свидания были очень скромными: мой отец обычно приглашал маму куда-нибудь попить чаю с печеньем. Кроме того, они посещали спортивные мероприятия, проводившиеся в колледже. Два года спустя родители поженились. Они до сих пор вместе и очень счастливы. Конечно, если бы их встречи были более разнообразными, они смогли бы лучше узнать город, однако баскетбола и крекеров им вполне хватало, чтобы хорошо провести время и понять, что они на самом деле созданы друг для друга. Эту историю я привожу в качестве рецепта успешного тестирования с привлечением пользователей.

Проводите тестирование с самого начала работы

Даже если план вашего сайта представляет собой лишь рисунки на салфетках (рис. 3.1), вполне можно начинать тестирование с привлечением пользователей.

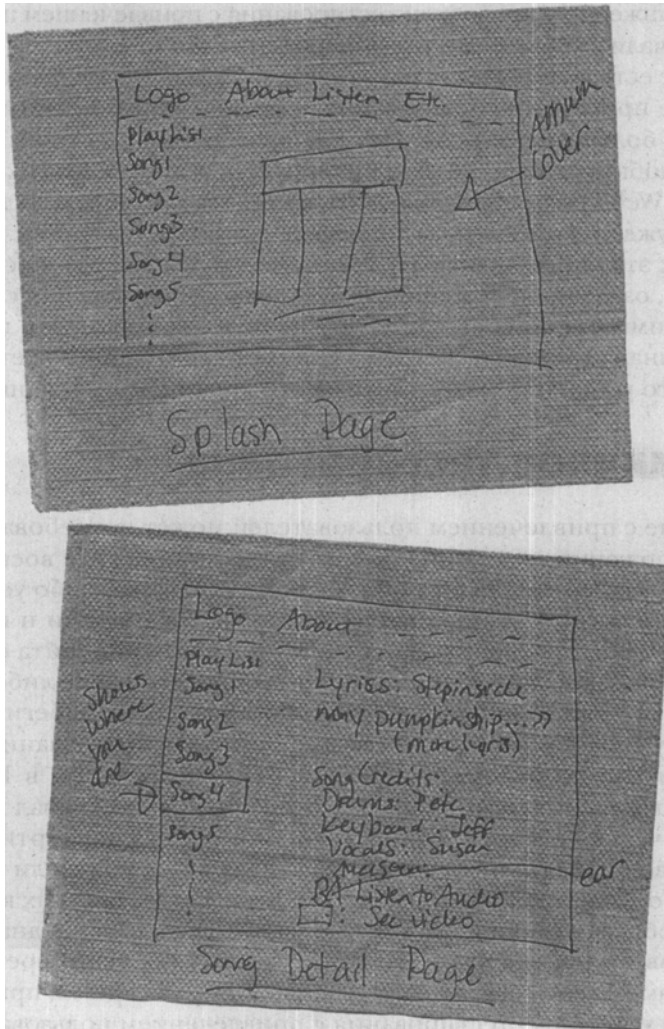


Рис. 3.1. Как только ручка коснулась обычной салфетки, можете начинать театирование с привлечением пользователей. Вы сэкономите время и деньги, если позволите потенциальным пользователям определить недостатки ваших идей относительно интерфейса. Чем раньше вы получите комментарии насчет вашей работы, тем меньше времени вы потратите на исправление ошибок

Как только у вас появятся идеи относительно формы и построения сайта, можете смело приступать к опросу потенциальных посетителей.

В 2000 году, когда я создавала сайт для Нью-Йоркской публичной библиотеки, пришлось обратиться к фирме Small Company (www.smallco.net) с просьбой разместить электронную версию печатного календаря на сайте. Прежде чем заняться планированием или производством, Альберт Харум-Альварес (Alebert Harum-Alvares), основатель Small Company, посетил несколько отделов библиотеки, чтобы пообщаться с посетителями.

Он использовал печатную версию календаря в качестве основы для бесед. На этом этапе работы его интересовал не интерфейс, а даты мероприятий и использование календаря. Он просидел половину дня за маленьким столом в одном из залов Нью-Йоркской публичной библиотеки, разговаривая с представителями разных групп читателей: детьми, подростками, взрослыми, взрослыми с маленькими детьми - и задавая им следующие вопросы:

- как далеко вы готовы поехать, чтобы принять участие в каком-либо мероприятии?
- что для вас главное при выборе мероприятия?
- планируете ли вы посещение мероприятия, либо это происходит спонтанно?

От посетителей удалось узнать несколько важных вещей. Подростки, возможно, наиболее раскованные из всех посетителей, были готовы ехать далеко и хотели бы ознакомиться с полным списком всех мероприятий, происходящих в городе. Взрослые с маленькими детьми, наоборот, не хотели ехать куда-либо, их интересовали мероприятия, происходящие поблизости от дома. Наш интерфейс должен был удовлетворить самые разные потребности.

Подобные результаты удивили: мы ожидали, что желания наших пользователей будут совпадать. Различные результаты по разным возрастным группам нашли свое отражение в дизайне сайта, позволяющем угодить нашим посетителям.

Стоимость такого предварительного тестирования была небольшая; нам пришлось заплатить за несколько часов работы Альберта. Выгода заключалась в том, что еще перед началом работы мы немного изменили направленность сайта, так как наши ожидания относительно потенциальных пользователей не оправдались (рис. 3.2).

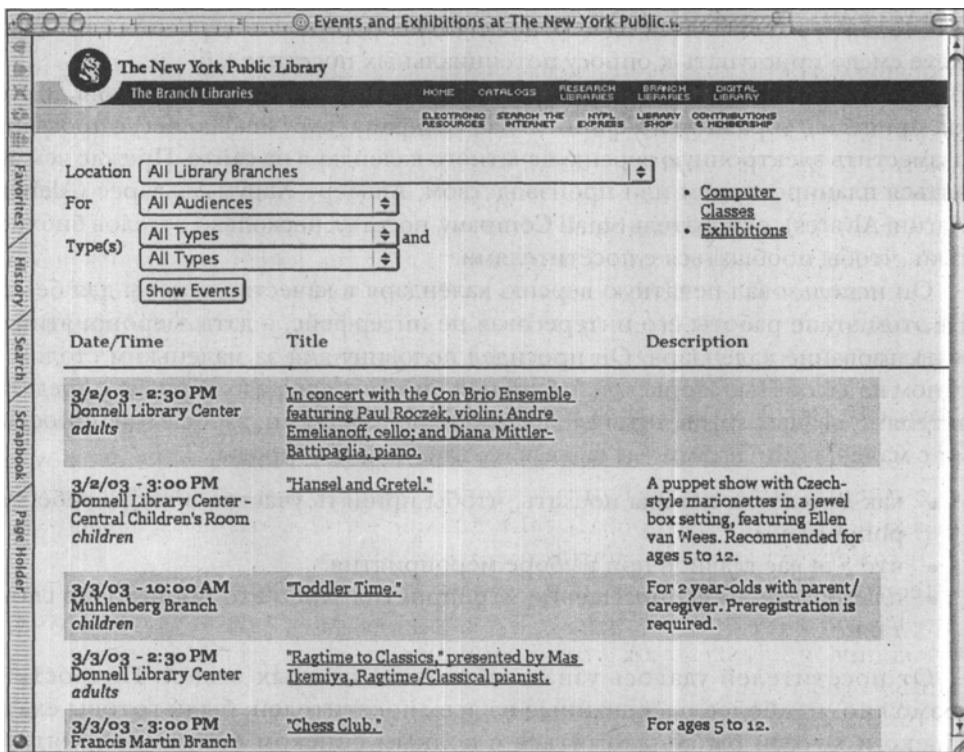


Рис. 3.2. Прежде чем запустить виртуальный календарь событий, наш консультант провел несколько часов в одном из залов библиотеки, разговаривая с потенциальными читателями на предмет того, как они обычно пользуются печатной версией календаря событий

Г Как получить золото из песка

Проведя исследование в самом начале, мы подкорректировали важные положения относительно пользовательских потребностей и исправили несколько организационных ошибок. Вы можете избежать затрат на исправление интерфейса в процессе производства (или даже после него), если выполните тестирование с помощью пользователей.

Намылить, смыть, повторить

Хорошие новости для вашего бюджета: достаточно протестировать 6-12 пользователей. После опроса такого количества людей задача по уменьшению расходов будет выполнена. В своей замечательной книге «Не заставляйте меня думать!» Стив Краг дает следующий совет: «Если вы действительно хотите узнать, хорош ваш сайт или нет, попросите кого-нибудь из соседей посетить его, а сами понаблюдайте за ним (угостив соседа пивом)».

Если попросить трех друзей или коллег в течение 15 мин побродить по сайту, это поможет выявить проблемы интерфейса. Тестирования надо проводить часто. Каждый раз, добавляя на сайт что-то новое, выполняйте тестирование, основанное на заданиях, описываемых в следующих разделах.

Простое пользовательское исследование

Соберите небольшую группу пользователей. Опять-таки, 5-6 человек будет достаточно, хотя даже два или три смогут сообщить вам самое необходимое. Неважно, кто будет тестировать сайт, главное - чтобы это не были ваши коллеги, тоже работающие над этим проектом.

Тестирование проводите с партнером. Один из вас должен разговаривать с пользователями, другой - делать заметки. Если у каждого члена группы есть копия списка заданий, кто-то один сможет делать пометки рядом с названиями заданий.

Пригласите пользователей к компьютеру. Поблагодарите их за то, что они не пожалели времени на ваше задание. Скажите им, что сейчас они просматривают неоконченную версию вашего сайта и вы бы хотели, чтобы они проверили сайт на предмет выполнения нескольких задач. Введите людей в курс дела - они должны понимать, что тестируется именно сайт, отвечать честно и не чувствовать себя глупо. (Если люди чувствуют себя глупо, они часто неискренни.)

Давайте пользователям реальные задания. Например, если вы работаете над Интернет-магазином, можете попросить пользователей найти какой-либо предмет, добавить его в корзину, а затем купить его. Если сайт представляет собой галерею изображений с возможностью поиска, попросите кого-либо найти то или иное изображение и добавить его в папку Избранное. Это поможет определить, в порядке ли метаданные и справляется ли со своими задачами машина поиска. Количество и продолжительность заданий должны

быть короткими; пятнадцатиминутного сеанса будет более чем достаточно. Отведите на выполнение каждого задания определенное время. Здесь заключается наиболее важная часть: попросите, чтобы помощники проговаривали свои мысли вслух. В это время ваш коллега, ответственный за запись, должен зафиксировать несколько ключевых идей. Если какое-то из предложенных заданий кажется пользователям невыполнимым, пусть они переходят к следующему заданию.

Подумайте о символическом вознаграждении: это может быть сертификат участника мероприятия, небольшая сумма денег - все, что вы сможете предложить. Пользователи, тестиовавшие ваш сайт, потратили свое время, и вам следует их отблагодарить.

Изучив данные, полученные при тестировании, измените ваш сайт. В дальнейшем, если тестирования проводить регулярно, подобные преобразования будут менее масштабными и обойдутся дешевле.

Малобюджетный инструментарий для создания интерфейса

В последних классах школы и в колледже я работала официанткой и, должна вам признаться, плохо выполняла свои обязанности. Помимо того что я была медлительной и неорганизованной, я мешала посетителям обедать. Мое надоедливое присутствие не позволяло клиентам в полной мере насладиться едой и общением - тем, ради чего люди приходят в подобные заведения.

Хочется верить, что Web-профессионал из меня получился более грамотный, чем официантка. Однако, поработав официанткой, я поняла, чего хочет потребитель, а именно: наслаждаться процессом, а не думать о нем. Став Web-профессионалом, я долго трудилась над тем, чтобы создавать интерфейсы, которые служат пользователям, а не привлекают к себе внимание.

Если пользователям приходится размышлять об интерфейсе или о пригодности официантки, это должно настораживать. Хороший интерфейс, как и хорошая официантка, незаметен. Неважно, где вы работаете - в «Елках-палках» или «Метрополе», - о качестве вашего обслуживания посетители задумываться не должны.

Много времени уходит на то, чтобы стать хорошей официанткой, - примерно столько же, сколько требуется, чтобы стать хорошим разработчиком интерфейса. Умение создать удобный интерфейс вырабатывается далеко не у всех. Если вы трудитесь над малобюджетным проектом, если у вас нет денег, чтобы

нанять специалиста по интерфейсу, и если сами вы таким специалистом не являетесь, придется учиться это делать.

Я хочу помочь вам стать настоящим специалистом и предлагаю воспользоваться следующими советами. Как бы я хотела получить подобные советы, когда начинала работать официанткой!

Советы по меню и навигации

Самое главное в интерфейсе Web-сайта - отсутствие особых правил. Однако несколько важных советов могут пригодиться. Назову мои самые любимые.

Пусть в вашем окне всегда горит свет

Не все посетители зайдут сразу на главную страницу вашего сайта. Некоторые придут с других страниц через поисковые системы. Убедитесь, что ваш логотип, название сайта и элементы навигации присутствуют на каждой странице. Таким образом, пользователи, которые заходят на ваш сайт через «черный вход», смогут сориентироваться так же просто, как и те, кто открывают «главную дверь».

Правило трех щелчков

Правило трех щелчков полезно для навигации. Многие специалисты по интерфейсу скажут вам, что пользователи не должны щелкать более трех раз, чтобы найти то, что они ищут. Например, посетитель, щелкнув один раз, может попасть с главной страницы виртуального магазина в отдел женской одежды, затем - в отдел женской обуви, а оттуда - в секцию обуви необходимого стиля. Правило трех щелчков - это основной совет, однако вполне вероятно, что более важным, чем число щелчков, является простота поиска каждой ссылки. Стив Краг в книге «Не заставляйте меня думать» тоже обсуждает этот вопрос: «Если ваш сайт наделен интуитивной навигацией и информация подается на должном уровне, а также если пользователь может без труда найти не только основную, но и специфическую информацию, то количество щелчков вам простят».

Живой и понятный язык

Манеру письма для Web-сайта мы рассмотрим в следующей главе, а здесь я скажу, что язык повествования должен быть простым, живым и понятным. Пользователи не особенно одобряют применение специальных терминов и сложных фраз.

Правило пяти пунктов меню

Последний совет - это правило пяти. Оно рекомендует создателям Web-сайтов использовать меню, которое будет состоять не более чем из пяти пунктов. Причина проста: очень часто пользователи бегло просматривают пункты меню, когда сталкиваются с бесконечным списком. «Слишком большой выбор» равносителен полному отсутствию выбора. Существует ряд исключений из этого правила.

Например, владельцев электронного аукциона eBay.com мало беспокоит то, что его посетители теряются в огромном списке меню. Но сами посетители очень заинтересованы в использовании данного сайта. Если же вы на своем сайте продаете электронных собак Sony Aibo по \$200, правилом пяти лучше не пренебрегать.

Многоразовая HTML-библиотека

В конце 2003 года в Нью-Йорке прошел небывалый снегопад. Моя коллега поскользнулась по пути на работу и сломала правую руку. Через пару дней после перелома мы общались с ней по телефону и пытались договориться о встрече. «Подожди, я посмотрю, могу ли я еще пользоваться программой-календарем», - шепотом сказала она. Поскольку правая рука у нее была сломана, пользоваться мышкой она не могла, поэтому всю навигацию по календарю ей пришлось осуществлять при помощи клавиатуры. Задача была не из легких. До некоторых частей экрана, необходимых ей, можно было добраться с большим трудом, используя в качестве перехода клавишу табуляции. К несчастью для моей коллеги, перехода при помощи табуляции интерфейс календаря не предусматривал, вот почему ей пришлось пролистать длинный список меню до тех пор, пока она не добралась до команды **Назначить новую встречу**.

- Как получить золото из песка

От того, как вы пишете HTML-код, может сильно зависеть простота использования вашего сайта. Сайты со сложной разметкой страниц загружаются медленно. Плохо написанный HTML-код не годится для работы. Подобного рода проблемы легко исправить при помощи грамотной разметки. Грамотно написанный HTML-код в дальнейшем сэкономит массу времени, которое можно будет потратить на улучшение интерфейса. Более подробно о плюсах хорошей разметки мы поговорим в главе 7.

HTML-код у сетевого календаря был несовершенен, поэтому возникли трудности при его использовании. Такие недоработки дорого стоят. Следует заметить, что простота применения Web-сайта во многом зависит от хорошей разметки. Потратив время на создание многоразовой HTML-библиотеки, у которой будет функция перехода по клавише табуляции, вы сможете в дальнейшем использовать созданные разметки любое количество раз, экономя средства и собственные силы.

Разметка: простота использования и доступность Web-сайта

Хорошо продуманная HTML-библиотека поможет сделать ваш сайт простым и доступным, даже для пользователей с ограниченными физическими возможностями.

Web-специалисты часто работают с клиентами, которые думают, что им не по карману оплатить создание доступного сайта. Действительно, эта услуга может стоить очень дорого, поскольку для того, чтобы сделать сайт доступным для пользователей с ограниченными физическими возможностями, вам придется потратить больше времени, а время - это деньги. Однако есть несколько приемов, используя которые можно заложить основы доступного сайта с самого начала работы:

- доступность поможет не только тем пользователям, физические возможности которых ограничены постоянно, например слепым или людям с повреждениями опорно-двигательного аппарата, но и тем, чьи физические возможности ограничены временно, как, например, моей упавшей на лед коллеге. Не нужно создавать сайт, доступный избранному числу посетителей;
- доступность сделает ваш сайт более популярным среди посетителей, использующих нетрадиционные устройства, такие как карманные компьютеры и мобильные телефоны. Мы поговорим о создании сайтов для нетрадиционных устройств в главе 7;
- для некоторых владельцев недоступность их сайтов может обернуться серьезными судебными тяжбами, а мы все знаем, как это дорого обходится;
- лучше сразу позаботиться о доступности вашего сайта. Если включить доступность в требования к еще не созданному сайту, это займет всего 3%

от бюджета, а переделка сайта будет стоить намного больше. Например, на малобюджетном сайте, созданном без CMS, добавление возможностей навигации при помощи перехода по клавише табуляции или альтернативной функции для посетителей, не имеющих возможности использовать JavaScript, займет несколько дополнительных минут на страницу. Однако время и цена, потраченные на преобразования, выполняемые каждый день, - ничто по сравнению с полной переделкой сайта. Добавлять небольшие изменения ежедневно все равно, что откладывать деньги на черный день;

- некоторые сайты по закону должны быть доступными для людей с физическими недостатками. В статье 508 американского закона по реабилитации говорится о том, чтобы электронные информационные источники федеральных учреждений были доступны людям с ограниченными физическими возможностями. Работа над .gov-сайтами или над .edu- или .org-сайтами, финансируемыми из федеральных источников, вполне возможно, потребует принять во внимание статью 508. Сейчас очень многие правительственные учреждения начинают предъявлять подобные требования к такого рода сайтам¹.

-Угрозе бюджету—

Доступность. Если вы интуитивно чувствуете, что в дальнейшем вам придется сделать сайт доступным для людей с ограниченными физическими возможностями, но предпочитаете экономить деньги и отложить этот вопрос на потом, подумайте хорошенько еще раз. Позаботившись о таких людях с самого начала, вы затратите из своего бюджета дополнительные 3%. При этом переделка готового сайта будет стоить намного дороже. Воспользуйтесь мудрым советом - включите доступность в список основных задач с первого дня работы.

Следующие советы по разметке помогут вам создать сайт, доступный максимальному числу пользователей. Если обычно вы не заостряли внимание на HTML, должно пройти некоторое время, прежде чем вы поймете, что тщательная работа над HTML не отнимает лишнее время. Потерпите немного. Скоро вы освоите этот способ работы, и ваша работа станет более эффективной и контролируемой, чем когда-либо. Потраченное время принесет свои плоды в виде большего числа посетителей и меньшего количества жалоб.

¹ Несмотря на то что за рубежом в последнее время предъявляются строгие требования к сайтам с точки зрения их доступности, российских сайтов это касается мало. Обеспечивать доступность или нет - дело Web-профессионала, создающего сайт. - *Прим. науч. ред.*

Порядок перехода по клавише табуляции

Вернемся к моей коллеге, чья временная травма помешала ей использовать виртуальный календарь. Порядок перехода по клавише табуляции - это необходимая составляющая правильного интерфейса, которую многие создатели сайтов часто забывают.

Проведите следующий эксперимент: отключите мышку от компьютера на некоторое время и попробуйте поработать в Интернете, пользуясь исключительно клавиатурой. Вы сразу узнаете, как это сложно, и быстро поймете, как просто это должно быть. Переключение с одного элемента сайта на другой при помощи клавиши табуляции поможет пользователям перемещаться по сайту быстро и эффективно. У пользователя должна быть возможность обходиться без мышки. Также он должен иметь возможность перемещаться по сайту при помощи клавиши табуляции, пропуская ненужные элементы сайта, с такой же скоростью, с какой пользователь воспринимает сайт визуально.

Посмотрите на рис. 3.3, где изображен один из моих почтовых ящиков.

Если бы мне нужно было просмотреть входящие сообщения, то сначала пришлось бы пройти по всем пунктам с левой стороны страницы, а затем - сверху. Подумайте о моей коллеге, которая временно не могла работать правой рукой. Подумайте о людях, которые никогда не смогут пользоваться мышкой! Такой сайт, как этот, потребует непомерных затрат времени и будет очень утомительным в использовании. Помните о том, что сайты вы создаете для того, чтобы помогать людям.

Существует простой способ решения проблемы перехода по клавише табуляции: подумайте об этом заранее. Еще раз взгляните на рис. 3.3 и на коды ниже. Оба кода представляют собой разметку для почтового сайта, однако во втором продуман порядок перехода по табуляции.

```
<table>
<tr>
<td><input type="checkbox" name="somename" value="001" /></td>
<td><a href="openmail.html">Carrie Bickner</a></td>
<td>03/01/2 003</td><td>2313 8 6</td><td>SOW Migration Project</td>
</tr>
```

Чтобы добраться до первого сообщения (рис. 3.3), пользователь вынужден пролистать все меню левой панели.

```
<table>
<tr>
<td><input type="checkbox" name="somename" value="001"
tabindex="1" /></td>
```

```
<td xhref="openmail.html" tabindex="2">Carrie Bickner</td>
<td>03/01/2003</td><td>2313 86</td><td>SOW Migration Project</td>
</tr>
```

Проблема порядка перехода по клавише таблицы легко поправима при добавлении атрибута `tabindex` в элемент формы или ссылки, которая, по вашему мнению, должна быть выделена, когда посетитель начнет просмотр. Ссылка с атрибутом `tabindex="1"` - это первая ссылка, с которой у посетителя начнется переход по клавише таблицы. Когда атрибут `tabindex` станет частью пополне-

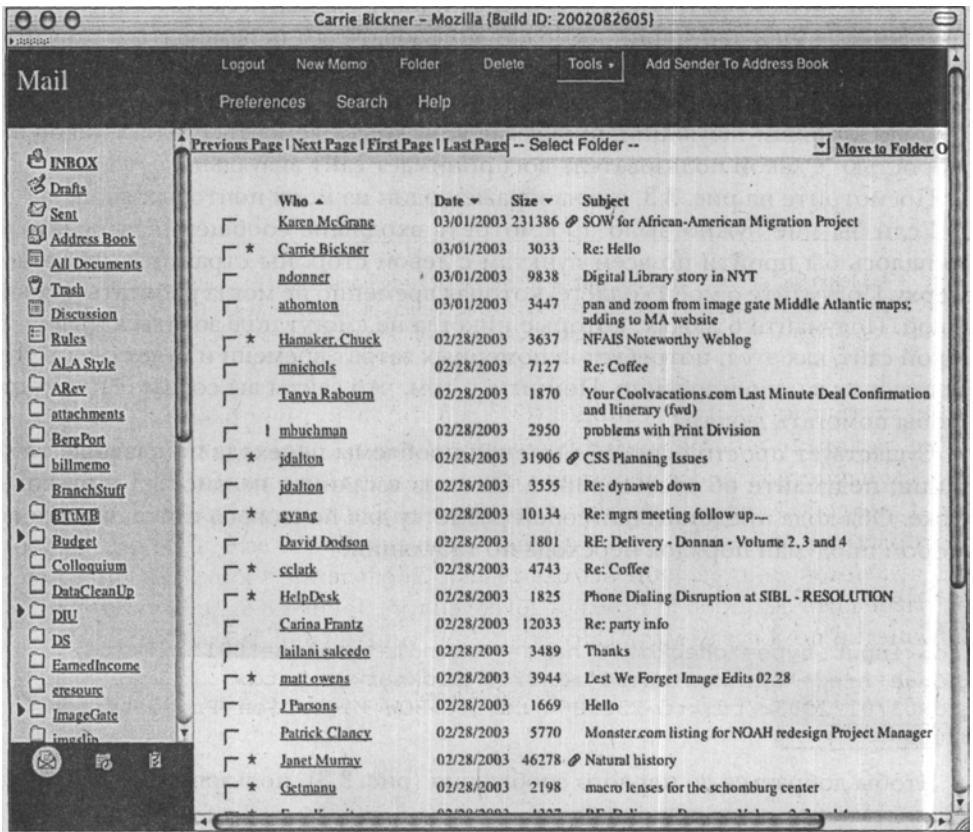


Рис. 3.3. Доступность - одна из особых проблем интерфейса. Здесь изображена почтовая страница, в которой порядок перехода по клавише таблицы далек от идеала: для того чтобы просмотреть новые письма, пользователю приходится пролистывать все пункты меню сверху, а затем слева. Такую проблему легко решить, если позаботиться о порядке перехода по клавише таблицы

мой HTML-библиотеки, вы сами убедитесь в его необходимости. Включение подобных атрибутов разметки значительно сократит затраты на создание HTML-кода.

Формы

Парадокс заключается в следующем: невероятно *легко* создавать формы, *сложные* для всех, кто не может работать с мышкой. Посмотрите на форму, показанную на рис. 3.4.

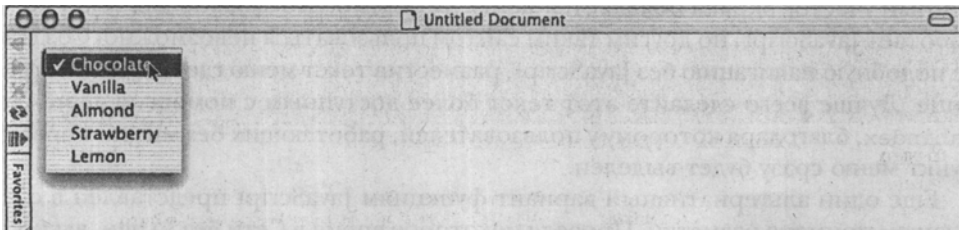


Рис. 3.4. Меню без кнопки типа Submit, если вы не можете пользоваться мышкой, напоминает дверь без ручки. Когда вы выбираете первый пункт меню - в данном случае Chocolate, форма передается. У пользователей, работающих с клавиатурой, нет возможности выбрать другой вкус, сразу несколько вкусов или изменить выбор

Отсутствие кнопки типа **Submit** (Отправить) затрудняет работу с меню, особенно людям, использующим клавиатуру. Выбрав в меню необходимый вам вкус, вы начнете процесс передачи формы, как только нажмете на первую опцию. Само собой, шоколад не единственный вариант, доступный посетителям, у которых нет возможности пользоваться мышкой. Но если бы вы захотели выбрать лимон, то понадобилась бы мышка. Добавление кнопки **Submit** позволит разрешить эту проблему без особых усилий. Такие незначительные нюансы разметки помогут сделать сайт более доступным (рис. 3.5).

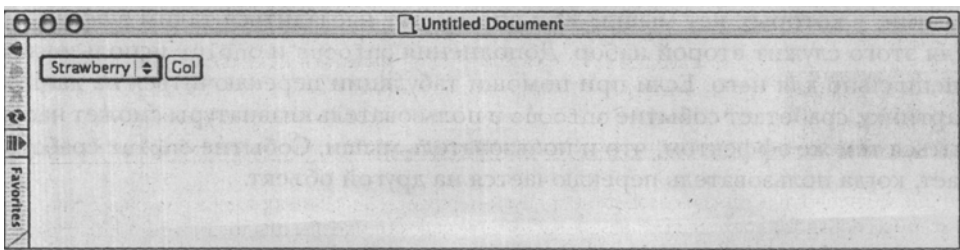


Рис. 3.5. Добавление кнопки типа Submit сделает форму доступной для людей, работающих без мыши

JavaScript

Добавляя команды JavaScript в библиотеку, имейте в виду, что не все браузеры поддерживают этот язык. Некоторые пользователи (как ни странно, около 10%) предпочитают браузеры с поддержкой JavaScript, но отключают этот язык в целях безопасности; многие используют устройства, не поддерживающие JavaScript. Особенно важно принимать это во внимание, создавая элементы навигации, содержащие JavaScript. Иногда элементы навигации бывают даже не видны, пока мышь просто перемещается по экрану; при наведении же мыши на определенный участок экрана появляется меню. Вероятно, это забавно для тех, у кого работает JavaScript, но другим таким сайтом пользоваться невозможно. Создайте подобную навигацию без JavaScript, разместив текст меню где-нибудь на странице. Лучше всего сделайте этот текст более доступным с помощью атрибута `tabindex`, благодаря которому у пользователей, работающих без мыши, первый пункт меню сразу будет выделен.

Еще один альтернативный вариант функциям JavaScript представлен в следующем примере разметки. Проведя некоторое время в Сети без мыши, вы поймете, насколько важна эта простая работа.

```
<a href="thepage.html" onmouseout="swapImgRestore()"
onblur="swapImgRestore()"
onmouseover="swapImage('Image1','susan2.gif',1)"
onfocus="swapImage('Image1','susan2.gif',1)" ximg
src="susan.gif"
name=Image1" width="200" height="200" alt="Susan" /></a>
```

Эти два набора функций JavaScript создают элемент *ролlover* (одновременное нажатие нескольких клавиш). Первый набор содержит два условия: `onmouseout` (наведение курсора) и `onmouseover` (курсор убран с объекта). Этот набор достаточно прост: когда пользователь наводит курсор на картинку, изображение `susan.gif` меняется на `susan2.gif`. Когда курсор убирается с картинки, условие `onmouseout` возвращает исходную картинку `susan.gif`. Но как быть с пользователями, у которых нет мыши? Как они смогут насладиться таким эффектом? Для этого служит второй набор. Дополнения `onfocus` и `onblur` использованы специально для него. Если при помощи табуляции переключиться на данную картинку, сработает событие `onfocus` и пользователь клавиатуры сможет насладиться тем же эффектом, что и пользователь мыши. Событие `onblur` срабатывает, когда пользователь переключается на другой объект.

Расширение HTML-библиотеки

С каждым новым проектом у вас появляется больше полезных вещей для добавления в HTML-библиотеку, а также больше возможностей, позволяющих пользоваться ресурсами, которые вы создаете для себя. Разрабатывая малобюджетные сайты, вы будете постоянно использовать эти ресурсы, постепенно понижая цену производства. Накопление подобных материалов займет определенное время, однако если внимательно относиться к создаваемым разметкам, то в дальнейшем они обязательно пригодятся.

ГЛАВА 4. Почему хороший текст имеет большое значение

Умные слова не так хороши,
как откровенный разговор.
Китайская поговорка

Основные идеи главы

1. Слова имеют огромное значение, когда у вас скромные ресурсы.

Язык сайта требует полного контроля, а также технологических и финансовых вложений.

2. Хороший текст может способствовать продвижению малобюджетного сайта.

Если не хватает средств на дизайн, уделите особое внимание тексту. Написанный простым и понятным языком текст может продвинуть ваш сайт, сделав его популярным и привлекательным. Текст способен произвести на посетителя такое же большое впечатление, как визуальный дизайн, а может быть, даже более сильное.

3. Пишите в простой и ясной манере.

Люди, читающие информацию в Сети, менее терпеливы и довольно скептически, чем при ознакомлении с любыми другими источниками. Избегайте использования огромных предложений и специальных терминов. Такой стиль может подойти для научно-публицистических текстов, однако в Сети он неприемлем. Выделяйте основные идеи и сообщайте посетителям только о том, что им действительно необходимо знать.

4. Текст должен быть простым, но не примитивным.

Опять-таки, старайтесь делать меньше. Используйте лаконичный и понятный язык. Придерживайтесь простых и содержательных предложений. Однако не упрощайте текст до такой степени, чтобы он был примитивным, поскольку в данном случае ваш сайт будет выглядеть глупо и вызовет негативную реакцию со стороны посетителей. Впрочем, помните также и о том, что чем короче текст, тем проще его будет редактировать.

5. Избегайте орфографических и грамматических ошибок.

Ничто так не портит прекрасно оформленный сайт, как орфографические и грамматические ошибки. Выделите ресурсы на редактирование нового текста.

6. Необходимые вспомогательные элементы.

Совместите преимущества необходимых вспомогательных элементов и силу, присущую исключительно словам. Используйте теги <PPLE>, файлы подписи e-mail, сообщения об ошибках и другие необходимые элементы в своих интересах.

Слова - это простой, независимый от устройства и недорогой «клей», который позволяет делать из вашего сайта единое целое. При правильном употреблении они заставят пользователей совершать покупки, щелкать по ссылке в рекламном письме, удержат и оставят посетителей на вашем сайте. Слова отличают ваш сайт от сайтов конкурентов и дают посетителям понять, что все-таки человек, а не программа стоит за этим сайтом.

Можно сказать, что слова - это единственная составляющая сайта, над которой у вас есть полный контроль. Воспроизведение графического дизайна обычно зависит от принимающего устройства пользователя, это может быть карманный компьютер, голосовой браузер либо обычный Web-браузер, средства которого могут просто не поддерживать все особенности вашего дизайна. Вы не можете контролировать предпочтения пользователей в области установленных плагинов, отображения слов и других опций. Между тем слова отображаются в точно такой же последовательности, как вы их напечатали.

Текст не только доносит до пользователя то, что вы хотите сообщить на своем сайте, но и дает пользователю представление о вас и том, что для вас важно. Если текст на сайте содержит много грамматических ошибок, у пользователя может сложиться впечатление, что вы невнимательны или сами не знаете, о чем пишете. Благодаря хорошо сформулированному тексту сайт покажется умным, доброжелательным, интересным, веселым - словом, каким вам будет угодно.

Разумный дизайнер малобюджетных сайтов внимательно относится к тексту и выделяет значительную часть ресурсов на его написание и редактирование. Это не значит, что ваш текст должен быть литературным шедевром, это даже не означает, что вам необходимо приглашать высокооплачиваемого писателя (хотя, если вы можете себе это позволить, сделайте это). Пристальное внимание к тексту заключается в том, что вы должны выделить достаточное количество времени на подготовку и редактирование материала и что предложения должны быть оригинальными, интересными и конкретными.

Эта глава специально написана для ответственных за многие аспекты производства Web-профессионалов, которые, возможно, не могут себе позволить нанять хорошего писателя и редактора.

Определение

Что значит хорошо писать. Ник Асборн (Nick Usborne), автор книги Net Words, а также эксперт, консультировавший меня во время написания этой главы, высказал свое мнение насчет того, что означает хороший текст и как его составить для малобюджетного сайта: «Если вы умеете писать, остальное само приложится. Хороший текст всегда простой и четкий. В нем нет ничего лишнего: ни драматизма, ни восклицательных знаков и излишнего пафоса. Если вы чувствуете, что предложению не хватает выразительности, постарайтесь составить его заново. Не стоит переполнять его громкими словами и многочисленными синонимами.

Может ли потенциально «дешевый» сайт выиграть за счет хорошего текста? Да. Основное достоинство сайта базируется на качестве письма. Хороший текст может продвинуть все! Однако в этом вопросе я не совсем объективен - прав, но не объективен».

Составляющие хорошего текста

Чтобы писать для сайта хорошие тексты, не обязательно быть самым талантливым или высокооплачиваемым писателем. Мы не пытаемся порадовать пользователей литературными шедеврами, мы просто стараемся дать то, что им нужно. При создании текста для сайта помните, что он должен:

- быть простым и понятным, но не примитивным;
- состоять из коротких, лаконичных частей, имеющих пометки и заголовки;
- говорить о пользователе, а не об авторе;
- быть выверен с точки зрения грамматики и пунктуации;
- быть написан одним стилем.

Размышляя о том, как соблюсти эти шесть требований к хорошему тексту для Сети, подумайте и о месте на сайте, где будут размещены:

- основной текст;
- выноски и смысловые выделения;
- заголовки;
- теги названия страницы;

- навигация;
- сообщения об ошибках.

Где бы текст ни появился, профессионал, располагающий небольшим бюджетом, должен извлекать максимум пользы из написанного текста и придерживаться требований к хорошему тексту для Сети.

(-Угроза бюджету_____);_____

Ничто не может помешать успеху вашего сайта так, как плохой текст. Пользователи не любят неясное изложение мыслей и с недоверием относятся к сайтам, на которых встречаются грамматические ошибки. Время и деньги на создание прекрасного дизайна или покупку мощного программного обеспечения окажутся потраченными понапрасну, если вы будете небрежно относиться к содержанию сайта.

Пишите проще и понятнее

Возвращаясь к мудрому совету Ника Асборна, заметим, что люди более скептически относятся к прочитанному в Сети, чем к прочитанному в книге, брошюре, газете или любом другом печатном источнике. Асборн подчеркивает, что загруженность текста сложными словами и предложениями, свойственная книгам, раздражает читателей, если они сталкиваются с такими текстами в Сети. Сложный язык с массой терминов не принесет хороших результатов, если будет использоваться в Сети.

Добавлю к уже написанному, что даже самый довольный пользователь Сети находится всего на один щелчок мыши от чувства неудовлетворенности и желания покинуть сайт. Собирая эти идеи в одно целое, вы можете представить, что большинство пользователей - довольно нежные создания, а в задачу Web-разработчика входит защищать их, чтобы они не чувствовали себя никому не нужными. Лучший способ это сделать - использовать простые и конкретные слова, касающиеся определенной темы, то есть сообщать пользователям о том, что им надо узнать.

Взгляните на рис. 4.1. Здесь изображен сайт Swimmers Guide (Справочник пловца) в том виде, в каком я его обнаружила. Обратите внимание на название темы. Дословный перевод будет выглядеть примерно так: «У нас больше информации о большем числе бассейнов в самых разных местах, чем на любом другом сайте! Огромное количество ссылок по сравнению с любым другим каталогом, посвященным этой теме!» Что такой текст сообщит читателю о самом сайте?

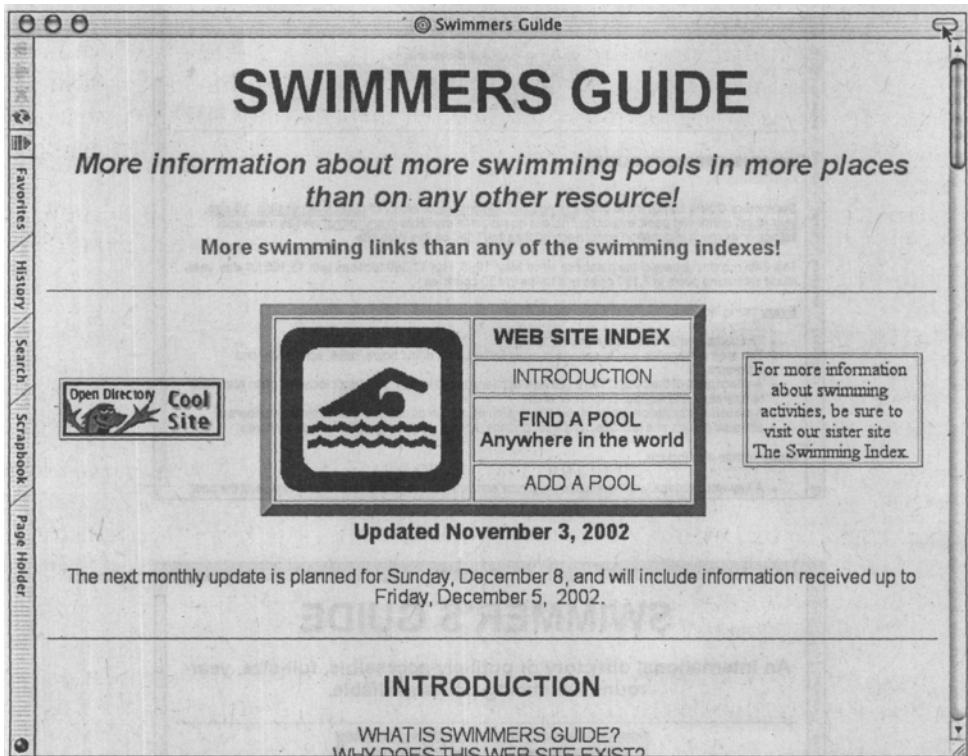


Рис. 4.1. Перегруженность никому не нужна. Строка или строки темы в случае со «Справочником пловца» (www.swimmersguide.com) - пустая болтовня. Этот текст даже не сообщает нам, о чем сайт: об общественных бассейнах, о строительстве бассейнов, о продаже бассейнов или о бассейнах для богатых и знаменитых. Кого интересует, что сайт предлагает исчерпывающую информацию о бассейнах? Посетители хотят знать, какого рода информация здесь размещена и насколько она полезна

- Как получить золота из песка — / —
- Очень часто самый важный текст скрыт на внутренних страницах сайта, в то время как второстепенная информация сразу бросается в глаза. Первостепенная информация должна присутствовать на первой странице.**

Прочитайте текст, размещенный на сайте (см. рис. 4.2), в отредактированном мной виде: «Международный справочник по общедоступным и круглогодично

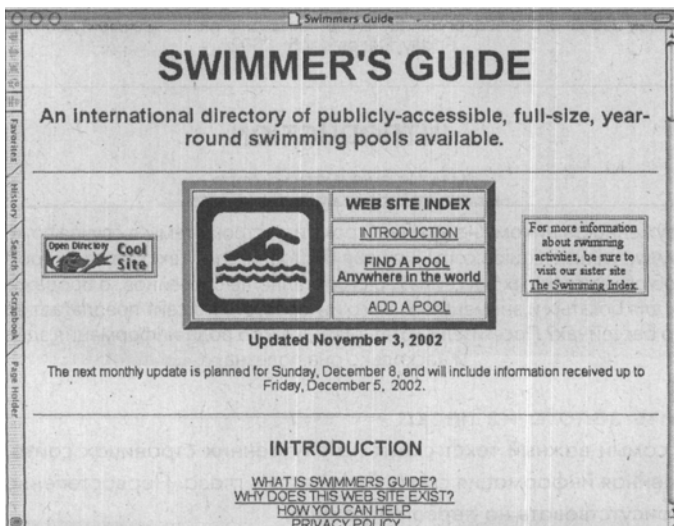
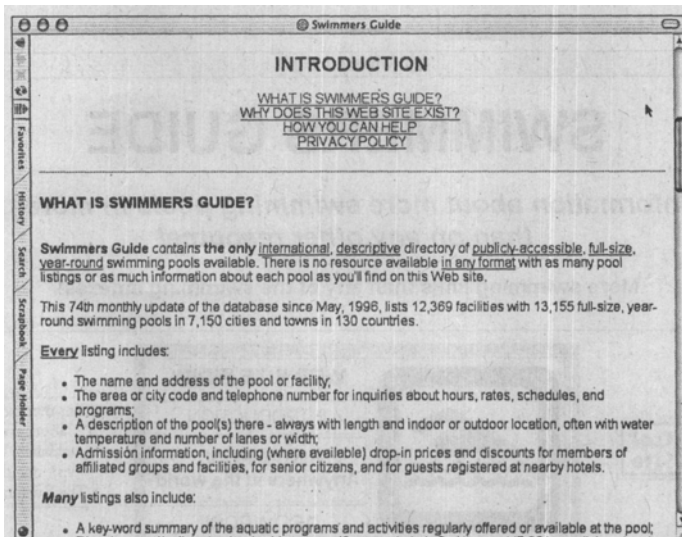


Рис. 4.2. Расскажите посетителям о том, кто вы и что собой представляет ваш сайт. Воспользовавшись фрагментами текста со страницы, рассказывающей о сайте (вверху), я отредактировала строку заголовка для исправленной версии сайта (внизу). Теперь мы знаем, что представляет собой сайт

открытым бассейнам». Этот текст я нашла на одном из уровней сайта. Очень часто наиболее привлекательная информация спрятана на сайте, как будто является незначительной. Возможно, это происходит потому, что, когда люди не думают в полную силу, хорошая, свежая идея приходит к ним не сразу. Помните об уже написанном тексте, который «пылится» где-то в темном углу вашего сайта. Если у вас есть подобный хороший текст - текст, в котором говорится о том, что собой представляет ваш сайт, - используйте его на главной странице. Это обойдется не дороже, чем если вы ее спрячете.

Перегруженная строка темы на рис. 4.1 может подойти для справочника «Желтые страницы», а не для сайта. Ориентируясь на нетерпеливую аудиторию посетителей Сети, сразу же сообщите им то, что необходимо. Пусть все будет просто и понятно. Сайт The Hendersonville County Public Library (Окружная публичная библиотека Хендерсонвиля), показанный на рис. 4.3, -

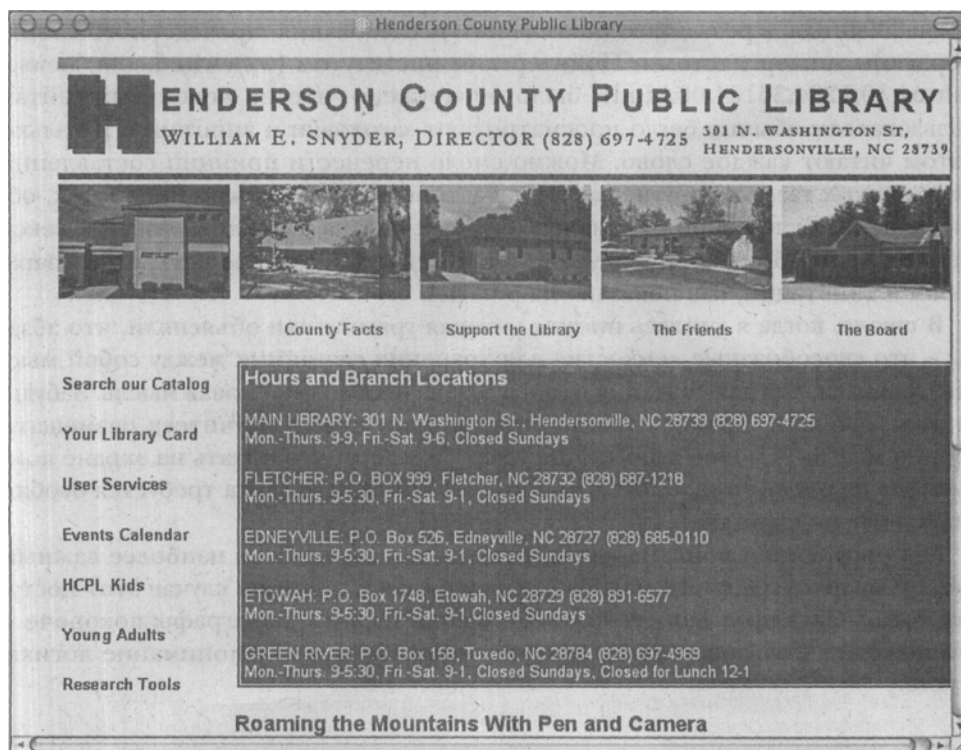


Рис. 4.3. Содержательный текст может быть так же прост, как номер телефона

прекрасный тому пример. На самом верху сайта находится телефон директора библиотеки. Таким образом, сайт предоставляет четкую информацию, в которой, скорее всего, нуждаются чаще, чем в информации о посетителях библиотеки.

Сократить текст стоит не дороже, чем оставить его чересчур длинным и запутанным. Специального программного или технического обеспечения для этого не требуется. Чем короче текст, тем меньше слов вам придется редактировать. Если при написании или редактировании текста вы платите за каждое отдельное слово, то компактный и в то же время содержательный текст сократит ваши расходы.

Разделите текст на маленькие части и чаще используйте заголовки

В мае 2002 года, в результате тщательного исследования, проведенного Стэнфордским университетом и Пойнтерским институтом (www.wired.com/news/culture/0,1284,36104,00.html), было обнаружено, что на новостных сайтах пользователи обычно бегло просматривают заголовки и аннотации и только потом читают каждое слово. Можно смело перенести принцип составления виртуальных газет на другие сайты, в частности сайты школ и библиотек, общин и церквей, предприятий малого бизнеса. Задача очевидна: разбейте текст на небольшие абзацы, и пусть у пользователей будет возможность читать заголовки и аннотации, как показано на рис. 4.4.

В школе, когда я училась писать, учителя грамматики объясняли, что абзацы - это своеобразные «емкости» для хранения связанных между собой мыслей. Я думала, что для создания нового абзаца необходима новая мысль. Забудьте это заученное правило, когда пишете для Сети, и научитесь размещать «новую мысль» в менее строгом порядке. Помните, что читать на экране компьютера не очень-то удобно и что посетителям вашего сайта требуется особая визуальная поддержка.

Мы определили понятие «делайте меньше» как одну из наиболее важных составляющих успешного малобюджетного сайта; в данном случае этот постулат может быть применим и к объему текста. Пишите параграфы покороче - это поможет пользователям сократить время и силы на понимание логики текста.

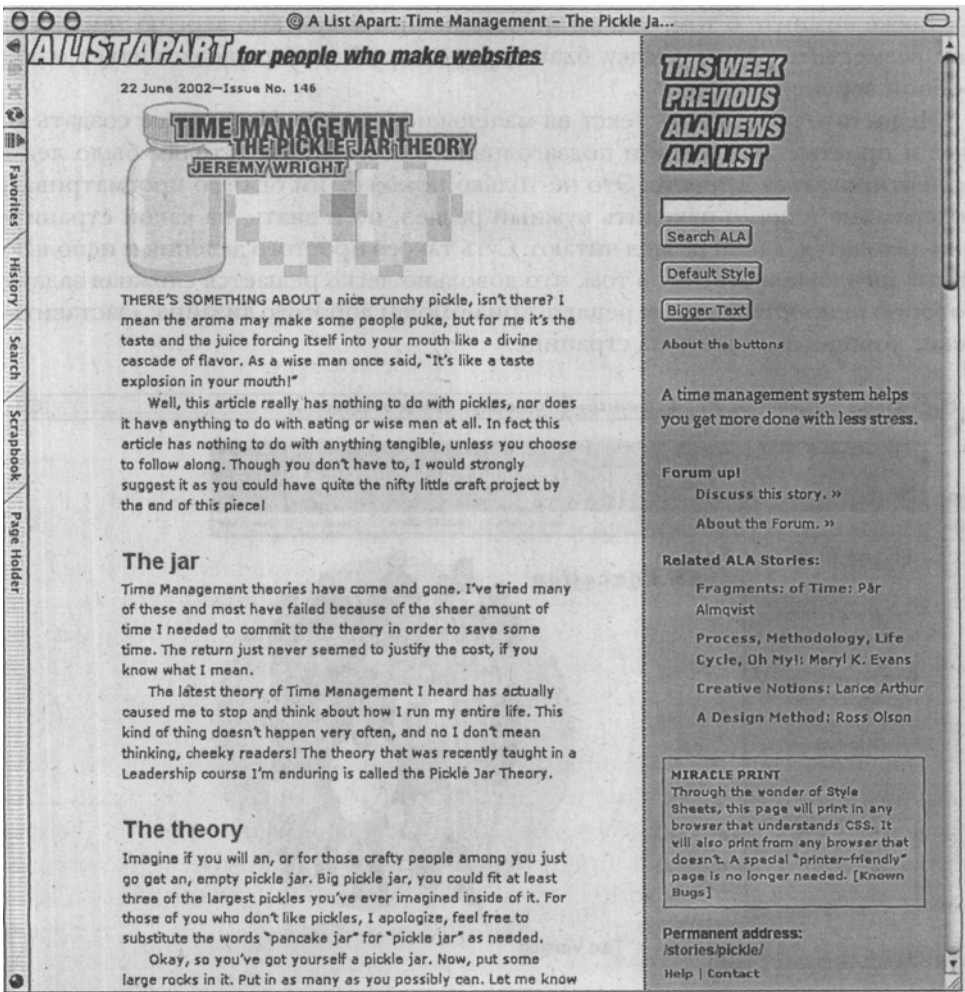


Рис. 4.4. A List Apart (www.alistapart.com) - это еженедельный журнал для специалистов, занимающихся созданием Web-сайтов. Являясь некоммерческим, журнал ALA практически не имеет собственного бюджета. Каждая статья поделена на небольшие параграфы и самостоятельные части с содержательными заголовками. Если вы хотите разместить на сайте много информации, разбейте текст подобным образом и напишите простые, но содержательные заголовки. Они помогут пользователям сориентироваться в структуре сайта

Также помните о том, что вы можете написать краткую версию темы, а затем разместить рядом ссылку, благодаря которой можно ознакомиться с подробной версией материала.

Недостаточно разбить текст на маленькие кусочки. Нужно еще создать ясные и простые заголовки и подзаголовки, чтобы пользователям было легко ориентироваться в тексте. Это не только поможет им быстро просматривать содержание и легко находить нужный раздел, но и знать, на какой странице они находятся, какой раздел читают. Суть такого простого деления и использования заголовков состоит в том, что довольно легко решается сложная задача, которую непросто было бы решить при помощи дорогого дизайна: «заставить» текст хорошо выглядеть на странице.

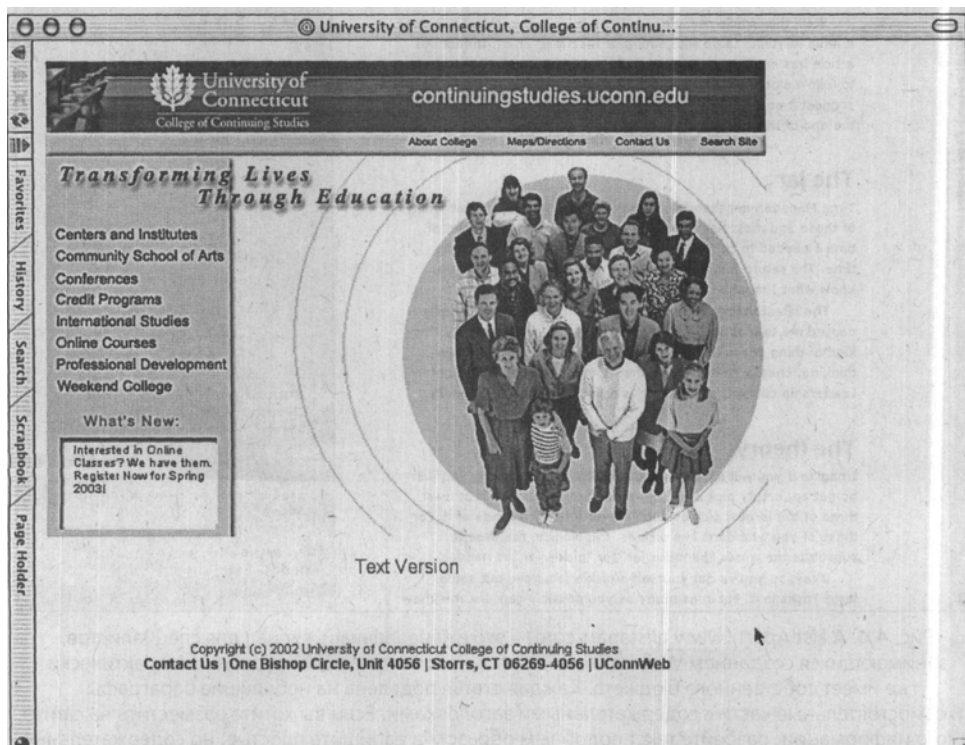


Рис. 4.5. На этом сайте рассказывается о некоторой программе (<http://continuingstudies.uconn.edu>).

Даже строка темы, **Transforming Lives Through Education** (Изменяя жизнь с помощью образования), посвящена программе, а не студентам. От этой привычки трудно избавиться, но все же постарайтесь писать о пользователе, а не о программе

Пишите о пользователе, а не о себе

Если бы за каждый сайт, при просмотре которого кажется, что он «разговаривает с самим собой», мне бы давали по 5 центов, я быстро бы разбогатела. Использование слов в манере, направленной на себя, а не на посетителя, встречается повсюду. Лучше всего проиллюстрировать это на примере навигации сайта.

На рис. 4.5 показан сайт для студентов, обучающихся в Университете Коннектикута. Обратите внимание на первую ссылку панели навигации. Пункт **Centers and Institutes** (Центры и институты) кажется вполне безобидным, но взгляните на аналогичную ссылку сайта UCLA (рис. 4.6) - **Fields of Study** (Области изучения). Здесь мало информации о структуре школы, зато подробно рассказывается о возможностях, предоставляемых студенту.

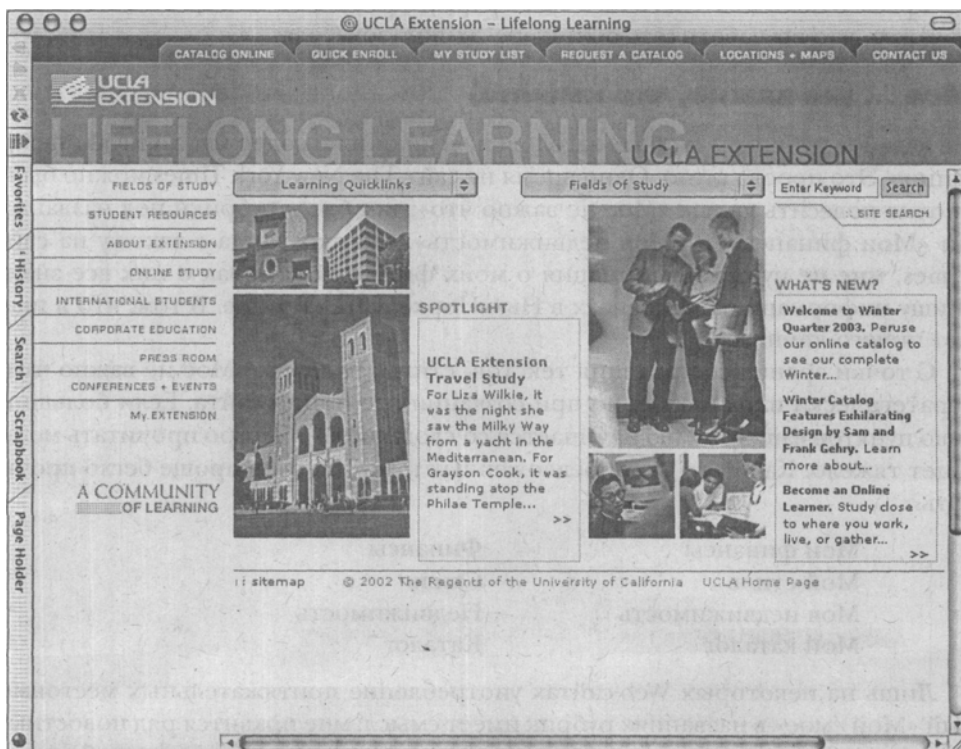


Рис. 4.6. У UCLA тоже есть программа продолжения образования (www.uclaextension.org). Строка темы, **Lifelong Learning** (Обучение в течение всей жизни), информирует об учебном процессе или о жизни студента, а не о программе

Практически каждое учебное заведение предлагает программу обучения за рубежом. На сайте Университета Коннектикута ссылка на эту программу называется International Studies (Обучение за рубежом) - это описание программы. Ссылка на подобную программу на сайте UCLA выделена в центре страницы; текст, размещенный в рамке, - это рассказ студента. Этот рассказ - свидетельство того, что сайт UCLA служит интересам пользователя.

Еще одно сравнение, и я перестану говорить об Университете Коннектикута. Почти каждое учебное заведение предлагает удаленное online-обучение. Ссылка на данный раздел сайта Университета Коннектикута находится в меню слева и носит название Online Courses (Online-курсы). На сайте UCLA ссылка на эту же информацию размещена в нижнем правом углу под названием Welcome an Online Learner (Стань online-учеником).

Фразы о пользователе, а не о вас, сформулировать ничего не стоит, зато это покажет, что вы заботитесь об интересах пользователя.

Мое ... (не важно, что именно)

Не давайте рубрикам личные заголовки, если они не содержат личную информацию. Это просто глупо. Одно время на сайт The New York Times можно было вполне повесить ярлык «Мое не важно что»; там были рубрики под названиями «Мои финансы» и «Моя недвижимость». Однако, когда я захожу на сайт Times, мне не нужна информация о моих финансах. Я и так о них все знаю. Я ищу информацию о финансах в Нью-Йорке, США и мире. В том, что я ищу, нет ничего личного.

С точки зрения составления текстов, стиль заголовка «Мое не важно что» стратегически плох, особенно при создании навигации сайта. Если большинство пунктов вашего меню начинается со слова «мое», быстро прочитать меню будет тяжело. Какой из двух предложенных ниже списков проще бегло прочитать?

Мои финансы	Финансы
Мой стиль	Стиль
Моя недвижимость	Недвижимость
Мой каталог	Каталог

Лишь на некоторых Web-сайтах употребление притяжательных местоимений «Мой/мое» в названиях рубрик имеет смысл: мне нравится ряд новостных сайтов, на которых можно выполнить настройки таким образом, чтобы научные статьи появлялись раньше, чем статьи на спортивную тему. Я не призываю

полностью отказаться от персонализации; это всего лишь небольшое предупреждение о том, что не стоит часто употреблять слова «мой», «мое».

Текст должен быть простым, но не примитивным

Следите за тем, чтобы ваши предложения были короткими и касались заданной темы. Старайтесь избегать употребления сложных предложений, особенно если вы предлагаете продукты и услуги. На рис. 4.7 приведен отличный пример от Почтовой службы США (www.usps.com/zip4).

UNITED STATES POSTAL SERVICE

ZIP Code Lookup

It's a snap to find a ZIP Code and more.

Do you need a ZIP + 4 Code(s) for a city? Or all cities in a ZIP Code? Select your task from the menu bar below.

What are you looking for?
ZIP+4 Code

Please fill in the following information: (* Required fields)

- * Address 1
- Address 2
- * City
- * State

ZIP Code

Submit >

[Required when City/State are not provided.]

For integrating ZIP Code Lookup capabilities into your web site or application system, please visit our [Web Tools \(APIs\)](#) or find additional addressing products at [Address Information System Products \(AIS\)](#).

ZIP Code FAQ's

Magazine Subscriptions Online

POSTAL INSPECTORS Preserving the Trust

site map contact us government services

Copyright © 1999-2003 USPS. All Rights Reserved. [Terms of Use](#) [Privacy Policy](#)

Рис. 4.7. Текст на этой странице поиска ZIP-кодов (www.usps.com) смотрится просто превосходно.

Фразы «Вам необходим код ZIP + 4 какого-то определенного города? Или список всех городов с данным ZIP-кодом?» простые, и они сообщают пользователю о том, что он может сделать на этой странице. Однако взгляните на то, что идет далее

При чтении дальнейшего текста создается впечатление, что его написал программист, когда сайт еще находился в процессе разработки:

«Для активации функции поиска ZIP-кода на вашем сайте или в прикладной системе, пожалуйста, воспользуйтесь нашими Web-сервисами (APIs) или смотрите дополнительные продукты по адресации на Address Information System Products (AIS)».

Я не обвиняю программистов: то, что возникает в голове при разработке приложений, иногда очень сложно передать доступным языком. Мне приходилось наблюдать, как подобные вещи случались и с самыми внимательными людьми. Если вы одновременно занимаетесь и программированием, и написанием текстов, отведите определенное время на редактирование написанного, когда сайт уже готов. Простое изменение фразы может иметь большое значение. Рассмотрим это на примере абзаца о ZIP-кодах:

«Добавьте функцию поиска ZIP-кода на ваш сайт. Смотрите другие наши продукты по адресной системе».

Маститые писатели иногда думают, что текст в Сети должен быть непринужденным, даже шутивным. Это не так. На самом деле такой подход может сбить пользователя с толку.

Представьте, что Web-сайт IRS, показанный на рисунке 4.8, - это предупреждение (<http://www.irs.gov/individuals>):

«Где возмещение моих убытков?»

Соберите полную информацию о процессе возмещения своих возможных убытков. Безопасный доступ в любое время из любого места. Что за сделка!»

Помните также о том, что слова, которые вы используете, создают впечатление о компании или организации, для которой вы пишете. Я знаю, что IRS пытается быть максимально дружелюбным и общедоступным агентством, но у нас никогда не возникнет желания обратиться к нему и такой несоответствующий текст немного раздражает. Кто пытается шутить? Зачем тратить время на составление шутивного текста, который кажется неуклюжим и не представляет никакой реальной ценности для клиента?

Я бы переписала текст IPS следующим образом, а шуточки приберегла бы для встреч с друзьями:

«Где возмещение моих убытков?»

Проверьте состояние процесса возмещения убытков прямо сейчас в режиме on-line».

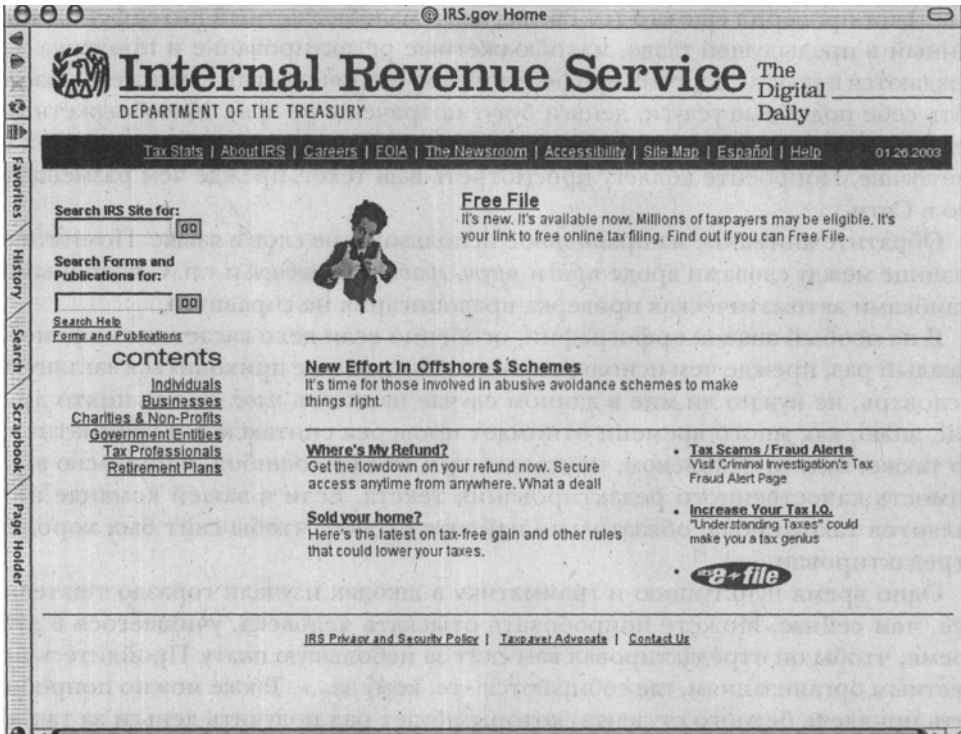


Рис. 4.8. В самом деле, что за сделка (www.irs.gov)!
 Как полная информация поможет мне найти материал о возмещении убытков?

Избегайте грамматических ошибок

Лучший способ потерять пользовательское доверие к вашему продукту и к словам - разместить сайт, который содержит уйму грамматических ошибок. Пользуйтесь автоматической проверкой правописания, но помните, что она не поможет исправить все ошибки. Не пожалейте времени на то, чтобы проверить текст самостоятельно, самым тщательным образом. Попросите, чтобы

ваш текст проверил еще кто-то. Так же, как и малобюджетный интерфейс, описанный в предыдущей главе, малобюджетные редактирование и проверка не нуждаются в услугах дорогих профессионалов (впрочем, если вы можете позволить себе подобные услуги, деньги будут потрачены не зря). Малобюджетные редактирование и проверка основываются на том, что одна голова хорошо, а две лучше. Попросите коллегу просмотреть ваш текст, прежде чем размещать его в Сети.

Обратите внимание на правильное использование слов в языке. Помните о разнице между словами вроде *пруд* и *прут*, *грибной* и *гребной* и т.п. С подобными ошибками автоматическая проверка правописания не справится.

Я не особый знаток орфографии, особенно если дело касается омофонов. Каждый раз, прежде чем использовать слово *плот*, мне приходится заглянуть в словарь, не нужно ли мне в данном случае написать *плод*. Я, как никто другой, знаю, как много времени отнимает проверка синтаксиса и грамматики, но также, будучи человеком, пострадавшим от этих ошибок, я понимаю значимость качественного редактирования текста. Если в вашей команде нет талантов такого рода, обязательно найдите способ, чтобы сайт был хорошо отредактирован.

Одно время пунктуацию и грамматику в школах изучали гораздо тщательнее, чем сейчас. Можете попробовать отыскать человека, учившегося в это время, чтобы он отредактировал вам сайт за небольшую плату. Пройдитесь по местным организациям, где собираются «те, кому за...». Также можно попробовать привлечь бедного студента, который будет рад получить деньги за такую работу. !Н |

Если вы не можете себе позволить услуги редактора или даже можете, все равно воспользуйтесь этим советом - прочитайте книгу "The Elements of Style", авторами которой являются William Strunkjr., E.B. White, and Roger Angell. Она научит вас, как избежать самых распространенных грамматических и стилистических ошибок.

Бесплатные и недорогие элементы страницы

Теперь, когда вы знаете, как написать хороший текст *для* Сети, давайте подумаем о том, как его лучше оформить. Объем кода для сайта гораздо больше, чем только строки навигации и текст страницы.

Ter title

Тег <title> описывает строку, обычно располагающуюся вверху браузера и отображающую название вашей страницы (рис. 4.9). Я подробно расскажу о том, где должен быть размещен этот элемент в HTML-документе, в главе 7. Элемент названия располагается среди первых строк разметки в HTML и выглядит следующим образом:

```
<title>The Rogue Librarian - - Shooshin' and stampin' </title>
```

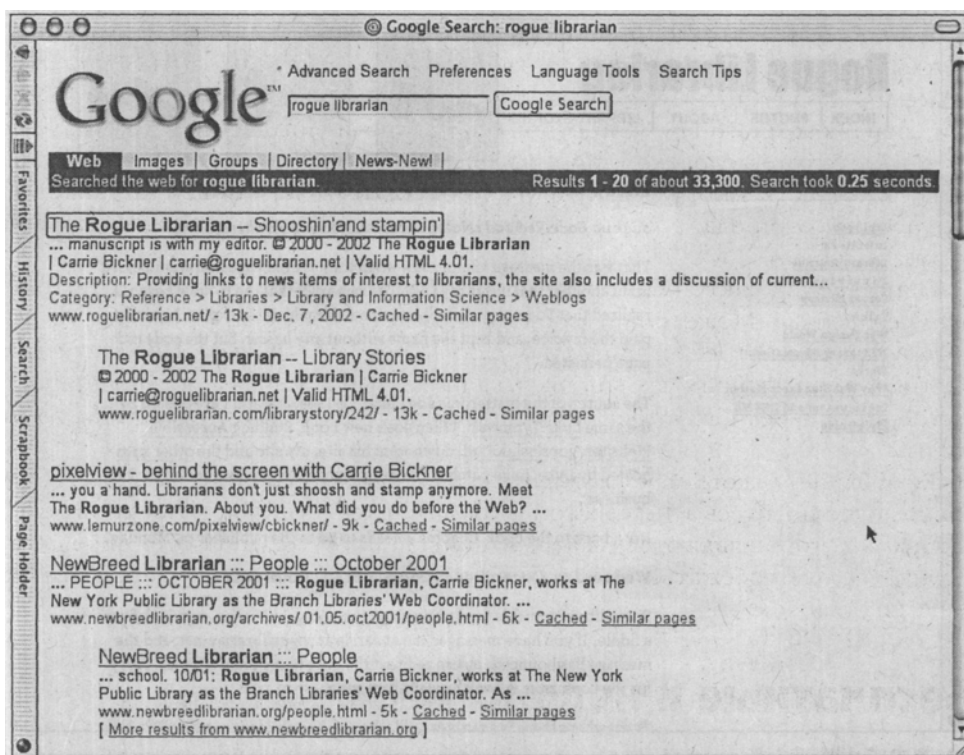


Рис. 4.9. Текст, который вы пишете для тега <title>, появляется в результатах поиска на специальных сайтах (например, www.google.com). Эта строка отображается на многих поисковых машинах Сети, а также на внутренних поисковиках сайтов. На рисунке показано, что получится, если вы не будете тратить время на написание элемента <title> для каждой страницы, - вы даете пользователю меньше шансов отличить один раздел вашего сайта от другого

Содержимое элемента `<title>` отображается в самых значимых местах: в результатах поисковых машин (рис. 4.10), в строке заголовка браузера (рис. 4.11) и в закладках браузера. Элемент названия также можно использовать в различных Web-приложениях, таких как новости, или в окнах Отправить Другу по e-mail. Таким образом, правильное использование этого элемента крайне необходимо, когда вам нужно описать содержимое сайта на других сайтах или в рассылках. Такой небольшой элемент страницы имеет очень большое значение, это все равно что снять дешевую квартиру в Манхэттене. Пользуйтесь этим!



Рис. 4.10. Содержимое элемента `<title>` отображается в строке заголовка большинства браузеров, позволяя вашим посетителям увидеть, где они находятся (www.roguelibrarian.com). Подумайте о значимости этой информации для пользователей, которые любят одновременно открывать сразу несколько окон браузеров. Составляя этот текст, будьте кратки. Некоторые браузеры обрезают названия. Постарайтесь, чтобы количество символов в строке названия не превышало 45. Как уже говорилось ранее в этой главе, лучше быть лаконичным. Используйте слова экономно

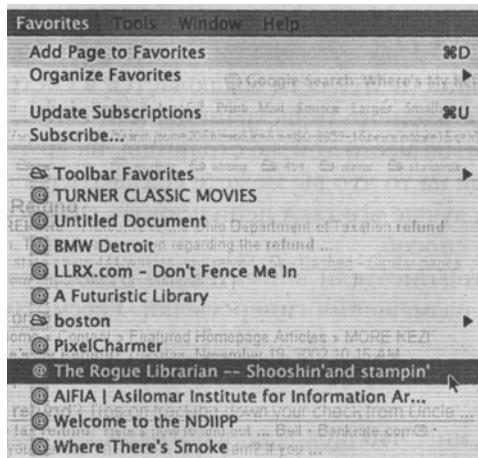


Рис. 4.11. Еще одно место, где отображается содержание тега < title > , - это папка Избранное.

Если вы рассчитываете на то, что пользователи вашего сайта добавляют в Избранное не одну страницу, позаботьтесь, чтобы у каждой страницы было свое название. Можно включить в них название компании, но постарайтесь уложиться в рамки небольшого текста

Иногда про элемент <title> забывают. Порой этот элемент остается просто неиспользованным. Подчас названием сайта является элемент <title>, одинаковый для каждой страницы сайта. В таком случае пользователям трудно сориентироваться в разделах вашего сайта, если они заходят на него через ссылки в папке Избранное (Favorites), либо через ссылки поискового сервера. Если вы потратили время и деньги на установку внутренней поисковой машины для своего сайта и забыли написать содержательные элементы названия для каждой страницы или раздела сайта, считайте, что поисковая машина была установлена зря. Если у вас небольшой online-магазин и ваши пользователи ищут обувь, наилучшим вариантом названия для страницы с обувью будет **Название компании: Обувь** - так будет гораздо лучше, чем просто написать название компании.

Взгляните снова на рис. 4.9, где изображены результаты поиска моего личного сайта, Rogue Librarian. Первые два результата - различные части сайта, тег названия каждой из них разный. У главной страницы - общее название и строка темы, а у подраздела Library Stories - название, соответствующее этому разделу. Позаботьтесь о том, чтобы теги названия максимально соответствовали каждому разделу вашего сайта.

Ссылки

Самая главная особенность Web-сети заключается в системе гиперссылок. *Гиперссылкой* называется элемент Web-страницы или сайта, который указывает (или отсылает) на другое место той же страницы, на другую страницу либо на другой сайт. Несмотря на то что вы можете создавать гиперссылки на картинках либо внутри них, лучше все-таки использовать гиперссылки в тексте - это самый эффективный способ.

Ответственные Web-писатели внимательно относятся к словам, которые они используют для размещения гиперссылки. Рассмотрим следующие примеры:

Моя мама сказала мне, что шоколад необходим для поддержания здоровья женщины.

Моя мама сказала мне, что шоколад необходим для поддержания здоровья женщины.

Моя мама сказала мне, что шоколад необходим для поддержания здоровья женщины.

Расположение каждой ссылки вызывает различные ожидания пользователя относительно того, что он обнаружит, отправившись по ссылке. Ссылка в первом примере, возможно, отправляет на страницу, где темой является моя мама. Пользователи также предполагают найти информацию о том, что еще моя мама говорит. Во втором примере темой является шоколад, и мы предполагаем, что переход по ссылке позволит узнать интересную информацию, как, например, рецепт вкуснейшего шоколадного пирога. В третьем примере мы ожидаем увидеть информацию из журнала "The New England Journal of Medicine", в последнем номере которого опубликована статья о пользе шоколада для женщин, - таким образом, оправдывается надежда моей мамы на то, что я буду есть шоколад каждый день.

Обратите внимание, что выбор места для ссылки не стоит денег, и если у вас это хорошо получается, старайтесь грамотно формулировать ссылки для удобства перемещения по сайту.

Атрибут title

Выбор удачного фрагмента текста - не единственный способ использования языковых ресурсов. Есть еще вариант - добавить атрибут title, благодаря которому вашим сайтом можно будет легко пользоваться. Атрибут title можно использовать в гиперссылке для предоставления о ней большей информации.

В следующем примере атрибут `title` сообщает пользователю о том, что ссылка указывает на шоколад и его полезность для женского здоровья. Текст атрибута `title` в большинстве браузеров отображается в виде всплывающей подсказки (рис. 4.12):

```
<p>Моя мама сказала мне, что шоколад необходим для  
<a href="http://www.somesciencejournal.com"  
title="Новые исследования показали, что шоколад богат  
железом.">поддержания здоровья женщины</a>.</p>
```

My mother told that chocolate is essential for women's healths

[New study finds chocolate is high in iron](#)

Рис. 4.12. Текст «Новые исследования показали, что шоколад содержит железо» - это атрибут `title` в теге гиперссылки, который высвечивается при наведении курсора на ссылку (www.somesciencejournal.com). Используйте атрибут `title` для предоставления дополнительной информации. Отметим, что не все браузеры его поддерживают, особенно старые, вот почему пользователи Netscape 4 не смогут увидеть этот текст

Добавление атрибутов `title` позволит сделать ваш сайт более информативным. Здесь можно разместить дополнительный текст для описания страницы, либо текст, полезный для ваших посетителей. Также благодаря атрибутам `title` пользователям будет проще путешествовать по сайту, а информация, размещенная на нем, станет более понятной.

Подписи в электронных письмах

Если в своей электронной переписке вы не используете файл подписи, начните прямо сейчас (рис. 4.13). Многие приложения для работы с электронной почтой позволяют создать этот небольшой образец текста, который автоматически прикрепляется к любому отправляемому сообщению. Это бесплатная реклама и возможность развлечься. Используйте девиз вашей компании, если он есть, а также адрес вашего сайта. Если вы хотите сообщить о новинках на сайте, напишите аннотацию, которая будет отправляться с каждым посылаемым сообщением. Не бойтесь - это бесплатно! Впрочем, будьте осторожны и не перестарайтесь.

Помните о том, что людей, читающих ваши письма, могут утомить многословные подписи. Главное - чтобы ваша подпись не содержала более четырех строк; просто посчитайте эти строки. Чем меньше вы пишете, тем выразительнее

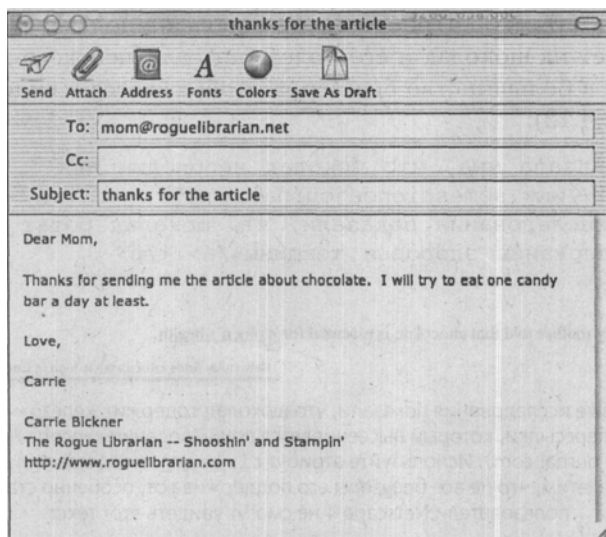


Рис. 4.13. Подписи, используемые в электронной почте, - это бесплатная реклама.

В своих сообщениях я использую файл подписи, который содержит название моего сайта, его тематику и адрес. Вся эта информация автоматически прикрепляется к любому моему сообщению

каждое слово и тем проще читать ваше сообщение пользователям Интернета, у которых нет времени на ознакомление с длинными текстами.

«Ошибка404, страницаненайдена» и другие сообщения об ошибках

Практически на всех сайтах появляются сообщения об ошибках. Это может быть дефект приложения, вызванный неправильным программным кодом, результат неверного поиска или ошибка «Страница не найдена». Не пожалейте времени на настройку сообщений, которые возвращают пользователей туда, откуда они пришли. Сообщения «Страница не найдена» дают возможность размещения бесплатной рекламы, которой не пользуются многие Web-разработчики либо пользуются в ограниченном объеме. Некоторые сайты не переделывают сообщение об ошибке 404 «Адрес не найден». На рис. 4.14 приведено отображаемое по умолчанию сообщение 404, которое вы можете увидеть на многих сайтах.

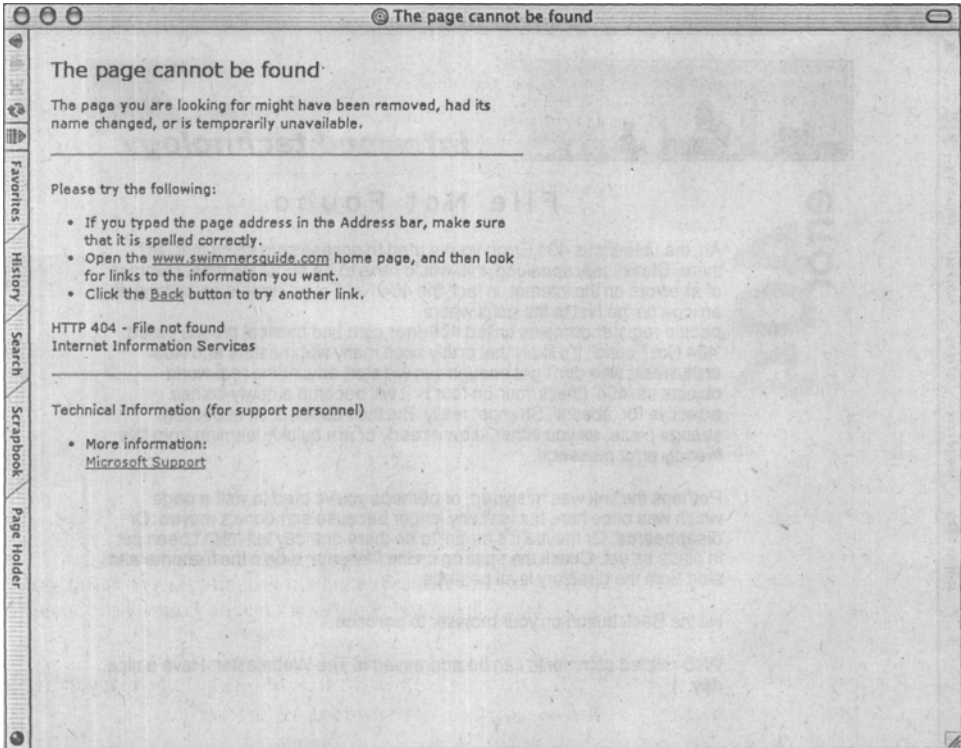


Рис. 4.14. Настраивайте все сообщения об ошибках. Настроенное сообщение 404 - это бесплатная реклама, и, если вы не используете возможность разместить рекламу, могут возникнуть нежелательные последствия. Когда пользователи попадают на несуществующие страницы, вы рискуете потерять своих пользователей. Если вы не настроили сообщения «Страница не найдена» и другие, не пожайте времени и настройте их прямо сейчас

Многие Web-разработчики определенным образом настраивают эти сообщения. И надо сказать, что некоторые посетители действительно пользуются такой возможностью размещения рекламы (рис. 4.15 и 4.16). При написании подобных сообщений не стоит предоставлять пользователю слишком большой выбор. Оставьте только самое необходимое - ссылку на главную страницу, возможность связаться с вами, ссылки на часто посещаемые страницы, а также короткое и простое для понимания объяснение, почему пользователь оказался

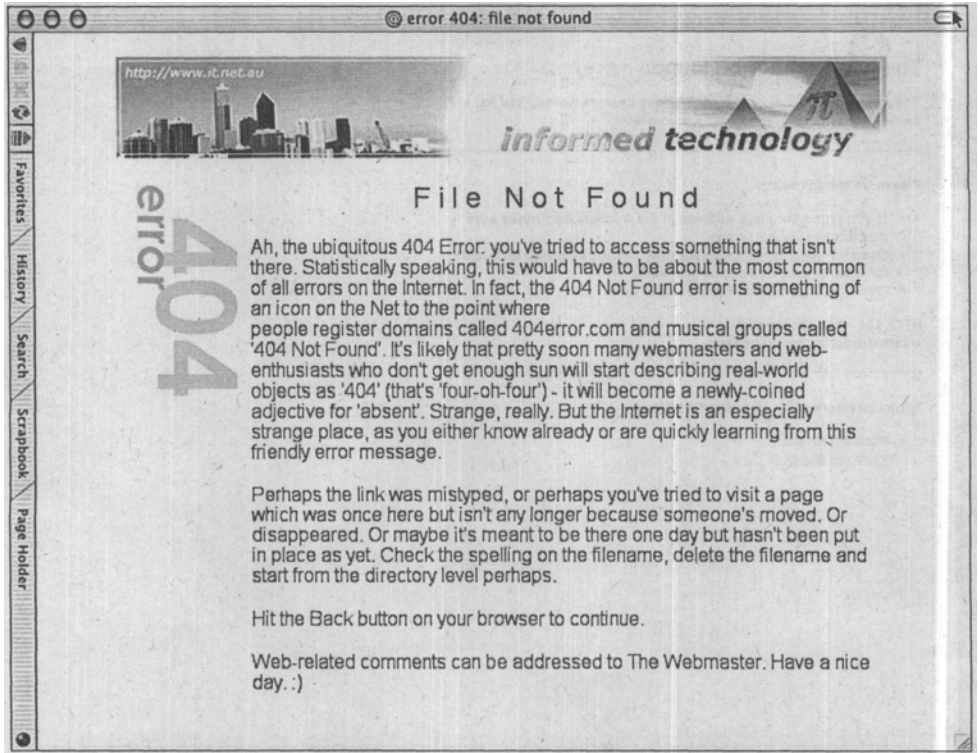


Рис. 4.15. Удостоверьтесь, что все сообщения об ошибках содержательны. Ошибка «Страница не найдена» - это предложение, но с кем разговаривает эта страница? Не с посетителем. Первый абзац целиком читается так, будто разработчик сайта смотрит в другую сторону либо, что еще хуже, разговаривает сам с собой. Во втором абзаце появляется информация, которая может иметь значение для читателя. Больше всего шансов у пользователя найти то, что он ищет, содержится в двух последних параграфах, однако эта информация достаточно скудна. Неужели кнопка **Назад** и ссылка на электронный адрес Web-мастера и есть та самая пресловутая помощь, которую мы можем оказать пользователю? Я так не думаю. Задача любого текста, особенно из серии «Страница не найдена», должна заключаться в возможности дать пользователю то, что ему необходимо

на странице 404. Задача всех сообщений об ошибках должна заключаться в том, чтобы удержать пользователя на вашем сайте.

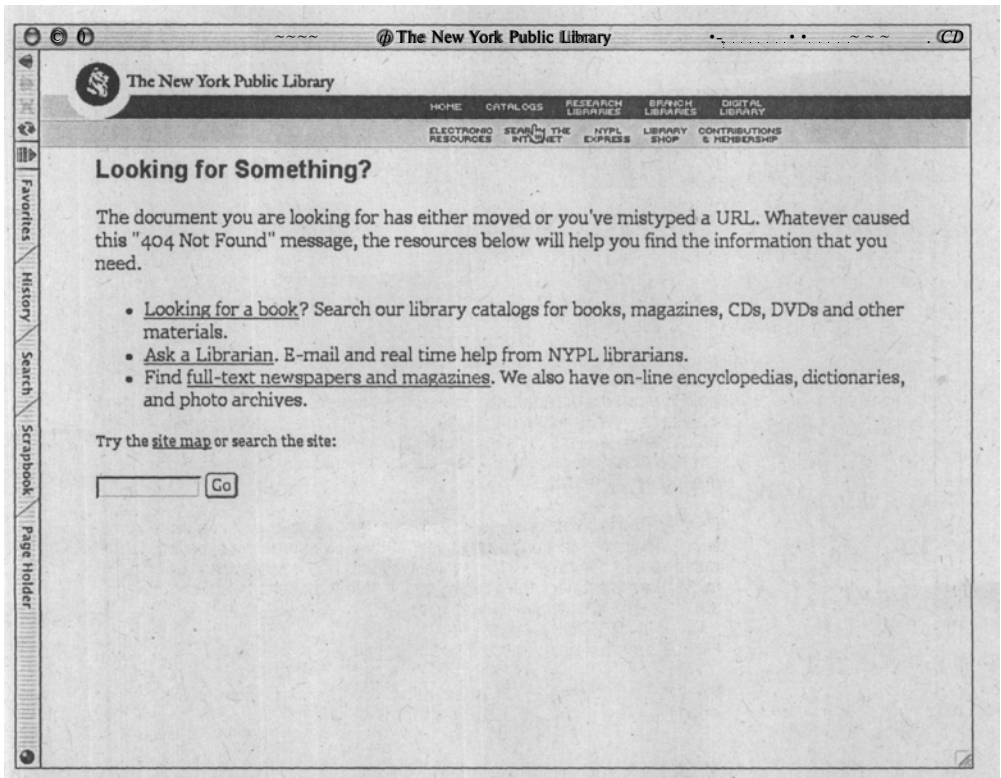


Рис. 4.16. Сообщение «Страница не найдена» написали я и моя коллега Кэтрин Джонс, когда вместе работали для Нью-Йоркской публичной библиотеки. Большая часть текста написана на основе опыта разочарования, который был у наших пользователей, и на основе ресурсов сайта, которые, по нашему мнению, имели самое большое значение для наших пользователей

Ресурсы

Если вам не хватает вдохновения для создания сообщений об ошибках, посетите сайты, посвященные лучшим и худшим страницам на эту тему, например:

- комната отдыха 404: www.404lounge.net;
- исследовательская лаборатория 404: www.plinko.net/404.

ГЛМШ I. Как хорошо выглядеть с минимальными затратами

Знаешь, как ты выглядишь со своей дорогой сумкой и дешевыми ботинками? Ты выглядишь, как деревенщина. Отмытая наглая деревенщина, у которой нет ни капли вкуса.

*Ганнибал Лектор Клариссе Старлинг
в фильме «Молчание ягнят» (1991)*

Основные идеи главы

1. Экономия - ключ к хорошему оформлению текста.

Используйте ограниченное число шрифтов.

2. Остановитесь на двух или трех правильно подобранных шрифтах.

Удостоверьтесь, что они совместимы с браузерами пользователей Сети и что они хорошо смотрятся вместе. Избегайте конфликтующих друг с другом шрифтов и тех, что напоминают письма с требованием выкупа.

3. Не тратьте деньги на плохие шрифты.

Если у вас есть какой-то особенный шрифт, используйте его в местах логического выделения либо в титульных данных, а остальной текст пусть будет написан обычным шрифтом.

4. Сэкономьте время и деньги, используя хорошую разметку и внешние таблицы стилей.

Если возложить задачи по оформлению текста и цвету на CSS, ваш сайт будет быстрее загружаться, а управлять оформлением и цветами будет проще, чем при использовании сложных шрифтовых тегов (более детально этот вопрос рассматривается в главе 7).

5. Для того чтобы сайт выглядел элегантно, выберите два или три подходящих цвета.

Даже с бюджетом \$5000 ваш сайт будет выглядеть на миллион, если вы используете простую и незатейливую палитру цветов.

6. Вам не надо тратить много денег на фотографии и иллюстрации, для того чтобы ваш сайт выглядел великолепно.

Выбирая стиль, остановитесь на каком-то одном. Разумно подбирайте и размещайте изображения. Начните с самых лучших, которые у вас есть.

88 Г ПАВА 5. Как хорошо выглядеть с минимальными затратами

Должна признаться: свой рабочий гардероб я пополняю в магазинах со сниженными ценами. Впрочем, хочется верить, что никто этого не замечает. Возможно, я обманываю себя - если нет, то у меня есть три секрета, как сделать так, чтобы никто об этом не узнал.

Секрет успешных покупок в магазинах по сниженным ценам состоит из трех частей:

1. Определите свой стиль и следуйте ему.
2. Цветовая палитра должна быть простой.
3. Линии должны быть четкими.

На таких же принципах основывается хорошо выглядящий Web-сайт. Эти принципы гарантируют, что даже при небольшом бюджете вы сможете сделать блестящий, профессиональный Web-сайт, который будет выглядеть на миллион долларов.

Можно с легкостью заглянуть в какой-нибудь магазин одежды по сниженным ценам, например в «Сток», и выйти оттуда с одеждой, которая вроде бы вам подходит, но не подчеркивает вашей индивидуальности. Прогуливаясь по рядам распродажи, легко сбиться с толку из-за обилия всевозможных предложений и купить что-то, что плохо на вас сидит, либо совсем не вашего цвета. Ничего страшного, если вы просто купили летние шорты. Кого волнует, что вы выглядите немного смешно, если вы будете носить их, только прогуливаясь по лужайке? Однако Web-сайт - это прежде всего выход на публику, причем на очень большую аудиторию, поэтому вы должны быть уверены, что не вышли в одежде, в которой гуляете на лужайке.

В случае с Web-дизайном извинения относительно того, что вы выглядите плохо, не принимаются, даже если у вас маленький бюджет. Начиная работать над малобюджетным сайтом, поступайте так же, как поступают отоваривающиеся в магазине по сниженным ценам модники: определите свой стиль, придерживайтесь простой цветовой палитры и четких линий. Давайте посмотрим на то, как даже с небольшим бюджетом определенный стиль, простые цвета и четкие линии могут дать хорошие результаты в трех основных областях дизайна: в оформлении текста, цвета и графики.

Принципы оформления текста

Представьте следующую ситуацию: вы только что начали работать в качестве Web-дизайнера на дому, и первое, что просит ваша начальница, - привести в порядок Web-сайт. На это она выделила вам одну неделю (и ноль рублей). Это

не должна быть полная переделка дизайна - ею вы займетесь, когда появятся деньги. На данный момент сайт надо лишь немного привести в порядок.

- Как получить золото из песка _____

Благодаря грамотному оформлению текста ваш малобюджетный сайт превратится из пестрого в утонченный, и вам не придется платить за это лишние деньги. Если вы ограничите число шрифтов сайта двумя или тремя хорошо гармонирующими друг с другом гарнитурами и используете четкие линии при расположении текста, ваш сайт будет выглядеть более элегантно.

Предположим, что сайт, который вам нужно привести в порядок, был создан три года назад. Выглядит он так, будто его сделал племянник начальника, в результате чего ваша компания производит непрофессиональное впечатление.

Начните с оформления текста и используйте его для определения общего стиля: упростите цветовую схему и сделайте более четкими визуальные линии вашего сайта. Подумайте о разнообразии шрифтов (как часто должен использоваться каждый шрифт) и о размещении текста на экране. Придумайте, как вы могли бы изменить эти три принципа оформления текста - разнообразие, частоту использования и размещение - с целью устранения визуального беспорядка и улучшения вашего сайта. Такой анализ начинается с трех шагов, которые помогут вам придать элегантность и стиль любому дизайну, даже если у вас недостаточно средств.

Правильно подобранное разнообразие шрифтов. Наиболее дешевый способ создать единство и структуру - выбрать две или три хороших гарнитуры. Постарайтесь выбрать шрифты, которые создают сильный визуальный контраст и хорошо смотрятся на экране компьютера:

1. Частота использования шрифтов. После того как шрифты будут выбраны, составьте небольшое резюме по визуальному стилю, в котором будет определено, как должен использоваться каждый из шрифтов. Это резюме должно содержать набор правил, затрагивающих такие элементы, как заголовки, навигация сайта, основной текст, ссылки и подписи. Совершенно очевидно, что вам придется изменять эти правила по ходу работы над сайтом. Ничего страшного, просто не забывайте обновлять каждый раз руководство по стилю.
2. Создание четких линий с помощью правильного размещения и выравнивания текста. Размещая текст на экране, уверенно создавайте четкие линии. Что это значит? Это значит, что элементы на вашей странице помогают посетителям сайта зрительно перемещаться с одного места на

90 ГЛАВА 5. Как хорошо выгнать детей с минимальными затратами

другое благодаря правильному размещению текста. Выравнивание каждого фрагмента текста должно быть четким и визуально соответствовать другим фрагментам текста на странице. Четкое размещение стоит не дороже, чем неряшливое, однако оно сильно улучшает дизайн.

Представим, что ваш краткосрочный проект - это сайт, подобный тому, что представлен на рис. 5.1. Вспомните три пункта - умеренное разнообразие шрифтов, последовательное их использование и четкие линии посредством размещения - и начните исправления.

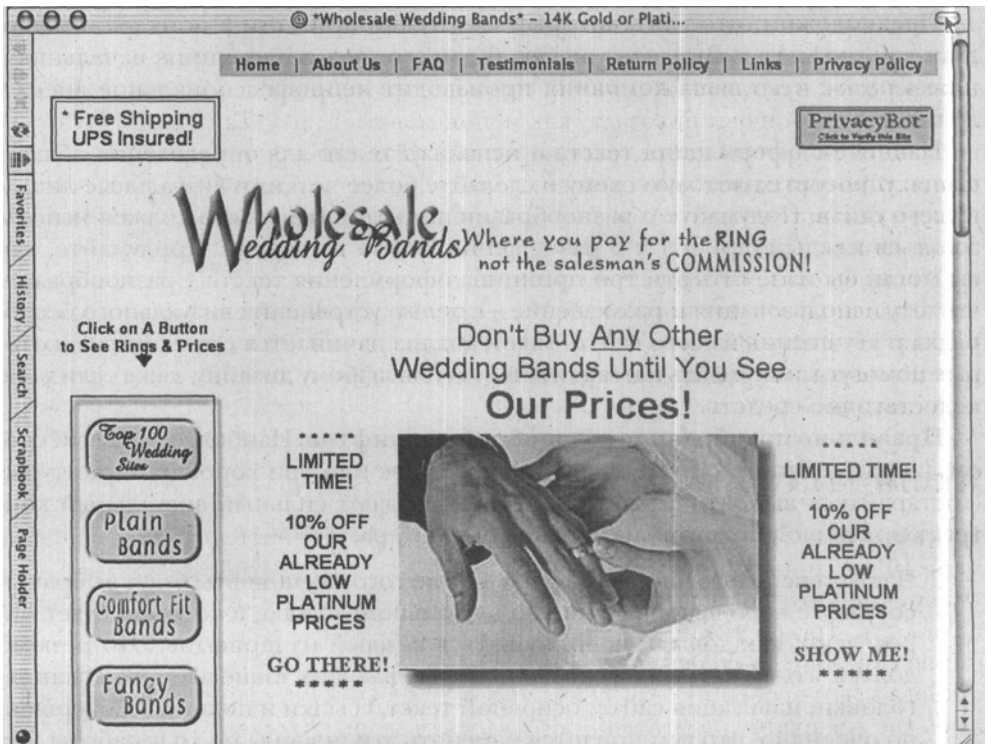


Рис. 5.1. На сайте оптовой торговли свадебными товарами (www.WholesaleWeddingBands.com) слишком много шрифтов

Использование трехшагового метода при корректировке сайта на рис. 5.1 пойдет ему на пользу. Здесь настолько много шрифтов, что трудно сказать,

какая часть сайта является навигацией, какая - заголовком, а какая - основным текстом или врезкой. В дизайне отсутствуют визуальные линии и другие структурные связующие. Пользователю приходится растерянно бродить по странице, пытаясь самостоятельно разобраться, где что находится.

-Примечание

Если вам любопытно увидеть, какой у этого сайта на самом деле потенциал, посмотрите на рис. 5.2, на котором показан сайт после нашей трехшаговой корректировки.

Этот сайт напоминает мне 14-цветный набор теней для глаз, который моя тетушка Морин подарила мне, когда я была подростком. Мои глаза разбежались от большого количества цветов, поэтому я использовала семь цветов на правом веке, а остальные семь - на левом. Морин заметила мою ошибку, отвела меня в сторонку и показала, как использовать всего два хорошо сочетающихся друг с другом цвета, которые правильно подчеркнут мои глаза. Принципы, о которых она мне сообщила, подходят для нашего сайта: вернитесь назад и оставьте только те визуальные элементы, которые имеют особое предназначение.

Владелец сайта является также его дизайнером. В случае с сайтом на рис. 5.1 его владелица купила книгу по HTML и применила на сайте все, о чем она прочитала. Мой первый сайт был не лучше этого. Сайт оптовой торговли свадебными товарами я использую в качестве примера сайтов первого и второго поколения, обреченных на переделку.

Количество шрифтов

На сайте оптовой торговли свадебными товарами используется по крайней мере 10 различных шрифтов (гарнитур). Вместе они создают визуальное противоречие, которое подрывает смысловое наполнение сайта и его функциональность. Трудно различить логически выделенные места, заголовки и навигацию сайта среди общего текста.

Беда этого особенного сайта состоит в том, что текст на нем используется в качестве декорации. Создатель сайта оптовой торговли свадебными товарами настолько вдохновился свадебной темой, что слишком увлекся рукописными и декоративными шрифтами. Можно выбрать один шрифт, который будет утонченно подчеркивать настроение, однако использовать шрифт в качестве декорации - глупо. Если вам не жаль своего времени и усилий на реализацию подобных «концепций», такой декоративный подход - первый шаг к тому, чтобы сайт выглядел дешево. Старайтесь не применять такой подход.

Когда вы определите два или три шрифта, которые вы будете использовать для сайта, ознакомьтесь со следующей традицией, которой обычно следуют типографские и Web-дизайнеры: выберите шрифт с засечками, такой как Times или Georgia, для текста, а шрифт без засечек, такой, как Verdana, Helvetica или Arial для заголовков и элементов навигации. (Однако не забывайте о проблемах, описанных ниже, см. определения в рамках.) При желании вы можете сделать наоборот: шрифт без засечек используйте для текста, шрифт с засечками – для заголовков. В умеренных дозах можно использовать декоративный шрифт при оформлении логотипа.

г-Определение

Шрифт с засечками (Serif Typeface). Шрифт с засечками отличается наличием *подстрочного* элемента *литеры* - короткой угловой линии, которая отходит от верхнего и нижнего концов линий буквы. Шрифты с засечками, которые также называются прямыми (roman) шрифтами, хороши для многих целей в печатном мире, особенно для больших фрагментов текста. Впрочем, на экране монитора они могут выглядеть как бы с зазубринами - из-за низкого разрешения по сравнению с печатной страницей. Для хорошего монитора шрифты с засечками становятся неплохим решением применительно к любому элементу Web-сайта; хотя на данный момент я предпочитаю использовать их в таких элементах страницы, как заголовки и названия рубрик. Впрочем, у любого правила есть свои исключения: шрифт с засечками от Microsoft под названием Georgia специально был разработан для хорошего отображения на компьютерных экранах с низким разрешением, он установлен по умолчанию на большинстве операционных систем Windows и Macintosh. Примеры шрифтов с засечками - Times New Roman, Georgia и Palatino.

г-Определение

Шрифт без засечек (Sans Serif Typeface). *Sans* по-французски означает «без». В шрифте без засечек отсутствуют линии подстрочного элемента *литеры*, исходящие из концов ее линий. То, что правильно в печатном мире, необязательно верно на экране компьютера. В печати шрифты без засечек обычно используются для заголовков и выносков. В сети их четкие, простые линии чрезвычайно читабельны, поэтому они отлично подходят для основного текста. Запомните, что из всех приведенных ниже примеров Verdana - наиболее разборчивый и простой для восприятия, особенно при небольших величинах кегля. Подобно шрифту Georgia, Verdana был создан специально для экранного отображения дизайнером Мэтью Картером (Matthew Carter), и он доступен на большинстве компьютеров Windows и Macintosh. Impact, который также поставляется с большинством компьютеров Windows и Mac, может эффектно выглядеть в заголовках, но не подходит для основного текста. (В CSS установите обычное начертание для Impact, чтобы избежать «двойной жирности», так как это уже полужирный шрифт.) Примеры шрифтов без засечек - Arial, Verdana и Impact.

-Определение

Декоративный и рукописный шрифт (Decorative and Script typeface). В данном случае рассматриваю декоративный и рукописный (на сленге называемый *скриптовым*) шрифты как одно и то же. Помните, однако, о том, что декоративные шрифты - отдельный случай, который окончательно не вписывается в какую-либо другую категорию шрифтов. Их цель соответствует названию: украшать. Рукописные шрифты имитируют написание от руки, и, подобно декоративным, лучше всего их использовать в малых дозах. Зона использования декоративных и рукописных шрифтов должна ограничиваться написанием логотипов и небольших брендов; основной текст и навигация, набранные этим шрифтом, слишком тяжело читаются.

Примеры декоративных и рукописных шрифтов - Bickham Script, Bermuda LP и Hobo Std.

Последовательное использование шрифтов

Когда вы отобрали основные шрифты, самое время хорошо подумать над тем, как последовательно использовать эти шрифты. Чтобы успешно справиться с этой задачей, проанализируйте содержание вашей страницы:

- какой самый важный элемент сайта на рис. 5.1? Название компании? Тематическая строка? Может, здесь больше чем одна тематическая строка? **Является ли строка Wholesale Wedding Bands, Where you pay for the ring, not the salesman's commission!** (Оптовая торговля свадебными товарами там, где вы оплачиваете кольцо, а не комиссию продавца) тематической? Это сообщение очень важно, но из-за неудачно подобранного шрифта сила его заметно уменьшилась;
- что является элементами навигации сайта? Как их организовать и подобрать шрифт таким образом, чтобы посетителю было сразу понятно, что это - элементы навигации, а не основной текст или подписи?

Когда вы составите свое руководство по стилю, в котором будет определено, где какой шрифт должен использоваться, ответы на подобные вопросы будет очень легко найти.

Ваше руководство по стилю может выглядеть так, как в табл. 5.1. Обратите внимание, что я выбрала три шрифта: шрифт с засечками для заголовков, названий рубрик и особых частей текста; шрифт без засечек для навигации и основного текста; декоративный шрифт для логотипа и названия бренда. Каждый шрифт будет использовать по-разному. Разнообразие создаст контраст, необходимый пользователям для того, чтобы различать элементы страницы. Основной текст, ссылки и навигация слева и наверху будут написаны одним и тем же шрифтом, однако цвет, начертание и размер будут различными.

Заметьте также, что я сделала столбец справа под названием «Правило CSS» ЭТИМ столбцом мы воспользуемся в главе 7.

Таблица 5.1. Образец руководства по стилю

	Шрифт*	Размер	Цвет**	Начертание	Правило CSS
Заголовок/ Название рубрики	Georgia, Times New Roman, Times, с засечками	Заголовок 1: 18 пт Заголовок 2: 16 пт Заголовок 3: 15 пт Заголовок 4: 13 пт	#330066 (пурпурный)	Полужирный	
Навигация наверху	Verdana, Arial, Helvetica, без засечек	14 пт	#000066 (синий)	Полужирный	
Навигация слева	Verdana, Arial, Helvetica, без засечек	12 пт	#000066 (синий)	Полужирный	
Логотип/ Бренд	Zapfino	18 пт, слабое сглаживание	#6633CC (бледно- лиловый)	Обычный	
Выделенные части текста	Georgia, Times New Roman, Times, с засечками	14 пт	#330066 (пурпурный)	Полужирный	
Основной текст	Verdana, Arial, Helvetica, без засечек	12 пт	#000033 (темно-синий)	Обычный	
Ссылки в основном тексте	Verdana, Arial, Helvetica, без засечек	12 пт	#6633CC (бледно- лиловый)	Полужирный, подчеркнутый	

* В графе «Шрифт» я указала несколько гарнитур. Это потому что в различных операционных системах и браузерах шрифты проявляют себя по-разному. Подробнее мы остановимся на этом позднее, в разделе «Реализация оформления текста».

** Для цветов приводятся не только их названия, но и их шестнадцатеричные значения. Так я сделала потому, что большинство цветов в языках разметки выражены в шестнадцатеричном виде. Благодаря руководству по стилю вам будет легко не запутаться в этих шестизначных числах.

Помните, что один шрифт имеет много разных вариантов. Начертание, размер и цвет ждут того, чтобы вы с ними поиграли. Используйте их - они добавят необходимый контраст вашему дизайну. Такой контраст не только визуально

приятен - он также информативен: он позволяет занятому пользователю бегло просмотреть страницу и выделить наиболее важные элементы.

Размещение текста

Оформление текста на рис. 5.2 эффективно и экономично (это моя быстрая переделка сайта оптовой торговли свадебными товарами).

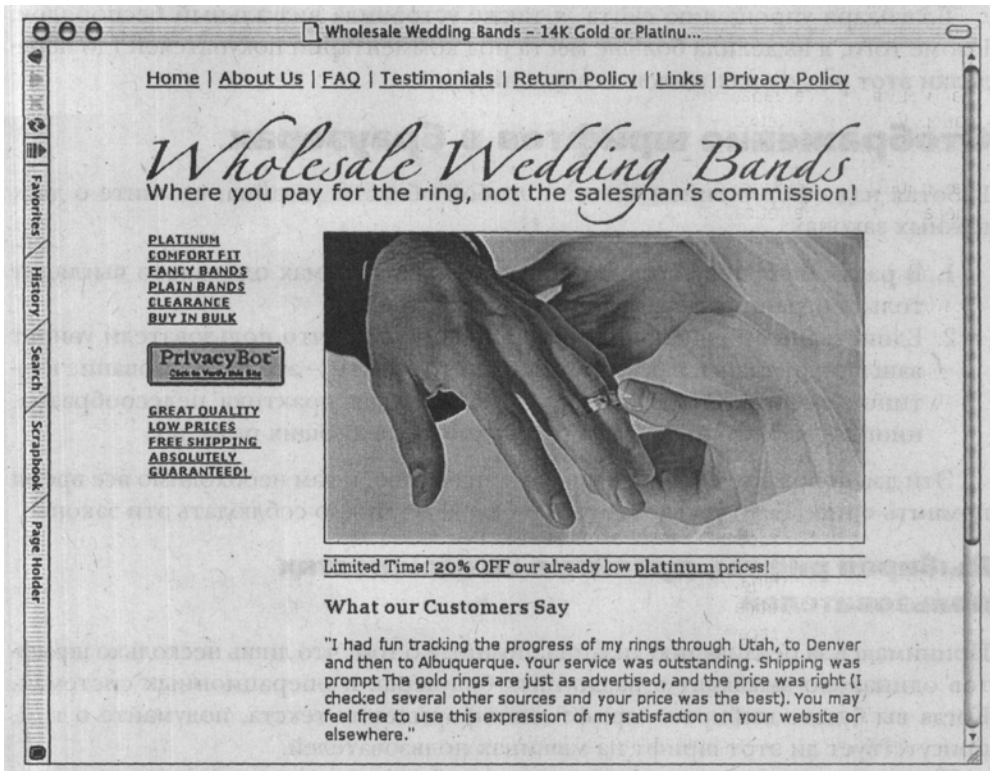


Рис. 5.2. Переделанный сайт оптовой торговли свадебными товарами. Всего лишь ограничив количество шрифтов и создав четкие линии текста, я успешно применила три принципа хорошего дизайна. Я создала простой стиль и последовательно использовала его, я упростила цветовую гамму и разместила текст так, чтобы создать четкие линии

Путем уменьшения количества гарнитур до трех, а количества столбцов до двух были созданы три четкие линии. Одна линия - крайняя левая граница сайта, другая линия образована левой границей фотографии и комментариями

покупателей и последняя линия - правая граница страницы. Выравнивание текста по комментариям покупателей и строке навигации сверху позволяет оставить правую линию четкой. Обратите внимание, как выравнивание текста сайта упрощает задачи по созданию четких вертикальных линий. Так, сайт становится более легким для чтения и использования. Этот переделанный сайт вполне может быть создан с небольшим бюджетом, однако дешево он не выглядит.

Благодаря упрощению сайта, я также устранила визуальный беспорядок. Кроме того, я выделила больше места под комментарии покупателей (до переделки этот раздел находился в самом низу).

Отображение шрифтов в браузерах

Работая над оформлением текста в любой области дизайна, помните о двух важных законах:

1. В различных браузерах и операционных системах одинаково выглядит только ограниченное количество шрифтов.
2. Единственный способ быть уверенным в том, что пользователи увидят ваш шрифт таким, как вы хотели бы его видеть, - это использование картинок вместо XHTML-текста. Иногда такая практика целесообразна, иногда - нет. Об этом будет рассказано в следующих разделах.

Эти два положения равны по своему значению, и вам необходимо все время помнить о них. Если вы заботитесь о бюджете, нужно соблюдать эти законы.

Выбирая рифты, думайте о возможностях пользователей

Принимаясь за реализацию дизайна, помните о том, что лишь несколько шрифтов одинаково выглядят в различных браузерах и операционных системах. Когда вы будете выбирать шрифт для оформления текста, подумайте о том, присутствует ли этот шрифт на машинах пользователей.

Вернемся ненадолго к табл. 5.1, в которой перечислены шрифты для каждого элемента текста. Заголовки будут оформлены шрифтом Georgia, если у пользователя есть этот шрифт на компьютере. (Практически у всех пользователей Windows и Mac он есть.) Если нет, заголовки будут оформлены Times New Roman, а если у пользователя нет ни одного шрифта из этого списка - любым имеющимся у него на компьютере шрифтом с засечками. В чрезвычайно редких случаях, если у пользователя нет шрифтов с засечками, будет использован

шрифт, стоящий в системе по умолчанию. Не забудьте назначить несколько шрифтов для каждого элемента текста.

В зависимости от доступных на данном компьютере шрифтов, вы должны отметить прежде всего предпочтительный шрифт, затем альтернативные шрифты, а потом - шрифт по умолчанию. Например, классическая операционная система Мае обладает шрифтом Helvetica, однако для мелкого текста многие пользователи Мае предпочитают Geneva, который чаще всего недоступен в Windows. Шрифтом по умолчанию стоит сделать Geneva, чтобы быть уверенным в том, что у пользователей Мае не появится Helvetica, который, как правило читается хуже.

Управление шрифтами посредством изображений

В примере с сайтом оптовой торговли свадебными товарами я выбрала шрифт Zarfino в качестве декоративного шрифта для логотипа сайта. Держу пари, что Zarfino есть далеко не у всех посетителей сайта. Тем не менее мне хочется быть уверенной в том, что все посетители сайта увидят логотип именно в Zarfino. Единственный гарантированный способ - создать логотип в графическом редакторе, таком как Adobe Photoshop или Macromedia Fireworks (рис. 5.3), и экспортировать его в формат GIF благодаря функции Photoshop под названием **Save for Web**. Затем нужно включить это изображение в XHTML-страницы.

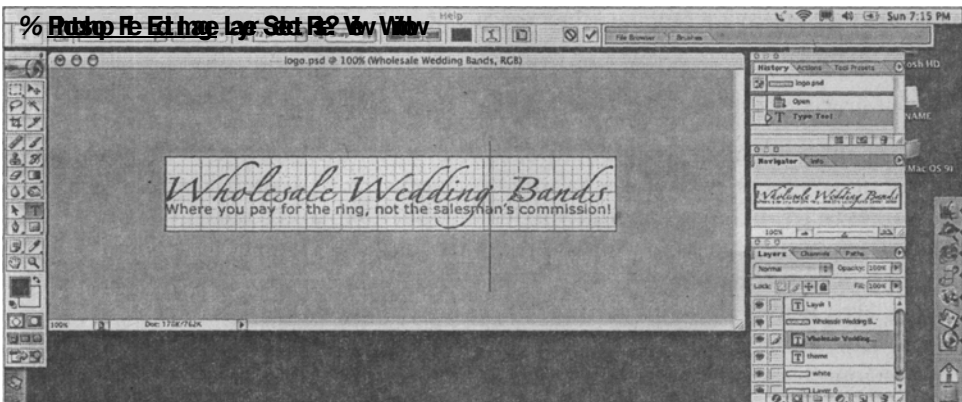


Рис. 5.3. Это файл изображения логотипа в том виде, в каком я работаю над ним в Photoshop. Я могу создать изображение, используя любой шрифт, имеющийся у меня на компьютере, а затем экспортировать его в GIF-файл, который будет использоваться у меня на сайте. Такая практика широко используется всеми Web-дизайнерами

р Угроза бюджету

Ни Adobe Photoshop, ни его альтернатива, Macromedia Fireworks, - программы не из дешевых. Однако ничто не может с ними сравниться, когда речь идет о создании профессионально выполненной графики. В случае с графическими редакторами я не могу предложить никакой дешевой альтернативы; придется потратиться. Из-за экономии на графических редакторах пострадает ваш сайт.

Если вы заменяете текст картинками, позаботьтесь об альтернативном (alt) тексте на странице для компьютеров, не поддерживающих графику (или в которых отключены картинки), так чтобы название страницы или имя компании не отсутствовало полностью на их мониторах. Замена каждой картинки атрибутом текста alt поможет вам и вашим посетителям, а в некоторых случаях и содержанию вашего сайта стать более доступным для некоторых поисковых машин.

II

-Угроза бюджету

Графические файлы могут потребовать более высокую пропускную способность канала, необходимую для загрузки вашего сайта, что равносильно потерянной времени посетителей вашего сайта с низкоскоростным соединением. В ряде случаев, когда на сайте слишком много изображений, некоторые посетители устают от ожидания и покидают сайт. В таком случае не реализуется основная задача сайта, а в случае с деловыми и коммерческими сайтами уменьшается потенциальный доход. Также, если ваша хостинг-компания ограничивает трафик данных, чрезмерное количество больших по размеру картинок скажется на расходах на хостинг.

Еще одна статья ваших расходов - создание и размещение этих картинок. Создание, оптимизация, обновление, затем опять оптимизация - все это требует времени. В любое время, когда мне понадобится изменить текст на рис. 5.3, мне придется открывать картинку в редакторе, сохранять изменения, оптимизировать картинку, а затем отправлять измененный GIF-файл снова на сервер. Если бы я могла использовать текст в виде шрифта, времени потребовалось бы меньше. Это имеет мало значения, когда на вашем сайте находится лишь несколько картинок, но по мере добавления новых картинок, требующих частого обновления, вы сразу же заметите, как возрастут расходы на производство.

Кто-то скажет, что я преувеличиваю проблемы, связанные с использованием изображений вместо текста. Это будет вполне справедливо. Существуют способы работы с изображениями, которые не вызывают сложностей, связанных с большим трафиком и расходами на производство. Во-первых, размер изображения должен быть как можно меньше. Функция Photoshop Save for Web

легко справляется с этой задачей без потери качества. Во-вторых, помните о том, что повторяющиеся изображения сохраняются на жестком диске пользователя, экономя время пользователя на передачу данных (и ваши расходы на хостинг) при посещении более чем одной страницы вашего сайта. Это не смертельный, даже не страшный грех - использовать картинки таким образом; удостоверьтесь, что вы не применяете текстовые изображения неоправданно. Используйте их разумно!

Текстовые GIF-изображения подходят для логотипов, а в некоторых случаях - даже для подзаголовков. Однако большую часть Web-текста, включая подзаголовки, лучше всего оформлять в виде форматированной XHTML-разметки при помощи CSS.

Сэкономьте время и деньги при помощи CSS

Для контроля над оформлением текста используйте CSS - Cascading Style Sheets (каскадные таблицы стилей). Это не только уменьшит расходы на хранение и трафик при эксплуатации сайта, но и упростит поддержку сайта, а также возможную и неизбежную его переделку.

CSS обеспечивают общий способ управления компоновкой и дизайном Web-документов. Более подробно мы рассмотрим CSS в главе 7, но основная идея состоит в следующем: уберите всю работу по представлению элементов из XHTML и позвольте таблицам стилей сделать эту работу за вас. Кроме того, уберите теги шрифтов, разделительные GIF-изображения и другие теги и атрибуты представления данных и также дайте таблицам стилей выполнить эту работу за вас.

Г Определение

Представление (Presentation). Говоря о представлении на Web-сайте, я имею в виду то, как ваши страницы выглядят в браузере. В некоторых случаях представление на одном устройстве, платформе или в браузере может отличаться от других. Я знаю, как это контролировать, но не совсем точно.

(-Определение _____; _____)

Структура (Structure). Говоря о «структуре» XHTML-страницы, в идеале я имею в виду логическую структуру разметки. Идея состоит в том, чтобы убрать из XHTML-кода все теги и атрибуты, касающиеся представления, и свести разметку к простейшим, наиболее логичным элементам. CSS затем добавляет слой представления в разметку, когда страница открывается в браузере. Если сейчас вам пока не все понятно, запомните эти идеи и приберегите их для главы 7.

Такой способ форматирования отличается от классического метода, который использовался на ранних этапах развития Web-дизайна. В былые дни для стилизации каждого абзаца, каждого заголовка требовался отдельный тег шрифта. Это было все равно что ходить в школу, каждый день поднимаясь на гору по пути туда и обратно. Благодаря CSS ваш XHTML-код будет кратким и простым, а определить оформление абзаца или заголовка вы сможете в CSS-документе. Для выполнения этой работы нужно совсем немного времени, зато создание, поддержка и переделка будут осуществляться гораздо быстрее - для перегруженных, слабо финансируемых дизайнеров всего мира это как бальзам на душу.

Если вы разумно создаете ваш сайт - используете внешние таблицы стилей вместо встроенных и хорошую разметку, - ваши страницы будут меньше. Это сэкономит расходы на передачу данных (и необходимую пропускную способность). Вы заметите реальную экономию при создании HTML-разметки; этот процесс протекает заметно быстрее, если использовать CSS для выполнения работы по представлению страницы.

Использование CSS

Из этого раздела вы узнаете, как можно использовать CSS, чтобы выполнить работу в недельный срок и уменьшить нужную пропускную способность. Во-первых, удалите все теги шрифтов. На рис. 5.4 показана HTML-разметка сайта оптовой торговли (см. рис. 5.1) свадебными товарами до его переделки.

Как видно из строк 54 и 63, теги шрифтов присутствуют повсюду на сайте, «съедая» пропускную способность, хотя их присутствие здесь необязательно. Если они имеются на вашем сайте, удалите их и позвольте CSS выполнить оформление шрифтов за вас.

Этот сайт содержит достаточно много ненужных HTML-тегов, которые «пожирают» ресурсы. Переделанный XHTML уменьшит разметку с 28 до 4 Кб, плюс 2 Кб на таблицу стилей. В главе 7 мы вернемся к этому примеру и используем тот же самый сайт в качестве упражнения по созданию более компактной и менее дорогой разметки.

В строке 52 на рис. 5.4 вы видите вездесущий разделительный GIF-файл. Начиная с браузера Netscape 1.1 дизайнеры использовали разделительные картинки для того, чтобы добавить края и пробелы на сайты. Разделительные теги появились в NS Navigator 3.0, но и в 1995 году дизайнеры использовали прозрачные GIF-изображения для скрепления элементов дизайна. Большая проблема, связанная с этими изображениями, состояла в том, что, в зависимости от конфигурации, Netscape пытался загрузить их каждый раз, когда они

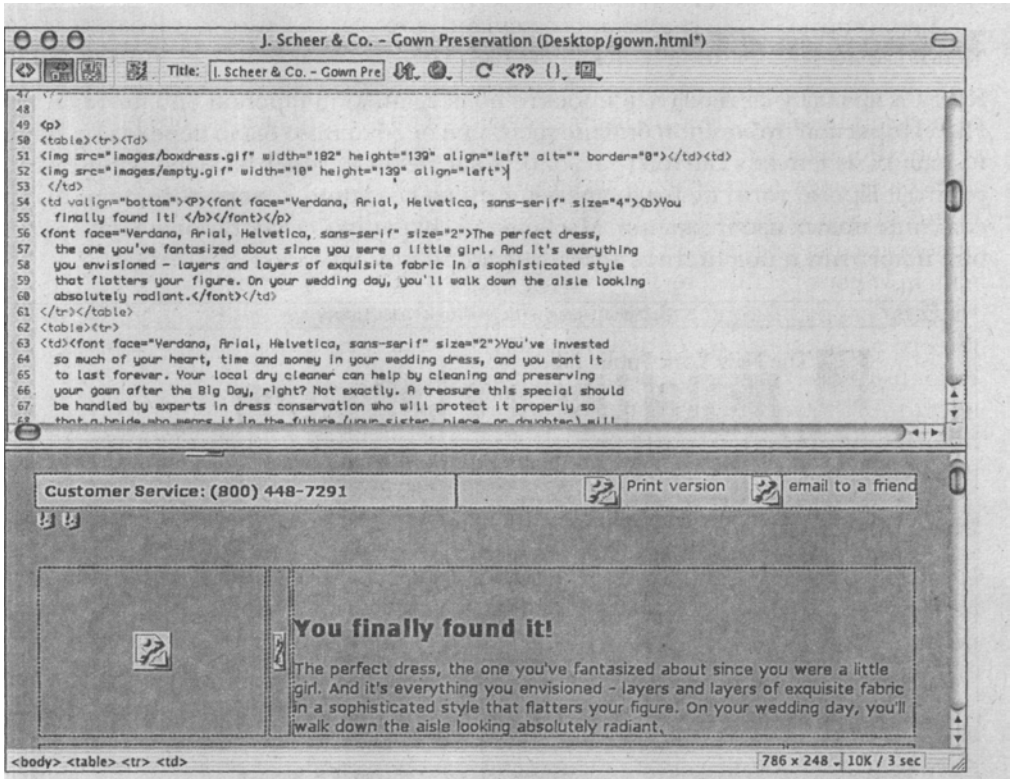


Рис. 5.4. Исходный код на рис. 5.1 выявляет сайт, переполненный громоздкими, «съедающими» время и трафик тегами шрифтов и разделительных GIF-изображений. Более экономичный подход - переложить работу по дизайну на Cascading Style Sheets

присутствовали на странице. Сайт, использовавший прозрачные GIF-файлы, требовал большой пропускной способности.

На практике пропускная способность до сих пор зря расходуется из-за присутствия в Сети миллиардов этих пустых маленьких картинок и из-за написания ссылающейся на них разметки. Вот совет по экономии: удалите их.

Если вам необходимо разделение между элементами страницы, можете создать его в таблице стилей. Если вам необходим отступ справа всего лишь для одной картинки и вы не хотите, чтобы при этом на вашем сайте появлялись еще какие-то изображения, определите для этого встроенный стиль. Мы коснемся данной темы *более* подробно в главе 7.

Экономия с CSS: пример из жизни

Когда я начала участвовать в проекте по созданию цифровой библиотеки для Нью-Йоркской публичной библиотеки, мне необходимо было переделать один из наших маленьких сайтов (рис. 5.5). У нас не было денег на выполнение этой работы. Кроме того, не было времени ни на создание нового логотипа, ни на создание новых изображений. Мы решили сохранить существующие изображения и логотип и подчистить текст сайта при помощи таблицы стилей.



Рис. 5.5. Так выглядел сайт цифровой библиотеки до двухдневной очистки

Распорядок двух дней производства выглядел следующим образом:

День первый:

- изменение существующей таблицы стилей (2 часа);
- подчистка старой, громоздкой HTML-разметки и замена ее на элегантную, структурную XHTML-разметку (4 часа).

День второй:

- тестирование на различных браузерах и операционных системах. Просмотр того, как сайт выглядит в различных браузерах и операционных системах (подробнее о браузерах см. в главе 2). Как вы увидите в главе 7, при использовании CSS тестирование проходит быстрее и эффективнее (4 часа);
- запуск.

В результате получился более четкий, оформленный в одном стиле сайт (рис. 5.6). Для ссылок и заголовка мы использовали цвет, который был в предыдущей версии сайта. Мы увеличили интерлиньяж и отступы основного текста, чтобы сделать текст более читабельным. Работы заняла около 10 часов и была распределена между двумя людьми, которые выполнили ее в свободное

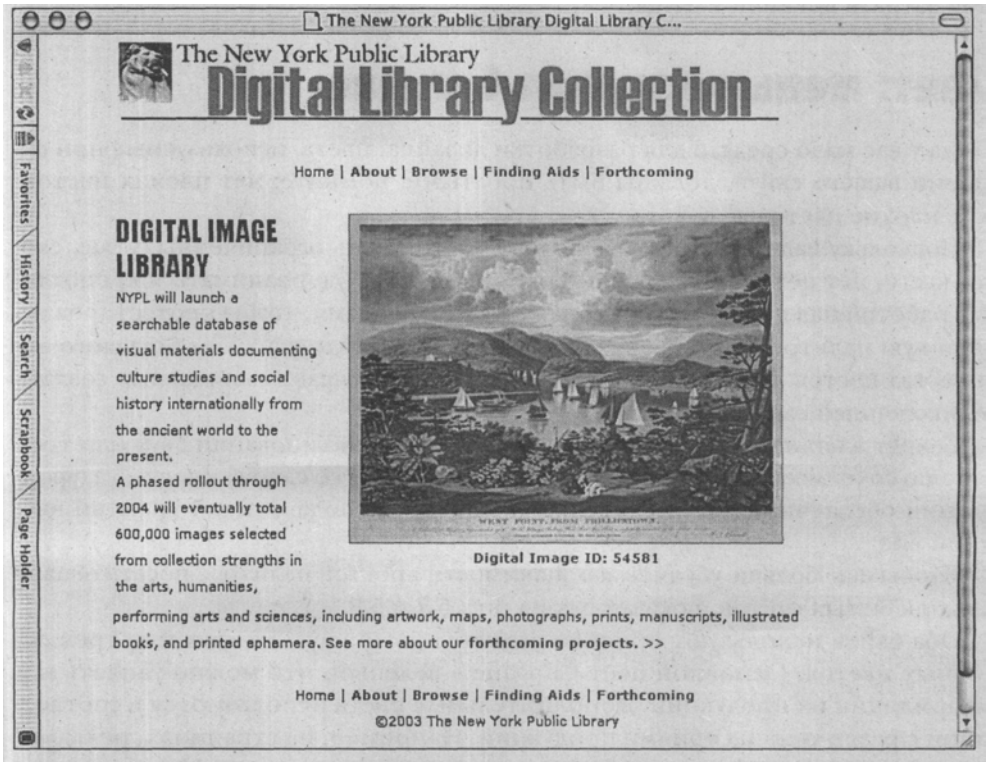


Рис. 5.6. Это тот же самый сайт цифровой библиотеки после простой двухдневной очистки, основанной на CSS

от основной работы время. Таблица стилей основывалась на таблице предыдущего проекта, поэтому создание прошло быстро.

Как уже отмечалось, мы изменяли дизайн сайта временно; через несколько месяцев мы будем работать с дизайнером, чтобы создать новый логотип и совершенно новый внешний вид. Если бы при выполнении этой временной работы мы не использовали бы CSS, я бы сочла это задание достаточно длительным по времени. Но теперь, когда мы подчистили XHTML-разметку и сделали более дружелюбный, читабельный и удобный дизайн на основе CSS, мы лучше подготовлены к созданию окончательной, улучшенной версии дизайна. Использование CSS и очистка разметки не только сэкономили наши время и деньги при создании этого временного, переходного дизайна; это также сэкономило время и деньги для создания окончательной версии, благодаря тому что самые большие его проблемы были решены в домашних условиях при небольшой затрате времени и денег.

Цвет: меньше значит больше

Когда у вас мало средств для разработки дизайна, цвета, используемые при создании вашего сайта, должны быть простыми; помните: нет плохих цветов, есть плохие цветовые сочетания.

Поскольку ваши финансовые возможности очень ограничены, у вас, скорее всего, нет целой команды по дизайну, которая будет заниматься кропотливой работой над цветом; вряд ли также у вас есть время, чтобы протестировать цветовую палитру на разных компьютерах. Избегайте слишком большого количества цветов. Использование 2-3 оттенков повышает ваши шансы создать великолепный сайт.

Секрет элегантного внешнего вида состоит в использовании двух или трех хорошо сочетающихся цветов. Такая техника вместе со сдержанностью в оформлении обеспечит вашему сайту изысканный и неподвластный времени внешний вид.

Чтобы еще больше убедиться в значимости простой палитры, посетите пару моих любимых сайтов, показанных на рис. 5.7 и 5.8.

Оба сайта используют простую палитру, состоящую из двух или трех основных цветов. Основной цвет Clinique - зеленый, что можно увидеть и в оформлении их продукции. Дополнительные цвета используются в соответствии с различными сериями продукции. Например, на страницах, рекламирующих помаду, используются цвета новых оттенков сезона. На страницах,



Рис. 5.7. Сайт Clinique придерживается простых цветов (www.clinique.com). Торговая марка зеленого цвета сразу выделяется, а осторожное использование коричневого, бежевого и розового обеспечивает визуальный контраст

представляющих коллекцию пудры и кремов, используются бежевый и коричневый, соответствующие цветам данной продукции. Такое изобилие цветов может быть чрезмерным для небольшого бюджета; создание и поддержка цветовой гаммы для каждой страницы могут оказаться вам не по карману. Я предполагаю, что у людей, читающих эту книгу, нет достаточного количества времени, чтобы так детально разрабатывать цветовую гамму сайта. Недорогой способ добиться такого же сбалансированного эффекта - работать с той же цветовой схемой, что и у Линды Уэйнман. Заголовки на сайте Линды бледно-лилового цвета, а ссылки - близкого к нему голубого. Я не призываю вас копировать цвета с ее сайта, но надеюсь, что вы позаимствуете ее опыт использования небольшого количества родственных цветов.

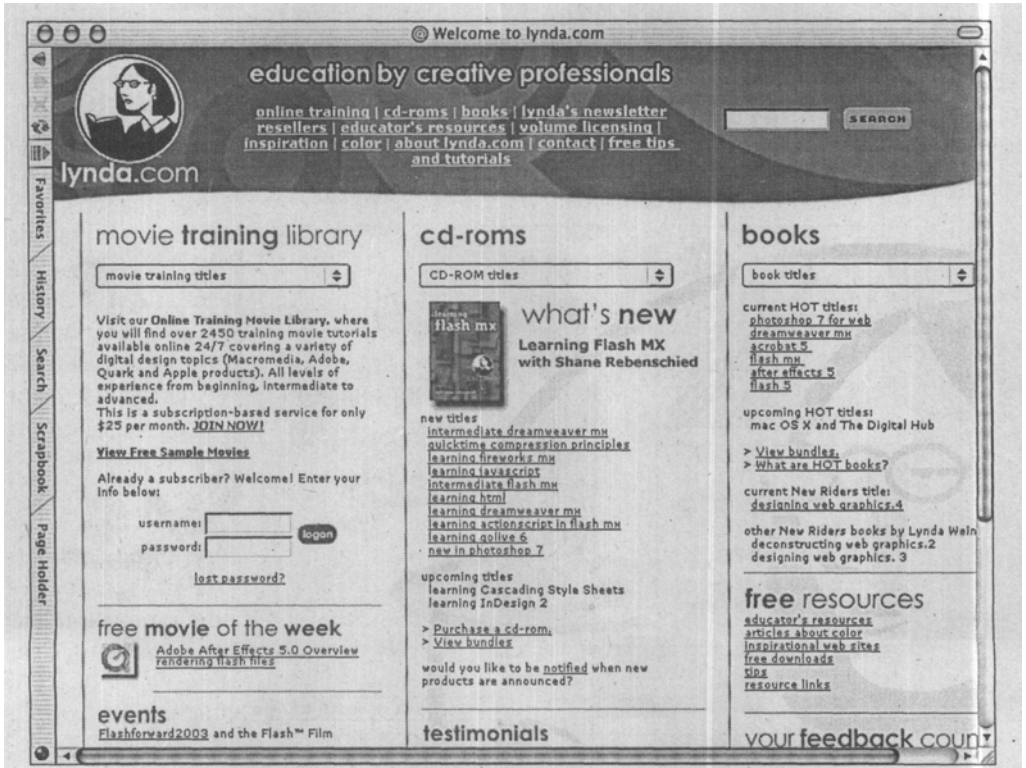


Рис. 5.8. У дизайнера и преподавателя Линды Уэйнман (Lynda Weinman) всегда был великолепный сайт (www.lynda.com/resources/inspiration). Сайт менялся несколько раз, но каждый раз Линда крайне экономно использовала цвета. Она тяготеет к использованию двух связанных цветов одновременно. Также она поддерживает сайты, использующие цвета таким же образом

Цвет: проще при помощи CSS

Использование CSS - это лучший способ управлять цветами вашего сайта. В главе 7 вы увидите, как назначать цвета текста, ссылок, фона и даже цвет ролловеров при помощи CSS. Когда времени на переделку сайта очень мало, значительная часть работы может быть сделана за несколько минут путем изменения таблицы стилей.

Рисунки и фотографии

Рисунки и фотографии могут дорого стоить. Часто, когда дизайнеры пытаются сэкономить в этой области, сайт выглядит дешево. Рис. 5.9 служит примером того, как неудачно реализовалась идея экономии денег. Фоновая картинка представляет собой фотографию кружевной ткани, из которой пошиты свадебные платья. Выглядит это так, будто дизайнер отсканировал кусочек кружева на планшетном сканере. Кружево немного переэкспонировано и не сфокусировано. Использование изображения кружева привлекает ненужное внимание к банальному элементу дизайна. И что в результате? Уберите его, сайт выглядит немного уныло. Впечатление малобюджетности, которое производит этот сайт, вряд ли позволит мне доверить одежду ценой в тысячу долларов в руки этой компании: кажется, они любят экономить. Значит, они сэкономят и при обработке моего платья?

К тому же в использовании такой картинки есть еще один недостаток: ее размер - 140 Кб. Это огромный файл для загрузки и большой файл для хостинга, если вы платите за количество трафика (потребление пропускной способности). Возможно, вы не сможете себе позволить каждый раз пересылать 140 Кб (см. главу 8, чтобы узнать о трафике и о расходах на пропускную способность). Даже если вы можете себе это позволить, ваши посетители с коммутируемым соединением, вероятно, не смогут. Размер 140 Кб в 40 раз превышает размер файла, который следует использовать для изображения фона.

Я не хочу сказать, что использование изображений для фона - это всегда плохо. Меня беспокоит то, как вы создаете файлы фоновых изображений. Вот несколько советов:

- удостоверьтесь, что качество изображения максимально высокое. Если вы не можете обеспечить фокус и экспозицию достаточно высокого качества, найдите небольшую контору, где вам смогут это сделать;
- удостоверьтесь в том, что размер файла небольшой. Постарайтесь, чтобы размер вспомогательных изображений, которые не влияют на содержание сайта, был не более 3 Кб, а то и меньше;
- если вы используете фон, следуйте советам по использованию внешних таблиц стилей для отображения фоновых картинок в главе 7. Это более экономично с точки зрения расходов на хостинг и гораздо удобнее для пользователей с низкоскоростным подключением к Интернету.



Рис. 5.9. Дешевые фотографии очень портят впечатление о сайте и качестве предлагаемых услуг (www.after-the-event.co.uk). Фоновая картинка в виде кружева на странице, посвященной свадебным платьям, выглядит непрофессионально - разве потребитель отдаст одежду, представляющую фамильную ценность, в ремонт любителю? Недорогие изображения не должны выглядеть дешево, однако, когда так происходит, страдает ваш сайт и торговая марка, а следовательно, и ваш бюджет

Будьте осторожны с использованием клипартов

Клипарт - это хитрая вещь¹. Декоративным потенциалом этого жанра легко увлечься. Постарайтесь, чтобы этого не произошло. Напротив, помните о том, что клипарт лучше всего использовать в небольших дозах для создания

В этой книге, как и в большей части переводной литературы, под понятием *клипарт* подразумевается платно или бесплатно распространяемый набор рисунков, используемых в оформительских целях, в том числе при создании сайта. - *Прим. науч. ред.*

приятных глазу значков, которые помогают пользователям бегло просмотреть ваш сайт. Помните о принципах для достижения хорошего внешнего вида с минимальными затратами: выберите определенный стиль и следуйте ему, пусть цвета будут простыми, создавайте четкие и согласующиеся линии. В итоге использование клипартов должно подчиняться двум принципам:

1. Создавайте визуальную гармонию. Найдите изображения, сочетающиеся по стилю, размеру, ракурсу и цвету. Также постарайтесь отобрать картинки, которые будут выглядеть так, будто принадлежат кисти одного и того же художника.
2. Работайте с высококачественными изображениями. Найдите изображения максимально высокого качества, а затем подстройте их под необходимые размеры и разрешение. Проще работать с изображениями, изначально обладающими высоким качеством и высоким разрешением.

Давайте посмотрим несколько примеров (рис. 5.10-5.12), где эти задачи выполнены удачно и неудачно.

Как найти недорогой клипарт

Проблемы, существующие на сайтах, подобных изображенным на рис. 5.10 и 5.11, возникают в том случае, если дизайнеры ищут клипарт в Сети. Попробуйте ввести в поисковике Google строку **Free Library Clipart** (Бесплатная библиотека клипарта), и вы поймете, что я имею в виду. Бесплатных наборов так много, что трудно найти что-нибудь стоящее⁴. К тому же, очень сложно выбрать несколько изображений высокого качества, которые могут создать визуальную гармонию, если вы имеете ограниченный доступ к Сети. Придется выискивать хорошие картинки среди массы плохих, а это дорогой и длительный процесс. Благодаря некоторым источникам, этот процесс может стать несколько проще.

Клипарт в Сети

Когда мне необходимо найти клипарт в Сети, я обращаюсь к следующим ресурсам, включая Veer (рис. 5.13). Что-то бесплатно, за что-то приходится платить (табл. 5.2). Как вы можете догадаться, чем больше платите, тем выше качество. Однако, пользуясь дорогими источниками, можно переплатить лишнее.

Библиотеки картинок

Библиотеки картинок можно купить на сайте CD. Art Explosion (www.hallogram.com/artexplosion) - это коллекция, которую я использовала много раз. Это несколько дисков, читающихся в Windows или Mac, которые обычно стоят \$100-200,

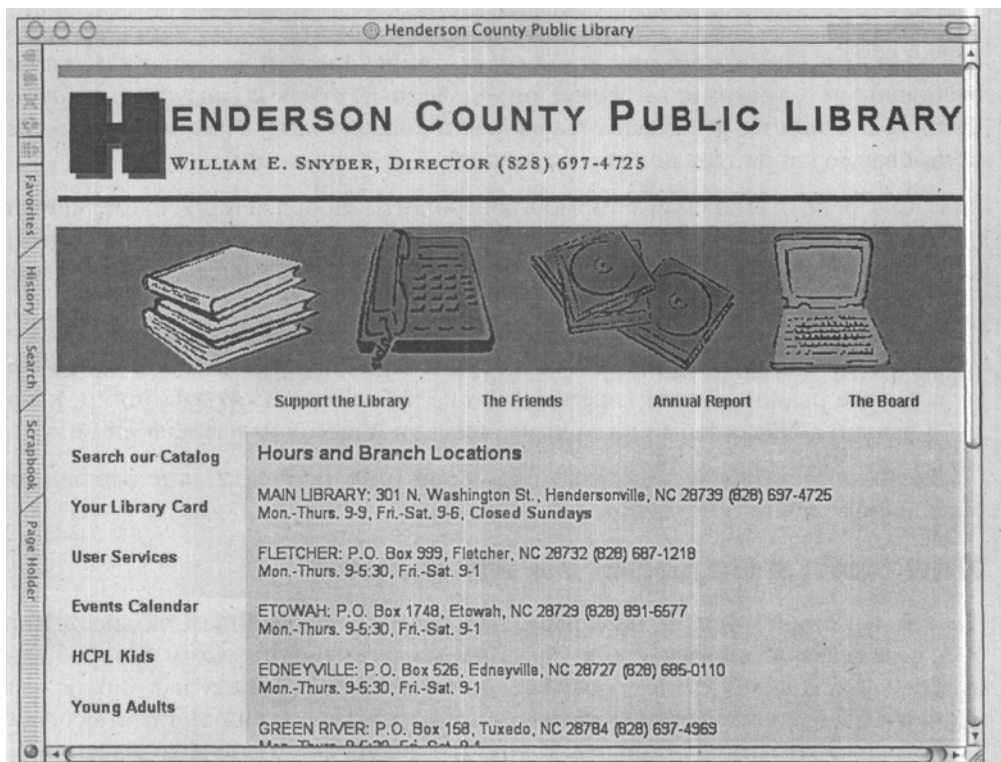


Рис. 5.10. Сайт публичной библиотеки графства Хендёрсонвилл (www.henderson.lib.nc.us). С первой задачей клипарт справился хорошо: каждая иконка соответствует требованиям стиля, размера, ракурса и цвета. Однако качество рисунков оставляет желать лучшего. Выглядит это так, будто дизайнер начал работать с изображениями изначально низкого качества, а затем сделал их качество еще хуже. В результате картинки выглядят расплывчато; а надо, чтобы они выглядели более отчетливо. Доводилось ли вам когда-нибудь слушать музыку на кассете, переписанной с другой кассеты, которая, в свою очередь, тоже была переписана с кассеты? Наверное, вы обращали внимание на значительные потери в качестве звучания. То же самое случается и при использовании изображения низкого качества; если у вас есть исходное изображение с высоким разрешением, файл, который получится в результате, будет гораздо лучшего качества. Прежде чем перейти к следующему примеру, я должна отметить кое-что важное: телефон директора библиотеки удачно размещен в верхней части страницы. Но теперь по этому телефону производится обслуживание клиентов

в зависимости от размеров коллекции. На eBay (www.ebay.com) я видела коллекцию из 10000 изображений примерно за \$10. Покупайте самую большую,



Рис. 5.11. Вместо того чтобы набирать картинки разных стилей, дизайнер сайта публичной библиотеки Мансфилда и Ричмонда (www.mrcpl.lib.oh.us) мог бы остановиться на каком-то одном стиле и использовать его последовательно. Иконки с календарем и компьютером, по идее, должны помогать пользователю быстро просматривать и отбирать информацию, но из-за того, что каждый рисунок разного размера, освещенности, ракурса и цвета, иконки только портят и без того дисгармоничный внешний вид сайта. Дизайнер хорошо смог реализовать второй принцип использования клипарта: качество изображений высокое, но сайт проигрывает из-за невыполнения первой задачи: ему крайне необходима визуальная гармония

которую можете себе позволить; эти картинки вы будете использовать много раз.

Декоративные графические шрифты

Пытаясь сэкономить деньги, не забудьте о декоративных графических шрифтах или о шрифтах символов, которые, вполне возможно, уже есть на вашем

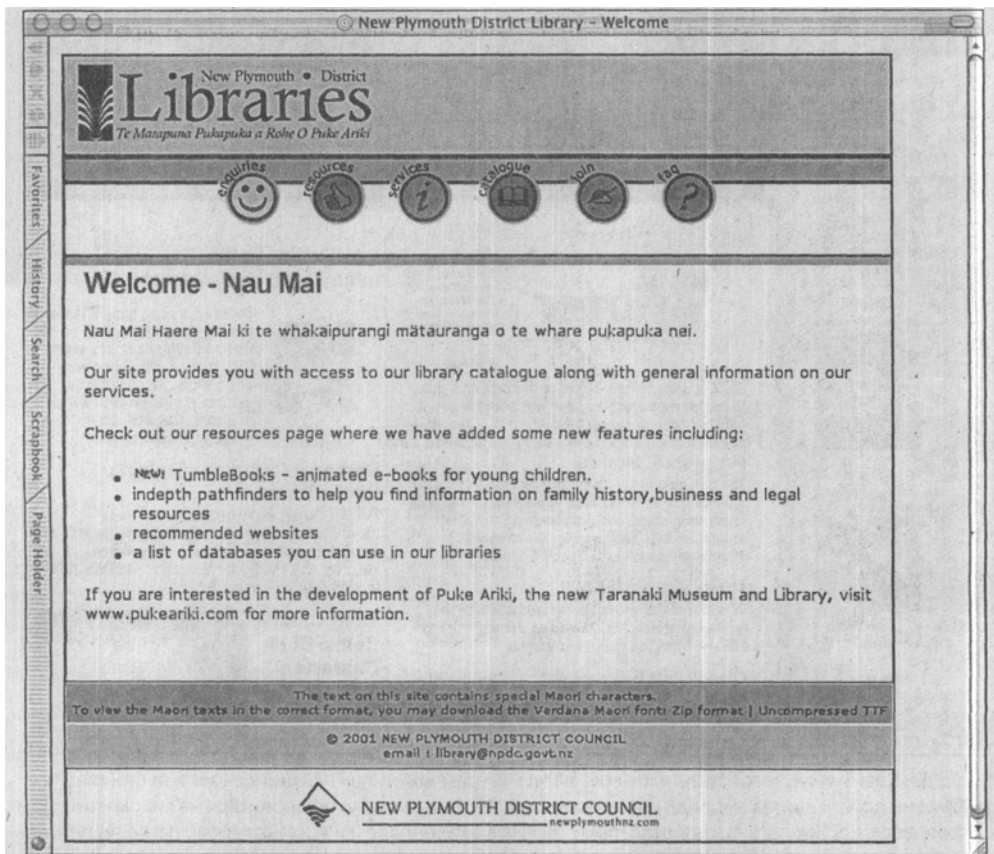


Рис. 5.12. Сайт новой областной библиотеки Плимауса (www.pukeariki.com/npd1) использует высококачественные клипарты, создающие визуальную гармонию. При более внимательном рассмотрении становится ясно, что у каждой иконки разное происхождение: символы взяты из набора шрифтов, картинки - из набора клипартов. Несмотря на это, они отлично смотрятся вместе, потому что дизайнер придал им одинаковый размер, одинаковые округлые формы. Даже оригинальное размещение заголовков рубрик над каждой иконкой придает дизайну сбалансированность и последовательность. Наиболее эффективно клипарт работает при гармоничном сочетании его элементов; рисунки выполняют возложенные на них задачи навигации

компьютере. Это шрифты, которые содержат символы, а не буквы. Webdings и Wingdings - это два шрифта, которые есть в большинстве персональных компьютеров. Набор декоративных графических шрифтов вроде Big Cheese или



Рис. 5.13. Veer (www.veer.com) - это своего рода собрание лучших картинок, которые вы можете найти на других сайтах. Правда, здесь уже собраны не столько клипарты, сколько настоящие произведения искусства, хотя некоторые из них вполне могут быть использованы в качестве клипарта. Например, одна из коллекций называется Корпоративный Панк (Corporate Punk). За \$350 на сайте Veer вы можете купить 40 галерей. Это может показаться дорого, но вы получите высококачественные картинки, которые в целом создают визуальную гармонию. Подумайте, сколько бы времени занял тщательный отбор бесплатных картинок. Можете ли вы позволить себе заниматься этим? Готовые коллекции картинок вы можете использовать неоднократно в разных целях

ZeitGuys можно купить примерно за \$60 у компании Emigre (www.emigre.com). Также можно найти пусть менее интересные, зато бесплатные декоративные графические шрифты, набрав в строке поиска **freeware dingbat** или **symbol fonts**.

Преимущество использования декоративных графических шрифтов перед библиотеками клипартов состоит в том, что достичь визуальной гармонии

Таблица 5.2. Руководство по клипартам

Компания	Цена	Хорошие новости	Плохие новости
ClipArt.com (www.clipart.com)	Зарегистрированные пользователи платят примерно \$8 в неделю, либо \$150 за год. Вы можете скачивать все что угодно (и сколько угодно). Отдельной платы за каждое изображение нет	Цена небольшая, выбор большой, можете скачивать картинки высокого качества. Сайт также предлагает шрифты, фотографии и другие ресурсы	Плохих картинок здесь больше, чем хороших. Поэтому придется потратить достаточно большое количество времени, отсеивая лишнее, а время – это деньги. Также предстоит долгая работа по отбору изображений с точки зрения стиля, размера, ракурса и цвета
Veer (www.veer.com)	Одна картинка может стоить \$100 и более, а коллекция картинок – \$200 и выше	Легко найти серию картинок, которая соответствует обеим нашим задачам (визуальной гармонии и высокому качеству изображения). Veer также предлагает фотографии, шрифты и множество других визуальных ресурсов	

становится гораздо проще: один декоративный графический шрифт содержит набор символов, которые отлично смотрятся вместе. Еще одна положительная черта состоит в том, что, поскольку декоративные графические шрифты созданы на векторной основе, они могут быть настолько высокого качества и разрешения, насколько вам понадобится. Можете использовать их в графическом редакторе (Photoshop, Fireworks), а затем придать им любой размер, и это не повлечет за собой никаких ухудшений качества. Поэтому, работая с ними, удостоверьтесь, что шрифты, которые вы покупаете, масштабируются. Иными словами, покупайте шрифты PostScript, TrueType или OpenType.

Высококачественные растровые шрифты, такие как на сайте Joe Gillespie's MiniFonts (www.minifonts.com), превосходно подходят для некоторых областей дизайна, но в силу своих особенностей растровые шрифты не масштабируются: они могут быть использованы только в одном определенном размере, либо быть кратными этому размеру. (Например, растровый декоративный графический шрифт размера 10 пт также может быть использован в размерах 20 пт, либо 30 пт, а размера 14 пт быть не может.)

Как найти подходящие фотографии и рисунки

Клипарт - дело хорошее, и он может стать лучшим другом дизайнера, работающего в условиях малобюджетности. Однако клипарт по-своему ограничен. Если вам необходимо сверкающее изображение логотипа, маленькими значками тут не обойтись. Однако профессиональные фотографии и рисунки стоят дорого. Рисунки для хорошего 10-страничного сайта могут стоить около \$1000. Как избежать таких расходов?

Что находится в ваших картотеках и ящиках письменного стола? Просмотрите ежегодные отчеты вашей организации и брошюры по изобразительному искусству. Спросите у коллег, может быть, есть фотографии, сделанные для каких-то других целей. Наверняка найдется подходящий материал и для вас. Если вы ищете уже готовый материал, помните о двух вещах:

1. То, что ваша компания использовала какое-то произведение искусства в печати, еще не значит, что вы имеете право использовать его в Сети. Узнайте, кто на самом деле владеет этим предметом, убедитесь, что у вас есть необходимое разрешение. Помните о том, что подобное вторичное использование интеллектуальной собственности может обернуться для вас кошмаром, связанным с авторскими правами; если художник узнает о том, что вы нелегально использовали его произведение, вам придется раскошелиться. Если вы не уверены в том, что на использование данного произведения у вас есть разрешение, лучше забудьте о нем. Те, кто занимался производством брошюр и ежегодных отчетов, могут сказать вам, какие контрактные соглашения имеются с фотографами в случае работы с ранее опубликованными материалами организации.

2. Будьте внимательны при сканировании рисунков из бумажной брошюры - по неосторожности они могут получиться плохо. Лучше всего найти оригинальный фотографический слайд, либо файл с изображением в цифровом виде.

-Определение

Изображения без оплаты роялти (Royalty-Free). *Изображения без оплаты роялти* - не бесплатные. Без оплаты *роялти* означает, что вы можете купить и полностью обладать правами на использование изображения. Вы можете использовать изображение сколько угодно раз, при этом не платить каждый раз дополнительно. У вас нет исключительных прав на изображение; любой заплативший денежный взнос может использовать эту же фотографию или рисунок.

Г Определение

Управляемые права (Rights-Managed). *Изображения с управляемыми правами* - это изображения, за каждое использование которых вы должны платить отдельно. Стоимость использования зависит от того, как вы используете изображение и как долго вам необходимо его использовать. Я использовала изображения с управляемыми правами на крупных Web-сайтах, и это стоило достаточно дорого. Нам пришлось нанять отдельного работника на полный рабочий день, задача которого состояла в том, чтобы вести отчетность об использовании этих изображений; это дорогая работа. Для дизайнеров, работающих с небольшими бюджетами, более экономичный выбор - использование фотографий и рисунков без оплаты роялти.

Таблица 5.3. Гид по фотографиям и рисункам

Компания	Цена	Хорошие новости	Плохие новости
Цифровая галерея Нью-Йоркской публичной библиотеки (The New York Public Library's Digital Gallery) (http://digital.nypl.org/imagegate/)	Бесплатно	Это большая база данных, и она постоянно растет. К 2004 году сайт будет содержать примерно 600000 изображений. Новые изображения добавляются регулярно. Практически все изображения не защищены авторским правом	Поскольку практически у всех изображений отсутствуют авторские права, вы не найдете современных фотографий. Если вам понадобится изображение высокого разрешения, это может занять время, плюс, возможно, за это придется дополнительно заплатить

Таблица 5.3. Гид по фотографиям и рисункам (окончание)

Компания	Цена	Хорошие новости	Плохие новости
Проект «Американские воспоминания» в Библиотеке Конгресса (The American Memory Project at the Library of Congress) (http://memory.loc.gov/)	Бесплатно	Эта обширная коллекция посвящена истории США. Естественно, вы найдете здесь изображения. Также есть книги, музыка и фотографии 3D-объектов. Я обращаюсь к этому сайту, когда мне нужна какая-нибудь иллюстрация, например изображение банджо	Как и в предыдущем случае, новые материалы здесь отсутствуют
Rebel Artist (www.rebelartist.com)	Около \$3 за изображение	Rebel Artist предлагает достаточно хорошую коллекцию клипартов и шрифтов, а также фотографий	Здесь вы не найдете ничего настолько же хорошего, как в более дорогих библиотеках произведений искусства
Istockphoto (www.iStockphoto.com)	Загрузка одного рисунка стоит 25 центов	Вы можете купить коллекции стоимостью \$10, \$20, \$50 или \$100. Это коллекции недорогих файлов хорошего качества без оплаты роялти, которые были собраны членами сайта. Загрузка одного изображения высокого разрешения стоит 25 центов и разрешает вам использовать фотографию в коммерческих Web- и печатных публикациях. Если вы фотограф, можете стать членом сайта. Если вы разместите свои собственные изображения, то будете получать проценты от их продажи	Несмотря на то что коллекция постоянно растет, выбор небольшой по сравнению с Getty Images или Veer.com и здесь искать особо нечего. Та же организация содержит коллекцию Istockpro (www.iStockpro.com), где изображения более высокого качества
Getty Images (www.gettyimages.com)	От \$25 и выше	Вы можете найти прекрасные фотографии и иллюстрации, благодаря которым ваш сайт будет выглядеть на миллион долларов. Потратив немного сейчас, вы достойно окупите ваш сайт в дальнейшем	Только цена. Ограничьте свой поиск изображениями без оплаты роялти. Изображения с управляемыми правами стоят слишком дорого
Veer (www.veer.com)	Примерно такая же, как и на Getty Images	Veer напоминает Getty Images и был основан людьми, ранее работавшими в Getty. Качество и выбор потрясающие	Такие же, как и на Getty Images

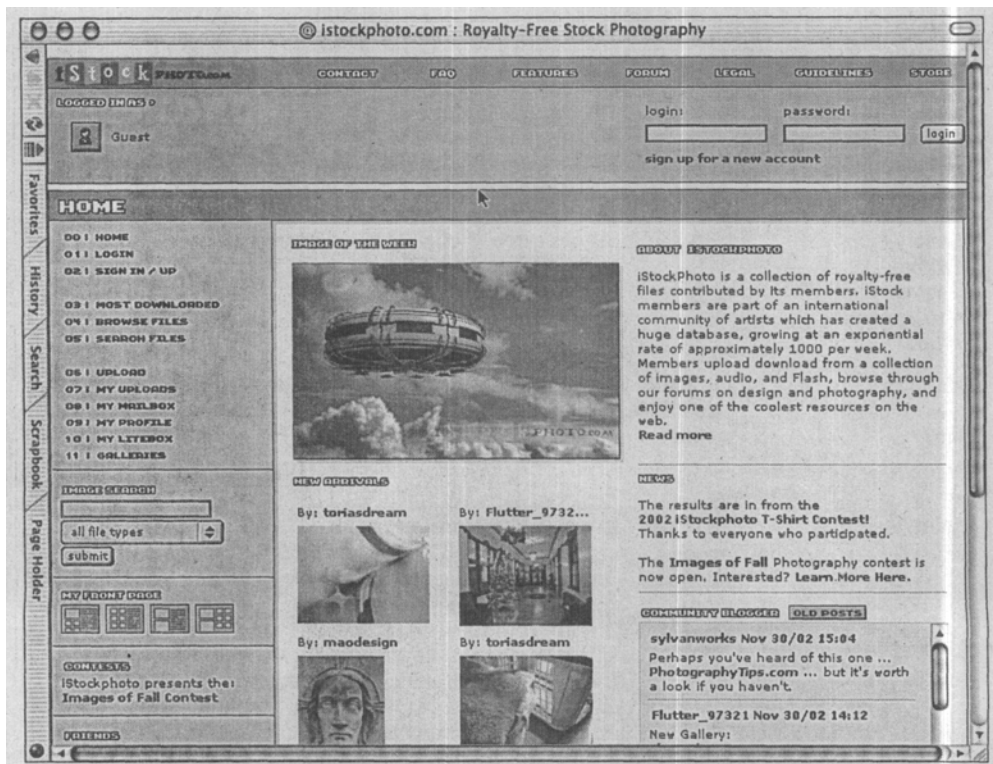


Рис. 5.14. Istockphoto (www.istockphoto.com) - мой любимый источник изображений без оплаты роялти. Если вы сами начнете заниматься фотографией, можете продать пару снимков. Удачная фотография может принести фотографу пару сотен долларов

Если дома вам не удалось найти необходимые фотографии, попробуйте поискать их в недорогих и бесплатных библиотеках изображений (табл. 5.3). На большинстве клипарт-ресурсов, которые я упоминала ранее, также продаются рисунки и фотографии (рис. 5.14 и 5.15).



Рис. 5.15. Getty Images (www.gettyimages.com) продают изображения как без оплаты роялти, так и с управляемыми правами; если у вас мало средств, ограничьте свой поиск изображениями без оплаты роялти

Г- Кок получить золото из песка

Малобюджетные дизайнеры, мужайтесь! Если цена - это ограничивающий фактор, а у вас есть доступ к цифровой камере, вы можете самостоятельно сделать несколько подходящих изображений с помощью графических редакторов Photoshop или Fireworks.

Для того чтобы ваши изображения выглядели профессионально, используйте свои фотографии творчески и последовательно. Если вы столкнулись с различиями в освещении, а цветовой баланс различается от фотографии к фотографии, диалоговые окна Photoshop **Variations** (Вариации) или **Hue/Saturation** (Тон/Насыщенность), которые находятся в разделе **Adjustment** (Корректировки) в меню **Image** (Изображение), помогут сделать эти изображения одинаковыми по насыщенности.

Или вы можете использовать в Photoshop корректировку **Desaturate** (Обесцветить), выполнив последовательность команд **Image => Adjustments => Desaturate**, чтобы убрать цвет из ваших изображений, а затем применив команду **Variations** для того, чтобы окрасить все фотографии в один или несколько цветов, которые соответствуют палитре вашего сайта.

Помимо последовательности в освещении и цвете, будьте последовательны и в том, как вы кадрируете ваши фотографии. Если вы сильно кадрируете три фотографии и получаете изображения крупным планом, а из четвертой фотографии делаете общий план и располагаете все четыре рядом, ваш дизайн может утратить привлекательность и целостность. Если кадрировать все фотографии одинаковым образом, содержание изображений достигнет такого профессионального уровня, что никто не догадается, что на его создание было потрачено немного средств.

Недорогая цифровая камера позволяет делать вполне хорошие фотографии, разрешение которых достаточное для размещения в Сети. Возможно, вам захочется заняться фотографией для того, чтобы создать свою собственную коллекцию. Брюс Ливингстон (Bruce Livingstone), основатель студии Istockphoto, недавно опубликовал в Digital Web Magazine статью «Дизайнерское руководство по созданию своей собственной коллекции фотографий (для нефотографов)» (A Designer's Guide to Making Your Own Stock Photography, см. [www.digital-Web.com/features/feature_2002"03.shtml](http://www.digital-Web.com/features/feature_2002)). Он пишет о том, как, используя фотоаппарат стоимостью \$20, можно создавать свои собственные изображения.

Как из плохой графики получить приемлемый вариант

Маленький бюджет подразумевает, что вам приходится работать с плохими рисунками. В данном случае аккуратно относитесь к имеющимся у вас материалам.

Недавно я общалась с дизайнером и автором нескольких книг Сэнди Коэн (Sande Cohen) - рис. 5.16. Мы говорили о том, что рисунки и фотографии могут стоить дорого, Я спросила, есть ли у нее какие-нибудь хитрости, чтобы сэкономить в этой области. Что делать, если вы не хотите, чтобы ваш сайт выглядел, как уцененный товар, но и платить сотни долларов за фотографии тоже не можете?

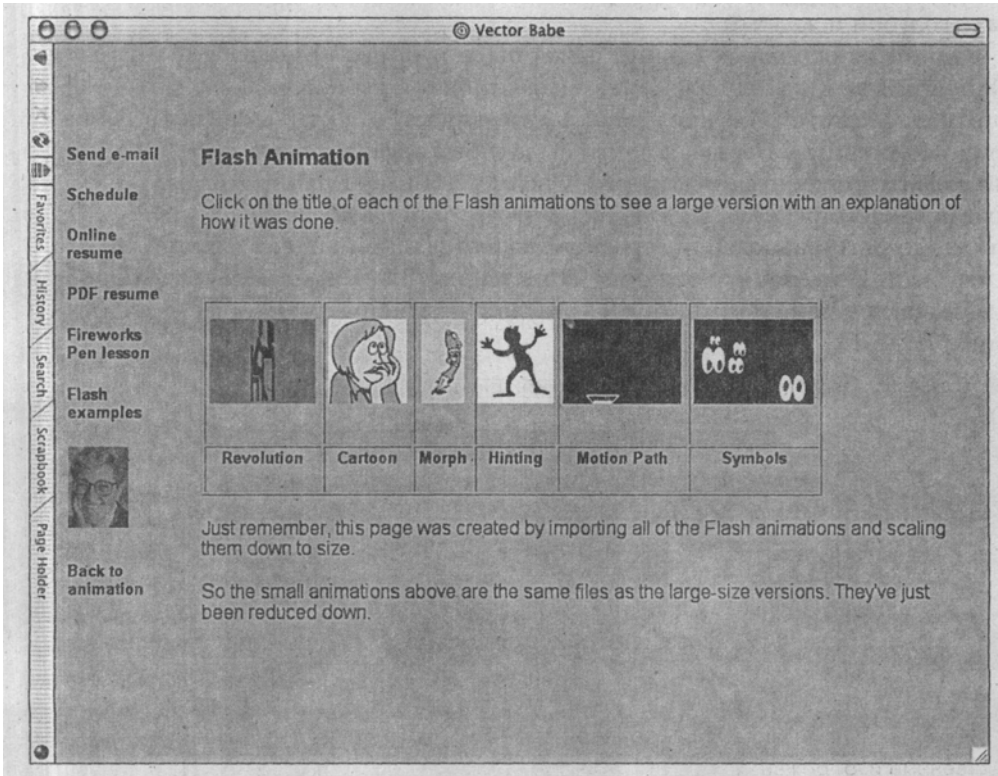


Рис. 5.16. Сэнди также является автором ежемесячной колонки советов и хитростей под названием Digital Dish (www.creativepro.com). Она выступает лекциями по всей Америке, а также поддерживает свой персональный сайт Vector Babe (www.vectorbabe.com). Мне кажется, что практические советы Сэнди особенно подходят тем, кто ограничен в финансах

Ответ Сэнди был следующий: «Это не тонкая художественная работа, но вот что тут надо делать. Существует масса способов преобразить посредственные клипарты и рисунки в Photoshop и других графических редакторах. Возможно,

я самый экономный человек на планете. Помню, много лет назад я купила CD за \$20, на котором была куча картинок. Сами по себе они выглядели дешево. Однако, если пропустить их через несколько фильтров, они выглядят отменно».

При работе над малобюджетным образовательным сайтом мы применили советы Сэнди. При оформлении текста и при цветовом оформлении мы руководствовались принципами, выделенными ранее; у нас получился приятный на вид сайт с четкими линиями и двумя дополнявшими друг друга цветами: зелено-вато-голубой и желтый.

Одной из основных секций сайта была рубрика «Биографии учителей». Основным визуальным элементом каждой такой страницы была фотография учителя. Можете себе представить, что качество и стиль снимков были не очень хорошими. Большая часть была снята на цифровые фотоаппараты с низким разрешением, некоторые учителя прислали нам фотографии 3x5 из их последнего резюме. Ни одна из фотографий не выглядела достойно. Если бы мы просто разместили эти фотографии в таком виде, в каком они были, у нас бы получилось что-то вроде сайта на рис. 5.9. Но мы не пожалели времени на двухступенчатый процесс, результаты которого вы можете видеть на рис. 5.17–5.19.



Рис. 5.17, 5.18 и 5.19. Попробуйте следующее, если вам приходится работать с неоднородными фотографиями низкого качества. Эти фотографии были подрезаны так, чтобы у них были одинаковые форма и размер. Затем они были единообразно оформлены: вокруг каждой



фотографии была добавлена рамка шириной 1 пиксель. Она помогает фотографиям слиться с остальными элементами сайта и вносит гармонию в коллекцию фотографий, которые до этого выглядели по-разному (www.nyp.org)

Что мы сделали:

1. Подрезали фотографии таким образом, чтобы они были одинаковых размеров, а фигуры учителей были изображены в более или менее одинаковых пропорциях.
2. Заключили фотографии в рамку.

Результаты оказались не самыми лучшими. Если бы у нас была возможность работать с собственными фотографиями, а не с теми, которые нам прислали люди, мы могли бы разместить учителей на одном и том же фоне, с одинаковым освещением, сфотографировать их примерно с одного расстояния, используя одинаковый фокус. Если бы у нас была возможность это сделать, мы бы подрезали фотографии одинаковым образом, а не так, чтобы добиться одного и того же размера. Тем *и* менее, учитывая то, с чем нам пришлось работать, у нас получился неплохой сайт. Даже если ваш бюджет не позволяет вплотную заняться изображениями, вы можете потратить немного времени для того, чтобы выжать максимум из имеющихся материалов.



Инструменты

ГЛАВА 6. Управление сайтом при небольшом бюджете

ГЛАВА 7. Сэкономьте время и деньги благодаря Web-стандартам

ГЛАВА 3. Экономичные домены и хостинг

Глава 6. Управление сайтом при небольшом бюджете

Здесь форма – это содержание, а содержание – это форма.
Сэмюэл Бэкетт о «Поминках по Финнегану».

/ М М в. Управление сайтом при небольшом бюджете

Здесь форма - это содержание, а содержание - это форма.
Сэмюэл Бэкетт о «Поминках по Финнегану».

Основные идеи главы

1. Определите задачи системы управления сайтом.

Найдите время, чтобы определить действительно необходимые задачи CMS; не платите за решение проблем, которых у вас нет.

2. Выбирайте CMS в зависимости от квалификации вашей команды.

Выбирая CMS, убедитесь, что ваша команда обладает достаточными техническими навыками для работы с ней. Если вы выбираете CMS, требующую навыков, которыми не обладают ваши сотрудники, деньги будут потрачены зря.

3. Подходите творчески к поиску инструментов.

Подходите творчески к поиску и отбору полезных инструментов. Ресурсы, которые уже имеются под рукой, например некоторые серверные вставки, могут быть достаточными на данном этапе.

4. Создайте CMS сами.

Помните также о том, что вы можете сами создать CMS. Можете начать с малого: создать CMS, которая будет заниматься управлением маленького сайта, а затем добавлять туда новые модули и возможности. Вы можете столкнуться с ситуацией, когда ни одна CMS не отвечает всем вашим потребностям и вам придется тратить время на переработку CMS, за которую вы заплатили. Если вам не удастся найти CMS, которая будет соответствовать всем вашим потребностям, возможно, вам захочется создать свою собственную систему.

5. Автоматизируйте CMS по максимуму.

CMS может сэкономить драгоценное время, которое уходит на администрирование. Попробуйте подсчитать, сколько умственной и административной работы уходит на управление сайтом, а потом подумайте о том, как CMS могла бы сэкономить ваши деньги.

6. Не забывайте про CMS с открытыми кодами.

CMS-инструментами с открытыми кодами можно обзавестись бесплатно (либо за небольшую плату) и таким образом сэкономить деньги. Выбирая инструменты с открытыми кодами, проверьте, что они активно используются другими разработчиками, что вашему разработчику (возможно, это будете непосредственно вы) они нравятся и что любое нужное приложение может расширяться по мере роста вашего сайта.

Как видно из предыдущих глав и как вы знаете из собственного опыта, нам, Web-профессионалам, работающим при небольшом бюджете, приходится нелегко. Наши бюджеты не позволяют нам делать серьезные ошибки и не дают нам свободного для творчества времени на то, чтобы экспериментировать с разными версиями дизайна. Роскошь вроде оригинальных фотографий или обширных коллекций шрифтов - не для нас. Мы не можем себе позволить тщательные исследования интерфейса, не всегда имеем возможность протестировать сайт на различных устройствах и платформах. После обращения с бесконечными просьбами к коллегам составить содержание сайта либо после самостоятельного его написания мы часто занимаемся редактированием текста, корректируем его и разбиваем текст на части.

Однако наши проблемы не заканчиваются, когда сайт готов. Они только начинаются. При написании дополнительного содержания для сайта нам приходится вручную вносить изменения. Web-дизайнеры, работающие при небольшом бюджете, привыкли обновлять содержание сайта либо вносить небольшие изменения в структуру путем редактирования вручную каждой страницы сайта. Эта работа не только утомительная, но и дорогая: при каждом изменении сайта на ее выполнение приходится тратить часы. Будучи Web-профессионалами, работающими при небольшом бюджете, часто мы не можем уложиться в отведенное время. Часто случается, что сайт обновляется с опозданием, либо не обновляется вообще. Должен существовать лучший способ по управлению и поддержке малобюджетного сайта.

Системы управления сайтом

Управление сайтом (также говорят *управление контентом*) предлагает решение этой проблемы. CMS (Content Management System), или система управления сайтом, предоставляет широкий спектр возможностей. Сам термин CMS может означать совокупность дорогих приложений, которые управляют всеми аспектами создания Web-содержания, публикации и поддержки. Однако CMS также означает сравнительно недорогие (или даже бесплатные) инструменты публикации, которые делают немного больше, чем простое написание кусков текста для статических HTML-страниц. Практически все сайты могут извлечь выгоду из автоматизированной в некоторой степени Web-публикации. Обновление содержания отнимает достаточно много времени, поэтому приходится делать

все вручную. Хорошая CMS позволит вам работать быстрее и логичнее. За такие преимущества обычно приходится дорого платить. Но, как показано в этой главе, существует масса способов, благодаря которым дизайнеры, работающие при небольшом бюджете, могут сэкономить деньги при автоматизации управления сайтом.

CMS - это не волшебная палочка, которая удивительным образом решает все проблемы. Если система, которую вы покупаете или устанавливаете, не соответствует вашим потребностям, она может создать такое же количество проблем, какое она может решить. Чтобы избежать проблем, связанных с использованием CMS, прежде чем установить систему, разберитесь, что вам нужно, а что - нет. Несколько лет назад я занималась управлением сайтом Нью-Йоркской публичной библиотеки. Это был большой сайт, созданный на основе маленьких сайтов-спутников. Этот сайт есть и сейчас. Каждый из маленьких сайтов был посвящен определенной теме или услуге. У каждого спутника был собственный управляющий или команда, ответственная за новизну и точность содержания этого маленького сайта. Такая схема может показаться вам хорошо знакомой. Многие большие публичные сайты организованы таким образом.

Когда какой-то раздел требовал обновления, человек, ответственный за это, часами редактировал содержание. К сожалению, он обновлял текст предыдущих версий, а не работал с самой последней версией. Затем он отдавал его моей группе, которая вручную обрабатывала текст и загружала на сайт. Мы быстро обнаружили ошибку. Вместо того чтобы просто выкладывать материал, нам приходилось переписывать и переписывать. Приличный контроль над версиями (version control) - инструмент, который иногда является частью CMS, - мог бы помочь нам избежать этой проблемы.

Система контроля над версиями следит за файлами и их авторами. Система создана так, чтобы можно было избежать одновременной работы двух авторов над одним и тем же файлом, потраченных зря часов на редактирование ненужной версии документа и других проблем, которые возникают, когда большая группа людей работает над одним проектом.

Возможно, вы помните, в начале книги я описывала случай, как библиотека потеряла финансирование, в результате чего мы не смогли купить то, что, по нашему мнению, нам было необходимо. Крупная CMS была одним из инструментов, который мы почти купили, но вдруг это приобретение сорвалось. Эта CMS была мощной системой управления сайтом, стоимость которой была примерно \$300000 плюс расходы на обучение и другие профессиональные

услуги - короче говоря, это система, недоступная профессионалу, работающему при небольшом бюджете. Эта CMS должна была содержать усиленный контроль над версией, который помог бы избежать ошибок в редактировании, которые нам пришлось исправлять вручную.

Но разве это было бы лучшим решением в данном случае? Вспомните про цену в \$300000. Решение проблемы контроля над версиями заняло у нас 6 часов, которые мы разбили на три дня. Всего один сотрудник занимался тем, что сравнивал версии сайта, чтобы удостовериться, все ли содержимое сайта новейшее. Давайте подсчитаем: вместе с зарплатой работника и дополнительными расходами, изменения стоят организации \$50 в час. В результате получается \$300.

Конечно, ручное исправление ошибок, которых, в принципе, не должно быть, требует времени и денег. Тем не менее разве потеря в \$300 может сравниться с \$300000 потраченными на покупку CMS? Вряд ли. Такая система может быть превосходной покупкой для тех, кто может себе это позволить, но для нашей маленькой проблемы это *слишком крупное решение*.

Угроза бюджету

Если вы не определитесь с тем, какие именно задачи должна выполнять CMS, будет очень легко переплатить за систему управления сайтом. Не пожалев времени на определение потребностей и рассмотрение альтернативных вариантов, которые уже могут быть у вас под рукой, вы сможете хорошо сэкономить на CMS. ^В

Возможно, вы удивитесь, почему в качестве примера я привела такую дорогую CMS. Я делаю это потому, что эта короткая история очень хорошо демонстрирует три самые большие угрозы, которые CMS представляет небольшому бюджету:

- незнание того, что вам нужно;
- переплата за решение проблемы;
- незнание того, что некоторые CMS-подобные инструменты уже имеются в вашем распоряжении.

Основное правило экономии на CMS: прежде чем купить, точно определите, что для вас является проблемой, а что не является.

Если вы располагаете маленьким бюджетом, тщательное планирование поможет вам определиться с потребностями, и вы, таким образом, сможете найти верное решение и не платить за *слишком крупное решение*.

Не платите за то, что вам не нужно

Недостаточно знать, что вам нужно от системы управления сайтом, - необходимо также знать, *что вам не нужно*. Если вы этого не знаете, высока вероятность потерять деньги на CMS. У меня возникают подобные проблемы, когда я иду за покупками, особенно если дело касается компьютерных магазинов. Меня прельщают новейшие приспособления, решающие проблемы, о наличии которых я и не подозревала. Например, я и не знала, что мне нужен Apple iPod, пока я не увидела его в магазине. Осведомленный потребитель с ограниченным бюджетом точно знает, что ему действительно нужно, фокусирует внимание на самом важном (например, на мониторе, если он сломался) и не растрачивается на привлекательные, но совершенно не нужные ему вещи. Таким образом, прежде чем пойти за покупками, давайте посмотрим, какие у вас могут быть потребности.

Нужно ли вам контролировать план страницы при помощи шаблонов?

Одна из основных особенностей CMS - возможность управления сайтом при помощи шаблонов (templates). При использовании шаблонов вы можете заблокировать некоторые области Web-страницы, такие как навигация и область размещения логотипа, предоставляя возможность их редактирования только некоторым сотрудникам, так чтобы создатели содержания, не имеющие специальной технической подготовки, не смогли случайно все испортить. Менеджеры сайта и дизайнеры, само собой, смогут редактировать заблокированные области, если им понадобится обновить навигацию или изменить логотип. Специалисты по содержанию смогут редактировать исключительно те разделы страниц, которые соответствуют их работе. Простые в использовании шаблоны позволяют вам разделять доступ к различным кускам ваших Web-страниц (рис. 6.1 и 6.2).

Существует масса других способов создания шаблонов страниц, и для этого не обязательно использование CMS. Хорошей малобюджетной альтернативой служат HTML-редакторы, обладающие инструментами работы с шаблонами, такими как Macromedia Dreamweaver и Macromedia Contribute. Вполне возможно создать сайт на основе шаблонов, используя подобное программное обеспечение,



Рис. 6.1. Главная страница Коллекции картин (Picture Collection), снабженной поиском по базе данных с изображениями и принадлежащей одному из отделов Нью-Йоркской публичной библиотеки, в том виде, в каком она предстает перед посетителями (<http://digital.nypl.org>)

и не тратить ни копейки на настоящую CMS. Важно отметить, что Macromedia Dreamweaver и Contribute - недорогие решения основных проблем по управлению содержанием. Настоящая экономия будет в том случае, если вы используете эти программы и в других целях. Многие инструменты CMS предлагают управление содержанием и установками сайта прямо в Сети; Contribute также выполняет эту задачу. Это должно иметь значение для людей с ограниченным бюджетом, поскольку единственное программное обеспечение, которое необходимо создателю содержания, - это Web-браузер. Позднее мы рассмотрим еще несколько примеров управления содержанием в Сети.

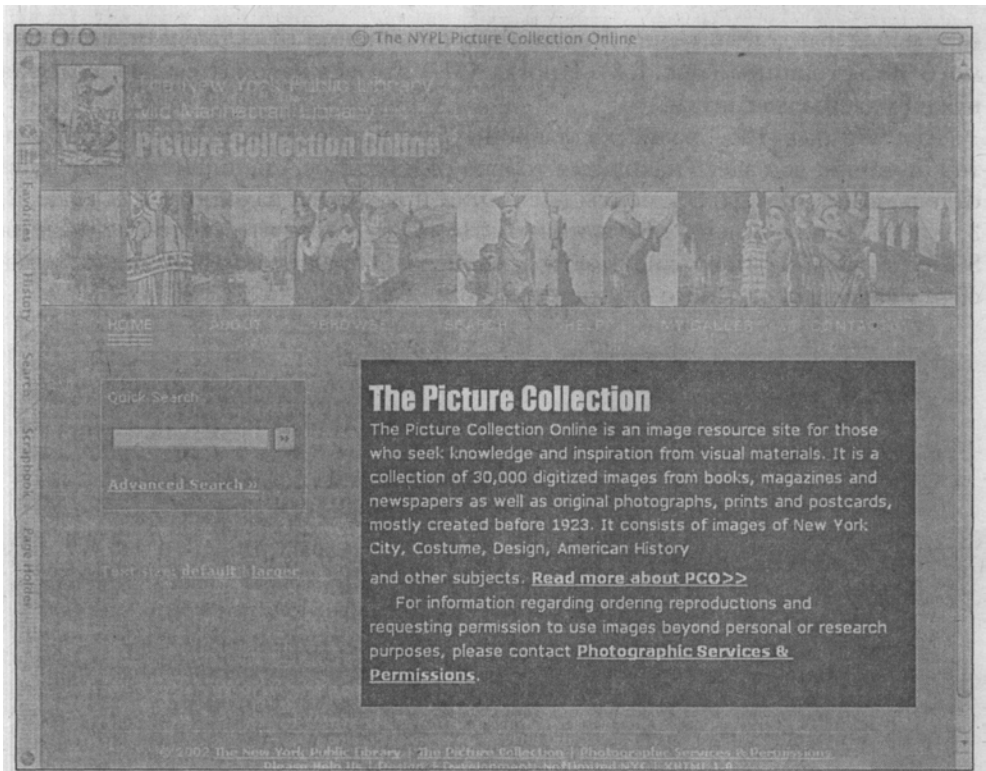


Рис. 6.2. Шаблон этого сайта позволяет людям, ответственным за содержание, редактировать только темно-серую область. Бледные области (светло-серые) - части шаблона, которые может редактировать только владелец сайта. При обновлениях содержания сотрудник может работать только в той области шаблона, к которой у него есть доступ. Поменять навигацию или логотип может только владелец сайта

Другое не-CMS решение по шаблонам - использование бесплатной технологии под названием *server-side include*, или *SSI*-серверные вставки. Название может вам показаться сложным, но суть этой технологии на самом деле проста. Многие Web-разработчики оставляют страницы, материалы навигации и логотип в отдельном Web-документе, который находится в специальном файле на сервере, но появляется как часть Web-страницы, когда сайт отображается в браузере. Вы с таким не знакомы? Не беспокойтесь, сейчас мы рассмотрим один пример. Поверьте мне, SSI позволяет разработчикам управлять набором

отдельных файлов логотипа и навигации независимо от остального содержимого Web-страницы (рис. 6.3). Иногда SSI - это вся автоматизация, которая нужна маленьким сайтам.

Листинг на с. 135 - разметка главной страницы Коллекции картин. Обратите внимание, что здесь находится только содержимое, специфичное для этой отдельной страницы. Элементы навигации и логотипа находятся в строках 2, 3, 4 и 19 (они выделены полужирным). Помните о том, что способ написания SSI будет различным в зависимости от сервера. О правильном синтаксисе спросите у вашего системного администратора.

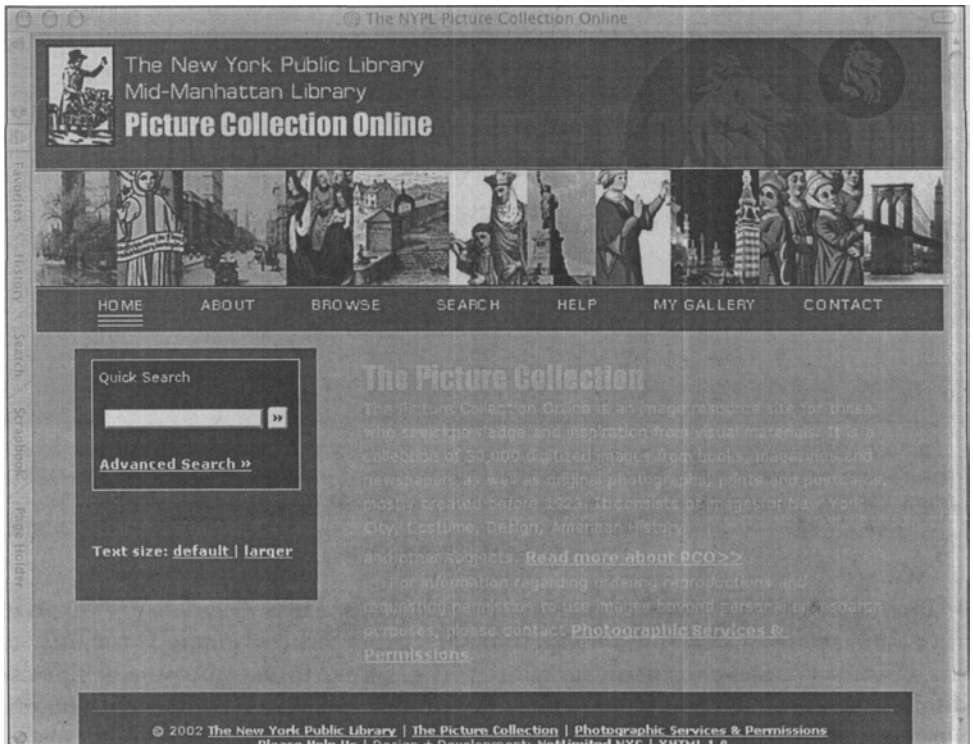


Рис. 6.3. Шаблонами этого сайта управляют не с помощью CMS, а при помощи серверных вставок. SSI - это файлы, которые на сервере расположены независимо от страниц содержания. Три темно-серые области: навигация сверху, слева и внизу - это разные файлы, которые появляются на главной HTML-странице благодаря использованию серверных вставок

рОпределение

Серверные вставки. Область определения термина *серверные вставки (SSI)* достаточно широка. В самом общем значении SSI - это HTML-комментарий, который дает Web-серверу распоряжение о динамическом заполнении Web-страницы содержанием, внешним по отношению к ней. Внешнее содержание находится на отдельной HTML-странице, подобно тому как CSS- или JavaScript-документы расположены отдельно от страницы или страниц, на которых они используются. SSI может заполнить Web-страницу строкой текста или порцией данных, даже запустить программу. Например, SSI может определить день, дату и время, а затем вставить эту информацию в HTML-страницу. Для нас SSI - это HTML-комментарий, который дает серверу распоряжение перетащить содержание из другого документа.

Один из признаков успешного малобюджетного сайта состоит в том, что при создании такого сайта максимально используются ресурсы, имеющиеся под рукой. Если ваш сайт небольшой и его управлением занимается лишь несколько Web-профессионалов, вполне возможно, что SSI - это все, что вам нужно, чтобы управлять различными кусками содержания сайта. Чтобы в этом лучше убедиться, еще раз взгляните на рис. 6.3.

```

1. <body>
2. <!--#include virtual="/mainnav.txt" -->
3. <!--#include virtual="/subnav.txt" -->
4. <!--#include virtual="/search.txt" -->
5. <p1>Коллекция Картин</Ы>
6. <p> Коллекция картин Online - это сайт, содержащий коллекцию
7. изображений для тех, кто ищет знания и вдохновение в
8. виртуальных материалах. Коллекция содержит более
   30,000 цифровых
9. изображений из книг, журналов и газет, а также
10. оригинальные фотографии, гравюры и открытки, главным образом
    созданные
11. до 1923 года. Основные темы коллекции: Нью-Йорк,
12'. Костюмы, Дизайн, история Америки и другие. <a
13. href="about.cgi">Узнайте больше о Коллекции Картин</a>.</p>
14. <p>Чтобы получить информацию, касающуюся заказа репродукций и
15. получения разрешения на использование изображений в личных или
16. исследовательских целях, пожалуйста, свяжитесь с <a
17. href="http://www.nypl.org/admin/pro/copies/cs.html">
18. Центром фотографических услуг</a>.</p>
19. <!--#include virtual="/footer.txt" -->
20. </body>

```

Возможности для написания и редактирования содержания

Великолепная возможность некоторых CMS - простой в использовании интерфейс для написания и редактирования содержания. Некоторые CMS-интерфейсы работают, как визуальные редакторы; другие, например Zope (рис. 6.5), требуют знания HTML.

Возможно, вам понадобится интерфейс, который будет прост в обращении и рассчитан на авторов, не имеющих специальной технической подготовки. Не пренебрегайте такой возможностью только потому, что у вас небольшой бюджет. Вы можете найти доступный инструмент, который будет соответствовать этому требованию. Обратите внимание на список возможных средств:

- утилиты по проверке орфографии;
- WYSIWYG-интерфейс, который позволяет людям, не знающим язык разметки, вводить содержание сайта и не требует при этом использования или понимания HTML (рис. 6.4 и 6.5);
- возможности отмены и повтора (команды Undo и Redo);
- поддержка копирования и вставки;
- возможность ручного или автоматического создания метаданных (см. рубрику «Определение»);
- возможность форматирования страницы в рамках шаблона;
- контроль над расположением изображений;
- необходимые области сайта (такие, как дополнительный текст для изображений).

Duct Tape Clothing, fashion, wallets, hats, purses, and more!...
Duct Tape hats shoes backpacks dresses and more!... Cheap! **Duct Tape** Book Only 55 Get them while they last! Random **Duct Tape** Use of the Moment. Submit yours today!...
 Description: **Duct tape** wallets, purses, adjustable belts, hats and straps. Keith Drone's creations are unique.
 Category: [Shopping](#) > [Clothing](#) > [Accessories](#)
www.ducttapedfashion.com/ - 15k - Mar 28, 2003 - Cached - [Siml](#)

Рис. 6.4. Когда сайт одежды Duct Tape отображается на сайте Google (как здесь) или в других поисковых машинах, в результатах поиска отображаются метаданные. Web-профессионалы, работающие при небольшом бюджете, используют метаданные для того, чтобы улучшить отображение в подобных поисковых машинах, таким образом не позволяя поисковой машине показывать случайный текст стела сайта

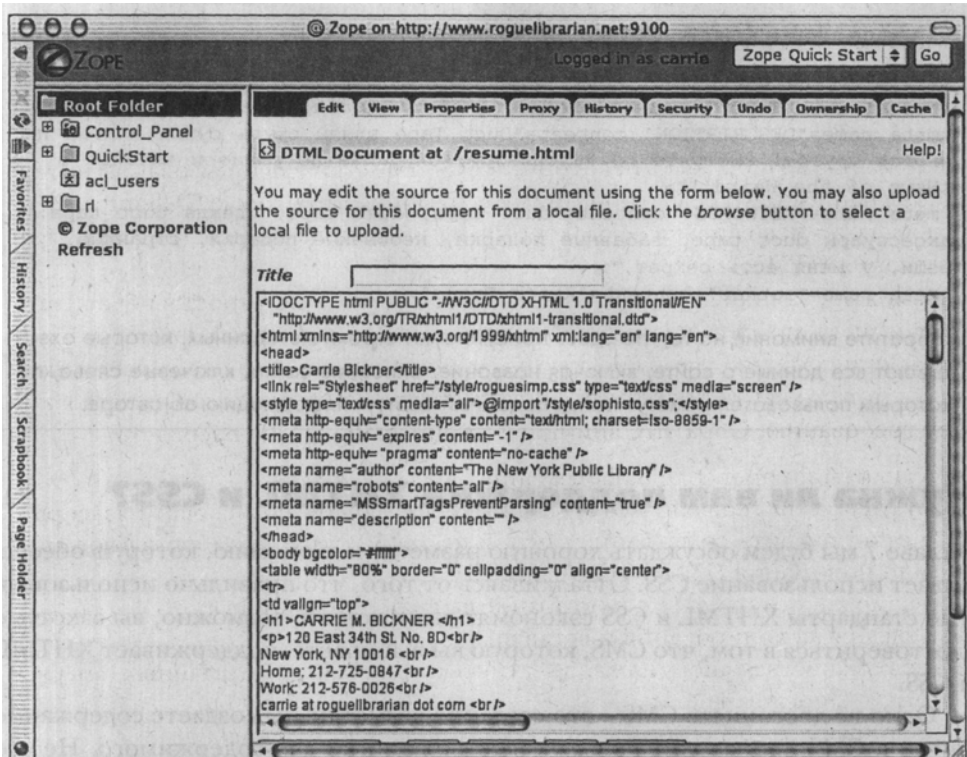


Рис. 6.5. Если графический интерфейс - одно из ваших требований к CMS, доступная конфигурация Zope - открытого CMS-инструмента (www.zope.org) - может не соответствовать вашим потребностям. Для того чтобы использовать Zope в его первоначальной форме, создателю содержания потребуется умение писать разметку. Однако, как вы увидите позже, Zope можно модифицировать так, чтобы у него был простой в использовании графический интерфейс. Также обратите внимание на то, что Zope позволяет загружать HTML-страницы, созданные в любом удобном для пользователя HTML-редакторе

-Определение-

Метаданные. *Метаданные (Metadata)* - это данные о данных. Метаданные могут быть использованы с различными настройками и в различных целях. Для нас метаданные полезны тем, что они описывают содержание и назначение вашей Web-страницы. Многие поисковые машины используют созданные вами метаданные в качестве описания вашего сайта, таким образом, эта информация является важной. Ниже приводятся метаданные для сайта моды Duct Tape (www.ducttapefashion.com):

Определение (окончание)

```
<title>Ofle>Kfla Duct Tape, мода, кошельки, шляпы, сумочки и многое
другое! Duct Tape - модная одежда, аксессуары и многое другое!</title>
<meta name="DESCRIPTION" content="Duct Tape шляпы обувь сумки платья и
многое другое! Смотрите по телевизору, слушайте по радио и читайте в
'News of the Weird!'">
<meta name="KEYWORDS" content="duct tape, duck tape, одежда duct tape,
аксессуары duct tape, забавные подарки, необычные подарки, странные
вещи, у меня есть секрет,">
<meta name="author" content="Keith Drone">
```

Обратите внимание на то, что здесь присутствуют строки метаданных, которые охватывают все данные о сайте, включая название, описание сайта, ключевые слова, по которым пользователи смогут найти подобный сайт, и информацию об авторе.

Нужна ли вам поддержка XHTML и CSS?

В главе 7 мы будем обсуждать хорошую разметку и экономию, которую обеспечивает использование CSS. Отталкиваясь от того, что правильно использованные стандарты XHTML и CSS сэкономят ваши деньги, возможно, вы захотите удостовериться в том, что CMS, которую вы выбираете, поддерживает XHTML и CSS.

Одно из достоинств CMS - это создание разметки; вы создаете содержание сайта, а CMS производит большую часть разметки для содержимого. Не все CMS одинаково хорошо справляются с разметкой. Некоторые системы делают это более искусно, создавая превосходный, простой, соответствующий стандартам XHTML. Другие же системы делают такую разметку, которую даже самые плохие HTML-редакторы сделали бы не хуже.

Как будет показано в главе 7, простая, безупречно созданная разметка имеет огромное значение; сайт получается более маленьким и менее дорогим с точки зрения трафика и дискового пространства. Также сайт становится доступным более широкой аудитории пользователей, включая тех, кто использует нетрадиционные Интернет-устройства, как, например, карманные компьютеры.

Составляя список требований, примите во внимание экономичность хорошей разметки.

Нужны ли вам проверка разных версий содержания и управление рабочим процессом?

Пример управления версиями, который я привела в начале главы, - это одна из проблем, которую можно решить при помощи CMS. У нас не было механизма, который мог бы отслеживать версии страницы или компонентов, у нас также не было возможности вернуться к более ранней версии той или иной страницы или компонента либо более ранней версии целого сайта. CMS с функциями управления версиями и последовательностью выполняемых действий будет отслеживать версии и контролировать процесс редактирования.

Если проблемы, связанные с версиями сайта, немногочисленны из-за небольшого штата сотрудников и слаженной работы, контроль над версиями сайта, возможно, вам не нужен. Тем не менее отсутствие инструментов автоматизации контроля над версиями и управления последовательностью выполняемых действий может повлечь за собой непредвиденные расходы. Вашим разработчикам придется каждый вечер скачивать целые директории, прежде чем пойти домой? Вам приходилось слышать из-за стены что-нибудь вроде: «Грейс, не трогай директорию about примерно в течение часа. Я сейчас в ней работаю»?

Подобные взаимодействия сотрудников выливаются в расходы, которых вы изначально не замечаете. Предположим, что Грейс - разработчик, и лучше всего она работает, когда слушает в наушниках тайваньскую поп-музыку. Ее коллеги заходят к ней, хлопают ее по спине и ждут, пока она снимет наушники, чтобы попросить ее какое-то время не работать в той или иной части сайта. За год время, потерянное на такие взаимодействия, превращается в напрасно потраченные деньги. Простая CMS функция **Check-in/Check-out** (Контроль входа/выхода) позволила бы Грейс более спокойно работать, слушая музыку, и сэкономила бы *деньги для* команды. (Благодаря функции **Check-in/Check-out**, когда один человек работает с каким-то файлом или директорией, доступ к этому файлу или директории закрыт для других до тех пор, пока работа над файлом не закончится и его не поместят обратно в систему.)

Если над вашим сайтом работает больше чем 2-3 человека, вы можете избежать значительных расходов, используя инструмент **Check-in/Check-out** или другие возможности управления версиями. Пока у вас маленькая команда, то, возможно, такие инструменты вам совсем не нужны.

Нужна ли вам возможность работы с базой данных?

Большинство CMS используют базу данных; в таких случаях база данных в том или ином варианте хранит содержимое сайта. Если ваша CMS требует использования базы данных, уточните, совместима ли ваша база данных с системой управления контентом. Помните, что базу данных и CMS* возможно, придется покупать отдельно. Многие (бесплатные и бесплатно лицензированные) CMS-инструменты с открытым кодом можно использовать вместе с языком с открытым кодом. Мы обсудим эти возможности в разделе «CMS с открытым кодом» в этой главе. Часто ваш отдел информационных технологий или главный сервер имеют одну или две уже установленные базы данных. Если вашей CMS требуется база данных, удостоверьтесь, что она будет работать с имеющейся у вас базой данных.

Нужны ли вам области для подготовки и для производства?

Многие Web-профессионалы работают на нескольких серверах и в нескольких рабочих областях для того, чтобы управлять различными задачами по производству и развитию. Простой пример такого рода администрирования представлен в табл. 6.1.

Таблица 6.1. Задачи рабочих областей для простого содержания

	Содержание
Область разработки или подготовки	Создание и проверка содержания
Область производства	Динамическое содержание

В данном случае есть две рабочие области: одна - для создания и редактирования текста, а другая - для реально работающего (динамического) содержания. Когда текст создан и отредактирован, он перемещается в область производства. В некоторых системах управления сайтом это перемещение происходит автоматически. Например, CMS позволяет создателю текста переместить его в область разработки одним щелчком мыши. Затем редактор работает в той же самой области разработки, проверяя новое содержание. Когда работа по редактированию завершена, редактор может переместить содержание в область производства, опять-таки только нажав одну кнопку. По

завершении этой работы содержание становится динамическим и появляется в Сети.

Другая, более сложная система использования областей подготовки и производства встречается при разработке программного обеспечения вместе с содержанием сайта. Предположим, у вас есть техническая команда, которая разрабатывает программное обеспечение для чата. В то же время создатели содержания работают над текстом. Возможно, вы захотите, чтобы управление разработкой программного обеспечения осуществлялось по той же схеме, по которой вы управляли созданием содержания в предыдущем примере, используя различные области разработки для создания, тестирования и устранения ошибок. Когда программное обеспечение для чата готово, вы можете переместить его в область производства.

Если создание (содержания) происходит в то же самое время, что и разработка программного обеспечения, возможно, вам захочется, чтобы эти процессы проходили отдельно друг от друга. В таком случае области подготовки и производства будут выглядеть, как в табл. 6.2.

Таблица 6.2. Задачи рабочих областей при отдельной работе над программной частью

	Программное обеспечение	Содержание или данные
Область разработки или подготовки	Разрабатываемое и тестируемое программное обеспечение	Создание и проверка содержания
Область производства	Работающее программное обеспечение	Находящееся в общем доступе содержание

Области подготовки и производства могут быть разделены по-другому. В некоторых проектах одна область подготовки отводится под данные, другая - под Web-страницы. Некоторые CMS-инструменты позволяют работать одновременно в областях подготовки и производства, автоматизируя процесс передачи содержания из области разработки в область производства. Помните о том, что CMS-инструменты не покрывают расходы на использование различных областей - они просто позволяют работать в этих областях одновременно.

Даже если сейчас вы не имеете дело с такой организацией работы, возможно, когда-нибудь вам придется. Если вы предвидите такую возможность в будущем, имейте в виду, что процесс перемещения файлов из одной области в другую - не всегда простой и дешевый. Подумайте, нужен ли вам CMS-инструмент, который будет заниматься перемещением файлов из области разработки в область производства.

Нужен ли вам механизм RSS?

Rich Site Summary (Расширенное изложение сайта), или RSS, - это XML-формат, который позволяет давать доступ к Web-содержанию других сайтов. Не беспокойтесь: вам не придется создавать XML-разметку. RSS - это инструмент, который позволяет использовать какую-то определенную часть вашего Web-содержания одновременно в нескольких местах (рис. 6.6). Механизм RSS позволяет пользователям, включая и других Web-разработчиков, выбирать, сохранять и распространять любые понравившиеся куски содержания.

г- Как получить золото из песка

Рассылки новостей. Многие сайты общаются со своими сообществами через рассылки новостей (рис. 6.7). Это может быть эффективным маркетинговым инструментом; не стоит пренебрегать возможностью влиять на посещаемость сайта с помощью еженедельных новостных писем. Когда я работала над сайтом, написанным с помощью JavaScript на about.com, я отправляла рассылки новостей раз в две недели. В те дни, когда эти письма отправлялись, количество посещений сайта увеличивалось примерно в два раза. Ник Асборн в NetWodrs рассказывает подобную историю и дает совет авторам, работающим при небольшом бюджете, которые хотят получить максимум пользы от рассылки новостей:

«Меня всегда поражал рост активности посетителей, вызванный каждым новостным письмом, которое я отправлял. Я не использую новостные письма, чтобы что-то продавать, - по крайней мере, не делаю это открыто. Отправка каждого письма вызывает наплыв новых посетителей на мой сайт. Если содержание вашего сайта весьма полезно, то рассылка новостных писем будет особенно эффективна.

Среди всех строк в новостных рассылках есть одна, которая может оказаться особенно важной: Расскажите своему другу или коллеге об этой рассылке».

У некоторых пользователей есть свой собственный RSS-список просмотров, и когда они видят, что сайты, добавленные в этот список, были обновлены, они посещают эти сайты. Многие используют бесплатные (или недорогие) программы типа News Reader, которые показывают новые или измененные RSS-заголовки и текст с сайтов, на которые они подписались. (Подписка бесплатная; вы сообщаете News Reader, RSS каких сайтов вы хотите отслеживать; это делается путем добавления того или иного адреса в диалоговое окно **News Reader**. Например, www.scripting.com/rss.xml - это один из RSS-механизмов Web-сайта Scripting News.) News Reader просматривает сайты, поддерживающие RSS, а затем загружает те разделы, на которые вы подписались.

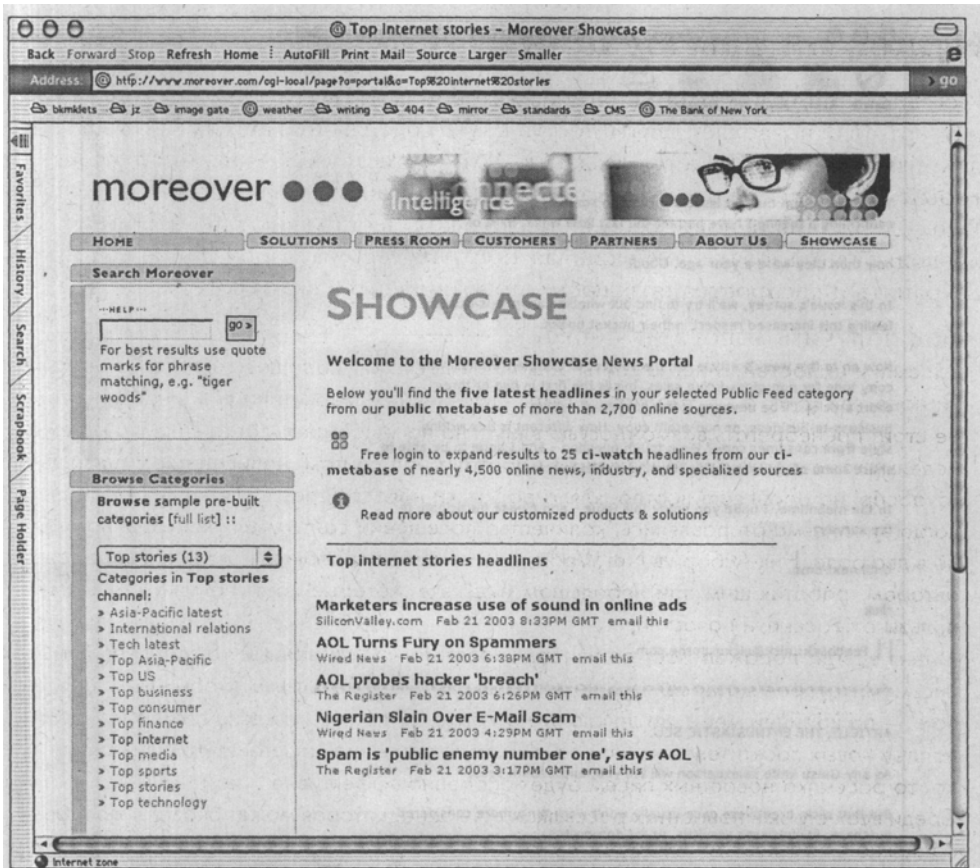


Рис. 6.6. Moreover (www.moreover.com) - это информационная брокерская фирма, которая использует механизм RSS для отображения новостей. SiliconValley.com, Wired News и The Register - все эти сайты поддерживают технологию RSS; на этих сайтах Moreover отбирает новости. Благодаря такой технологии можно значительно увеличить количество посетителей. Если вы занимаетесь поставкой содержания (например, новостей), вам следует обратить внимание на RSS. Если вам нужна RSS-стратегия, выбирайте какие-нибудь CMS-инструменты, которые могут использовать несколько механизмов RSS

Полезность механизма RSS состоит в том, что он позволяет недорого, но при этом более широко распространять ваше содержание и тем самым увеличить посещаемость сайта. Если вы предоставляете вашим посетителям больше

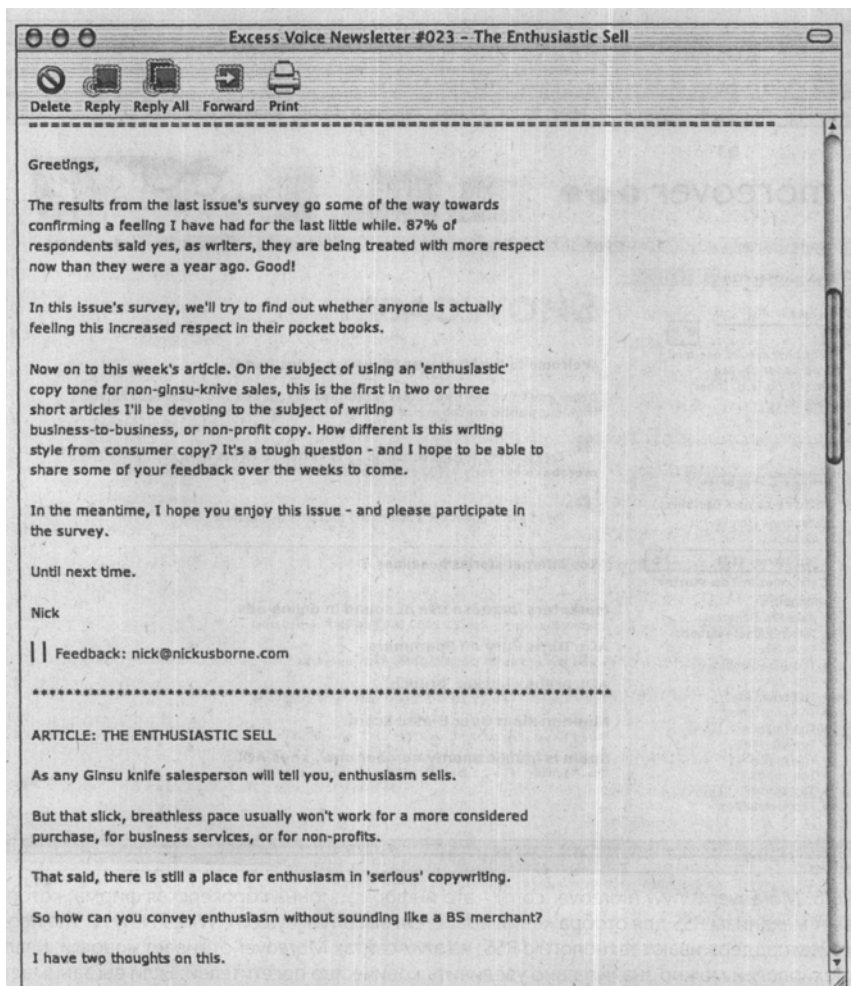


Рис. 6.7. Рассылка новостей может оказать значительное влияние на увеличение посещаемости сайта. Об обновлениях на сайте можно сообщать кратко в форме новостного письма. Если вам это нравится, при покупке CMS обратите внимание, содержит ли она инструмент по автоматическому (или упрощенному) производству новостных писем

возможностей узнать о том, что ваш сайт был обновлен, вы даете им больше шансов и стимулов для того, чтобы посетить ваш сайт снова.

Лучшие CMS для вашего бюджета

Теперь, когда вы определились с тем, что CMS может сделать для вас, самое время приступить к ее покупке. Прежде чем определиться с наиболее подходящей вашему бюджету CMS, вам необходимо сделать три вещи:

1. Определите функциональные требования. Здесь отталкивайтесь от списка, приведенного ранее.
2. Определите бюджет. Помните, что эта сумма должна покрыть не только расходы на покупку CMS, но и связанные с CMS расходы. Эти расходы могут включать в себя обучение, профессиональное обслуживание, а также расходы на импортирование уже существующего содержания. Дополнительные расходы могут включать в себя хостинг, дополнения, обновления, поддержку. Противопоставьте эти расходы сумме, которую вы можете сэкономить благодаря CMS на коротком отрезке работы и в дальнейшем. См. также рубрику «Угроза бюджету» ниже.
3. Удостоверьтесь в том, что ваши сотрудники обладают необходимыми техническими навыками. Будьте осторожны: учитывайте только тех сотрудников, которые намерены остаться на долгий срок. Если у вас работает только один или два технических специалиста и вы не уверены в том, что они останутся в организации надолго, следует это учесть, думая о CMS. (Готовьте себя к худшему, и вы сможете этого избежать.) Впрочем, возможно, вы недооцениваете уровень навыков ваших сотрудников и вам не нужно дополнительно нанимать консультанта.

Если у вас работает один технический специалист, чьи навыки в некоторых областях подвергаются сомнению, и он недавно сказал вам, что ищет другую работу, вы, возможно, передумаете отдавать предпочтение CMS с открытым кодом, которая требует технических знаний для установки и поддержки.

г- Угроза бюджету

На самом деле стоимость CMS не ограничивается ценой, указанной в прейскуранте. Иногда новая CMS требует дополнительных профессиональных услуг по установке и загрузке старого содержания в новую систему. Также могут понадобиться новые сотрудники, владеющие необходимыми техническими навыками для запуска новой системы, либо понадобятся вспомогательные приобретения, как, например, дополнительные серверы, базы данных или лицензии на использование программного обеспечения. Прежде чем выбрать CMS, проверьте, посчитали ли вы все дополнительные расходы.

Я организованная девушка, и, когда я планирую что-то, я люблю составлять схемы. Это помогает мне держать в одном месте все факторы, влияющие на принятие решения. При планировании расходов на CMS я бы обязательно составила список всех необходимых мне возможностей, свои технические требования и, конечно, определила бы свой бюджет. ^{^В}

Если вы будете делать то же самое, вам будет проще принимать решения. Если вы не распорядитесь бюджетом и вам необходимо получить одобрение сверху, составление четкого и подробного плана поможет вам и в данном случае. Я бы советовала учитывать три фактора, влияющих на принятие решения (табл. 6.3).

Таблица 6.3. План требований и возможных решений

Необходимость наличия тех или иных возможностей	Ответ	Примечания
Нужно ли вам управлять разметкой страницы при помощи шаблонов?	Да	Нам нужно управлять разметкой страниц сайта при помощи шаблонов. Уже давно необходимо изменить верхнюю панель навигации: чтобы поменять одну ссылку или одно изображение, нам приходится редактировать 20 различных файлов. На эту работу уйдет слишком много времени, и в процессе ее выполнения может возникнуть множество ошибок
Нужны ли вам возможности по написанию разметки и визуальному редактированию?	Отчасти	На данный момент это не обязательно. Однако, когда большое количество людей, не являющихся техническими специалистами, начнет добавлять содержание на сайт, безусловно, для этого им понадобится простой интерфейс. Если такая возможность появится у нас сейчас, нам будет проще привлечь к проекту специалистов нетехнического профиля
Нужна ли вам поддержка XHTML и CSS?	Да	Мы потратили большое количество усилий и средств на разработку хороших (X)HTML и CSS. Теперь нам надо, чтобы эта работа не оказалась напрасной
Нужна ли вам возможность работы с базой данных?	Да	Нам надо, чтобы наша CMS работала с базой данных MySQL, которая недавно стала нашим стандартом
Нужны ли вам области для подготовки и для производства?	Возможно	Было бы неплохо, но в ближайшие 2–3 года нам это не понадобится. Это будет нужно только в том случае, если у нас что-то не будет получаться

Таблица 6.3. План требований и возможных решений (продолжение)

Необходимость наличия тех или иных возможностей	Ответ	Примечания
Нужен ли вам механизм RSS?	Да	Наши конкуренты использовали механизм RSS в течение последних восьми месяцев, и, согласно нашим исследованиям, именно благодаря этой возможности количество посетителей у них значительно увеличилось
Команда в 7–10 человек	Очень нужно	Мы работали с некоторыми хорошими приложениями по управлению процессом Web-производства. Например, инструмент Check-in/Check-out пакета Dreamweaver помог нам отслеживать, кто над чем работает. Сейчас наша команда увеличилась с трех до семи человек, нам нужны более серьезные инструменты для управления файлами
Управление версиями	Отчасти	Было бы неплохо, если бы мы смогли найти CMS, которая смогла бы управлять версиями, но, если мы найдем доступную CMS, которая будет вполне соответствовать всем остальным нашим требованиям, мы найдем и отдельное недорогое программное обеспечение для управления версиями
Функциональные требования		
Работает под управлением UNIX	Очень нужно	Это стандарт нашей компании, мы не можем себе позволить расходы на новую операционную систему. У нас работают только специалисты по UNIX, ни у кого нет навыков в управлении другими ОС
Работает с базой данных MySQL	Очень нужно	MySQL – это наш новый стандарт. Помните, что в ближайшие два года нам также необходимо поддерживать совместимость с уже существующей базой данных MS Access
Простая в установке и поддержке	Очень нужно	У нас великолепная IT-команда, но наши люди и без того перегружены. Необходимо, чтобы мы могли произвести установку самостоятельно, а за помощью IT-поддержки обращались только в крайних случаях
Другие факторы		
Общий бюджет	\$1000	Эта сумма должна покрыть все расходы, включая обучение, поддержку и установку

Таблица 6.3. План требований и возможных решений (окончание)

Необходимость наличия тех или иных возможностей	Ответ	Примечания
Навыки штатных специалистов	PHP/Perl	
	XML/XSLT	
	Графический дизайн	
	(X)HTML и CSS	
	Apache/UNIX	
	SQL	
	MySQL, Access	Мы можем работать и с другими базами данных, однако нам необходимо, чтобы наша CMS поддерживала MySQL – наш новый стандарт

Теперь, когда у вас есть список необходимых предметов, классифицированных по степени важности, можно отправляться покупать CMS. Давайте обсудим некоторые варианты CMS из тех, что существуют на рынке. В каждой категории я выделила несколько CMS, а затем более детально исследовала по одной CMS в каждой группе. Пожалуйста, не стоит думать, что мой основной пример – это лучший вариант; вам необходимо провести собственное исследование. Не ограничивайтесь изучением только той CMS-продукции, которая здесь упоминалась; включите в исследование другие CMS-системы. Я уверена, что к моменту выхода в свет этой книги на рынке появятся новые продукты.

CMS с открытым кодом

Рабочее определение: программное обеспечение с открытым кодом – обеспечение, распространяемое по открытой лицензии. В большинстве случаев это означает, что программное обеспечение (ПО) может быть скомпилировано, использовано, передано и изменено любым пользователем, но при условии что он будет действовать в рамках лицензии на это ПО. Принцип движения открытого программного обеспечения состоит в том, что оно может свободно распространяться и изменяться членами открытого сообщества, а значит, со

временем будет улучшаться, - в этом его большое преимущество перед обычным программным обеспечением.

Открытая операционная система Linux - один из самых известных (и успешных) примеров этого принципа. Другой пример - бесплатный, расширяемый серверный язык PHP, используемый на сотнях тысяч сайтов (включая Yahoo.com).

Не все продукты с открытым кодом бесплатные. Open Office - это программный продукт с открытым кодом, который соперничает с Microsoft Office и работает под Linux, UNIX, Windows и Mac OS. Эта программа стоит гораздо меньше, чем Microsoft Office, но не выполняет всех задач MS Office (да и не претендует на это).

Когда код части программного обеспечения открыт, его разработчики создают общедоступную лицензию, которая описывает условия использования этого программного обеспечения. При появлении того или иного ПО возникает сообщество разработчиков. Члены этого сообщества начинают модифицировать и расширять оригинальную версию, улучшая и развивая ее. Так произошло с Linux, Apache, PHP, Mozilla и XHTML (открытым кодом, на котором работают браузеры Konqueror и Safari). Тот факт, что программное обеспечение с открытым кодом по лицензии можно использовать бесплатно, делает подобные продукты очень привлекательными для людей, работающих при небольшом бюджете.

Программное обеспечение с открытым кодом бывает самым разным, в том числе и коммерческим: текстовые процессоры, ПО для сервера, почтовые программы, Web-браузеры и, конечно же, системы управления сайтом. В этой главе мы сосредоточим наше внимание на CMS с открытым кодом.

Не все продукты с открытым кодом одинаковы по качеству и имеют одинаковые лицензии, поэтому вам надо быть очень осторожными при выборе. Действительно, CMS с открытым кодом могут подойти не всем. Несмотря на их низкую (либо вообще нулевую) цену, продукты с открытым кодом могут требовать большего участия специалистов, поскольку подобное ПО часто нуждается в том, чтобы хотя бы один человек в вашей команде обладал необходимыми навыками для установки и изменения CMS.

При выборе CMS с открытым кодом, примите во внимание пять следующих характеристик выигрышного продукта с открытым кодом:

1. Активное сообщество разработчиков. Чем активнее будет сообщество, тем больше улучшений и поддержки вы сможете получить.
2. Расширяемость. Удостоверьтесь, что CMS-инструмент, который вы присмотрели, сможет расширяться вместе в вашими потребностями. Не тратьте

время, энергию, и, конечно же, деньги на CMS, которая не будет расширяться по мере роста вашего сайта. (Например, открытый язык PHP расширяется в соответствии с требованиями Yahoo! - это хороший знак.)

3. Поддержка. Возможно, вам придется нанять консультанта, когда вы начнете использовать свою CMS с открытым кодом. Убедитесь, что в этой области поддержка доступна.
4. Привлекательность для ваших программистов, разработчиков и для вас. Верный индикатор успеха любой новой технологии - ее привлекательность для разработчиков. Если вашим программистам она не нравится, они не будут ее успешно использовать.
5. Соответствие вашим нуждам. Наконец, нужно удостовериться в том, что CMS, которую вы выбрали, соответствует вашим нуждам. Не тратьте время и деньги, устанавливая CMS, которая не предлагает необходимые вам функции и вынуждает понапрасну тратить время, пытаясь выжать что-то хорошее из плохой ситуации, или подстраивать ваши деловые потребности под эту ситуацию, чтобы оправдать использование системы, которая на самом деле не помогает вам.

Существует масса самых разных CMS-инструментов - от широкомасштабных инструментов уровня предприятия до небольших приложений. Начнем с CMS уровня предприятий.

CMS-решения с открытым кодом на уровне предприятия

Говоря понятным языком, CMS уровня предприятия - это CMS, которая отвечает сложным потребностям и выдерживает огромную посещаемость. Подумайте об IBM, Microsoft, Amazon, Yahoo!, Apple и Google. Это гигантские Web-представительства, и они пользуются услугами гигантов мира CMS. В данном разделе вы ознакомитесь с самыми лучшими предложениями данного класса CMS с открытым кодом. Мы рассмотрим несколько примеров, а затем остановимся на одной системе и используем ее в качестве примера тестирования CMS на расширяемость, трафик и функциональные требования, необходимые для предприятий.

CMS-предложение: Midgard

Наш первый пример, Midgard, не является CMS в полном смысле этого слова - это каркас управления содержанием (www.midgard-project.org). Если вы разработчик, возможно, вам захочется использовать Midgard в качестве каркаса для

CMS, которую вы разрабатываете. Обычно потребители используют CMS-системы, такие как Aegir, Asgard и PHPmole, которые созданы на основе каркаса Midgard. Midgard построен на Apache, PHP и компонентах MySQL, для его использования нужен набор навыков, который мы обрисовали ранее, когда обсуждали техническое понимание штатных сотрудников; которое требуется для работы с продуктами с открытым кодом. Достоинство Midgard состоит в поддержке последовательности действий редакторов, механизмов утверждения и его способности автоматически прикреплять метаданные ко всему содержанию.

CMS-предложение: PHPmole

PHPmole использует Midgard в качестве каркаса, однако, помимо этого, он обладает прекрасными инструментами для разработки, включающими в себя визуальный HTML-редактор и другие простые в использовании графические инструменты разработки. PHPmole, как и многие проекты с открытым кодом, - это продукт любви разработчиков, и он постоянно эволюционирует. Если ваш офис или организация активно пользуется MS Word для создания содержания, обратите внимание на другую CMS, основанную на Midgard, - это Aegir CMS (www.aegir-cms.org) - рис. 6.8. Также Aegir поддерживает несколько кодировок языков, включая Unicode. Если вы пишете содержание на таких языках, как финский, немецкий, китайский, русский или маори, вам следует попробовать эту CMS, потому что она имеет встроенную поддержку языков.

При составлении функциональных требований для CMS в этой главе (см. выше) простая установка была отмечена как большой плюс. Здесь может подойти OpenCMS (www.opencms.com). OpenCMS поставляется вместе с HTML-мастером установки, который поможет вам подключиться к базе данных. Также вместе с ним поставляются инструменты, которые помогут вам управлять последовательностью действий; это позволяет вам проще администрировать и разделять задачи по созданию содержания, редактированию и публикации, которые ранее обсуждалось в разделе «Нужны ли вам области для подготовки и для производства?» Также вместе с OpenCMS поставляется визуальный редактор, который поможет людям нетехнического склада ума легко создавать и обновлять содержание.

Исследование: Zope

Из широкомасштабных CMS-инструментов уровня предприятия Zope - наиболее известный. Должна признаться: для одного из моих сайтов я использую Zope. Я думаю, вы понимаете, почему я выбрала именно его в качестве

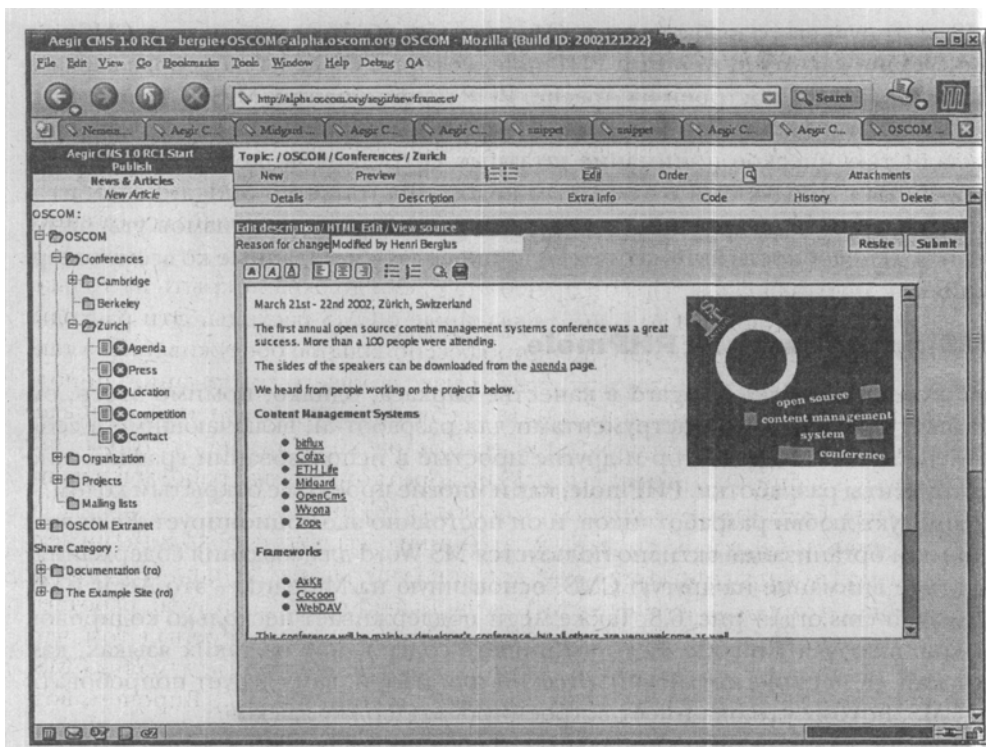


Рис. 6.8. Aegir (www.aegir-cms.org) доступен бесплатно по открытой лицензии. В качестве ядра он использует CMS-каркас Midgard и предлагает некоторые простые для использования графические интерфейсы, которые выглядят, как HTML-редактор. В предыдущем разделе по планированию мы сказали, что такой простой визуальный интерфейс на Web-основе современем понадобится для предполагаемого проекта, чтобы создатели содержания, не имеющие специальной технической базы, могли легко работать с сайтом. Такой интерфейс очень хорошо соответствует этому требованию

основного примера CMS с открытым кодом уровня предприятия: потому что я знаю его лучше всего.

Среди CMS-инструментов с открытым кодом Zope появился одним из первых - его выпустила в 1998 году Zope Corporation. Zope также включает в себя свой собственный сервер приложений, что может быть ценным дополнением либо, наоборот, лишней вещью - в зависимости от того, с какой точки зрения на это посмотреть. Zope включает программное обеспечение Web- и FTP-серверов, которые извлекают содержание из базы данных, например из Oracle, MySQL или SQL Server 2000.

Системные требования. Zope работает под управлением большинства ОС, включая Linux, Solaris, Windows и Mac OS.

Для каких сайтов подходит лучше всего. Zope был разработан для крупных сайтов предприятий. Среди его пользователей: Министерство сельского хозяйства США, Viacom, AARP, Военно-морской флот США и НАТО. Но Zope также может эффективно работать и для меньших сайтов. Я использую его для управления своим персональным сайтом, Rogue Librarian, который представляет из себя шоу одного человека на менее чем 300 страницах. Поэтому не думайте, что вы «слишком маленькие» для того, чтобы пользоваться Zope.

Поддержка. Поддержку вы можете получить бесплатно на сайте Zope, который называется Euro Zope (www.eurozope.org), а также из почтовых рассылок. Заплатив, вы также можете получить поддержку непосредственно от Zope Corporation или от компаний, которые работают с Zope.

Источник. Вы можете скачать Zope на www.Zope.org/Products.

Необходимые навыки. Скрытая стоимость Zope состоит в том, что вам придется нанять талантливого специалиста, чтобы поддерживать систему. Большая часть Zope написана на языке Python, поэтому будет необходимо, чтобы в штате сотрудников был человек, знакомый с этим языком, если вы хотите настроить Zope, потому что, как мы уже обсуждали ранее, большинство CMS необходимо настраивать. Программистов на языке Python найти не так легко, как других программистов, поэтому вы можете столкнуться с требованиями, выполнить которых вы не можете. Впрочем, если ваши планы по использованию этой CMS не слишком претенциозны, вы можете воспользоваться имеющейся конфигурацией Zope и по ходу работы изучать Python, выполняя небольшие настройки по мере необходимости. Однако такой подход не подойдет, если вам необходимы решения на уровне предприятия. Само собой, вам надо хорошо разбираться в базе данных, с которой вы работаете.

В заключение надо добавить, что при работе с Zope неплохо знать представление страниц на основе тегов и языки сценариев. Zope использует язык DTML (Document Template Markup Language - Язык разметки шаблонов документов) для того, чтобы автоматически создавать, управлять и форматировать содержание. DTML - это строительный блок системы Zope, поэтому вам придется выучить этот язык, если вы хотите сделать что-нибудь необычное. Если у вас есть опыт работы с представлением страниц на основе тегов и языками сценариев, такими как ColdFusion, вам, наверное, будет несложно разобраться в DTML. Если у вас мало опыта в этой области, вам понадобится больше времени, чтобы разобраться в Zope. Если вы владеете основами PHP, это поможет вам разобраться в DTML.

Поддерживаемые базы данных. Под UNIX вы можете использовать базы данных Oracle, PostgreSQL, MySQL, Sybase, InterBase и DB2. Под Windows вы можете использовать любую базу данных, поддерживающую ODBC, включая SQL Server 2000. Более подробную информацию вы можете получить на сайте www.zopenewbies.net/faq.html, вопрос 2.11 («Which databases can I use with Zope? - Какие базы данных можно использовать с Zope?»).

Особые возможности. Побочные продукты Zope - это полезные приложения, которые можно использовать для добавления мощных специфических возможностей на ваш сайт. Доски предложений работы, платежные системы, портал предприятия, доски сообщений и календари - это лишь некоторые возможности. Полный список продукции Zope Products смотрите в списке продуктов сообщества Zope (www.zope.org/Products).

Zope-ресурсы. Одно из преимуществ работы с Zope состоит в том, что Zope - это активное сообщество разработчиков. Ресурсы, которые могут помочь вам узнать больше о продукции, весьма качественны. Существует также много разработчиков, занимающихся созданием небольших приложений с целью расширения возможностей Zope. Обратите внимание на следующие ресурсы, с которых можно начать:

- Zope Community (www.zope.org);
- Zope Newbies (www.zopenewbies.net);
- The Zope Book (www.zope.org/Documentation/Books/ZopeBook).

Является ли Zope хорошим выбором? Теперь, когда мы произвели небольшое исследование, можно обновить таблицу планирования, которую мы начали в предыдущем разделе главы (см. табл. 6.3). Благодаря этому вы сможете понять, соответствует ли Zope вашим особым нуждам. Посмотрим, чего стоит Zope (табл. 6.4).

Помните, что вы работаете с гипотетическим бюджетом в \$1000. Конечно, Zope распространяется бесплатно, но, как вы уже, наверное, поняли, установить его будет не слишком просто. Представьте также, что ваши технические навыки ограничены. Бюджет в \$1000 пригодится, если вы будете обращаться к услугам профессионалов за помощью в установке и запуске Zope. Неплохо будет заключить договор на поддержку, это будет стоить немного дешевле.

Поскольку бюджет у вас небольшой, ваши функциональные требования достаточно скромны (возможно, слишком скромны) для Zope, поэтому вы можете потратить весь имеющийся бюджет на поддержку и обучение, заодно и узнаете, не является ли Zope *слишком крупным решением* ваших проблем. Вам следует также обратить внимание на открытые CMS-решения меньших размеров, а также ознакомиться с некоторыми недорогими коммерческими системами.

Таблица 6.4. Исследование: подходит ли нам Zope или нет?

Необходимость наличия тех или иных возможностей	Ответ	Как с этим справляется Zope
Нужно ли вам управлять разметкой страницы при помощи шаблонов?	Очень нужно	Zope позволяет достаточно легко создавать шаблоны и управлять ими
Нужны ли вам возможности по написанию разметки и визуальному редактированию?	Отчасти	Без необходимой модификации Zope не выполняет этой задачи. Дополнение под названием IE Editor for Zope (http://vsbabu.org/webdev/zopedev/ieditor.html) решает эту проблему
Нужна ли вам поддержка XHTML и CSS?	Очень нужна	При необходимой конфигурации Zope поддерживает XHTML и CSS
Нужна ли вам возможность соединения с базой данных?	Очень нужна	Zope работает с нашим стандартом, базой данных MySQL
Нужны ли вам области для подготовки и для производства?	Не уверен	На Web-сайте AARP система Zope сконфигурирована для параллельной работы в областях подготовки и производства. Читайте об этом на www.zope.com/CaseStudies/aarp
Нужен ли вам механизм RSS?	Очень	Такой механизм можно создать достаточно легко. Один из способов рассмотрен в статье на XML.com, «Advanced XML Applications in Zope» (www.xml.com/pub/a/2000/02/23/zope)
Команда из 7–10 человек	Очень нужна	Zope можно использовать и при работе с большей командой
Управление версиями	Отчасти	Zope позволяет пользователям использовать функции Check out (Выход) и Lock (Заблокировать) при работе с документами, таким образом, только один человек может работать на странице в определенный момент времени. Zope также обладает инструментом Rollback (Откат), который позволяет нам вернуться к более ранней версии данной страницы
Функциональные требования		
Работает под управлением UNIX	Очень нужно	Zope очень хорошо работает под управлением UNIX
Работает с базой данных MySQL	Очень нужно	Для Zope это не проблема

Таблица 6.4. Исследование: подходит ли нам Zope или нет? (окончание)

Необходимость наличия тех или иных возможностей	Ответ	Как с этим справляется Zope
Простота в установке и поддержке	Очень нужна	Установить и поддерживать Zope не так просто. Некоторые сайты Zope внушают, что Zope очень легко установить. Мне кажется, что эти сайты сильно стараются склонить пользователей в сторону Zope. В других источниках я читала, что Zope установить достаточно трудно
Другие факторы		
Общий бюджет	\$1000	Выбирая Zope, вам придется потратиться на специалистов, которые смогут помочь установить Zope, а затем на обучение, чтобы научиться расширять возможности Zope
Навыки штатных специалистов	PHP/Perl	
	XML/XSLT	Это может пригодиться при установке механизма RSS
	Графический дизайн	
	(X)HTML и CSS	
	Apache/UNIX	
	SQL	
	MySQL, Access	Мы можем работать и с другими базами данных, однако нам необходимо, чтобы наша CMS поддерживала MySQL – наш новый стандарт

Открытые CMS меньших размеров

В предыдущем разделе вы узнали о крупномасштабных открытых CMS. Оценив потребности вашего воображаемого сайта, вы могли понять, что решение уровня предприятия - это, возможно, слишком крупное решение вашей проблемы. Теперь давайте рассмотрим несколько маленьких и средних открытых систем и определим, подходят ли они вам. Как и ранее, мы кратко рассмотрим несколько примеров, а затем более подробно остановимся на одном, чтобы продолжить изучение.

CMS-предложение: Textpattern

Textpattern (www.textpattern.com) - это новая CMS, созданная дизайнером и разработчиком Дином Алленом (Dean Allen). Textpattern - хороший пример простой CMS, которая добросовестно выполняет несколько задач (рис. 6.9). Одно из преимуществ состоит в великолепной поддержке (X)HTML и CSS; в нашем гипотетическом списке нужд это было одним из основных требований.

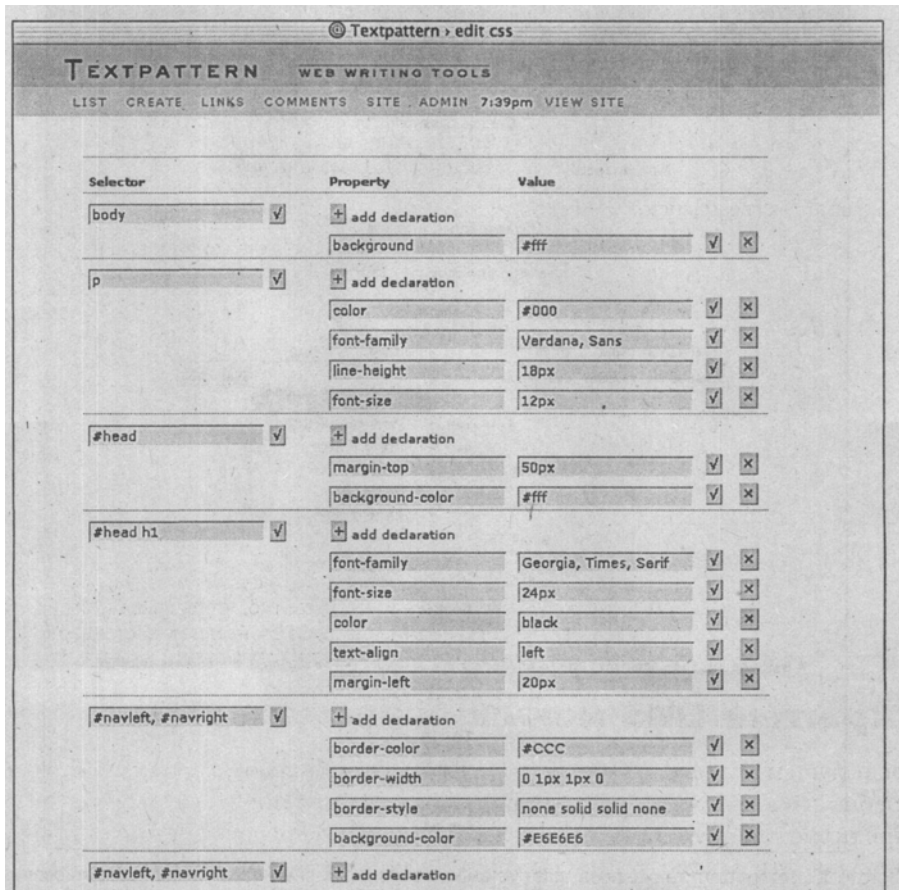


Рис. 6.9. Textpattern (www.textpattern.com) - новая стильная CMS с открытым кодом, которая позволяет устанавливать и управлять CSS-установками на вашем сайте. CSS - это экономичный способ управления и оформления дизайна сайта. Значимость CSS мы рассмотрим в следующей главе

Textpattern позволяет установить механизм RSS и работает с MySQL. Таким образом, она соответствует некоторым требованиям воображаемого сайта. Однако, имея лишь пять заранее установленных уровней администрирования (рис. 6.10), этот инструмент может оказаться недостаточно мощным с точки зрения управления.

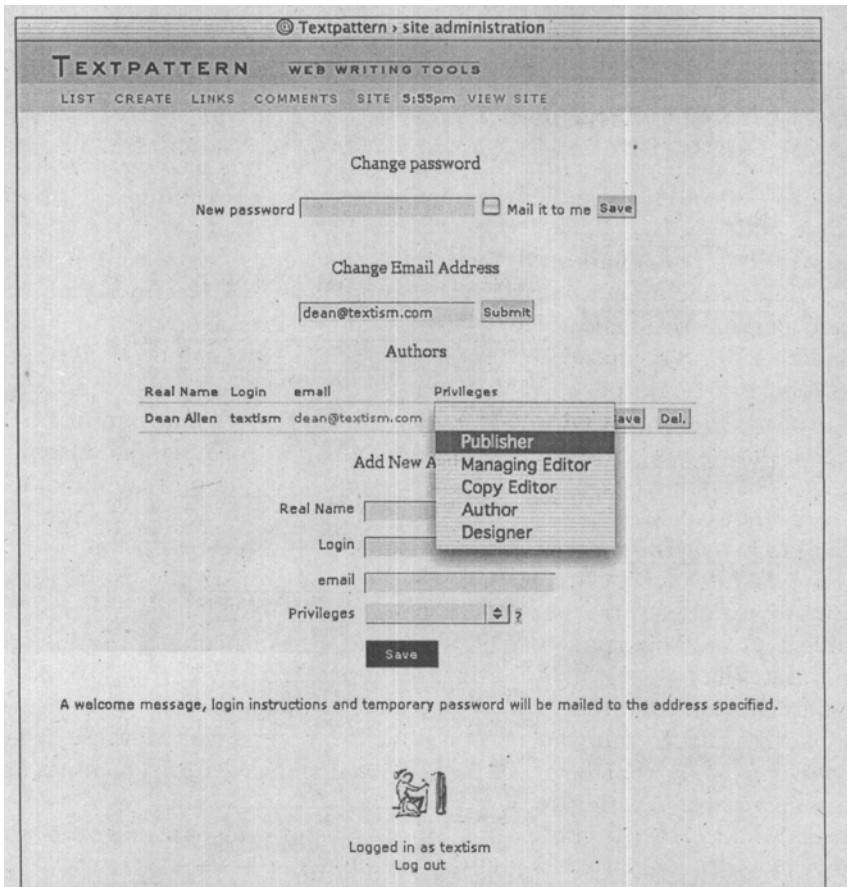


Рис. 6.10. Textpattern также позволяет устанавливать различные уровни доступа для разных пользователей, например для издателей, ответственных редакторов, литредакторов, авторов и дизайнеров

CMS-предложение: Greymatter

Более сложная CMS, чем система Textpattern, - это Greymatter (www.noahgrey.com/greysoft). Эта система была создана специально для управления сайтами ведения журналов и протоколов, впрочем, вы можете использовать эту CMS и для сайтов другой направленности. Greymatter дает возможность добавлять содержание на сайт, отсылая на сервер email-сообщение; она также обладает автоматическим RSS-механизмом и инструментом по проверке орфографии, который вы можете добавить в интерфейс.

Исследование: Movable Type

Одна из наиболее интересных открытых CMS средних размеров, появившихся на рынке, - Movable Type. Посмотрим, насколько этот продукт соответствует вашим требованиям.

Системные требования. Вам понадобится сервер, который поддерживает MySQL или менее известную базу данных Berkeley DB. Несмотря на то что она менее известна среди дизайнеров, многие хостинговые компании поддерживают Berkeley DB. Вам также понадобится сервер, который позволяет использовать CGI-скрипты и предоставляет минимум 25 Мб свободного пространства. Требования к программному обеспечению не являются постоянными. Загляните на www.movabletype.org/requirements.shtml для того, чтобы узнать о последних требованиях и о том, поддерживает ли ваша хостинговая компания необходимые технологии.

Для каких сайтов подходит лучше всего. Movable Type лучше всего подходит для персональных сайтов и для журналов, которые регулярно обновляются. Несмотря на то что персональный сайт - идеальный для применения Movable Type, это программное обеспечение может использоваться и в других целях (рис. 6.11).

Exploit Boston!, бостонский справочник по искусству, культуре и развлечениям, - сайт, который не является журналом, использует Movable Type для работы календаря, который группирует события по дате и категории и предлагает пользователям различные мероприятия.

Поддержка. Члены сообщества и создатели программного обеспечения осуществляют поддержку Movable Type в форумах.

Источник. Вы можете скачать данную CMS на Web-сайте проекта Movable Type по адресу www.movabletype.org/download.shtml. Учебник также находится на сайте www.movabletype.org по адресу www.movabletype.org/docs/mtinstall.html.

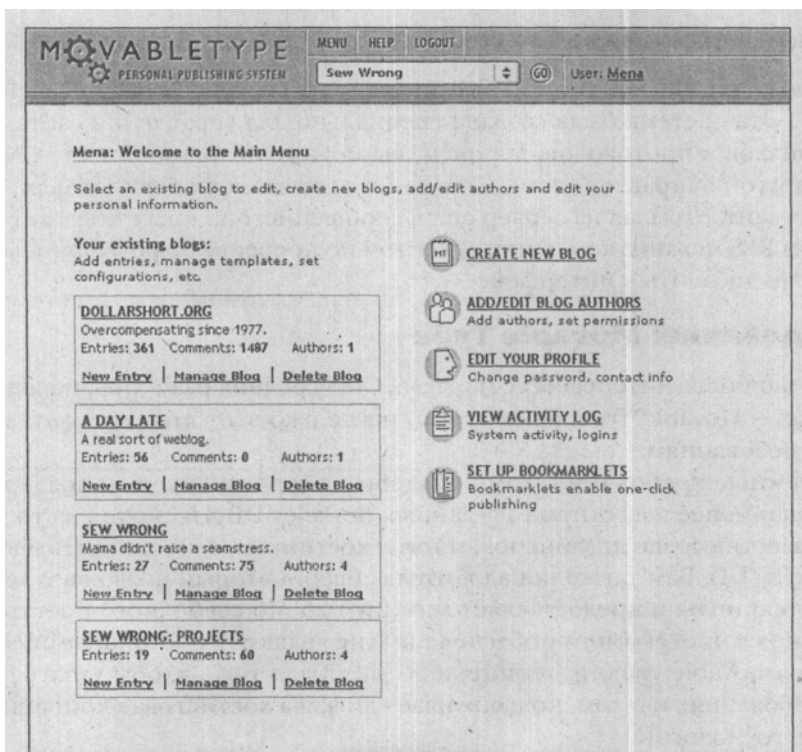


Рис. .6.11. Популярная CMS Movable Type с открытым кодом требует по крайней мере базовых знаний по HTML для создания и редактирования содержания

Необходимые навыки. Если вы не являетесь специалистом в технической области, вам, возможно, понадобится помощь в установке Movable Type на сервер. После установки вы можете относительно спокойно использовать CMS. Если у вас есть опыт программирования в Perl, можно написать собственные динамические приложения, используя библиотеки Movable Type, - это позволит вам добавить на сайт новые функции.

Специальные возможности. К этим возможностям относятся следующие: механизм RSS; поисковая машина; встроенная система создания комментариев; система email-оповещения; авторизация входа, которая классифицирует каждый вход как посещение; функция **Trackback** (которая упрощает перекрестное соединение сайтов) и функция состояния **Hold/Release** (Заблокировать/Открыть), которая позволяет заблокировать страницу при внесении изменений, а затем открыть ее, когда изменения будут внесены.

Расширяемость - одно из главных достоинств Movable Type. Подобно Zope и многим другим CMS-продуктам с открытым кодом, Movable Type поддерживает дополнительные приложения, которые можно использовать для улучшения вашего сайта. В эти дополнения входят инструменты RSS-календаря, категоризация Web-сайтов, модуль Smarty Pants, который автоматически создает типографически правильные кавычки и апострофы при цитировании, и много других интересных вещей. Сообщество развития Movable Type - очень активное, поэтому со временем ждите новых дополнений.

Давайте обновим таблицу и посмотрим, как Movable Type соответствует счету в \$1000 (табл. 6.5).

Таблица 6.5. Исследование: подходит ли нам Movable Type или нет?

Необходимость наличия тех или иных возможностей	Ответ	Как с этим справляется Movable Type
Нужно ли вам управлять разметкой страницы при помощи шаблонов?	Очень нужно	Movable Type позволяет достаточно легко создавать шаблоны и управлять ими
Нужны ли вам возможности по написанию разметки и визуальному редактированию?	Отчасти	Movable Type поставляется только с одним простым интерфейсом для добавления содержания, однако существует множество дополнительных модулей для графического редактирования
Нужна ли вам поддержка XHTML и CSS?	Очень нужна	При необходимой конфигурации Movable Type поддерживает XHTML и CSS
Нужна ли вам возможность соединения с базой данных?	Очень нужна	Movable Type работает с нашим стандартом, базой данных MySQL
Нужны ли вам области для подготовки и для производства?	Не уверен	Movable Type не поддерживает это
Нужен ли вам механизм RSS?	Очень	Movable Type легко поддерживает этот механизм
Контроль над версиями	Отчасти	Это не поддерживается в полной мере, но Movable Type обладает способностью блокировать страницу, когда происходит ее редактирование
Функциональные требования		
Работает под управлением UNIX	Очень нужно	Movable Type очень хорошо работает под управлением UNIX
Работает с базой данных MySQL	Очень нужно	Для Movable Type это не проблема
Простота в установке и поддержке	Очень нужна	Movable Type гораздо проще устанавливать и поддерживать, чем Zope

Таблица 6.5. Исследование: подходит ли нам Movable Type или нет? (окончание)

Необходимость наличия тех или иных возможностей	Ответ	Как с этим справляется Movable Type
Другие факторы		
Общий бюджет	\$1000	Movable Type соответствует многим вашим требованиям и не должна разорить ваш бюджет. Для некоммерческого использования продукт распространяется бесплатно (хотя если вам того захочется, создатели не откажутся от скромного дара где-нибудь в\$20). Коммерческая лицензия стоит примерно \$150
Навыки штатных специалистов	PHP/Perl	
	XML/XSLT	
	Графический дизайн	
	(X)HTML и CSS	
	Apache/UNIX	
	SQL	
	MySQL, Access	

Является ли Movable Type хорошим выбором? Допустим, что наш гипотетический сайт не будет Web-дневником или чьим-то персональным сайтом; это значит, что Movable Type придется выполнять работу, немного отличающуюся от той, для которой он был создан. Тем не менее Movable Type - достаточно мощная и хорошо настраиваемая программа для того, чтобы управлять различными видами сайтов. Exploit Boston (www.exploitboston.com) - прекрасный пример сайта, управляемого при помощи Movable Type и оснащенного дополнительными модулями Movable Type. На рис. 6.12-6.14 показано, как Exploit Boston использует некоторые из дополнительных легко доступных инструментов календаря.

Для того чтобы заставить Movable Type работать как нежурнальное программное обеспечение, придется потрудиться, но, поскольку у нас гипотетический пример, представим, что у вас в штате есть сотрудники с необходимыми

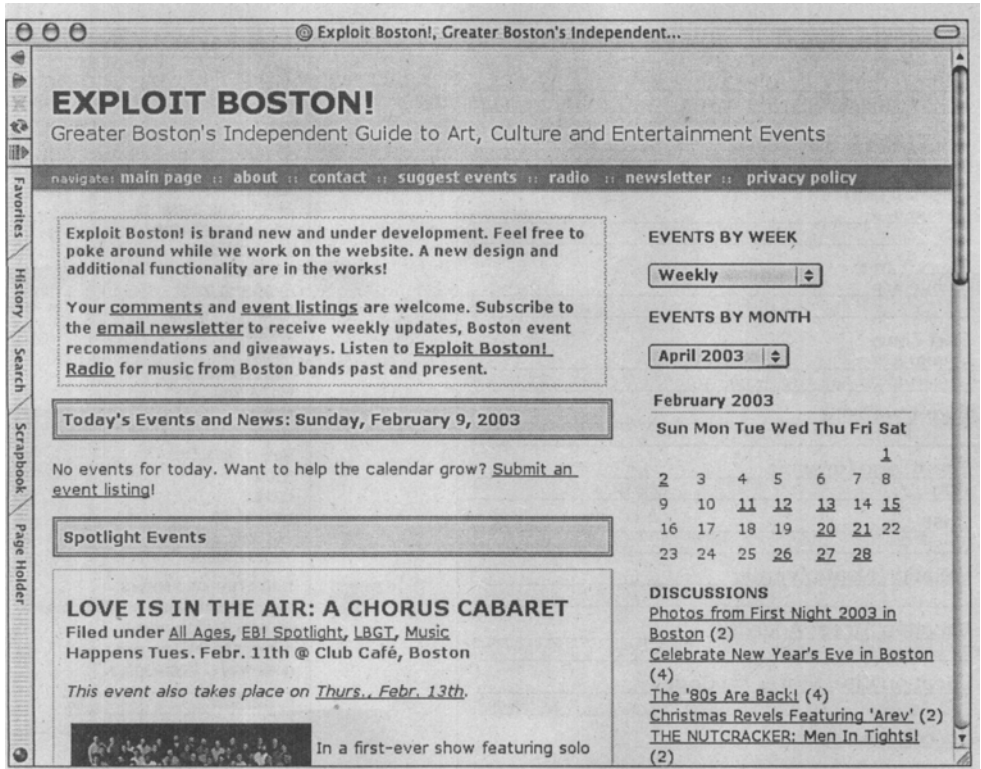


Рис. 6.12. ExploitBoston (www.exploitboston.com/) - сайт, который посвящен культуре и искусству в Бинтауне и при этом является недорогим. Создатель сайта, Сооз Кауп (Sooz Каир), использует Movable Type - CMS для управления персональными сайтами в виде дневника - для поддержки online-директории событий. Каупдобиласьуспеха, используябесплатныеподручныересурсы

навыками. Ни Zore, ни Movable Type полностью не соответствуют указанным здесь требованиям, однако вместе они выполняют все необходимое, за исключением одного: возможно, вам будет трудно установить Zore самостоятельно; из-за более простой установки вам, скорее всего, захочется в первую очередь попробовать Movable Type. Если вам удастся ее благополучно установить, вы сэкономите \$100, которые сможете потратить на что-нибудь еще, например на вечеринку в офисе.

Поскольку оба сообщества разработчиков весьма активны, Zore и Movable Type будут быстро развиваться. Так, если вы выберете Movable Type, не исключено, что в скором времени появится возможность работы в отдельных

About the event ...

Title

Date (Click to Select Date)

Doors Open
 hh:mm a/p

Start Time
 hh:mm a/p

End Time
 hh:mm a/p

Event website

Event logo/graphic
 http://

Cost

Location Name/Venue

Location Street Address

Location City, State, Zip Code

Location website

Рис. 6.13. Мощности Movable Type хватает Кауп для предоставления электронного бланка, заполнив который, пользователь может посоветовать другим людям поучаствовать в каком-либо мероприятии. Это удобно и приносит успех сайту с постоянно меняющимся содержанием. В своем административном интерфейсе Кауп может просматривать, редактировать и публиковать сообщения об интересных событиях

EVENTS BY CATEGORY

[18+](#)
[21+](#)
[All Ages](#)
[Art](#)
[Benefit](#)
[Crafts](#)
[Dance](#)
[EBI Announcements](#)
[EBI Photo Albums](#)
[EBI Spotlight](#)
[Festivals](#)
[Food and Drink](#)
[Free](#)
[Humor](#)
[LBGT](#)
[MBTA Green Line](#)
[MBTA Red Line](#)
[Movies](#)
[Music](#)
[New Year's Eve](#)
[Other](#)
[Poetry](#)
[Smokefree](#)
[Theater and Plays](#)
[Variety Show](#)

UPDATES AND NEWS
 Sign-up with your email address to receive Boston event recommendations, news and giveaways. ([more info](#))

Рис. 6.14. Каун модифицировала Movable Type таким образом, чтобы он поддерживал систему сортировки событий по категориям. Movable Type также предоставляет пользователю возможность подписаться на email-рассылку с перечнем грядущих событий

областях подготовки и производства, а также более серьезное решение проблемы управления версиями. Поэтому вы должны угадать, точно ли появятся эти возможности у Movable Type и насколько они важны для ваших долгосрочных потребностей.

Прежде чем выбрать, вам надо решить еще некоторые задачи. Если вы хотите выбрать открытую CMS, позаботьтесь о том, чтобы у вас в штате были квалифицированные сотрудники, которые смогут управлять этим программным обеспечением. Вам понадобятся постоянно работающие люди, которые будут отслеживать новые версии ПО, расширять и поддерживать приложения. Надо будет найти консультантов, которые будут выполнять эту работу, а это, в свою очередь, тоже будет стоить денег.

Подобные затраты, связанные с долгосрочным содержанием дополнительного персонала, побуждают многих выбирать коммерческие системы. Прежде чем принять решение, давайте рассмотрим несколько коммерческих CMS.

Коммерческие CMS

Обойдемся без дорогих продуктов. Хотя среди дорогих и средних по цене CMS есть из чего выбрать. Те, кому позволяют средства, могут выбрать из Interwoven (www.interwoven.com) и BroadVision (www.broadvision.com). Эти системы могут стоить сотни тысяч долларов. (Каждая копия CMS уникальна для каждой компании, поэтому цены различны.) Общая стоимость такого инструмента может быть больше, если добавить сюда стоимость профессионального обслуживания, обучения и импорта старого содержания. Также можно выбрать из средних по цене CMS-систем, которые стоят от \$40000 до \$100000. Среди них: Merant (www.merant.com), Ingeniux (www.ingeniux.com), Paperthin (www.paperthin.com) и Mediasurface (www.mediasurface.com).

Впрочем, мне кажется, что людей, которые держат в руках эту книгу, мало интересуют средние по цене CMS-решения уровня предприятия, тем не менее я считаю, что будет полезно знать, сколько в принципе можно заплатить за CMS. А теперь давайте обратимся к дешевым CMS. Продолжая работать с воображаемым сайтом, рассмотрим системы, которые стоят намного меньше, чем \$10000.

Подходящих нам коммерческих CMS-решений так же много, как звезд на небе. Ладно, конечно, не так много, но некоторые все-таки есть.

CMS-предложение: WebWorld

Стоимость системы WebWorld (<http://cms.webworld.biz>) колеблется от \$29 до \$159 с пользователя. Ролевой принцип администрирования в системе WebWorld позволяет вам устанавливать администраторский, пользовательский и гостевой уровни для сотрудников, таким образом, у каждого члена вашей команды будет свой уровень доступа, что позволит ему делать только то, что необходимо. Помимо управления созданными в WebWorld страницами, данный

продукт может обновлять страницы, созданные CMS других компаний или даже вручную. Также WebWorld поставляется вместе с машиной поиска, которую вы можете установить и на страницы, не созданные WebWorld. Благодаря тому что с созданным ранее содержанием можно так просто работать, вы сэкономите время ваших сотрудников: вместо импорта своего старого содержания в новую CMS вы легко можете его подключить.

WebWorld обладает WYSIWYG-редактором на основе браузера для создания и редактирования содержания, поэтому сотрудники, не имеющие специальной технической подготовки, могут с легкостью осуществлять обновления, даже не зная HTML. Также они могут выполнять свою работу в любом месте, где есть MS Internet Explorer 5.0 и выше. Выше в этой главе я уже говорила о том, как важна возможность составления содержания прямо в Сети. Некоторые CMS требуют, чтобы у пользователей было специальное программное обеспечение, установленное на их компьютере; это значит, что вам придется платить за ПО для каждого сотрудника и за время, потраченное на установку. Доступ к CMS посредством Сети, таким образом, является отличной возможностью для вас. WebWorld - не единственная CMS, которая предлагает доступ через Web; многие CMS работают таким образом, включая дешевые и бесплатные системы для управления персональными страницами, которые мы рассматривали немного выше. Но это совсем не значит, что такая возможность присутствует у всех CMS. Если эта возможность важна для вас, убедитесь, что ПО, которое вы выбираете, этой возможностью обладает.

CMS-предложение: AssetUow

Если названия Microsoft и Macromedia для вас - не пустые звуки, AssetNow (www.assetnow.com), возможно, как раз то, что вы ищете. Коммерческая версия AssetNow стоит от \$1000 до \$2500, в зависимости от типа лицензии. Существует также личная редакция того же самого инструмента, которую вы можете использовать для персональных и некоммерческих сайтов. Если вас заинтересует AssetNow, можете сначала попробовать личную редакцию для персонального сайта; это поможет вам ознакомиться с основной структурой инструмента. Если понравится, можете приобрести коммерческую версию для работы.

Требования AssetNow отличаются от инструментов с открытым кодом, которые мы рассматривали. Вместо языка PHP это приложение использует сервер приложений Macromedia ColdFusion MX/5.0. Помимо Linux, AssetNow работает под управлением операционных систем Microsoft Windows NT/2000/XP. И вместо баз данных с открытым кодом, таких как MySQL, AssetNow работает с Microsoft Access.

Выгода работы с Access состоит в том, что эта программа у вас уже, возможно, есть; вы купили ее вместе с MS Office. Недостаток Access состоит в том, что она не очень хорошо работает с интенсивно работающими сайтами, управляемыми базой данных; Access разрабатывалась как настольное приложение, а не как Web-приложение. Если вы и ваши пользователи производите частые передачи данных в Сети - например, если у вас Интернет-магазин - у вас не будет гарантии, что ваши данные будут в полной безопасности. Кроме того, если ваш сайт станет популярным, Access может не выдержать большого количества посетительских запросов. Если вы только начинаете, можете начать работать с Access, но затем переходите на MySQL; главное, не затягивайте с этим, иначе застрянете с системой, которая не отвечает вашим потребностям.

Использование AssetNow имеет смысл, если вы работаете с небольшими Intranet-приложениями. Если у вас небольшое количество пользователей и простая база данных, все будет в порядке. К тому же AssetNow предлагает хорошук? поддержку CSS и (X)HTML, поэтому можете создавать хорошо оформленные страницы, приятные на вид и радующие количеством трафика.

•Определение_____:

UNIX и LINUX. *UNIX*- это операционная система, которая была разработана в Bell Labs в начале 1970-х годов. С годами UNIX значительно развилась и стала популярной операционной системой в университетах. *Linux* - это более новая версия UNIX, которая невероятно популярна. Она является бесплатной операционной системой с открытым кодом, разработанной, главным образом, одним человеком по имени Линус Торвальдс (Linus Torvalds). Linux увеличил популярность UNIX, и вы можете увидеть, что многие люди используют UNIX/Linux не только для серверов, но и для персональных компьютеров. Операционная система Macintosh OS X базируется на одной из версий UNIX и содержит инструменты и протоколы с открытым кодом под приятным интерфейсом Mac.

•Определение_____:

Intranet. *Intranet*, подобно Internet, - это сеть компьютеров, которые обмениваются информацией, однако сеть Intranet является частной и замкнутой. Данные в Intranet в компании или организации доступны только членам данной компании или организации.

Исследование: Easy Publish

Зоре для обычных пользователей? Easy Publish (табл. 6.6) основана на бесплатном продукте Zore с открытым кодом, который обсуждался в этой главе выше. Easy Publish может быть проще, но это не бесплатная CMS. Причина, по которой

цена (примерно \$2300) может быть подходящей, заключается в том, что Easy Publish уже содержит некоторые дополнительные инструменты Zope, поэтому еще раз за них платить не придется. Короче говоря, вы экономите время на работе, а время - это деньги.

Операционная система. Easy Publish работает под Windows 95/98/NT/2000 или Solaris, Linux, BSD, Mac OS X и другими.

Для каких сайтов подходит лучше всего. Поскольку это продукт, созданный на основе Zope, вы можете использовать его для крупных сайтов с высокой посещаемостью. Конечно, Easy Publish можно использовать и для более простых сайтов, но, если одно из ваших требований состоит в том, чтобы CMS выдерживала высокую посещаемость, Easy Publish будет вам полезна. К пользователям системы Easy Publish относятся, например, следующие: Cap Gemini, Ernst & Young England, Датский технический университет BioCentrum и Управление гражданской авиацией Дании.

Поддержка. Easy Publish предлагает заключение договоров на техподдержку по различным ценам. Техническая поддержка может стоить \$35 за 15 мин.

Источник. Для того чтобы скачать Easy Publish, зайдите на сайт www.easypublisher.com.

Необходимые навыки. Как и в случае с Zope, может пригодиться знание языка Python, причем продукты, поставляемые вместе с Easy Publish, работают самостоятельно.

Поддерживаемые базы данных. Под Windows вы можете использовать любую базу данных, поддерживающую ODBC, включая SQL Server 2000.

Специальные возможности. Специальные возможности у Easy Publish примерно те же, что и у Zope: эта система поддерживает все дополнения на основе Zope, которые вы без труда сможете найти и установить самостоятельно, если у вас не возникало проблем с использованием Zope.

Цена. Корпоративная лицензия для Easy Publish для неограниченного количества пользователей стоит примерно \$2340. Индивидуальные, некоммерческие лицензии доступны бесплатно.

Не совсем CMS

Следует обратить внимание на еще один ресурс, позволяющий управлять сайтом, но это не совсем CMS. На самом деле это сокращенный визуальный редактор Web-страниц Macromedia. Продукт называется Contribute и стоит он \$100

Таблица 6.6. Исследование: подходит ли нам Easy Publish или нет?

Необходимость наличия тех или иных возможностей	Ответ	Как с этим справляется Easy Publish
Нужно ли вам управлять разметкой страницы при помощи шаблонов?	Очень нужно.	Как и Zope, Easy Publish позволяет достаточно легко создавать шаблоны и управлять ими
Нужны ли вам возможности по написанию разметки и визуальному редактированию?	Отчасти	Несмотря на то что Zope для этой задачи требует модификации, в случае с Easy Publish эта возможность встроена в CMS: система обладает WYSIWYG-редактором с поддержкой графических стилей, ссылок, изображений, функции отмены, маркированных списков и нумерованных списков
Нужна ли вам поддержка XHTML и CSS?	Очень нужна	Easy Publish поддерживает XHTML и CSS в случае правильной установки страничных шаблонов
Нужна ли вам возможность соединения с базой данных?	Очень нужна	Easy Publish работает с нашим стандартом, базой данных MySQL
Нужны ли вам области для подготовки и для производства?	Не уверен	Поддержка версий и различных рабочих областей доступна в качестве дополнения; придется платить за дополнительные услуги консультанта. Дополнения поставляются вместе с издательскими и технологическими модулями, которые необходимы пользователю для редактирования и публикации
Нужен ли вам механизм RSS?	Очень	Как и в Zope, такой механизм можно легко создать. Если Easy Publish поставляется без RSS, мы можем модифицировать Easy Publish либо нанять консультанта, который добавит механизм RSS в Easy Publish
Команда из 7–10 человек	Очень нужна	Easy Publish можно использовать и при работе с большей командой
Контроль над версиями	Отчасти	Zope позволяет пользователям использовать функции Check out (Выход) и Lock (Заблокировать) при работе с документами, таким образом, только один человек может работать на странице в определенный момент. Zope также обладает инструментом Rollback (Откат), который позволяет нам вернуться к более ранней версии данной страницы

Таблица 6.6. Исследование: подходит ли нам Easy Publish или нет? (окончание)

Необходимость наличия тех или иных возможностей	Ответ	Как с этим справляется Easy Publish
Функциональные требования		
Работает под управлением UNIX	Очень нужно	Easy Publish очень хорошо работает под управлением UNIX
Работает с базой данных MySQL	Очень нужно	Для Easy Publish это не проблема
Простота в установке и поддержке	Очень нужна	Это часть профессиональных услуг, которые мы оплачиваем при лицензировании продукта
Другие факторы		
Общий бюджет	\$1000	Стоимость – \$2340 плюс профессиональные услуги. Необходимо выделить определенную сумму на установку и поддержку. Так что бюджет нужно расширить, если мы имеем дело с коммерческим проектом
Навыки штатных специалистов	PHP/Perl	
	XML/XSLT	
	Графический дизайн	
	(X)HTML и CSS	
	Apache/UNIX	
	SQL	
	MySQL, Access	Мы можем работать и с другими базами данных, однако нам необходимо, чтобы выбранная CMS поддерживала MySQL – наш новый стандарт

за копию. При такой цене можно купить несколько копий. (Однако помните, что Contribute - это настольное приложение, не работающее в Сети, поэтому каждому отдельно взятому пользователю придется устанавливать это программное обеспечение и, кроме того, за каждого пользователя нужно будет платить.)

Систему Contribute может использовать и создатель содержания сайта, не имеющий специальной технической подготовки. Графический интерфейс позволяет легко редактировать Web-страницы. Для публикации страницы после

завершения работы над ней используется FTP-клиент, встроенный в ПО. Как и в полноценных CMS-системах, администраторы решают, какие части каких страниц пользователь может редактировать. Contribute не может взаимодействовать с базами данных, поэтому приходится использовать статические страницы. Впрочем, вы можете создать шаблоны при помощи SSI или шаблонов Dreamweaver. В отличие от некоторых CMS-продуктов, Contribute позволяет пользователю вернуться к предыдущей версии страницы.

Последний по порядку, но не по важности совет

Наконец, я бы хотела дать вам еще один совет, который я не упомянула в этой главе:

Не тратьте деньги на создание многочисленных версий одной и той же страницы.

Нет нужды платить за программное обеспечение, которое управляет многочисленными версиями содержания. Некоторые продавцы могут предлагать вам купить различные CMS для различных Web-браузеров, карманных компьютеров и других устройств. Как вы увидите в следующей главе, это ненужная вещь, поскольку благодаря Web-стандартам можно с легкостью создать сайт один раз, одним способом, и сайт будет работать хорошо на самых разных устройствах. Теперь можете глубоко вздохнуть: эта глава закончилась - и последовать за мной в следующую, где мы рассмотрим связь между Web-стандартами и небольшим бюджетом.

ГЛАВА 7. Сэкономьте время и деньги благодаря Web-стандартам

Для хорошей работы прежде всего нужны хорошие инструменты.

Китайская пословица

Основные идеи главы

1. Выделите немного времени и денег на создание сайта, использующего и поддерживающего стандарты.

Это окупит себя при создании, а тем более при поддержке сайта.

2. Помните о важности разделения структуры и представления.

Когда наступит необходимость переделывать сайт, не следует менять XHTML-структуру. Возможно, вам понадобится добавить какие-то новые разделы, идентификаторы или части, но основная структура новой версии дизайна должна быть такой же, как в первой версии. В идеале обновление внешнего вида (представления) должно ограничиваться изменением таблиц стилей.

3. Создайте одну версию сайта, но сделайте это хорошо.

Если вы используете разметку с хорошей структурой, вам никогда не придется создавать и поддерживать многочисленные версии вашего сайта. Один хороший сайт будет работать одинаково хорошо и в традиционных браузерах, и на карманных компьютерах, и с устройствами для людей с ограниченными возможностями. Считайтесь с нуждами пользователей, и пользователи будут вас любить.

4. Помните, что CSS - это гораздо более мощный инструмент дизайна, чем HTML, используемый для представления Web-страницы.

Вы можете определять размер полей, отступы и оформление текста, а также создавать ролловеры и другие интерактивные эффекты. Все это вы можете сделать при помощи всего нескольких строк CSS - самого лучшего малобюджетного решения из когда-либо существовавших.

Каким образом Web-стандарты помогут вам сэкономить

Моя любимая бабушка в последние годы сошла с ума. Боясь, что люди «доберутся» до ее денег, она сняла все свои сбережения со счета и подожгла их в гостере. Некоторые Web-сайты делают то же самое каждый день, используя устаревшую разметку и техники кодирования. Ни один сайт не может выдержать такого, а малобюджетный сайт особенно. Web-стандарты, если их правильно использовать, смогут спасти ваш бюджет от пожара.

На одной из конференций по дизайну известный пророк Web-стандартов Эрик Мейер, автор книги «Эрик Мейер о CSS» (Eric Meyer on CSS, New Riders Publishing, 2002), рассказал об одном хорошо известном сайте Интернет-магазина, который теряет терабайт трафика в год, понижая скорость загрузки сайта и замедляя передачу информации. Таким образом этот сайт рискует потерять клиентов, и все из-за перегруженного меню навигации слева.

Вместо того чтобы использовать простой и надежный HTML-код, связанный с CSS, каждый пункт меню сайта находится в столбце таблицы с прозрачным разделительным GIF-файлом и небольшой синей GIF-стрелочкой. Учитывая то, что меню содержит примерно 50 различных пунктов, стоимость обслуживания каждой синей стрелочки достаточно велика.

Мейер отметил, что, если бы на сайте упростили разметку и вместо синей стрелочки использовали бы CSS-стиль, а также не создавали бы отдельный столбец таблицы для каждого пункта меню, владельцы страницы смогли бы уменьшить общий размер страницы на 5,518 Кб. Учитывая то, что страницы сайта просматриваются примерно миллион раз в день, ежедневная экономия на трафике составила бы 5,518 Гб. Нужно всего лишь немного подчистить разметку, чтобы за год сэкономить 2014 Тб. Предположим, что владельцы сайта платят по центру за гигабайт трафика; экономия могла бы составить \$20000.

Создание сайта на основе Web-стандартов помогает разработчикам уменьшить объем трафика и тем самым понизить расходы на хостинг. Построение по стандартам делает страницы проще и позволяет их быстрее загружать. При этом увеличивается работоспособность сайта и потенциально понижаются расходы, например, на дополнительный сервер. Увеличивая скорость работы страницы, стандарты также помогают избежать раздражения пользователей, из-за которого они покидают сайт, не собираясь когда-либо на него возвращаться. Однако улучшенная работа и экономия - это не главное для вашего сайта. Экономия и работа создают основу для производства и поддержки сайта.

Часть I: Снижение расходов на производство

Если бы вашей работой было создание новых лампочек, вы бы придумали новый вид резьбы или воспользовались бы уже существующими стандартными размерами? Само собой, глупый вопрос, ведь мы живем в постиндустриальном мире. Мы не обращаем внимания на стандарты, а использование стандартной резьбы было бы для нас само собой разумеющимся.

Однако при создании Web-страниц значимость работы по стандартам нередко забывается. Мы, Web-дизайнеры, очень часто понапрасну тратим время и деньги, не придавая значения тому, какую экономию нам предлагают стандарты, существующие в нашей отрасли. Мы тратим кучу времени на создание лампочек для каких-то особенных люстр, хотя можем потратить те же (или даже меньшие) ресурсы на создание лампочек, которые будут работать с огромным множеством осветительных приборов.

Интернет-стандарты определены Консорциумом World Wide Web, также известным как W3C (www.w3.org). Выражаясь простым языком, советы W3C рекомендуют, как должна создаваться разметка гипертекстовых документов (Web-страниц) и как пользовательские программы - включая браузеры, карманные компьютеры, браузеры Брайля (для слепых) - должны эту разметку отображать.

Web-стандарты - это термин, область определения которого настолько широка, что он сам становится достаточно расплывчатым. Можно сказать, что в целом Web-стандарты связаны с требованиями и советами W3C, которые охватывают все аспекты разметки, внешнего вида (представления), безопасности и кодировки. В этой главе мы будем использовать термин *Web-стандарты* в этом значении, чтобы он был понятен дизайнерам. Для опытных Web-дизайнеров и разработчиков Web-стандарты - это емкая, значимая разметка, язык XHTML в сочетании с отделенными от разметки Таблицами стилей, которые берут на себя задачи, связанные с внешним видом. То же самое касается и терминов *поддерживающее стандарты устройством поддерживающие стандарты страницы* - они обычно используются при описании данного метода создания сайтов.

Используя последние стандарты - XHTML для структуры документа и CSS для внешнего оформления, разработчики могут создавать сайты, которые будут работать с современными браузерами, браузерами будущего и другими устройствами отображения, включая ассистивную технологию (технологии для людей с ограниченными возможностями). Примером ассистивной технологии

может быть Screen Reader, который озвучивает содержание браузера для слепых, и другие устройства, которые помогают людям перемещаться по сайту, даже если они физически не способны использовать мышь.

Г Определение

World Wide Web Consortium (W3C) - это международный промышленный консорциум. Будучи основанным в 1994 году Тимом Бернерсом-Ли (Tim Berners-Lee), в данный момент он насчитывает более 300 членов. Его цель состоит в том, чтобы упростить взаимодействие людей в Web благодаря общим и открытым спецификациям, которые W3C называет «Web-стандартами». Члены консорциума и приглашенные эксперты создают рабочие группы, на которые возлагается ответственность за написание таких спецификаций. Если идеи принимаются, они появляются в Рекомендациях W3C (W3C Recommendations).

С 1994 года W3C выпустил более 40 рекомендаций. HTML, XHTML и CSS - это всего лишь малая часть их работы. Другие рекомендации вы можете найти здесь:

- Web Content Accessibility Guidelines 1.0 (Указания по доступности Web-содержания): www.w3.org/TR/WAI-WEBCONTENT;
- HTML 4.0 Guidelines for Mobile Access (Указания по мобильному доступу): www.w3.org/TR/NOTE-html40-mobile;
- Extensible Markup Language (XML): www.w3.org/XML .

Членами W3C являются следующие корпорации: Sun, America Online, Microsoft, Apple, Ask Jeeves, Canon, DoubleClick, ERICSSON и многие другие.

-Определение

Пользовательский агент (User Agent). В нашем случае *пользовательский агент* - это виртуальная либо реальная машина, которая используется для доступа в Интернет. Web-браузеры, карманные компьютеры, например PalmPilot, браузеры Брайля, мобильные телефоны и голосовые браузеры - все это примеры пользовательских агентов, которые посетители могут использовать для того, чтобы попасть на ваш сайт. Вы сэкономите время и деньги, если будете считаться с пользовательскими агентами при создании сайта: подумайте как о современных агентах, так и об агентах будущего.

Правильно использованный XHTML-код создает четкую структуру для Web-сайтов, а CSS использует эту структуру для того, чтобы правильно эти сайты оформить. Выгода, которую вы можете извлечь при производстве и отображении сайта для широкой аудитории, состоит в том, что XHTML выполняет структурную работу, а CSS заботятся о дизайне и внешнем виде. Разделение

между структурой документа и внешним видом - это ключ при использовании этих стандартов. У принимающих устройств, поисковых машин, индексов, CMS и других инструментов должна быть возможность ощутить преимущества такого разделения.

Определение _____:

Структура (Structure). Когда мы говорим о *структуре* Web-страниц, мы говорим о семантической и иерархической природе разметки. Четкая, логически упорядоченная иерархия - вот что должна действительно выражать разметка. Так, когда вы заключаете часть текста в рамки абзаца, вы даете этому тексту место в иерархии документа - это абзац. Когда вы заключаете строку текста в рамки тега `<h1>`, вы даете этой строке текста структурное значение заголовка.

р Определение _____

Представление (Presentation). Когда мы говорим о *внешнем виде* Web-страницы, мы говорим о том, как она выглядит в браузере. Внешний вид должен быть отделен от структуры, и здесь мы определяем, как будут выглядеть элементы абзаца или заголовка. Самый лучший, самый эффективный и самый экономичный способ управлять внешним видом - держать его отдельно от структуры, позволяя HTML-коду выполнять структурную работу, а CSS - работу над внешним видом.

В работе с Web-стандартами есть два преимущества с точки зрения экономии:

1. Вы потратите меньше времени и денег на разработку сайта и гораздо меньше времени и денег на переделку сайта.
2. У вас получится сайт, который будет правильно работать и в Web-браузерах, и в карманных компьютерах, и в устройствах, поддерживающих ассистивные технологии.

Чтобы добиться этих преимуществ, вам придется следовать трем принципам, которые крайне необходимы для успеха малобюджетного Web-сайта:

1. При создании сайта старайтесь отделять структуру документа от его внешнего вида.
2. Считайтесь со всеми возможными пользовательскими агентами. (Не делайте сайт для одного определенного браузера.)
3. Пообещайте сами себе создавать качественные сайты, рассчитанные на долгий срок службы.

-Определение

Адаптивная технология (Adaptive Technology). Адаптивная, или ассистивная, технология - возможность работать на компьютере, предоставляемая людям с ограниченными физическими возможностями. Одна из самых объемных категорий ассистивных технологий - это программное обеспечение, которое выполняет такие вещи, как чтение текста вслух, увеличение текста, упрощение использования клавиатуры и других устройств ввода для людей с уменьшенной подвижностью. Из Web-статистики сложно узнать, когда ваш сайт посещают люди, использующие адаптивные технологии. Часто на ваш сайт заходят, используя обычный Web-браузер, например Netscape или Internet Explorer, но посетитель, использующий браузер, использует какую-либо адаптивную технологию.

В первой части данной главы вы узнаете о том, как важна работа по созданию фундамента. Во второй части я расскажу вам о том, как применить эти правила к работающему Web-сайту.

г-Определение

Суть HTML Эта глава предполагает, что у вас есть хотя бы некоторое представление о HTML. Хороший online-ресурс - это учебник W3Schools's HTML Tutorial (www.w3schools.com/html/). Также я рекомендую книгу Джеффри Зельдмана (Jeffrey Zeldman) «Дизайн по Web-стандартам» (Designing with Web Standards), НТ Пресс, 2005.

Хорошая разметка: разумные расходы

Многие из нас учились делать сайты в 90-е годы, в те дни широко использовались ужасные шрифты и вспомогательные таблицы, код по большей части применялся для того, чтобы задать семантику (для так называемой структурно значимой разметки), однако было мало способов повлиять на внешний вид. У сайта образца 1993 года был структурный смысл, но сделать так, чтобы он хорошо выглядел, было достаточно сложно. Когда были введены теги `` и `<table>`, а браузеры начали их поддерживать, дизайнеры просто лишились рассудка, добавляя эту новую разметку внешнего вида на свои страницы.

Для того чтобы кусочек текста отображался шрифтом Arial меньшего размера, необходимо было написать следующую строку в разметке:

```
<font face=arial size=-1>Здесь какой-то текст.</font>
```

Для каждого куска текста - будь то абзац, список или заголовок - приходилось писать нечто подобное.

Теги шрифтов было недостаточно в этом, теперь уже устаревшем методе. В главе б я рассказывала о вездесущих разделительных GIF-фалах и о том, как их использовали для создания полей. В дни браузеров 3.0 и 4.0 у вас не было особого выбора. Если вам необходимо было заполнение, пропуск или поля, приходилось отделять требуемое пространство при помощи белого или прозрачного изображения.

Web-дизайнерам даже удавалось извлекать выгоду из проблем с ранними браузерами. Во времена Netscape 1.0, если разработчик писал несколько тегов `<title>`, в результате в заголовке окна браузера получалось настоящее слайд-шоу. Следующие три тега отобразились бы в виде меняющегося текста:

```
<title>/КурНan Модные niTaHbi</title>
```

```
<title>Еженедельный журнал для модных любителей HiTaHOB</title>
```

```
<title>CaMbie модные шпаны в городе</title>
```

Игры с причудами и несоответствиями браузера стали очень важным опытом; написание разметки, которая по-новому использовала ошибки браузера, было работой дизайнера. Идущие подряд несколько тегов `<title>` - это классический пример из истории HTML. Привычка разработки дизайнера для отдельных браузеров сильна; большая часть разметки строилась на таких хитростях.

Несмотря на свой творческий характер, такой метод производства отнимал достаточно много времени, даже при использовании графических редакторов, таких как Microsoft FrontPage и Macromedia Dreamweaver. Тем не менее этот метод работал безотказно. Но такая разметка верно отображалась только на тех пользовательских агентах, о которых побеспокоились дизайнеры.

Почему презентационная разметка усложняла работу в Сети

Несмотря на то что презентационная, основанная на специфике браузера разметка была забавной и даже позволяла дизайнерам контролировать внешний вид сайта, все же она значительно усложняла работу в Сети:

1. Создание сайта занимало невероятное количество времени, а его переделка - еще больше.
2. HTML-разметка не выполняла своих непосредственных задач. Вспомните пример с тегом `<title>`. Этот элемент должен сообщать пользовательским агентам название данной страницы сайта. Если у вас три тега `<title>`, какой из них должна использовать поисковая машина?

3. Слишком чувствительная к браузеру HTML-разметка должна была переделываться каждый раз, когда появлялся новый браузер, таким образом, увеличивалась стоимость сайта.
4. Страницы были перегружены лишней разметкой, которая отнимала время и деньги дизайнеров и пользователей за хранение и трафик.

Г Угроза бюджету

Плохая, перегруженная разметка «съест» ваш бюджет несколькими путями: создавая большие HTML-страницы, медленно загружаемые и дорогие для обслуживания; делая поддержку и переделку сайта более трудной задачей; и заставляя вас создавать разные версии сайта для нетрадиционных устройств просмотра. Ответственный Web-профессионал, работающий при небольшом бюджете, сэкономит время и деньги, создавая хорошо сделанную разметку.

Многим из нас приходилось создавать сайты таким образом; для некоторых это была работа - делать так, чтобы сайт выглядел хорошо на некоторых специфических браузерах. Производство было тесно связано с тем, как сайт выглядел в ранних версиях Netscape и Internet Explorer. Практически с каждым выпуском нового браузера приходилось возвращаться назад и изменять разметку представления сайта так, чтобы в новых браузерах сайт выглядел хорошо. Нельзя было отставать от прогресса, но следовать ему было очень сложно.

Хорошая разметка облагораживает Сеть

Впрочем, сейчас вам уже не надо делать презентационную разметку - вы просто не можете себе этого позволить. Браузеры и разметка изменились, вам больше не нужно тратить время, загромождая разметку презентационным «мусором». Некоторые браузеры с версий 4.0 начали одинаково поддерживать разметку, а с появлением Tasman (ядро браузеров в I.E. 5.0 для Mac) и Gecko (ядро браузеров для Mozilla, версий Netscape после 4-й и многих других браузеров) различные браузеры начали интерпретировать разметку практически одинаково. Такая одинаковость - результат того, что производители браузеров начали создавать пользовательские агенты в соответствии со стандартами W3C, написанными относительно (X)HTML и CSS.

В то же самое время дизайнеры начали осознавать необходимость создания сайтов, которые работают на альтернативных пользовательских агентах, таких как карманные компьютеры, браузеры Брайля и мобильные телефоны. К счастью, рекомендации W3C по мобильным устройствам и доступности Web-сайта

делают упор на разделение кодов XHTML и CSS, а сайты, созданные согласно стандартам W3C, легко и уверенно работают на всех пользовательских агентах.

Работа сегодняшнего Web-дизайнера сильно отличается от работы дизайнера 90-х: вы должны создавать качественные сайты, которые будут работать на разнообразных устройствах отображения. Самое главное, что для этого вам придется потратить меньше усилий, чем раньше. Такой подход должен состоять в создании одной версии вашего сайта, но чтобы при этом сайт работал на всех браузерах и на всех устройствах, старых, новых и тех, что еще появятся. В книге «Дизайн по Web-стандартам» Джеффри Зельдман (Jeffrey Zeldman) называет этот новый, экономичный, основанный на стандартах подход к Web-дизайну, «заблаговременной совместимостью» (forward compatibility), а в его книге показано, как этот подход работает как на крупных, так и на более скромных сайтах.

Урок по разделению структуры и внешнего вида

В этом разделе рисунки и листинги иллюстрируют значимость разделения структуры документа и его внешнего вида. Рис. 7.1 и листинг 7.1 показывают внешний вид и HTML-код сайта, созданного устаревшим способом, а рис. 7.2 и листинг 7.2 - внешний вид и HTML-код сайта, созданного новым способом. Обратите внимание, насколько кратка разметка в листинге 7.2 по сравнению с листингом 7.1. Листинг 7.2 краткий и скромный, потому что в нем нет разметки внешнего вида.

Листинг 7.1. Разметка сайта на рис. 7.1

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<title>спММер CSS и (X)HTML 1</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body bgcolor="#660066">
<h1>
<font color="#FFFFFF" size="+1" face="Georgia, Times New Roman, Times,
```

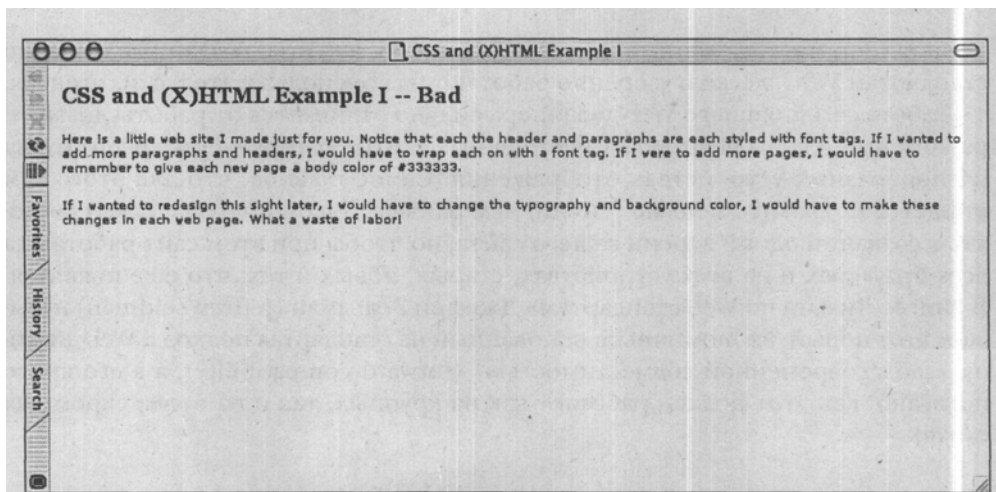


Рис. 7.1. Обратите внимание на то, что сайт был создан с использованием разметки внешнего вида.

Это плохо. HTML-код выполняет работу по представлению страницы в строке 7 (выделенной полужирным), где тег `body` содержит атрибут цвета. Кроме того, HTML-код выполняет презентационную работу в строке 8 (также выделенной полужирным); внутри тега `h1` вы заметите длинный тег шрифта, который задает цвет, размер и вид шрифта. Подобные атрибуты включены в тег абзаца. HTML-код данного сайта слишком трудный и неэкономный

serif">Пример CSS и XHTML 1 - плохо ****

```
<p> <font color="#FFFFFF" size="-2" face="Verdana, Arial,
Helvetica,
sans-serif"> Это небольшой Web-сайт, который я сделала специально
для вас. Обратите внимание на то, что заголовок и абзацы
оформлены при помощи тегов шрифта. Если бы я захотела добавить
еще заголовки и абзацы, мне бы пришлось каждый заключать в тег
шрифта. Если бы мне было надо добавить новые страницы, пришлось
бы дать каждой из них цвет body #333333.</fontx/p^
<pxfont color="#FFFFFF" size="-2" face="Verdana, Arial,
Helvetica,
sans-serif">Если бы потом мне захотелось изменить дизайн этого
сайта, мне бы надо было менять оформление текста и фоновый цвет,
и эти изменения пришлось бы делать на каждой странице. Это
невероятные потери времени!</font></p>
</body>
</html>
```

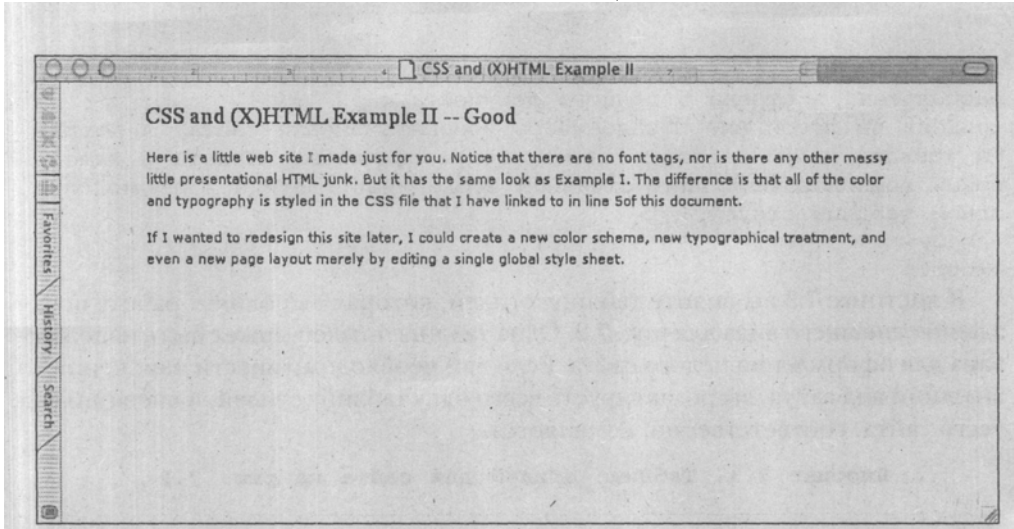



Рис. 7.2. Это практически тот же самый сайт, что и на рис. 7.1, только выглядит он немного лучше, а все потому, что HTML-код выполнял только структурную работу, а не презентационную. Это хороший пример. Выполнение презентационной работы было переложено с HTML-кода на CSS-документ. Ссылка, которая делает это возможным, выделена полужирным в листинге 7.2. Переделанный сайт выглядит лучше, потому что таблица стилей задает абзацам поля и произвольный интерлиньяж

Листинг 7.2. Разметка сайта на рис. 7.2

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1" />
<title> Пример CSS и (X)HTML II</title>
<link rel="stylesheet" media="screen" type="text/ess"
href="styles/intro.ess" />
</head>
<body>
<p1>Пример CSS и (X)HTML 2 - Хорошо</hl>
<p>Это небольшой Web-сайт, который я сделала специально для вас.
Обратите внимание на то, что здесь нет ни тегов шрифтов, ни
ненужных нагромождений презентационного HTML. Однако внешне сайт
выглядит так же, как и в примере 1. Разница состоит в том, что
```

```

цвет и оформление текста заданы в CSS-файле, ссылка на который
расположена в строке 5 данного документа.</p>
<p>Если бы потом мне понадобилось изменить 'дизайн сайта, я могла
бы создать новую цветовую схему, новое оформление текста и даже
нов*ый компоновочный план страницы всего лишь благодаря изменению
одной таблицы стилей.</p>
</body>
</html>

```

В листинге 7.3 вы видите таблицу стилей, которая выполняет работу по созданию внешнего вида для рис. 7.2. Одна таблица стилей может быть использована для оформления целого сайта. Если вам необходимо внести изменения во внешний вид сайта, вы редактируете всего одну таблицу стилей, и внешний вид всего сайта, соответственно, изменяется.

Листинг 7.3. Таблица стилей для сайта на рис. 7.2

```

body {
1.  color : #ffffff;
2.  background-color : #660066
3.  margin-left: 70px;
4.  margin-right: 70px;}
5.  p {
6.  font: 11px/18px verdana, geneva, arial, sans-serif;}
7.  hi {
8.  .font: 18px/24px Georgia, Times New Roman, Times, serif;}

```

Давайте посмотрим, почему такой подход делает оформление сайта гораздо более эффективным и менее дорогим для производства:

- вместо того чтобы заключать каждый элемент абзацев страницы в длинный, громоздкий тег шрифта, можно воспользоваться правилом в строке 7 таблицы стилей (см. листинг 7.3), которое сообщает браузеру о том, что ваши абзацы должны быть оформлены шрифтом Verdana размера 11 пт. (Если у пользователя нет шрифта Verdana, использоваться будет Geneva или Arial; если и этих шрифтов тоже нет, будет использован альтернативный шрифт без засечек, имеющийся на компьютере пользователя);
- все страницы сайта можно связать с одной таблицей стилей. Предположим, что сайт состоит из 40 HTML-страниц. Пользователь загружает эту таблицу стилей только один раз; затем браузер помещает эту таблицу стилей в кэш. Когда пользователь будет загружать оставшиеся 39 HTML-страниц сайта, таблица стилей к тому времени будет уже загружена. Вы (и ваши пользователи) сэкономите на потреблении трафика;

- если вам надо внести изменения во внешний вид сайта, вы просто меняете таблицу стилей. Предположим, что вам звонит клиент и сообщает, что абзацы, оформленные шрифтом Verdana, его раздражают; он хочет, чтобы абзацы были сделаны шрифтом Times New Roman или Georgia. При работе с дизайном, основанным на стандартах, вы редактируете строку 9 в таблице стилей - и требуемый внешний вид готов. С устаревшими тегами шрифтов подобные изменения заняли бы часы работы, в зависимости от количества страниц на сайте и разнообразия использованных начертаний шрифтов.

Такое использование таблицы стилей - это только начало; здесь мы создали всего три правила - для фона, заголовка и абзаца. В один CSS-документ вы можете добавлять различные вариации шрифтов и других элементов. Как сделать разнообразнее и полнее вашу таблицу стилей, вы узнаете в конце этой главы. Вы увидите, что здесь нет никаких пределов, что CSS дает вам возможность не только создать более легкий сайт, но и лучше управлять макетом страницы - так, как вы никогда и не надеялись бы сделать при помощи презентационного HTML. Самое важное - вы увидите, как значительно снизятся расходы на производство, когда вы будете использовать CSS для контроля над внешним видом сайта.

Пользовательские агенты

В предыдущей главе я рассказывала о том, как моя организация почти купила систему управления сайтом. Одним из основных критериев, от которых мы отталкивались при выборе CMS, была возможность создавать различные страницы для различных устройств просмотра: по одной версии для каждого браузера, одну для браузеров Брайля и других ассистивных технологий, версию для карманных устройств.

Сейчас стоимость производства различных страниц для различных устройств просмотра - это то, о чём надо забыть. Пусть никто вас не убеждает в том, что вам необходимо создавать разные версии сайта или что вам необходимы отдельные инструменты для этого. Web-стандарты решают эту проблему *бесплатно*.

- Как получить золото из песка —————

С помощью Web-стандартов вы создаете сайт, который хорошо работает в традиционных браузерах, карманных устройствах и устройствах, поддерживающих ассистивные технологии для людей с ограниченными физическими возможностями. Создавая *одну* простую версию сайта, вы заботитесь о работе всех этих устройств просмотра.

Если в вашем XHTML-документе не будет презентационной разметки, а XHTML-разметка будет простой, ваш сайт, таким образом, будет доступен для большого количества пользовательских агентов. Пользовательские агенты на основе текста, такие как карманные устройства, мобильные телефоны и браузеры Брайля, сильно зависят от хорошей разметки и неоправданно перегружаются, когда им приходится иметь дело с презентационным балластом.

На рис. 7.3-7.5 вы видите три скриншота сайта The Web Standards Project (www.webstandards.org). На рис. 7.3 показан сайт в традиционном браузере с включенной таблицей стилей, на рис. 7.4 - браузер с выключенной таблицей стилей, а на рис. 7.5 - сайт в том виде, в каком он отображается на карманном компьютере. Везде показана одна и та же страница. Навигация сайта - список с самой простой разметкой из возможных в XHTML.

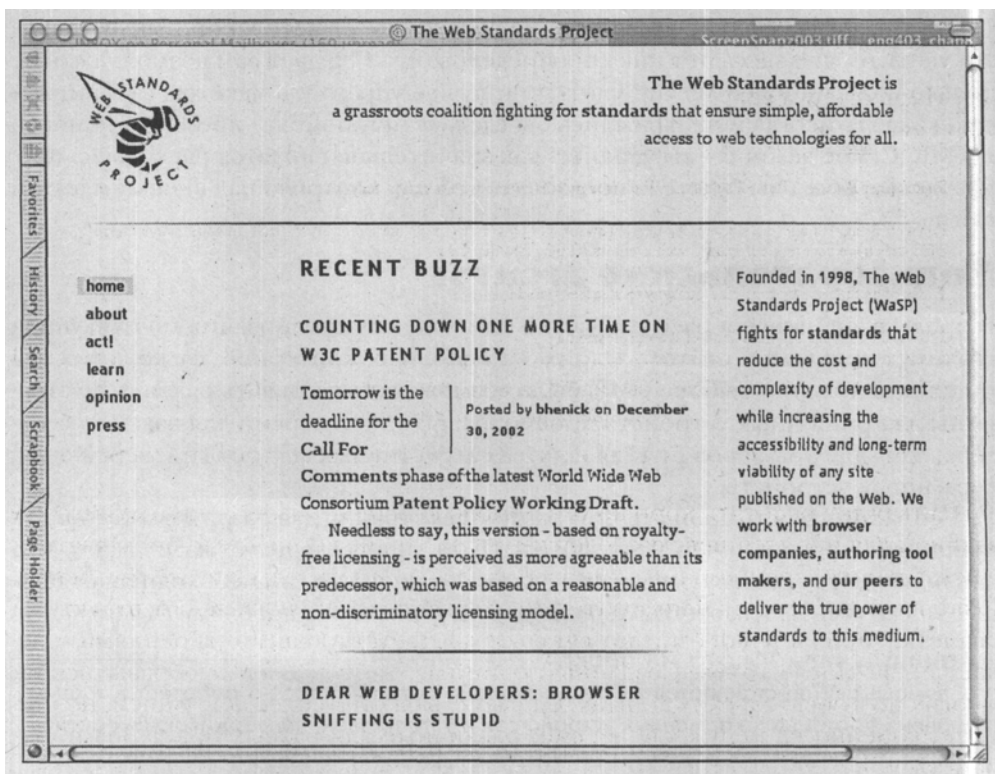


Рис. 7.3. TheWeb Standards Project (www.webstandards.org) повлиял на внедрение поддержки стандартов для Web-браузеров. С 1998 года авторы сайта убеждали производителей браузеров, разработчиков программного обеспечения и Web-дизайнеров поддерживать Web-стандарты. Этот сайт - один из лучших, где можно найти самые свежие новости из мира Web-стандартов

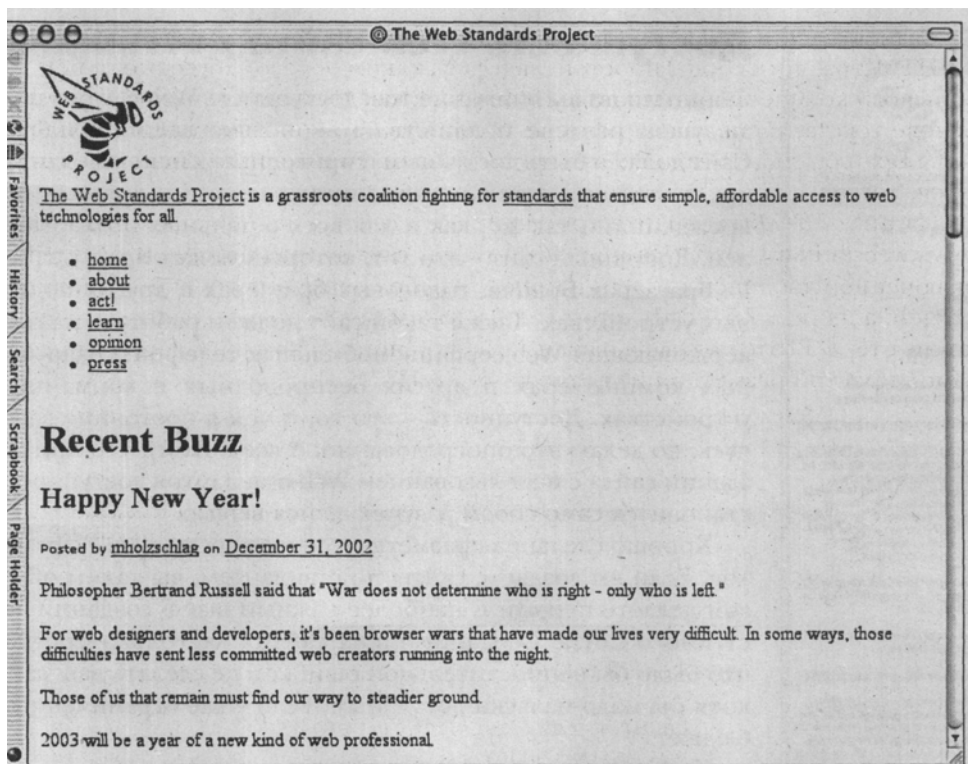


Рис. 7.4. Здесь вы видите, как отображается строгий HTML, когда таблицы стилей выключены

Сайт будет работать хорошо и с помощью Screen Reader - программы, созданной для прочтения вслух содержания на экране компьютера пользователя с ослабленным зрением. Как и многие другие пользовательские агенты, Screen Reader предоставляет пользователю содержание, отталкиваясь от структуры разметки. Screen Reader читает список с соответствующими ударениями и делает паузы после точек. Созданный один раз, сайт должен отображаться на многих пользовательских агентах. Сделайте ваш сайт более доступным, не тратя ни копейки на ненужные версии, различные для разных браузеров и устройств.

Я привела список для навигации в качестве примера, но это всего лишь вершина айсберга. Во второй части данной главы идея «Создай один раз - используй везде» излагается более подробно.

Доступность Web-сайта достижима

Обратили ли вы внимание, как доступность Web-сайта в предыдущем разделе повлияла на экономию ваших средств? Сайт должен быть доступным и пригодным к использованию для людей со зрительными, двигательными и другими повреждениями так же, как и для всех остальных пользователей. Доступный сайт - это тот, который может прочитаться на браузерах Брайля, голосовых браузерах и других подобных устройствах. Также такой сайт должен работать на поддерживающих Web-серфинг мобильных телефонах, карманных компьютерах и других беспроводных и карманных устройствах. Доступность - это то, о чем я постоянно беспокоюсь, но делаю это опосредованно. А все потому, что при создании сайта с использованием Web-стандартов доступность становится само собой разумеющейся вещью.

Хорошо сделанная разметка - основа доступных Web-сайтов. Если вы создаете сайты по стандартам, вы таким образом делаете первый и наиболее важный шаг в создании доступного сайта. Создание таких сайтов настолько просто, что было бы непростительной ошибкой не сделать ваш сайт хотя бы мало-мальски доступным, если у вас ограниченный бюджет.

Клиенты и начальники часто очень неохотно соглашаются на создание доступных сайтов. Это происходит потому, что очень многие люди, принимающие решения, ошибочно полагают, что доступность делает производство сайта более дорогим.

Вы, как Web-разработчик, должны знать о том, что, используя понятные стандарты и практикуясь, достигнете элементарного уровня доступности без дополнительных усилий. Учтите, что ваш начальник откажется давать дополнительные средства. У вашего сайта, созданного по Web-стандартам, будет по крайней мере минимальная доступность, и все это без дополнительных расходов.

Подумайте о том, как обрадуются ваши заказчики, когда они наконец начнут заботиться о доступности (если они еще не начали, то скоро начнут) и обнаружат, что ваш сайт является как раз доступным. Когда они поймут, сколько сэкономлено на том, что сайт не придется переделывать с самого начала, это наверняка положительно отразится на вашем благосостоянии.

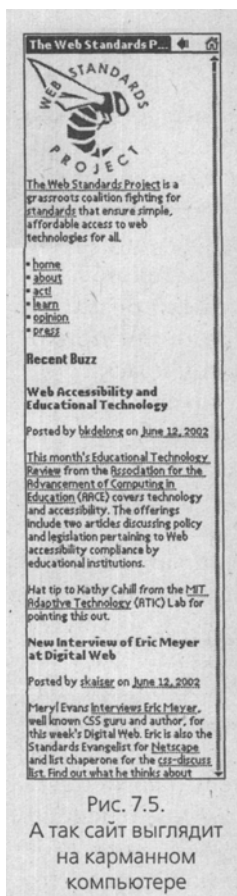


Рис. 7.5.

А так сайт выглядит на карманном компьютере

Создайте сайт с хорошей разметкой

Согласитесь, что, помимо доступности и экономических преимуществ создания грамотного XHTML-кода, всегда здорово потрудиться над чем-то, что приносит и пользу и удовольствие одновременно. В моей жизни был период, когда я отыскивала и приводила в порядок старую мебель. Если субботнее утро на блошином рынке проходило особенно хорошо, я возвращалась домой с великолепно сделанным столом или стулом. Мое приведение в порядок обычно ограничивалось чисткой и незначительной починкой найденных предметов мебели. Если я приходила домой с чем-то, что было сделано плохо, мне приходилось решать структурные проблемы и браться за клей, гвозди и краску.

Создать сайт с хорошей разметкой - это все равно что добротнo сделать стул. Сделайте его хорошо, и этот стул будет вас радовать долгие годы. Даже если каждые несколько лет вам приходится менять обивку, чтобы стул не выглядел устаревшим, вам никогда не придется чинить его с помощью гвоздей и клея.

Если такие ситуации не уязвляют ваше самолюбие, то, возможно, вас предостережет от них хотя бы чувство стыда за некачественно сделанные вещи. Несколько лет назад мои друзья заказали себе деревянную веранду, которую собирались пристроить к задней части дома. Они думали, что так их дом будет выглядеть лучше, и огорчились, когда эта веранда развалилась через год. Подрядчик использовал плохую древесину для строительства, неустойчивую к дождям, снегу и палящим солнечным лучам. Мои друзья предъявили подрядчику претензии, и ему пришлось заплатить штраф.

Трудно сказать, к какому виду ответственности могут привлечь нас, Web-работчиков, тем не менее мы должны быть ответственными за знание и использование самых передовых технологий в своем деле. Незнание Web-стандартов ставит нас в один ряд с подрядчиком, который построил дряхлую веранду. Это как минимум плохой пример. Конец урока.

Механика хорошего XHTML-кода

Прежде чем насладиться экономией бюджета за счет хорошего XHTML-кода, необходимо потратить немало сил на создание структурной разметки. Если вы еще не работали над такими разметками, вам придется сделать несколько простых механических настроек и одну, более интересную, емкую настройку.

Для того чтобы привыкнуть к этому новому (на самом деле довольно старому) подходу к разметке, вам, возможно, понадобится некоторое время. Не удивляйтесь тому, что поначалу продуктивность работы будет не очень высокой. Не зря говорят: тяжело в учении - легко в бою. Потом вам будет гораздо проще создавать и переделывать сайты. Вы будете гораздо меньше работать с HTML;

будете писать несколько строк разметки, так чтобы они выполняли всю оставшуюся работу. В результате у вас получатся страницы, которые хорошо работают во всевозможных браузерах и других устройствах просмотра.

Для того чтобы написать хорошую разметку, необходимо сделать пять простых механических вещей:

1. XHTML должен начинаться с тегов, которые говорят браузеру о том, как их надо интерпретировать. Начните страницу с объявления типа документа (Document Type Declaration ~ DTD) и с XHTML-объявления пространства имен.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd" ">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">
```

2. Пишите теги строчными буквами. Если у вас привычка писать <H1>, <BODY> и тому подобное, привыкайте писать <h1> и <body>. Многие HTML-редакторы имеют установку, которая позволяет писать теги строчными буквами по умолчанию.

P Определение

Объявление типа документа (Document Type Declaration) сообщает браузеру, какое определение типа документа (Document Type Definition ~ DTD) вы используете. Определение типа документа - это документ, который представляет собой список элементов и атрибутов, которые могут использоваться в SGML, XML или HTML-документах. Определение типа документа описывает, где каждый тег разрешен и какие теги могут появиться внутри других тегов. У каждой версии XHTML есть определение типа документа. Web-разработчики могут выбрать один из типов документов: XHTML Transitional (Переходный), XHTML Strict (Точный) и, если вы хотите разделить окно браузера на два или более фрейма, XHTML Frameset (Установка фреймов).

-Определение

Пространства имен (Namespaces) - это способ определить или квалифицировать элементы и атрибуты, используемые в XML, HTML или других документах разметки. Поскольку элементы и атрибуты могут приобрести двойной смысл в различных типах документов и программном обеспечении, вам нужно задать конкретное пространство имен, которое вы используете. Возможно, это покажется вам слишком нудной процедурой, но имейте в виду, что управление этими именами поможет предохранить ваши элементы от неправильного интерпретирования браузерами и устройствами просмотра. Это трудно понять, но легко сделать. Просто вставьте эту строку.

3. Возьмите все атрибуты в кавычки. Возможно, это лишняя перестраховка - все равно что держать закрытыми двери в шкафу. Раньше вам не приходилось брать в кавычки атрибуты ширины (width) и высоты (height). В HTML это было совершенно нормально:

```

```

XHTML требует, чтобы эти атрибуты были в кавычках, вот так:

```

```

4. Закройте все теги. Это будет самым трудным для выполнения изменением. Если у вас сформировались плохие привычки, как у большинства из нас, вам придется стать дисциплинированнее. Если у вас стоит открытый тег <body>, вам придется его закрыть тегом </body>. Если у вас открытый тег абзаца <p>, закройте его тегом </p>.

Необходимо знать одну хитрость при закрытии тегов: вы должны закрыть даже те теги, у которых нет отдельных закрывающих тегов. Такие теги называются элементами-одиночками, потому что они содержат только один тег; сюда входят метатеги, разрывы строки и теги изображений.

Определение

Метатеги (Meta Tags) в заголовке документа описывают документ. Метатеги содержат тему, автора, ключевые слова и другие описательные элементы о странице. Поисковые машины и индексаторы используют эти данные для того, чтобы описывать и ссылаться на сайт. Ниже приводится несколько метатегов, взятых с моего личного сайта и правильно созданных для XHTML с использованием закрывающего пробела и слэша:

```
<meta name="author" content="Кэрри Бикнер." />
```

```
<meta name="description" content="Web-дневник о работе в библиотеке, Web-стандартах и библиографических радостях.
```

```
Содержит библиотечные истории от читателей и фотографии Нью-Йорка." />
```

```
<meta name="keywords" content="Кэрри Бикнер,, фотографии, Web-дневник, сплетни, стиль, библиотеки, доступность, Web-стандарты, Нью-Йорк." />
```

В старые добрые времена HTML 4.0 и более ранних версий тег новой строки выглядел вот так:

```
<br>
```

В XHTML он должен быть закрытым. Добавьте к тегу пробел и слэш:

```
<br />
```

Тег изображения в ранних версиях HTML выглядел следующим образом:

```

```

Если вы используете язык XHTML, вам придется добавить пробел и слэш, прежде чем закрыть тег:

```

```

5. Проверяйте ваши страницы. Поначалу это может вас испугать, но вскоре вы поймете, что процесс проверки полезен, даже увлекателен. Просто напишите URL-адрес страницы, над которой вы работаете, в странице HTML-валидатор (контролер) на сайте W3C (<http://validator.w3.org>), как показано на рис. 7.6; нажмите кнопку **Submit** (Отправить) и узнайте, насколько хорош ваш XHTML-код. Этот инструмент обнаружит большинство XHTML-ошибок. Когда вы научитесь выявлять ошибки в CSS, этот инструмент станет вашим лучшим другом.

• Примечание:

Если вам кажется, что валидатор W3C работает медленно, используйте валидатор Web Design Group, который можно найти на сайте www.htmlhelp.com/tpols/validator.

При исправлении первых нескольких страниц у вас появится такое понимание разметки, какого, возможно, никогда и не было. В отчете валидатора на рис. 7.6 пункты 2, 3 и 4 сообщают о том, что элемент страницы `body` содержит нежелательные элементы `topmargin`, `leftmargin`, `marginwidth` и `marginheight`. Говоря простым языком, это означает, что тег `body` содержит четыре характеристики (атрибута) полей, которые не используются в текущей спецификации XHTML.

Вот плохая разметка:

```
<body bgcolor="#ffffff" topmargin="0" leftmargin="0"
marginwidth="0"
marginheight="0" onload="preloadImages();">
```

А вот хорошая разметка, которую вы напишете, прочитав отчет валидатора:

```
<body onload="preloadImages();">
```

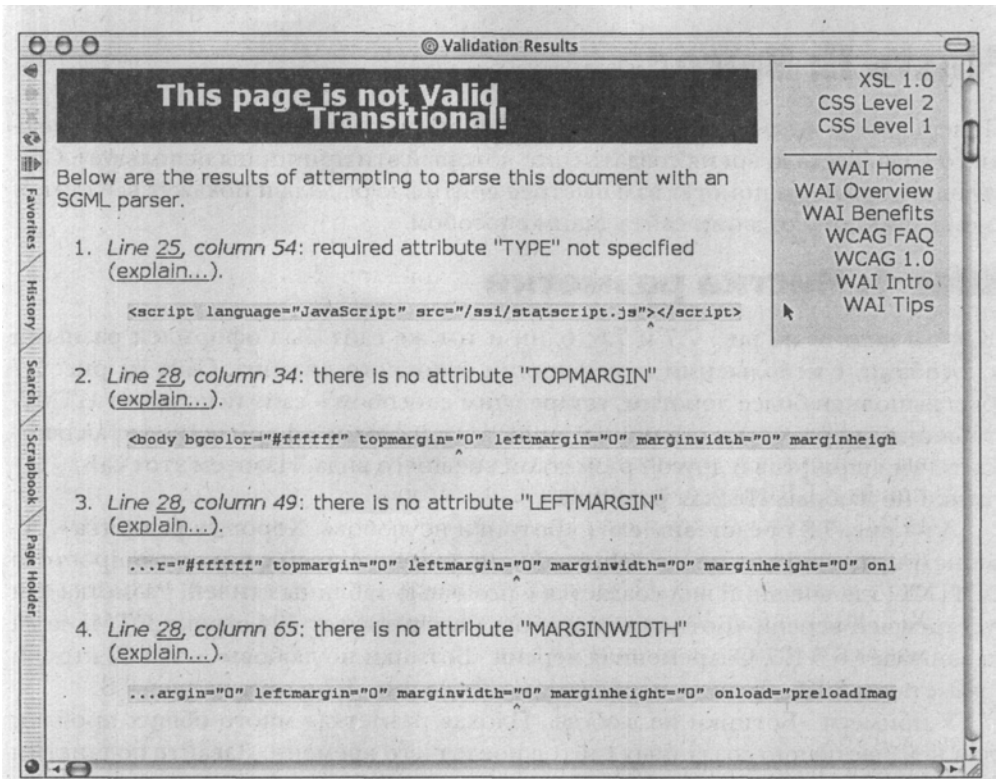


Рис. 7.6. Валидатор W3C - это инструмент, который проверяет, насколько правильно сделана ваша разметка. Он проверяет каждую строчку и сообщает о допущенных в разметке ошибках. При первом использовании валидатора вы его возненавидите; у вас сложится впечатление, что вы совсем не умеете писать разметку. Но по мере исправления ошибок на вашем сайте вы полюбите этот инструмент. Он станет вашим другом и помощником

Эти ненужные атрибуты тега `body` - теги, введенные Microsoft и Netscape, которые никогда не являлись частью формальной спецификации W3C. В качестве других ошибок могут быть выявлены устаревшие элементы и атрибуты - они были в предыдущей версии HTML, но сейчас от них отказались.

Устаревшие атрибуты тега `body`, такие как `marginwidth`, занимавшиеся управлением представления сайта, прекрасно и более экономично поддерживаются CSS. Вместо того чтобы тратиться на внесение этой разметки на каждую отдельную страницу, вы можете написать строчку в таблице стилей, которая позаботится об оформлении на всем сайте.

Часть II: Выгода

Теперь, когда вы познакомились с принципами отделения структуры от внешнего вида, настало время создать сайт, который эти принципы использует. Следующие примеры помогут вам быстрее войти в курс дела и покажут все статьи экономии при создании сайта таким способом.

Шаг 1: Чистка разметки

Как вы видите на рис. 7.7 и 7.8, один и тот же сайт был оформлен разными способами, с небольшими изменениями основного дизайна. Сайт на рис. 7.7 был выполнен более дорогим, устаревшим способом - сайт использует HTML-таблицы для разметки страницы, а цвета и текстовое оформление управляются тегами шрифтов и другой разметкой внешнего вида. Назовем этот сайт «Ботинки не любовь. Плохая разметка».

А на рис. 7.8 представим сайт «Ботинки не любовь. Хорошая разметка», демонстрирующий тот же самый дизайн, но выполненный с помощью простого XHTML, где внешний вид создается с помощью таблицы стилей. Разметка для устаревшей версии «Ботинки не любовь» насчитывает 104 строки HTML-кода и занимает 6,5 Кб. Современная версия «Ботинки не любовь» - это 62 строки разметки и 3 Кб. Давайте переделаем сайт на рис. 7.7 в сайт на рис. 7.8.

У примера «Ботинки не любовь. Плохая разметка» много общих проблем для некачественно созданных сайтов последнего времени. Давайте подчистим их один за другим.

г- Примечание о типах документов XHTML

Существует три вида XHTML 1.0: Strict (Точный), Transitional (Переходный) и Frameset (Установка фреймов). Во многих примерах данной книги используется DTD Точный, потому что он обеспечивает более четкое разделение структуры и внешнего вида и потому что он лучше подходит для того, чтобы предугадать будущие версии XML и XHTML. Воспользовавшись этим разделением и будущей совместимостью, вы можете значительно сэкономить. Если вы не можете отказаться от создания дизайна для страниц Netscape 4.0 или вы не готовы к быстрым изменениям, предлагаемым в этой главе, возможно, вы не сможете отделить структуру от внешнего вида так решительно, как сделала это я. Netscape 4.0 фрагментарно поддерживает CSS, особенно работу, которую CSS выполняет по макетированию (размещению). Если вас это устраивает, вы можете использовать подобный HTML 4.0 вид XHTML - Transitional XHTML. Вы также можете использовать таблицы для позиционирования по крайней мере некоторых частей вашей Web-страницы.

[- Примечание о типах документов XHTML (окончание) —————

Если вам приходится использовать таблицы, старайтесь чтобы их было немного. Не применяйте таблицы для компоновки страницы:

```
<table width="75%" border="1">
  <tr>
    <td>Какое-то содержание оформлено в виде таблицы. Это не очень
    хорошая идея.
      <table width="75%" border="1" align="left">
        <tr>
          <td>ВарнаНТ расположения одной таблицы внутри другой тоже не годится.</
          td>
        </tr>
      </table>
    </td>
  </tr>
</table>
```

Браузеры Netscape 4.0 обладают достаточно хорошей поддержкой CSS-описаний полей и отступов, поэтому вам совершенно ни к чему задавать этот аспект внешнего вида с помощью HTML

Online-руководство Нью-Йоркской публичной библиотеки по стилю (www.nypl.org/styleguide) - это учебник, в котором реализована идея поддержки CSS-описаний полей и отступов. Джефри Зельдман и я создали это руководство для Нью-Йоркской публичной библиотеки потому, что администрацию библиотеки, как и многих некоммерческих и образовательных организаций, было трудно убедить в необходимости перехода от Netscape 4.0 к более современному браузеру^ Если сейчас вы нуждаетесь в более легком способе перехода к современным стандартам, используйте это руководство, которое поможет вам упростить переход к отделению структуры от внешнего вида. В «Дизайне по Web-стандартам» Зельдмана более детально описывается, как организации могут использовать Этот подход, чтобы добиться более высокой совместимости с различными браузерами и доступности пользователям за невысокую цену.

Некоторые CSS-оформления без таблиц работают в Netscape 4.0, и эти компоновочные планы позволяют вам использовать тип документа «XHTML Strict» и продолжать радоваться посетителям, использующим Netscape 4.0. Возможно, вам больше придется по душе сделать так, чем использовать XHTML Transitional. Но достаточно скоро вам придется побеспокоиться насчет Netscape 4.0; со временем вам захочется разделить структуру и внешний вид насколько это возможно, чтобы все грядущие переделки сайта стоили дешевле. Крейг Сэйла (Craig Salla) составил перечень CSS-компоновок, которые работают в Netscape 4.0 (смотрите www.sailo.com/usage/layouts/nn4-layouts.shtml), а Марк Ньюхаус (Mark Newhouse) предоставляет некоторые из таких компоновок для вашего использования Hcrwww.realworldstyle.com.

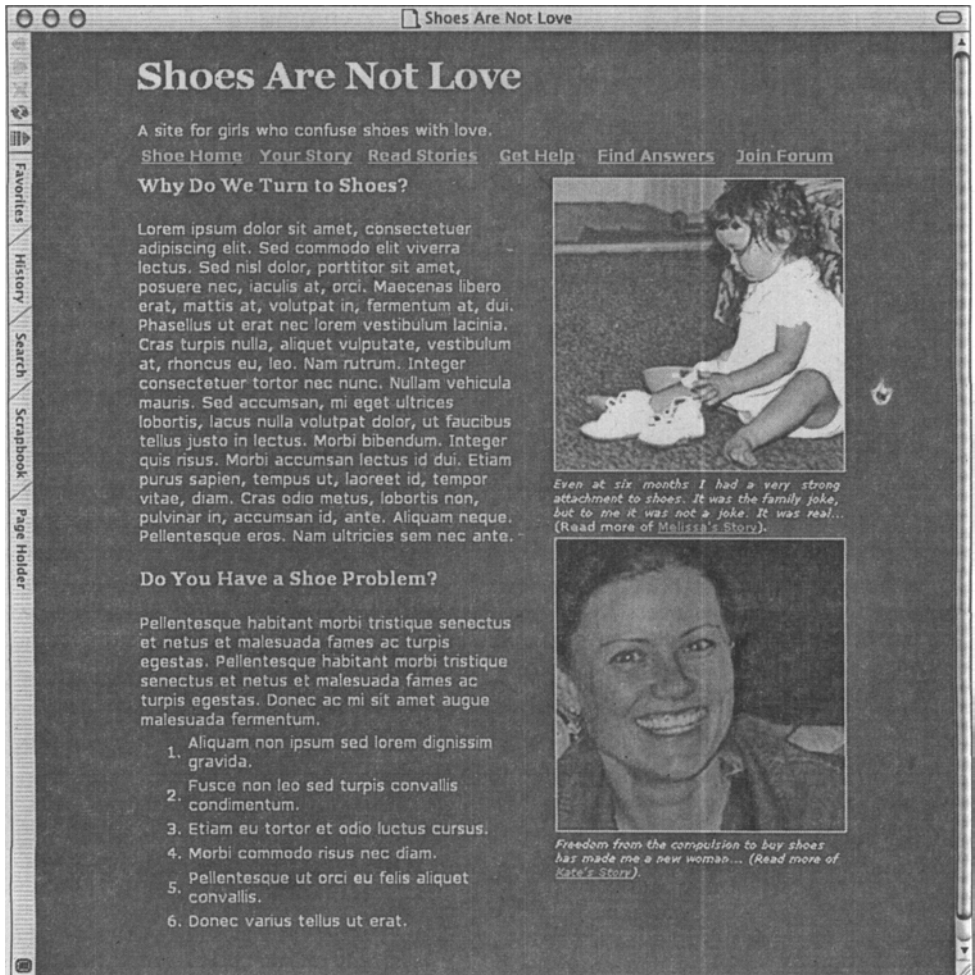


Рис. 7.7. Сайт «Ботинки не любовь. Плохая разметка»

Почистите разметку заголовка

Первое, что необходимо сделать, - это почистить заголовок HTML-файла. Листинг 7.4 показывает устаревший, плохой заголовок, а листинг 7.5 - новый, улучшенный заголовок HTML-файла.

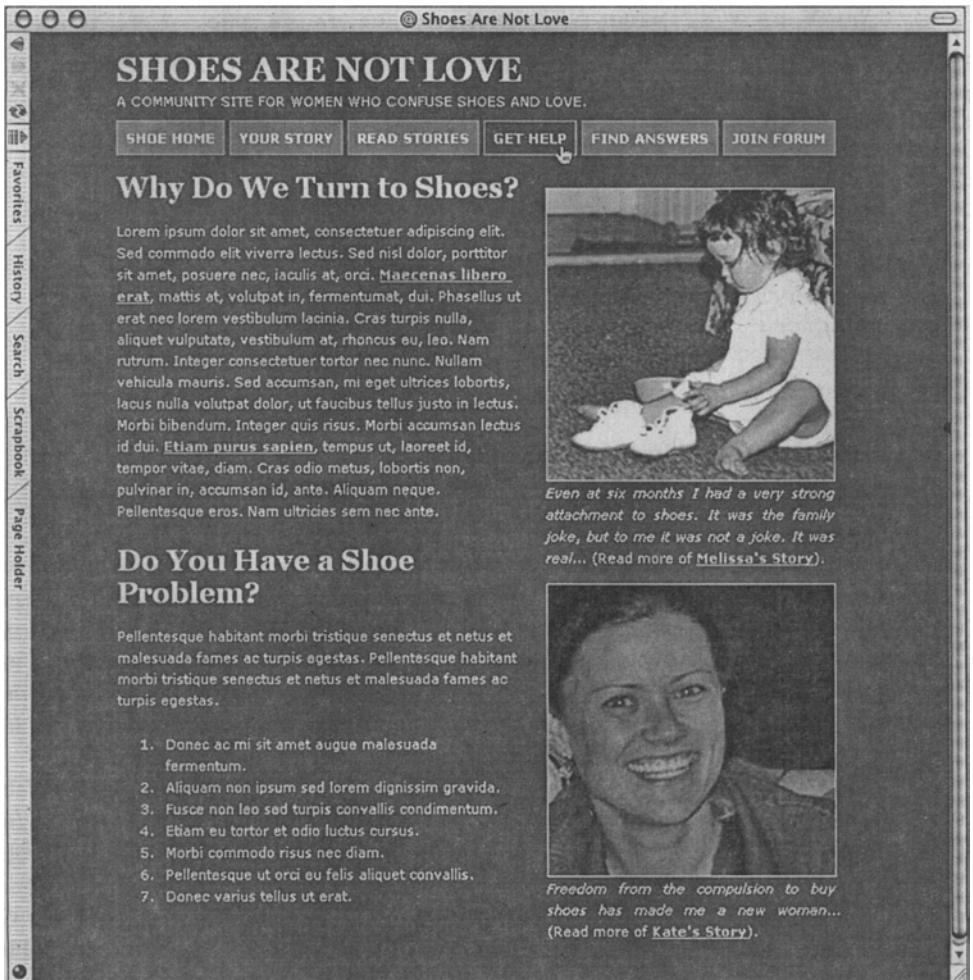


Рис. 7.8. Сайт «Ботинки нелюбовь. Хорошая разметка»

Листинг 7.4. Разметка заголовка сайта на рис. 7.7

```

<html>
<head>
<title>Ботинки нелюбовь</title>
</head>

```

Листинг 7.5. Разметка заголовка сайта на рис. 7.8

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://
www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1" /> .
<title>ВоТМНКМ не нioбoBb</tit:le>
<link rel="stylesheet" media="screen" title="Колонка 2 -
навигация слева" type="text/css" href="/styles/basic.ess" />
<style type="text/ess" media="all">@import */styles/fancy.ess";</
style>
</head>
```

Обратите внимание, что добавлено было следующее:

- XHTML-объявление типа документа в строке 1 и XML-пространство имен в строке 2 (листинг 7.5);
- метатег, который сообщает пользовательскому агенту, с каким набором знаков он должен работать. Тот факт, что разметка в листинге 7.5 длиннее, чем в листинге 7.4, сперва* может немного испугать. Ведь в конце концов Web-стандарты призваны уменьшать потребление трафика, а не увеличивать его дополнительными тегами и атрибутами. В самом файле отчищенная разметка намного короче разметки старого образца, но добавление объявления типа документа делает заголовок длиннее;
- тег <title> для сайта. Он присутствует и на рис. 7.7, и теоретически Web-дизайнеры старой школы знают о том, что тег <title> всегда необходимо включать в листинг. Но если бы мне давали по центу за каждую Web-страницу без тега <title>, которую я видела, я бы не стала писать книгу о малом бюджете. Тег <title> помогает поисковым машинам и указателям отыскивать и сортировать нужным образом содержание вашего сайта. Не пренебрегайте этим, иначе потеряете потенциальных посетителей;

Г Определение

Набор знаков (Character Set) - это определенный список букв и символов, узнаваемых устройствами и программным обеспечением. Каждой букве соответствует номер. Существует множество наборов знаков, поэтому для того, чтобы Web-страницы были правильно «поняты» браузерами и другими устройствами отображения, набор знаков должен быть объявлен.

- ссылка на две таблицы стилей. Вы можете назвать их как угодно; здесь они называются `basic.css` и `fancy.css`. Важно то, как вы вызываете эти таблицы стилей. Строка 6 вызывает таблицу стилей с помощью ссылки, а строка 7 - таблицу стилей методом импорта;
- вот почему вам это должно понравиться: ранее я упомянула, что Netscape 4.0 обладает фрагментарной поддержкой таблиц стилей. Поддержка настолько фрагментарная, что в действительности некоторые элементы CSS могут вызвать сбой Netscape 4.0. Однако Netscape 4.0 не понимает метод импорта, применяемый для вызова таблиц стилей. Поэтому, добавляя на ваш сайт что-то, без чего Netscape 4 было бы лучше обойтись, добавляйте такой стиль в CSS-документ, который вызывается методом импорта;
- закрытие элемента `<head>`.

Почистите тело документа

Теперь, когда есть приятный, чистый заголовок, можно обратиться к телу документа. Некоторые инструменты помогут вам это сделать. HTML Tidy - прекрасно для этого подходит (<http://tidy.sourceforge.net>). У Dreamweaver MX также есть инструменты для проверки и чистки разметки, которые работают, если вы используете объявление типа документа XHTML (www.macromedia.com/software/dreamweaver/).

Из листинга 7.6 видно, что сайт «Ботинки не любовь» оформлен при помощи тегов шрифтов. Обратите внимание, насколько громоздкой делают страницу теги шрифтов.

Листинг 7.6. Разметка тела сайта на рис. 7.7

```

1. <h1xF0NT COLOR=#FFFFFF SIZE=3 face=Georgia, Times New Roman,
Times,
2. serif>ЗАЧЕМ НАМ НУЖНЫ БОТИНКИ?</FONT></hi>
3. <PxF0NT COLOR=#FFFFFF SIZE=-1 face=Verdana, Arial,
Helvetica, sans-
4. serif> Lorem ipsum dolor sit amet, consectetur adipiscing
elit. Sed
5. commodb elit viverra lectus. Sed nisi dolor, porttotor sit
amet, posuere
6. nee, iaculis at, orci. <a href="erat.html"> Maecenas libero
erat</A>,
7. mattis at, volutpat in, fermentum at, dui. Phasellus ut erat
nee lorem

```

```

8. vestibulum lacinia. Cras turpis nulla, aliquot vulputate,
vestibulum at,
9. rhoncus eu, leo. Nam rutrum.. Integer consetetuer tortor nec
nunc. Nullam
10. vehicula mauris. Sed accumsan, mi eget ultrices lobortis,
lacus nulla
11. voluptat dolor, ut faucibus tellus justo in lectus. Morbi
bibendum.
12. Interger quis risus. Morbi accumsan lectus id dui.
13. <a href="sapien.html">etiam purus sapien</a>, tempus ut,
laoreet id,
14. tempor vitae, diam. cras odio metus, lobortis non, pulvinar
in, accumsan
15. id, ante. aliquam neque. pellentesque eros. nam ultricies
sem nec
16. ante.</FONT>
17. <h2><FONT COLOR=#FFFFFF SIZE=3 face=Georgia, Times New
Roman,
18. Times, serif>ЕСТЬ ЛИ У ВАС ПРОБЛЕМЫ С БОТИНКАМИ?</FONT></h2>

```

Листинг 7.7 – это практически тот же самый сайт, но эта версия оформлена с помощью CSS. Ссылка на CSS – это всего лишь одна строчка в заголовке документа (не показана здесь), а остальная часть разметки может спокойно выполнять только структурную работу. Обратите внимание, насколько меньше этот кусок разметки.

Листинг 7.7. Разметка тела сайта на рисунке 7.8

```

1. <h2>ЗАЧЕМ НАМ НУЖНЫ БОТИНКИ?</h2>
2. <p>Lorem ipsum dolor sit amet, consetetuer adipiscing elit.
Sed commodo
3. elit viverra lectus. Sed nisl dolor, porttotor sit amet,
posuere nec,
4. iaculis at, orci. <a href="erat.html"> Maecenas libero erat</
a>, mattis at,
5. volutpat in, fermentum at, dui. Phasellus ut erat nec lorem
vestibulum
6. lacinia. Cras turpis nulla, aliquot vulputate, vestibulum at,
rhoncus eu,
7. leo. Nam rutrum. Integer consetetuer tortor nec nunc. Nullam
vehicula
8. mauris. Sed accumsan, mi eget ultrices lobortis, lacus nulla
voluptat

```

```

9. dolor, ut faucibus tellus justo in lectus. Morbi bibendum.
Interger quis
10. risus. Morbi accumsan lectus id dui. <a
href="sapien.html">Etiam purus
11. sapien</a>, tempus ut, laoreet id, tempor vitae, diam. cras
odio metus,
12. lobortis non, pulvinar in, accumsan id, ante. aliquam neque.
Pellentesque
13. eros. Nam ultricies sem necante. </p>
14. <h2>ЕСТЬ ЛИ У ВАС ПРОБЛЕМЫ С БОТИНКАМИ?</h2>

```

Подчистить разметку просто, а когда вы достаточно попрактикуетесь, станет еще легче. Выполните следующие четыре простых шага:

1. Удостоверьтесь, что все XHTML-теги написаны строчными буквами. Вы можете сделать установку, чтобы HTML-редактор по умолчанию создавал разметку строчными буквами. Даже если вы не сразу перейдете от HTML к XHTML, вам следует начать писать разметку строчными буквами, чтобы выработать привычку и чтобы ваши страницы было проще переделывать, когда настанет время. В листинге 7.6 теги написаны строчными и прописными буквами вперемешку; это плохо. В листинге 7.7 все теги написаны строчными буквами; это хорошо.
2. Возьмите все атрибуты в кавычки. Большинство атрибутов в листинге 7.6 исчезнут, когда вы удалите из кода разметку внешнего вида; вы избавитесь от атрибутов шрифтов и размера. Другие атрибуты, как, например, атрибут href и элемент ссылки в листинге 7.6, нуждаются в кавычках, как это сделано в листинге 7.7.
3. Закройте все элементы. Правильнее всего говорить об этом, используя термины «начало» и «конец», поэтому будем следовать этому правилу. В листинге 7.6 есть начальный тег абзаца в строке 3, который должен закрываться в строке 16, но не закрывается. Стыд и срам! Листинг 7.7 заканчивается аналогичным тегом конца абзаца в строке 13.
4. Уберите всю разметку внешнего вида из вашего HTML-кода. Вы уже знаете, что вам нужно сделать такую вещь, как удаление презентационных атрибутов, подобных показанным ниже, из тега <body>:

```

<body bgcolor="#333333" text="#FFFFFF" link="#6699FF"
vlink="#6699FF">

```

Все, что вам надо, - это простой тег <body>, вот такой:

```

<body>

```

Вы также знаете о том, что вам необходимо удалить теги шрифтов, такие как в первых четырех строках и последних трех строках листинга 7.6.

Удаление этого балласта - лишь начало избавления вашего сайта от разметки внешнего вида. Также вы должны уничтожить HTML-таблицы.

Таблицы стоят слишком дорого

Одна из самых больших затрат трафика и времени - это HTML-таблицы. Старая версия сайта «Ботинки не любовь» (листинг 7.8) использует несколько таблиц для управления компоновкой страницы. Некоторые таблицы, как, например, панель для фотографий слева, вложены в другие HTML-таблицы. Хотя во времена примитивных браузеров вам и приходилось создавать сайты таким способом, сейчас существует хороший способ, как не использовать эту технику.

Эта глава постоянно напоминает об идее отделения структуры от внешнего вида; использование таблиц не позволяет этим частям разъединиться. Структурно HTML-таблицы созданы для данных. Подразумевается, что они должны содержать информацию, например список имен и телефонных номеров служащих или перечни продуктов и цен.

Листинг 7.8. Разметка для навигации на рис. 7.7

```
<tr>
<td wigth=16%xddiv align=center><strongxfont size=-1
face=verdana, arial, helvetica, sans-serifxa
href=two.html>Ботинки - rnaВHaН</ax/fontx/strongx/divx/td>
<td wigth=16%xddiv align=centerxstrongxfont size=-1
face=verdana, arial, helvetica, sans-serif><a
href=index.html>Вамa Историж/ax/fontx/strongx/divx/td>
<td width=17%xddiv align=centerxstrongxfont size=-1
face=verdana, arial, helvetica, sans-serif><a
href=two.1^t1>Читайте Исторшк/ax/fontxstrongx/divx/td>
<td width=15%xddiv align=center><strongxfont size=-1
face=verdana, arial, helvetica, sans-serif><a
href=onea.пШ1>Получите Помощь</ax/£on1><зIгодцх/clly></1(5>
<td width=18%xddiv align=centerxstrongxfont size=-1
face=verdana, arial, helvetica, sans-serif><a
href=two.html>НарIAMTe OTBeTби</ax/fontxstrongx/divx/td>
<td width=18%xddiv align=centerxstrongxfont size=-1
face=verdana, arial, helvetica, sans-serif><a
href=three.html>Почувствуйте в OopyMe</ax/fontxstrongx/divx/
td>
</tr>
```

При ограниченном бюджете большое значение имеет тот факт, что таблицы отнимают невероятно много времени на создание и переделку. Предположим, что вам захотелось поменять навигацию сайта на рис. 7.7, изменив пункт **Ваша история** на пункт **Поделитесь своей историей**. Учитывая, что каждый пункт меню находится в своей собственной ячейке таблицы, вы были бы должны изменять ширину каждой ячейки, в которой находятся пункты меню. Если бы вам понадобилось добавить новую фотографию и заголовок в панель слева, вам бы снова пришлось редактировать таблицы. Каждая фотография и заголовок в панели слева находятся в отдельной ячейке таблицы, поэтому, если бы вам понадобилось добавлять новые фотографии и заголовки в панель, вам пришлось бы создавать новые ячейки таблицы. Если вы хотите сэкономить время и деньги, не связывайтесь с таблицами.

На рис. 7.8 верхняя навигация была извлечена из таблицы и приобрела другую структуру: HTML-список (листинг 7.9). Если бы вы захотели сменить **Вашу историю** на **Поделитесь своей историей**, вам пришлось бы просто изменить этот пункт списка.

Листинг 7.9. Разметка навигации на рис. 7.8

```
<div id ="menu">
<li a href=" index.html ">Ботинки - Главнаж/ах/Н>
<li a href="/story/">Ваша историж/ах/П>
<li a href="/read/">Читайте истории<ах/л_>
<li a href="/help/">Получите помощь<ах/П>
<li a href="/answers/">Найдите ответы<ах/И>
<li a href="/forum/">Поучаствуйте в форуме<ах/И>
```

Очищенный и готовый к оформлению сайт

Когда вы закончите очистку сайта, у вас должна получиться страница, которая выглядит, как на рис. 7.9. Это совсем простая страница. Когда вы уберете балласт и создадите простой, хорошо сделанный (X)HTML-документ, вы сможете воспользоваться экономичной и удобной таблицей стилей.

Шаг 2: Создание таблиц стилей

Давайте создадим таблицу стилей, которая сделает на сайте «Ботинки не любовь. Хорошая разметка» такой же внешний дизайн, как на сайте «Ботинки не любовь. Плохая разметка». В главе 5 мы начали работать над таблицей для



Рис. 7.9. Перед созданием таблицы стилей сайт должен выглядеть, как простой старый HTML-сайт. На самом деле единственный способ убедиться в том, что вы провели хорошую работу по созданию разметки, - посмотреть, как сайт выглядит без таблиц стилей. Если сайт работает правильно, это знак того, что вы сделали страницу с основательной структурной разметкой. У вас будет хорошая возможность увидеть, что получают пользователи неграфических браузеров, таких как карманные компьютеры и браузеры Брайля

оформления текста (см. табл. 5.1). Можете использовать эту таблицу в качестве основы для создания своих таблиц стилей.

Таблицы стилей созданы из правил (rules), а правила составлены из селекторов (selectors), значений (values) и свойств (properties). Этот фрагмент текста - CSS-правило:

```
body {  
    color: #ffffff  
    background-color: #333333;  
    margin-left: 70px;  
    margin-right: 70px;}
```

Браузер знает, что это правило должно применяться по отношению к телу HTML-страницы благодаря первой строке - селектору тела (body selector). После селектора тела стоят фигурные скобки, которые говорят браузеру, что ему сейчас будут сообщены некоторые свойства. Первое свойство - это цвет (color) основного шрифта. Это свойство имеет шестнадцатеричное значение белого - #ffffff. Укороченная версия, #fff, может быть также использована для того, чтобы сэкономить несколько байтов трафика пользователя. Это очень просто. Удостоверьтесь в том, что вы используете двоеточие для отделения свойств от их значений. Некоторые люди используют знак равенства - это делается интуитивно и является ошибкой.

После свойства color в листинге указано еще несколько свойств стиля и их значения. Обратите внимание, что каждое свойство стиля отделено от своего «соседа» точкой с запятой. Вообще, последнее в правиле свойство стиля не нужно отделять точкой с запятой, однако использовать этот знак в данном случае все равно полезно в целях единообразного оформления листинга и во избежание ошибок, которые могут возникнуть, если в дальнейшем вы поменяете порядок, в котором перечислены свойства, или добавите другие свойства. Кроме того, в листинге есть свойство фона со значением #333333 (приятный темно-серый цвет), а также свойства ширины правого и левого полей, каждое из которых имеет значение 70 пикселей. Закрывающие фигурные скобки сообщают, что определение правила было закончено. Таким образом, теперь у вас есть документ с одним правилом, который делает сайт с белым текстом, темно-серым фоном и правым и левым полями по 70 пикселей.

Вы можете дополнять сайт, создавая новые правила и добавляя в них новые свойства и значения. Теперь, когда вы знаете, как называется каждая из частей CSS-документа, можете сделать такой документ для сайта «Ботинки не любовь».

Давайте создадим таблицу стилей, которая предложит нам такой же дизайн, как на рис. 7.8, с применением возможностей, которыми обладает только CSS. Начнем с простых вещей: цвета и оформления текста.

Цвет и оформление текста

Вам необходим темно-серый фон, белый текст и два вида шрифтов: Verdana для основного текста и Georgia для заголовков (листинг 7.10). Конечно, как обсуждалось в главе 6, если бы это была таблица стилей реального сайта, вам бы понадобились запасные шрифты, чтобы быть уверенным, что дизайн будет работать на различных платформах, включая те, которые не располагают шрифтами Verdana или Georgia. Поэтому включите и запасные шрифты в таблицу стилей.

Листинг 7.10. Таблица стилей для сайта на рис. 7.8

```

1. body {
2.   color: #ffffff;
3.   background-color: #333333;
4.   font-family: verdana, geneva, arial, sans-serif;
5.   margin-left: 70px;
6.   margin-right: 70px;}
7.
8. div, p, th, td, li, dd, dl, dt {
9.   color : #ffffff;
10.  font: 11px/18px verdana, geneva, arial, sans-serif;}
11.
12. h1, h2, h3, h4, h5, h6   {
13.  color : #ffffff;
14.  font-family: georgia, times new roman, times, serif;
15.  margin-bottom: 1px;}
16.
17.  hi   {
18.   font-size: 1.8em;
19.   text-transform: uppercase;
20.   text-align: justify;}

```

Строки 1-18 таблицы стилей basic.ess (листинг 7.10) задают цвет и параметры для оформления текста сайта (рис. 7.8).

Строка 1 начинается с правила для селектора тела. Строка 2 сообщает свойство цвета со значением #ffffff- иначе говоря, задает белый цвет. Строка 3 определяет цвет фона сайта (#333333, или темно-серый), а строка 4 - шрифты по умолчанию, такие как Verdana, Geneva, Arial или любой шрифт без засечек, который установлен в вашем компьютере.

Строки 5 и 6 добавляют к правилу тела сайта еще два свойства и, таким образом, берут на себя работу, которую раньше должны были выполнять таблицы:

они определяют края слева и справа от содержания сайта. Всего лишь две строки CSS выполняют работу, которую раньше делало множество HTML-таблиц. Это, дамы и господа, экономия.

Впрочем, CSS не полностью экономичны, по крайней мере, в том, что зависит от поддержки различных браузеров. CSS предлагают экономию, благодаря тому что свойства-потомки наследуют признаки от свойств-родителей. Правилу абзаца, например, следует унаследовать свойства и значения у правила тела. Таблица стилей не должна быть избыточной (то есть задавать одно и то же несколько раз). Однако наследование не всегда очевидная вещь для браузеров, особенно для Netscape 4.0. Поэтому в строке 8 длинное правило. Это правило для разделов, абзацев, заголовков таблиц, ячеек таблиц и других самых разнообразных элементов. Для различных браузеров вам следует подробно написать, что каждый из этих элементов должен отображаться белым, поэтому строки 9 и 10 определяют цвет, шрифт, размер шрифта для целой массы элементов. Все равно это намного эффективней, чем заворачивать каждую строку текста в тег шрифта.

Строки 12-15 определяют правило для заголовков. Эти свойства и значения сообщают браузеру, что заголовки должны отображаться белым шрифтом Georgia (или Times New Roman, или Times, в зависимости от имеющихся на компьютере пользователя шрифтов).

Значения и свойства - это все, что требуется. Вы можете добавлять какие угодно значения и свойства. Многие из этих значений и свойств делают вещи, которые простой старый HTML просто не мог сделать, потому что он никогда и не собирался это делать. Некоторые из этих свойств и значений описываются далее.

Поля, интерлиньяж и другие пустые пространства

Возможно, вы обратили внимание, что в строке 10 размер шрифта выражен как 11 пикселей/18 пикселей (11 px/18 px), а строка 15 определяет нижний отступ для всех заголовков как 1 пиксель (1 px). Здесь CSS выполняют тончайшую работу, которая просто не может быть сделана с помощью HTML, который используется для внешнего вида сайта. Строка 11 px/18 px сообщает браузеру о том, что текст должен отображаться размером 11 пикселей, но оставлять небольшой междустрочный пробел. Если вы вернетесь назад и сравните рис. 7.7 (старая школа, нет междустрочных пробелов) с рис. 7.8, вы увидите, насколько междустрочные пробелы улучшают читабельность текста в последнем примере. Также нижний отступ в 1 пиксель под элементами

заголовка позволяет удачно разместить текст относительно заголовка. Таким образом проще визуально идентифицировать абзац, чем если бы заголовок был сразу же над текстом. HTML-форматирование старой школы не может такого сделать.

Для получения более подробной информации относительно полей, отступов и междустрочных пробелов смотрите первую спецификацию Cascading Style Sheets на сайте W3C (www.w3.org/TR/REC-CSS1). Здесь многому придется учиться, но с этими знаниями вам будет намного лучше.

Преобразование текста

Давайте немного развлечемся, прежде чем перейдем от оформления текста к верстке страницы. При помощи CSS вы можете легко преобразовывать текст. Строка 17 содержит правило, которое определяет элемент `h1` более точно (листинг 7.10). Строка 19 содержит свойство `text-transform` со значением `uppercase` (заглавные буквы). Это сообщает Web-браузеру о том, что все элементы текста в `<h1>` должны быть отображены заглавными буквами. Вы можете использовать такое преобразование текста для того, чтобы, например, отобразить первые несколько слов абзаца буквами. Опять-таки, первая спецификация Cascading Style Sheets на сайте W3C расскажет вам о том, как использовать преобразование текста (см. www.w3.org/TR/REC-CSS1).

Верстка страницы

Теперь, когда мы разобрались с оформлением текста и цветом, давайте разберемся с компоновкой (версткой) сайта. Верстка - это просто. Для того чтобы создать макет страницы, необходимо разбить ее на три основных части: навигация, боковая панель и область основного текста. Затем нужно создать правила, которые разместят эти фрагменты в различных областях экрана.

Когда вы подчистите HTML-страницу, ваша навигация превратится в список. Теперь вы должны немного «поколдовать» над CSS, чтобы изменить отображение этого списка. Можно было бы создать правило для всех нумерованных списков, но тогда получится, что все они будут выглядеть, как навигация. А ваш сайт, возможно, содержит некоторые простые старые списки, которые не занимают навигацией. Поэтому вам надо создать специальное правило, которое распространяется исключительно на нумерованный список в навигационной области. Это делается благодаря созданию *идентификатора* (`identifier`), который позволяет таблице стилей создать специальный внешний вид для списка в навигации.

POпределение

CSS Селекторы класса (Class Selectors) и ID-селекторы (ID Selectors). CSS использует селекторы класса и ID-селекторы для того, чтобы отбирать и оформлять некоторые блоки HTML из Web-страницы. Работают они практически одинаково, но идентификаторы (они же ID-селекторы) используются только один раз на странице, а классы могут использоваться повторно. Используйте селекторы класса, когда вы хотите оформить элемент, который будет появляться на странице много раз, например абзац или заголовок. ID-селектор надо применять для единственного в своем роде элемента, например для панели навигации или логотипа.

ID-селектор в таблице стилей создается при помощи знака решетки (#). Синтаксическая структура выглядит следующим образом:

```
#someidentifier {
```

Таким образом можно задать любой элемент в HTML-документе, у которого есть id-атрибут:

```
<div id="someidentifier"></div>
```

Идентификаторы класса выглядят приблизительно так же, но вместо знака решетки здесь используется точка:

```
.someclass {  
}
```

Из HTML-документа берутся элементы с атрибутом class, а не с атрибутом id:

```
<div class="someclass"></div>
```

Селекторы могут и должны использоваться для всего сайта. Старайтесь давать селекторам значимые названия - так вам будет проще их запомнить. Когда вы создаете верхнюю и боковую навигацию, используйте селекторы с именами `trpnav` и `sidenav`. Один мой коллега давал селекторам изящные, но ничего не значащие имена, например `rimprkinbox`. Если бы за каждый раз, что мне приходилось возвращаться к таблицам стилей, чтобы вспомнить имя того или иного селектора, мне платили по центру, я бы отоваривалась исключительно в бутиках, а не на распродажах. При изменении дизайна вам будет значительно легче, если у вас будут подходящие названия селекторов в таблицах стилей и разметке; тот факт, что имена логичны и просты для запоминания, упростит вашу работу по переделке дизайна.

Обратите внимание, что это всего лишь часть информации о селекторах: существуют также типовые (type selectors) и универсальные селекторы (universal selectors). Использование каждого вида селектора определяется сложными правилами. Для более подробного ознакомления с ними см. страницу www.w3.org/TR/1999/REC-html401-19991224/types.html#type-id.

На XHTML-странице вам необходимо прикрепить идентификатор (атрибут id) к списку навигации. Листинг 7.11 показывает ненумерованный список, занесенный в тег раздела div.

Листинг 7.11. Простая разметка уменьшает количество ненужного балласта на сайте

```

14. <div id="menu">
15. <ul>
16. <li><a href="index.html">Ботинки - Главнаж</a></li>
17. <li><a href="/story/">Ваша истори</a></li>
18. <li><a href="/read/">Читайте истории</a></li>
19. <li><a href="/help/">Получите помощь</a></li>
20. <li><a href="/answers/">Найдите ответы</a></li>
21. <li><a href="/forum/">Почаствуйте в форуме</a></li>
22. </ul>
23. </div>
24.
25. <div id="sidebar">
26. <pximg src="images/carriel_6mo_c.gif" alt="Увлечение ботинками
    может коснуться даже молодых." width="239" height="243" />
27. <cite>Даже в шесть месяцев я была сильно привязана к
    ботинкам. Это была семейная шутка,
28. но для меня это не было шуткой. Это было по-настоящему...</
    cite> (читайте больше об <a href="story.html">истории
29. Мелиссы</a>).</p>
30. <pximg src="images/susan.gif" alt="Жизнь снова может стать
    хорошей" width="239" height="243" />
31. <cite>Отсутствe необходимости покупать ботинки сделала меня
    новой женщиной...</cite> (читайте больше об
32. <a href="story.html">истории Кейт</a>).</p>
33. </div>
34. <p2>Зачем нам нужны ботинки?</p2>
35. <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    Sed commodo elit viverra lectus. Sed nisi dolor, porttotor
36. sit amet, posuere nee, iaculis at, orci. <a
    href="erat.html">maecenas,libero erat</a>, mattis at, volutpat in,
37. fermentum at, dui. Phasellus ut erat nee lorem vestibulum
    lacinia. Cras turpis nulla, aliquot vulputate, vestibulum at,
38. rhoncus eu, leo. Nam rutrum. Integer consectetur tortor nee
    nunc. Nullam vehicula mauris. Sed accumsan, mi eget
39. ultrices lobortis, lacus nulla voluptat dolor, ut faucibus
    tellus justo in lectus.

```

```
40. Morbi bibendum. Interger quis risus. Morbi accumsan lectus
id dui. <a href="sapien.html">etiam purus sapien</a>, tempus ut,
laoreet id, tempor vitae, diam.
41. Cras odio metus, lobortis non, pulvinar in, accumsan id,
ante. aliquam neque. Pellentesque eros. Nam ultricies sem nec
42. ante. </p>
43.
44. <h2>Есть ли у вас проблемы с ботинками?</h2>
45. <p> Pellentesque habitant morbid tristique senectus et netus
et malesuada fames
46. ac turpis egestas. Pellentesque habitant morbid tristique
senectus et netus
47. et malesuada fames ac turpis egestas.</p>
```

Тег `div` содержит идентификатор (`id`) меню (`menu`). Это все, что вам нужно. Вы можете дать этому идентификатору любое название, однако лучше всего давать название, которое имеет смысл и которое легко запомнить и написать. Теперь CSS-правила, которые вы создаете для этого идентификатора, будут применены к этому разделу внутри тега `div`.

Поэтому давайте создадим правило или - в данном случае - правила. Для обеспечения необходимого внешнего вида вам понадобится три CSS-правила. Листинг 7.12 показывает, как выполняется эта задача. Пример продолжает CSS-документ, начатый в листинге 7.10, где мы оформили основные элементы страницы, такие как тело, абзацы и заголовки.

Листинг 7.12. Таблица стилей выполнит работу по оформлению

```
21. .first {
22.   margin-top: 0;
23.   padding-top: 0;
24.   text-transform: uppercase;}
25.
26. #menu ul, #menu li {
27.   display: inline;
28.   margin: 0;
29.   padding: 0;
30.   color: #333399;
31.   text-transform: uppercase;
32.   text-align: justify;}
33.
34. #menu ul {
35.   width: 800px;}
36.
```

212 ГЛАВА 7. Сэкономьте время и деньги благодаря Шеб-стандартам

```
37. #menu li a {
38. padding: 7px;
39. background: #666666;
40. color: #ffffff;
41. border: solid 1px #999999;
42. text-decoration: none;}
43.
44. #menu li a:hover {
45. background: #333333;}
46.
47. #sidebar {
48. width: 245px;
49. text-align: justify;
50. padding: 20px, 0px, 20px, 20px;
51. float: right;}
```

212 ГЛАВА 7. Сэкономьте время и деньги благодаря Шеб-стандартам

```
37. #menu li a {
38. padding: 7px;
39.- background: #666666;
40. color: #ffffff;
41. border: solid 1px #999999;
42. text-decoration: none;}
43.
44. #menu li a:hover {
45. background: #333333;}
46.
47. #sidebar {
48. width: 245px;
49. text-align: justify;
50. padding: 20px, 0px, 20px, 20px;
51. float: right;}
```

Правило, которое начинается в строке 26, устанавливает основные характеристики нумерованного списка. Это правило определяет внешний вид текста и устанавливает, что этот список будет отображаться горизонтально, а не вертикально, как это обычно принято:

- строка 27 содержит свойство отображения со значением `inline` (в ряд). Это сообщает браузеру, что вы хотите, чтобы нумерованный список и пункты списка, идентифицированного как меню, отображались в ряд (горизонтально, а не вертикально). Это не является гарантией того, что текст автоматически не перейдет на новую строку; это обеспечивает лишь то, что браузер не будет выставлять разрыв новой строки после каждого предмета в списке;
- строка 32 содержит свойство `text-align` (выравнивание текста) со значением `justify` (по ширине), которое отключает выравнивание по левому или по правому краю;

- строка 38 оставляет немного места между пунктами меню, чтобы текст мог располагаться более свободно;
- строка 39 придает кнопкам меню светло-серый фон;
- строка 40 делает текст белым;
- строка 41 добавляет каждой кнопке окантовку в 1 пиксель.

Чтобы задать включенное состояние ролловера, строки 44 и 45 сообщают браузеру о том, что, когда пользователь наводит курсор на ссылку, фон кнопок меню должен поменяться на темно-серый (#333333).

Теперь поговорим о боковой панели. Все, что вам нужно от этой панели, - чтобы она отображалась в правой части окна браузера. Опять-таки, надо создать специальный идентификатор `sidebar`. Прикрепите его к разделу страницы, который должен появляться в виде панели (листинг 7.11, строка 25). Затем опишите идентификатор в таблице стилей; в нашем примере он записан в строке 47 листинга 7.12. CSS-правила для панели говорят о том, что эта область должна быть шириной в 245 пикселей, выравнивать текст по ширине, давать тексту немного свободного пространства и находиться в правой части таблицы.

Когда вы создаете еще стили для сайта, удостоверьтесь, что они будут поддерживаться большим числом браузеров. Если вы создаете стиль, который не будет работать на Netscape 4.0, просто вытащите этот стиль из документа `basic.css` и поместите его в таблицу стилей `fancy.css`. Netscape 4.0 - не единственный браузер, у которого возникают проблемы с CSS-правилами, но у него самая слабая поддержка CSS из всех браузеров, которые сегодня используются. Перемещение неблагоприятных для Netscape 4 правил в импортируемую таблицу стилей `fancy.css` предохранит браузер Netscape 4 от сбоев и поможет людям, использующим этот браузер, быстро просматривать ваш сайт.

Шаг 3: Быстрое и простое изменение дизайна

Предположим, что клиентка, заказавшая сайт «Ботинки не любовь», в последний момент решила, что ей совершенно не нравится дизайн сайта. Правда, сначала она одобрила первую версию дизайнера, но примем во внимание то, что в первую очередь ее волнуют проблемы, связанные с ботинками, и простим ей непостоянство. Вас в клиенте должен интересовать платежный чек, а не постоянство. Теперь она хочет, чтобы сайт был розовым, с множеством маленьких пчел, которые летают вокруг, потому что пчелы означают власть, по крайней мере, она в это верит. Ей хочется, чтобы использовались необычные шрифты и чтобы боковая панель была слева. Вы хотите поскорей выполнить все



Рис. 7.10. Сайт «Ботинки не любовь», переделанный за считанные минуты

пожелания клиента. Поскольку для дизайна вы использовали CSS-стилизацию, решить эту проблему вы сможете быстро.

Вы потратите 30 минут на редактирование таблицы стилей и, поступившись своими эстетическими принципами, создадите сайт на рис. 7.10. Вот что вам пришлось бы для этого сделать:

- добавить свойство фоновой картинки в правило тела; для этого требуется написать одну дополнительную строку в CSS-документе. Обратите внимание, что в тег `<body>` HTML-страницы не добавляется никакого изображения.

```
body {
background-image: url(images/bees.gif) ; }
```

- изменить расположение панели с правой стороны на левую, изменив всего одну строку в таблице стилей. Свойство `float` в правиле панели было изменено с `left` на `right`:

```
#sidebar {
width: 245px;
text-align: justify;
padding: 20px 0px 20px 20px;
float: left;}
```

- поработать с еще несколькими CSS-правилами, изменив несколько свойств и значений. Измените несколько шрифтов, размеров и отступов, настраивая CSS-свойства и значения;
- предъявить клиентке счет за дополнительные 30 мин.

Шаг 4: Создание таблицы стилей для печати

Простая таблица стилей для печати контролирует процесс печати вашей страницы. Таковую таблицу стилей вы можете использовать для того, чтобы скрыть навигацию и другие элементы страницы, на которые напрасно расходуется бумага. Также вы можете выбрать шрифты, наиболее подходящие для печати. Листинг 7.13 показывает таблицу стилей, которая делает сайт «Ботинки не любовь» таким, как вы видите на рис. 7.11.

Листинг 7.13. Новая таблица стилей сайта «Ботинки не любовь» для печати

```
body {
color: #000;
background-color: #fff;
font-family: verdana, geneva, arial, sans-serif;
margin-left: 70px;
margin-right: 70px;}
div, p, th, td, li, dd, dl, dt {
color : #ffffff;
font: 11pt verdana, geneva, arial, sans-serif;}
```



Рис. 7.11. Если вы хотите проверить, как ваша страница будет выглядеть в напечатанном виде, вы быстро можете создать таблицу стилей специально для печати

```

hi, h2, h3, h4, h5, h6 {
    font-family: Georgia, Times New Roman, Times, serif;
    margin-bottom: 1px;}
hi {
    font-size: 24pt;
    text-transform: uppercase;
    text-align: justify;}
h2 {
    font-size: 20pt;}
h3 {
    font-size: 18pt;}
    
```

```
h4      {
        font-size:    14pt;}
h5      {
        font-size:    12pt;}
h6      {
        font-size:    11pt;}
a { '
color: #99cccc;
text-decoration: underline;
font-weight: bold;}
a:hover {
text-decoration: none;}
a:active {
color: #99ffff;}
img {
border: dotted 1px #cccccc;}
.first {
    margin-top: 0;
    padding-top: 0;
    text-transform: uppercase;}
#menu ul, #menu li {
    display: none;}
#sidebar {
    width: 245px;
    text-align: justify;
    padding: 20px, 0px, 20px, 20px;
    float: right;}
```

Листинг 7.13 показывает таблицу стилей, которая отформатирует документ для печати. Она основана на таблице стилей, переделанной для безумной клиентки, но сюда были внесены некоторые изменения, для того чтобы страница хорошо выглядела на бумаге:

- размеры шрифтов текста тела уже не выражены в пикселях и ems (относительных единицах), которыми лучше всего выражать размер шрифта в Сети; они выражены в пунктах. Пункты используются в печати;
- все, что было в рамках идентификатора menu, скрыто. Правило display: none выполняет это. Лучше не тратить бумагу на распечатывание панели навигации;
- убраны все фоновые цвета; остается только белый фон для тела. Пользователю не придется по душе, если по вашей вине ему придется тратить краску на печать фоновых цветов, которые при этом передадутся

неправильно. (К тому же некоторые браузеры вообще не печатают фоновые цвета.)

Когда пользователь напечатает сайт, он увидит страницу, отформатированную, как документ на рис. 7.11.

XHTML- и CSS-ресурсы

В этой главе я лишь чуть-чуть рассказала вам о силе CSS. Если у вас есть хорошая XHTML-разметка, с CSS вы можете сделать очень многое. Вот несколько ресурсов, куда вы можете обратиться, если захотите узнать больше:

- идея создания горизонтального списка, как в навигации сайта «Ботинки не любовь» была заимствована из статьи в журнале "A List Apart" под названием "Taming Lists" (Прирученные списки), написанной Марком Ньюхаусом (Mark Newhouse) - www.alistapart.com/stories/taminglists;
- как говорилось ранее в этой главе, Марк также управляет сайтом под названием Real World Style (<http://realworldstyle.com>). Он любит создавать проблемы верстки Web-страниц и находить для них CSS-решения. Решения зачастую великолепны и многие из них доступны далее для Netscape 4;
- the Web Standards Project (www.webstandards.org) - отличный ресурс, где можно узнать последние новости, найти учебники и, если вы чувствуете себя готовым, вступить в сообщество по работе над Web-стандартами;
- Эрик Мэйер создал массу фантастических ресурсов, включая несколько книг по теме: «Эрик Мэйер о CSS» ("Eric Meyer on CSS", New Riders Publishing, 2002), «Каскадная таблица стилей: Справочное руководство» ("Cascading Style Sheets: The Definitive Guide", O'Reilly & Associates, 2000) и «Каскадная Таблица стилей 2.0: Справочник программиста» ("Cascading Style Sheets 2.0 Programmer's Reference", The McGraw-Hill Companies, 2001). Он также поддерживает некоторые ресурсы на своем сайте Meyerweb (www.meyerweb.com/eric/css);
- Джеффри Зельдман - это мой любимый евангелист по Web-стандартам. В его журнале "A List Apart: A Magazine for People Who Makes Web Sites" - www.alistapart.com - регулярно появляются статьи по механике создания Web-сайтов, поддерживающих стандарты. В его ежедневном отчете проскальзывают новости, имеющие отношение к стандартам (www.zeldman.com), а в его книге «Дизайн по Web-стандартам» подробно освещаются все темы, кратко рассмотренные в данной главе. Я это знаю потому, что, когда я писала эту книгу, он сидел рядом и писал свою.

Глава в. Экономичные домены и хостинг

Гручо: Ну да, такой пункт есть в каждом контракте.

И это называется «sanity clause».

Чико: Ха, ха, ха, ха, ха! Вы меня не обманете.

Sanity Claus не существует.

*Гручо и Чико Маркс
в пьесе «Ночь в опере» (1935)*

Основные идеи главы

1. Покупатель, будь осторожен!

Несмотря на то что существует масса надежных компаний, регистрирующих домены и предоставляющих хостинг, будьте внимательны. Чтобы не разориться, будьте организованны и не останавливайтесь на первой попавшейся организации.

2. Помните о скрытых расходах.

Расходы, которые могут возникнуть из-за временной недоступности сайта и трудностей в получении поддержки, могут очень быстро «съесть» ваш бюджет.

3. Обращайте внимание на расход пропускной способности.

Многие хостинговые компании могут разорить вас, если трафик вашего сайта возрастет. Узнайте, какие у вас ограничения на пропускную способность и сколько вам придется заплатить, если вы выйдете за эти рамки. Следите за статистикой вашего сайта: если трафик возрастает, это хорошо, но вам не следует за это платить лишнее. Вы должны внимательно следить за тем, какова скорость передачи данных на вашем сайте и каковы расходы на пропускную способность. Вы сэкономите, если будете владеть этой информацией.

4. Разместите свой сайт на сервере самостоятельно.

Помните, что вы сами можете разместить свой собственный сайт. Если вы готовы как следует потрудиться, возможность собственного размещения сможет обеспечить вам большую экономию.

В 2002 году, когда я начала управлять небольшим профессиональным Web-сайтом, созданным для подруги, я зарегистрировала ей имя домена в компании easyDNS (www.easydns.com), надежном регистраторе имен доменов. Один или два месяца спустя я получила «уведомление об окончании срока действия имени домена» от другого регистратора; назовем его StinkyDomains («Вонючие домены»). Уведомление от StinkyDomains представляло собой форму для оплаты домена, который я только что зарегистрировала с easyDNS. Данный документ подразумевал, что я зарегистрировала домен со StinkyDomains, а не с easyDNS и что срок действия этого домена скоро истечет. Когда я более подробно изучила форму, я увидела, что дата истечения срока действия была закрыта строкой «Ответить до» и что дополнительная форма была не для обновления, а для передачи домена. Если бы лично меня так обманули, мне бы пришлось перенести домен на сервер StinkyDomains и дополнительно платить лишние \$35 в год за регистрацию домена.

Такие грубые приемы применять неэтично. В ряде случаев они даже могут оказаться незаконными (или почти незаконными), но тем не менее они периодически применяются. Многие дизайнеры и владельцы сайтов платят лишние деньги компаниям, отвечая на вводящие в заблуждение «уведомления об окончании срока действия». Никто не должен быть обманут таким образом, а дизайнер, работающий при небольшом бюджете, просто не может себе этого позволить.

Мои беды не ограничивались проблемами с регистрацией доменов; хостинг тоже был источником страданий. В 1999 году я разместила сайт на сервере хостинговой компании, назовем ее NastyHost («Мерзкий хостинг»). В первые два месяца никаких проблем не возникало, но на третий месяц сайт перестал регулярно загружаться; несколько дней он вообще не работал. Поэтому я прекратила работу с этой компанией, нашла новый хостинг и навсегда забыла об этом. По крайней мере, я так думала.

Год спустя я стала проверять отчет по своей кредитной карточке и обнаружила, что NastyHost совершенно неожиданно сняли с моего счета 1141 доллар 20-центов. Когда я стала выяснять, в чем дело, мне заявили, что у них нет никаких записей, подтверждающих аннулирование моего обслуживания, и что они решили, что я захотела продолжить содержание едва работающего сайта, но уже на новом, более дорогом тарифном плане. Тот тарифный план, за который я платила год назад, был уже недоступен, поэтому они выбрали «похожий» план, который стоит в четыре раза дороже, и спасли меня от тяжелых раздумий, как же мне потратить мои собственные деньги, сняв деньги с моего счета

напрямую. Кроме того, они зарегистрировали новое имя домена, и регистратор поверил им. Я платила за пользование сервером, которым я на самом деле не пользовалась. Люди, которые пытались посетить мой сайт, не могли этого сделать. Для того чтобы разобраться с этим безобразием и добиться возмещения убытков, мне понадобилось 10 дней; на одни только телефонные разговоры ушло в общей сложности 6 часов.

К несчастью, Web-профессионалы достаточно часто сталкиваются с подобными проблемами, связанными с регистрацией доменов и хостингом, поэтому я советую профессионалам, работающим при небольшом бюджете, развить мощную защиту против такого рода проблем. В этой главе я научу вас, как защититься. Не волнуйтесь: эта глава не вся из страшных историй. Мы отыщем хорошие компании по хостингу и регистрации имен доменов.

г-Определение

Регистрация имени домена (Domain Name Registration). *Имя домена* - это буквенно-цифровое имя, которое позволяет вам определить хосты внутри этого домена. Обычно, но не всегда хосты - это одиночные компьютеры. Имена хостов дают пользователям возможность обращаться на сайт с более простым для запоминания именем (например, www.example.com), а не сложным для запоминания IP-адресом (например, 66.45.6.196), посредством которого браузеры и Web-серверы соединяются на самом деле.

www.ebauxom - это имя домена, в котором определено имя хоста. Буквы www указывают на сервер, предоставляющий сайт: www.ebay.com. *Регистратор* (registrar) имени домена - это коммуникационная компания, которая позволяет вам зарегистрировать имя домена и затем определить имя хоста в этом домене, чтобы указать на место в Интернете, которое будет домом для вашего сайта.

•Определение

Web-хостинг (Web Hosting). Услуги Web-хостинга предоставляют пространство на сервере, где хранятся файлы, из которых состоит сайт. Сервер - это просто компьютер, который запускает программное обеспечение, отображающее ваши страницы для тех, кто их просматривает. Для объемных сайтов нужен мощный сервер, подключенный к Интернету с быстрым соединением и постоянно работающий. Для сайтов меньшего объема сервер может быть относительно маломощным; если ваш сайт имеет небольшой исходящий трафик, ему даже не нужно быстрое соединение (но он все равно должен работать всегда).

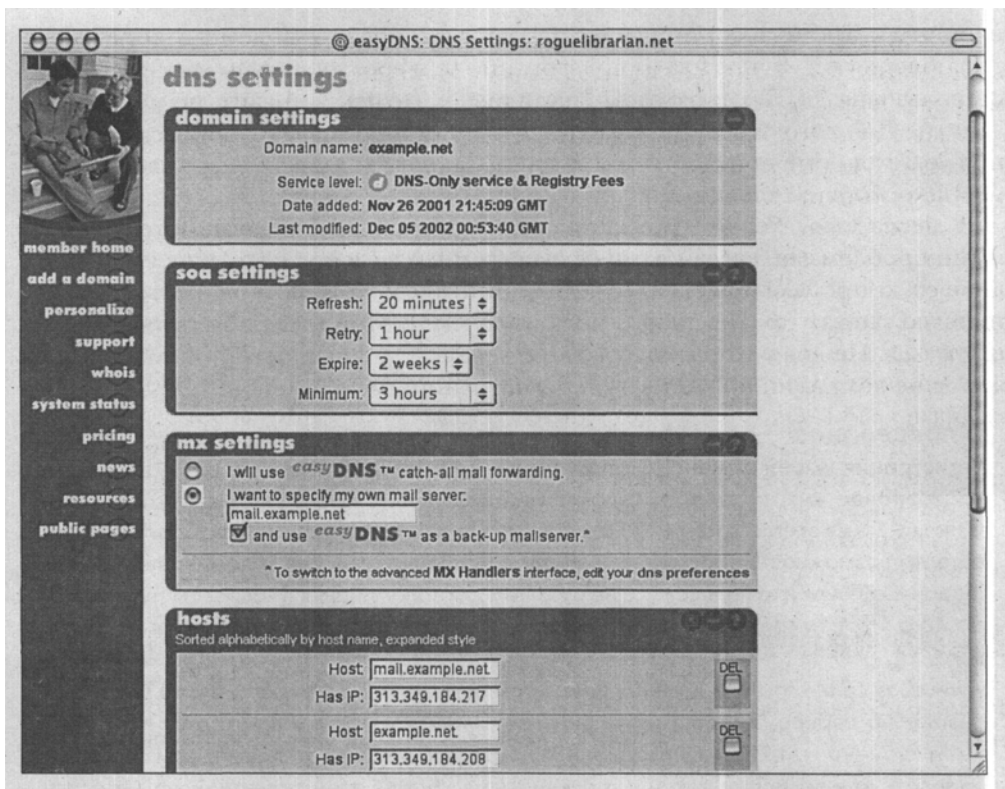


Рис. 8.1. Когда вы регистрируете имя домена, вы создаете запись, подобно той, что вы видите здесь: easydns.com. В конце страницы, в разделе **Hosts** (Хосты), вы можете увидеть, что имя домена - www.example.net - указывает на ведущий IP-адрес. В данном случае IP-адрес 313.349.184.208 - это хостинг-машина, и регистратор имени домена будет связывать www.example.net с этим IP-адресом

Покупатель, будь осторожен!

Я бы хотела сказать, что мой трагикомический опыт с доменом и хостингом, описанный ранее, является исключением из правила, но, к сожалению, это не так. Нет ни одного Web-профессионала из тех, что я знаю, кому бы не приходилось сталкиваться с подобными проблемами. Настоящий Web-профессионал внимательно следит за расходом трафика, периодами недоступности сайта, плохой технической поддержкой и поддержкой клиентов и прежде всего остается организованным.

Помните, что время - это деньги. Вернемся к моему шестичасовому разговору с NastyHost. Если бы клиент либо начальник платили мне около \$10 в час, мой разговор с NastyHost обошелся бы в \$60.

Также необходимо принять во внимание потенциальные потери прибыли из-за того, что сайт долгое время был недоступен. Вспомните, из-за NastyHost мне не только пришлось уплатить значительную сумму за сервер, который я не использовала; по их вине пользователи не могли посещать мой сайт.

Если вам приходится сталкиваться с вынужденной недоступностью сайта либо с поддержкой, на которую уходит много времени, помните, что это дорого обходится. При условии ограниченного бюджета такие затраты нежелательны, а то и просто разорительны.

Хорошие новости: существуют надежные, честные компании, занимающиеся регистрацией доменов и хостингом, существуют также вещи, которые вы можете и *должны* делать, чтобы предохранить себя от встречи с плохими компаниями. Если вы грамотно зарегистрируете хостинг и домен, вы не растратите понапрасну бюджет. Давайте рассмотрим подробнее регистрацию имен доменов и хостинг и поговорим о том, что вы можете сделать, чтобы сэкономить деньги и защититься от непредвиденных расходов.

Цена вашего доменного имени

Регистрация доменного имени - это не то, на чем надо *экономить*. Здесь основная задача - не *потерять* время и деньги. Избегайте сделок, которые кажутся слишком хорошими, чтобы быть правдой, ведите подробные записи.

Это звучит слишком громко, чтобы быть правдой

На момент публикации данной книги регистрация имени домена на год стоила примерно \$20. Если регистратор предлагает услугу за \$6, будьте осторожны - в дальнейшем вы можете столкнуться со скрытыми расходами, возможно, вам придется платить дополнительно «за любое движение» каждый раз, когда вы обновляете ваш адрес или другую информацию относительно домена.

Надо быть организованным

Настоятельно рекомендую вести записи. Вы должны знать, у кого вы зарегистрировали ваш домен, когда вы это сделали, сколько заплатили и какую кредитную карточку использовали.

Варварская практика вроде той, с которой я столкнулась в исполнении StinkyDomains, широко распространена. Можете поискать в Google или Index информацию о последних мошенничествах в этой области. В Интернете права корпораций очень хорошо защищены: только попробуйте использовать изображение, которым вы не владеете, или скачать музыкальный файл, за который вы не заплатили, и вы тут же очутитесь в суде. К сожалению, права обычных потребителей, фрилансеров и Web-профессионалов-одиночек защищены не так хорошо. Представьте, что вы на Диком Западе; даже если вы не ищете проблем, вы все равно вынуждены иметь при себе пистолет и знать, как с ним обращаться, просто для того, чтобы выжить.

Хорошие записи - это великолепное оружие. Конечно, вы должны хранить аккуратные записи в своем офисе, но также необходимо, чтобы у вашего *текущего* регистратора домена была ваша контактная информация. Даже самый порядочный регистратор не сможет вам ничем помочь, если вы смените адрес и телефон и никому не скажете об этом. Если у регистратора домена не будет вашей достоверной контактной информации, вы не получите уведомлений о необходимом продлении хостинга.

Записи также должны вестись в подходящей форме. Если у вас несколько доменов, а так у большинства Web-профессионалов и бывает, трудно держать все записи в голове. Если у вас есть один файл, вы быстро определите, какие формы из тех, что вы получаете, законные, а какие - нет. В дополнение к файлу и всем действиям ведите также таблицу, подобную табл. 8.1. Вы должны иметь возможность сразу узнать, где были зарегистрированы ваши доменные имена. Будьте разумны, когда регистрируете домен.

Если вы получаете непонятные e-mail-сообщения или письма-формы относительно вашего домена, внимательно изучите их, прежде чем посылать чеки и номера кредитных карт. Каждый месяц проверяйте состояние вашего счета, чтобы быть уверенным, что с вас не сняли денег за услуги, которые вы никогда не заказывали.

Решение: выбор регистратора доменного имени

Как говорилось ранее, регистратор доменного имени - это компания, которая позволяет вам «купить» домен и связать его с сервером. «Купить» - это не совсем точное слово, поскольку на самом деле вы *регистраете* домен, и только на ограниченный период времени. Наиболее распространенные регистрации - на один-два года и на пять лет; обычно существуют скидки, если вы заранее платите более чем за один год. Когда у вас установлен хост, вы используете

Таблица 8.1. Таблица для ведения отчетности по доменным именам

Имя домена	Регистратор	Дата регистрации	Дата истечения срока	Стоимость	Номер кредитной карты	Имя пользователя и пароль	Посредник (Да/Нет)	Примечания
Wishiwasher.com	SPECTRE DNS	30 июня, 2001	29 июня, 2002	\$150	6543 XXXX XXXX XX352	И: bornyesterday П: p0wd3rpuff	Да	Зачем я доверилась им? Они потеряли мою учетную запись, и теперь я не могу обновить домен. Придется регистрировать .net и .org, чтобы люди хотя бы смогли найти мой сайт снова
Wishiwasher.net	UFP DNS	22 мая, 2002	22 мая, 2003	\$35	4567 XXXX XXXX XX309	И: notthistime П: ssm00chk1tt3n	Нет	Пока все хорошо
Wishiwasher.org	UFP DNS	22 мая, 2002	22 мая, 2003	\$35	4567 XXXX XXXX XX309	И: notthistime П: d1rtys0ck3	Нет	Менеджер по работе с именами довольно милый. Может, послать ему секретный URL с моими фотографиями?

регистратор для того, чтобы задать имя хоста в вашем домене и связаться с сервером.

Старейший и самый известный регистратор - это Network Solutions (Netsol). Сейчас этим старинным регистратором владеет компания VeriSign (www.verisign.com), но большинство Web-профессионалов все еще называют этот регистратор Network Solutions, наряду с некоторыми менее лестными именами. Не будет преувеличением сказать, что с самого первого дня существования компании Netsol ее клиенты погрузились в бесконечные судебные тяжбы, связанные с потребительским обслуживанием. Этим тяжбам посвящен целый сайт (www.nsihorrorstories.com), содержащий много ужасных историй. Некоторые из них уже стали хрестоматийными.

В оправдание Netsol можно сказать, что чем больше компания, тем больше у нее клиентов и тем больше, таким образом, она получает от них жалоб. Поэтому, учитывая количество историй о некачественном обслуживании клиентов, связанных с этой компанией, вы должны искать лучшее место для вложения денег. Также несмотря на то что Netsol является самым большим и самым успешным регистратором, она не гнушается высылать вводящие в заблуждение уведомления о продлении хостинга людям, зарегистрировавшимся у конкурентов.

easyDNS (www.easydns.com) - более мелкий регистратор доменных имен. Стоимость регистрации та же, но, когда эта книга выйдет из печати, поддержка клиентов там все еще будет великолепной и общая значимость этих услуг останется высокой.

Фирма Go Daddy (<http://meet-the-makers.com/conversations/parsons>) - регистратор с высоким уровнем обслуживания клиентов и низкими ценами.

р Примечание

easyDNS и Go Daddy - не единственный удачный выбор. Чтобы познакомиться с полным списком регистраторов, смотрите директорию аккредитованных регистраторов компании Internic (www.icann.org/registrars/accredited-list.html). ICANN - это единственная организация, которая может аккредитовать регистратора для выдачи доменов высшего уровня com/[.net](http://net)/[.org](http://org).

Экономный хост

Забота о доменном имени - это очень хорошо, но имя мало что значит без Web-хоста, к которому оно подключено. Подобно регистрации имени домена, Web-хостинг также может быть «осиным гнездом» со скрытыми расходами, которые могут «ужалить» неосторожного разработчика сайта в самое уязвимое место

его бюджета. При выборе хоста профессионалы, работающие при небольшом бюджете, должны проводить такую же работу по оценке и планированию, как мы делали в главе 6, когда подбирали наилучшую систему управления сайтом для вашего бюджета:

1. Определите, чего вы хотите.
2. Узнайте, какой хост вам может это дать.
3. Не жалейте времени на изучение того, как эти услуги работают.

Определите, чего вы хотите

Прочитав предыдущие главы, вы не удивитесь, если я скажу вам, что люблю делать покупки. Существует только одна причина, по которой я не растратила все свои сбережения и не стала нищей. Прежде чем транжирить, я определяю свой бюджет и в голове составляю список того, что у меня есть. Затем я составляю список из новых покупок, которые помогут мне извлечь максимальную пользу из того, что у меня уже есть. Поступайте таким же образом, когда выбираете провайдера Web-хостинга.

Вместо того чтобы набивать шкаф всем подряд, посмотрите на то, что у вас есть: известный объем трафика, количество сотрудников, размер вашего сайта; затем обратите внимание на следующие пункты:

1. Каков годовой или месячный бюджет для хостинга? Примите во внимание, что сюда необходимо включить и стоимость труда в виде времени, которое вы потратите, разбираясь с технической поддержкой.
2. Будет ли у сайта большой трафик? Если так, то вам понадобится значительная пропускная способность и, возможно, придется учитывать объем трафика. (Позже, я покажу вам, как это работает.) Просмотрите отчеты статистики по уже существующему сайту. Если у вас нет этих цифр, постарайтесь узнать, сколько байтов передает подобный сайт.
3. Понадобится ли вам управляемый базой данных сайт? Какая база данных и промежуточное программное обеспечение вам понадобится? Содержание многих сайтов находится в таких базах данных, как MySQL, Oracle или Microsoft SQL Server, и отправляется на страницу при помощи промежуточного программного обеспечения, например PHP, ColdFusion или ASP. Программное обеспечение с открытым кодом, такое как MySQL и PHP, менее дорогое, чем их коммерческие двойники, включая MS SQL Server и ColdFusion. Однако, если вы уже работаете с коммерческим продуктом, вам придется слишком много потратить на переобучение персонала и преобразование данных.

г- Угроза бюджету _____:—

Многие хостинговые компании предлагают Microsoft Access в качестве программы Web-базы данных. Хотя повсеместное наличие Access делает его привлекательным, недорогим решением для сайтов, управляемых базой данных, эта программа не очень хорошо подходит для Web-задач; для ее загрузки может потребоваться большой трафик, и работа будет плохо выполняться. Если вы экономите на базе данных, ваш сайт вряд ли будет работать качественно. См. главу 6, чтобы больше узнать о проблемах, связанных с Access, и о недорогих сайтах, управляемых базами данных.

4. Если ваш сайт управляется базой данных, проверьте, насколько хорошо потенциальный хост управляет приложениями базы данных на серверах с общим доступом. Медленно действующие хосты предоставляют один и тот же сервер многим программистам ColdFusion одновременно; один сайт, над которым я работала, постоянно сбоил, потому что владелец сайта учился программированию ColdFusion на том же сервере. Хороший хост отслеживает и решает такие проблемы.
5. Понадобится ли вам большое количество почтовых ящиков, как это бывает нужно для сайта корпорации или организации? Потребуется ли почтовые ящики много места на сервере? Если вам понадобится хранение электронных писем, проверьте, отвечает ли ваш хостинговый план этой потребности. В противном случае это может обернуться чрезмерными затратами (или потерянными и недоставленными сообщениями). Помните также, что почтовый клиент с Web-интерфейсом хранит сообщения на сервере, пока вы вручную их не удалите..Почтовый клиент, такой как Eudora или Outlook Express, может и должен быть установлен таким образом, чтобы он уничтожал сообщения с сервера, когда они уже загружены на компьютер пользователя. При правильной установке этой опции вы сможете избежать чрезмерных затрат и заблокированных почтовых ящиков (когда клиенты пытаются вам написать, а их сообщения возвращаются, потому что ваш почтовый ящик переполнен).
6. Предположим, что ваши требования к поддержке будут высоки, и сделаем хорошую техническую и пользовательскую поддержку основным критерием. Если вы зависимы от бюджета, вы не можете себе позволить тратить много времени в ожидании. Маленькая местная хостинговая компания может обслуживать лучше, чем автоматизированный гигант, расположенный неизвестно где. Даже несмотря на то что большая компания за ежемесячное обслуживание может просить чуть меньше, ваше

время - это прежде всего деньги, а время, в течение которого ваш сайт работает, бесценно.

7. Предположим, что в службе технической поддержки вашей потенциальной хостинговой компании работают обезьяны, будьте готовы протестировать их. Придумайте какой-нибудь технический вопрос, на который вы знаете ответ. Посмотрите, как они ответят. Обращаясь к вашей компании за советом или поддержкой, спросите то же самое у кого-то, кто не связан с этой компанией. Люди из хостинговой компании могут быть плохо информированы. В некоторых случаях они даже могут дать неправильные ответы, чтобы не молчать. Найдите компанию, специалисты которой знают, о чем они говорят, а затем проверьте правильность их ответов.
8. Предположим, что вам необходима статистика Web-сайта для того, чтобы отслеживать трафик. Если у вас когда-либо возникали проблемы с чрезмерным расходом пропускной способности, такая информация может вам понадобиться. Спросите также, обеспечивает ли компания доступ к исходным журналам сервера, - некоторые люди хотят иметь более подробную информацию, чем та, которая предоставляется в стандартной статистике. Возможно, вы захотите обработать эти данные самостоятельно.
9. Определитесь, какой доступ к управлению вам необходим. Если вы хотите осуществлять более детальный контроль, удостоверьтесь в том, что у вас будет доступ через FTP-клиента к записям в базе данных. Если вам нужен только графический интерфейс, узнайте, можете ли вы администрировать ваш сайт при помощи Web-браузера. Будет обидно обнаружить, что интерфейс неподходящий, когда вы уже дали хостинговой компании номер вашей кредитной карты.
10. Спросите, есть ли гарантия возврата денег. Это тоже ценная информация.

Стоимость трафика и места на диске

Давайте рассмотрим реальный пример, чтобы разобраться со стоимостью трафика и места на диске. Сайт iwebhosting.com вполне конкретно говорит о ценах на пропускную способность (рис. 8.2).

Удостоверьтесь в том, что у вас есть вся необходимая информация и что вы понимаете, что она означает. Давайте возьмем базовый план в качестве примера: \$10 в месяц дают вам 25 Мб места на диске и 1000 Мб ежемесячного трафика.



Рис. 8.2. Расценки на пропускную способность компании iwebhosting (www.iwebhosting.com). Тарифные планы компании iwebhosting на хостинг достаточно конкретны: при увеличении объемов хранения и трафика увеличивается и цена. Обратите внимание, что каждый тарифный план включает дополнительную плату в случае превышения расходов пропускной способности. Это сумма, которую вы будете вынуждены доплатить, если превысите месячное ограничение. Давайте посмотрим, насколько различные тарифные планы подойдут двум сайтам в отношении трафика (рис. 8.3 и 8.4). Вы увидите, что необходимо очень внимательно отнестись к потреблению пропускной способности и выбрать тот тарифный план, который больше всего соответствует вашим потребностям

Если вы выйдете за эти рамки, вам придется платить примерно одну десятую цента за каждый мегабайт¹.

Отметим, что в России ситуация с оплатой хостинга несколько иная. На данный момент надежные хостинговые компании готовы предоставить 250 Мб места на диске и неограниченный трафик за \$10 в месяц. При этом необходимо, чтобы исходящий трафик был намного больше входящего, а российский - намного больше иностранного. Некоторые хостинг-провайдеры даже доплачивают за большой (около 100 Гб в месяц) российский трафик. - *Прим. науч. ред.*

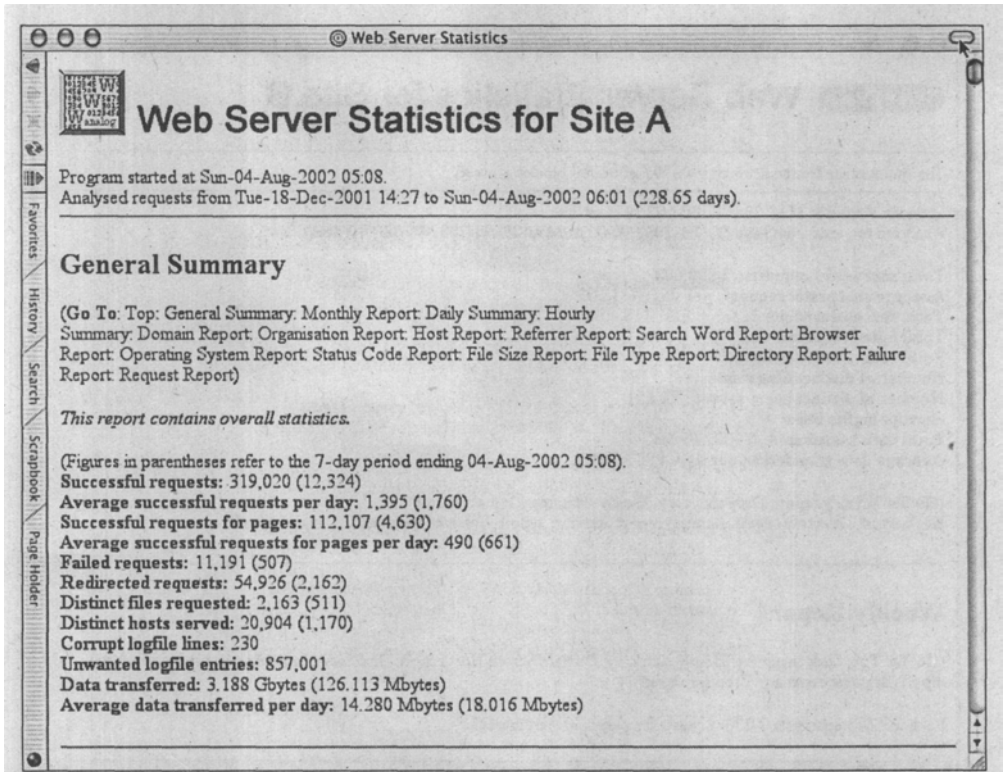


Рис. 8.3. Потребление пропускной способности для сайта А

Место на диске относительно дешево; это допустимый объем пространства на сервере. С другой стороны, пропускная способность стоит не так уж и дешево; это количество данных, которые передает ваш сайт за определенный период времени. Вам придется дополнительно платить за превышение трафика или места на диске. Узнайте, какова дополнительная плата на тот случай, если вы превысите допустимые ограничения. Также спросите, как провайдер это отслеживает, как он сообщит вам об этом, будете ли вы просто отключены, либо заплатите, когда вы уже превысили ограничения, и т.п.

Давайте посмотрим, как начисляется дополнительная плата за место на диске и пропускную способность, а также насколько важен правильный выбор тарифного плана для малобюджетного сайта.

Использование трафика для сайта А выражено двумя числами в последней строке в отчете под названием Data Transferred (Переданные данные) -

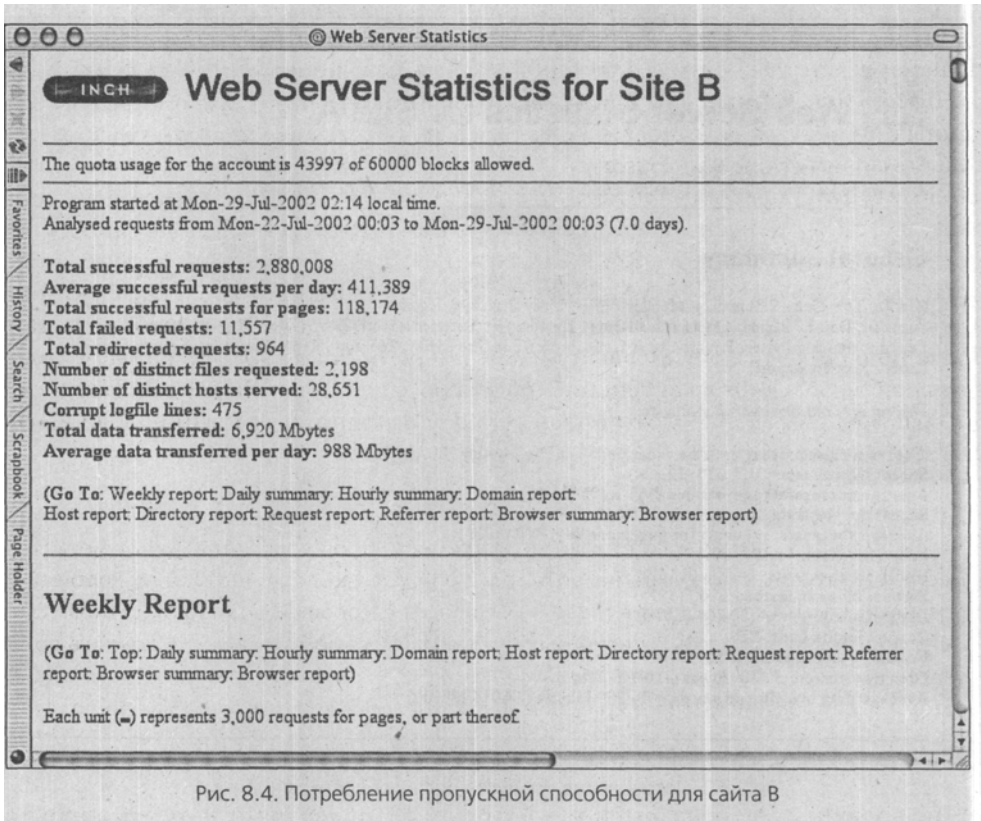


Рис. 8.4. Потребление пропускной способности для сайта B

рис. 8.3. Первое число - 3,188 Гб - это *общее* количество трафика за тот период, что ведется статистика (228 дней). Второе число - 126,113 Мб - за прошедшую неделю. Более надежный параметр - это 3,188 Гб за 228 дней, что в среднем дает 97 Мб в неделю. Однако тот факт, что за прошлую неделю потребление было на 28 Мб *больше*, чем среднее потребление за прошедшие 228 дней, говорит о том, что сайт становится популярнее и труднее для обслуживания. Использование большего количества трафика может говорить о том, что так будет и дальше.

Базовый план компании iwebhosting подойдет, если сохранится прежнее потребление пропускной способности, но, если сайт начнет передавать более 1000 Мб в месяц, - что, кажется, должно рано или поздно произойти - владельцу придется платить дополнительные \$0,008 за каждый мегабайт, выходящий за рамки тысячи.

Отчет по сайту В также показывает, сколько данных было передано за неделю; в данном случае это 6920 Мб данных, переданных за семь дней. 6920 Мб, умноженные на 4, составят 27680 Мб - это приблизительный объем данных, переданных за месяц. Если бы сайт В был на Бронзовом (Bronze) тарифном плане, ограничения по трафику данных были бы превышены на 24680 Мб. Так как вам придется платить \$0,008 за каждый превышенный мегабайт, плата за пропускную способность составит \$197,44 в дополнение к ежемесячной базовой плате в \$24,95. Таким образом, общая стоимость месячного хостинга составила бы \$222,39 в отличие от ожидаемых \$24,95. Очень важно следить за использованием пропускной способности и выбирать наиболее подходящий хостинговый план по мере роста вашего трафика. В этом случае передача данных в течение месяца обошлась бы сайту В дешевле, если бы сразу был выбран Золотой план (Gold) компании iwebhosting; \$149,95 составили бы все расходы.

- Как получить золото из песка _____

Чтобы уменьшить расходы на трафик, постарайтесь, чтобы ваш сайт занимал меньше места. Вы можете сэкономить на размере страниц благодаря написанию хорошей разметки, как мы обсуждали в главе 7. Страницы с хорошей разметкой становятся меньше и, таким образом, их использование обойдется вам дешевле. Вспомните поучительную историю в начале главы 7 о Web-сайте известного Интернет-магазина, который неоправданно теряет примерно терабайт трафика в год только из-за раздутой разметки.

Вы должны следить за вашим хостингом; если увеличиваются размеры и популярность вашего сайта, именно вы должны обратиться к хостинговой компании с просьбой сменить тарифный план. Как только цифры будут сообщать вам о том, что сайт в скором времени перейдет границы текущего плана, немедленно меняйте тарифный план на более дорогой. Переключившись на более дорогой месячный план с большим объемом трафика, вы сможете получить большую выгоду, чем если останетесь на более дешевом плане с меньшим трафиком данных.

Также вам необходимо знать, что некоторые компании не берут плату за трафик: им вы платите только за место на диске, которое вы используете. Если вы сможете найти хорошую компанию, которая взимает плату только за место на диске, будет просто замечательно. У сайта В коммерческий хостинг, где оплачивается только место на диске. Владелец сайта В платит около \$50 в месяц, что намного лучше, чем \$222,39.

Г-Измеряйте дважды: биты и байты

Бит. Бит представляет собой двоичный знак и является наименьшей единицей информации на компьютере. Когда мы говорим о нулях и единицах (0 и 1), мы говорим о битах. Один бит может иметь только одно или два значения: 1 или 0.

Байт, байт - это двоичное число, которое является единицей хранения, выражающей одну букву. Байт равен 8 бит и обозначается как б.

Килобайт. Обычно кило- означает 1000, но, поскольку компьютеры двоичные, кило- - это 2 в 10-й степени. Поэтому технически 1 Кб - это 1 024 байта. *Килобайты* обычно обозначаются как Кб.

ИВ |

Мегабайт. *Мегабайт* - это 2 в 20-й степени, или 1 024 Кб.

Гигабайт. *Гигабайт* - это 2 в 30-й степени, или 1 024 Мб. Разговаривая с хостинговыми компаниями, будьте внимательны с теми, кто говорит об оплате за гигабайты и кто - за гигабиты. Байт - это 8 бит, поэтому подсчеты могут разительно отличаться. Не удивляйтесь, когда увидите, что отчет в мегабайтах или гигабайтах не совпадает с отчетом в мегабитах или гигабитах. Сетевые инженеры обычно говорят об оплате в битах; люди, продающие место на диске, - об оплате в байтах.

Займитесь математикой. Кукла Барби была права: математика - это трудно. Может быть, это и так, но тем не менее заниматься подсчетами стоит. Удостоверьтесь в том, что вы знаете, за что платите, что означает «кило» или «мега», какой множитель используется: 1024 или 1000. Некоторые люди используют термин *килобайт* по отношению к 1000 байт, а не к 1024 байт, как надо. Люди, не сильно разбирающиеся в терминологии, могут использовать термин *мегабайт* по отношению к 1000 Кб, в то время как более точно следует использовать этот термин по отношению к 1024 Кб. Разница между 1000 и 1024 в течение года может вылиться в значительную сумму. Также имейте в виду, что вы не сможете измерить трафик простым перемножением размеров страниц, которые были просмотрены, на количество просмотров. При передаче трафика всегда существуют издержки. Когда передаются большие файлы, они могут быть разделены на меньшие части и каждая часть может подвергнуться так называемым «расходам за доставку».

Представьте, что вы заказали фарфоровый сервиз из 1 2 предметов. Масса нетто сервиза может составить 1 1 кг, но его могут разместить по 1 2 разным коробкам. Таким образом, вам придется заплатить за саму коробку, пенопласт и ленту. Что верно для посуды, верно и для данных: при передаче большого файла каждая часть требует маршрутизации, для того чтобы следить за этими частями. Эти расходы могут быть незначительными при передаче отдельного файла, но со временем вместе они могут составить значительную сумму. Конечно, заботиться об этих тысячных долях центов за «доставку упаковки» было бы чересчур, но ответственный за финансы Web-профессионал всегда должен помнить об этих расходах.

Поиск хостинга по подходящей цене

Теперь, когда вы знаете, что вам нужно, настало время найти хостинг, который сможет вам это предоставить. Начните с советов других; пообщайтесь со счастливыми владельцами сайтов и посмотрите, каким хостингом они пользуются. Также необходимо провести немалую самостоятельную работу. Вам необходимо знать детали тарифного плана, который вы покупаете, и быть внимательным.

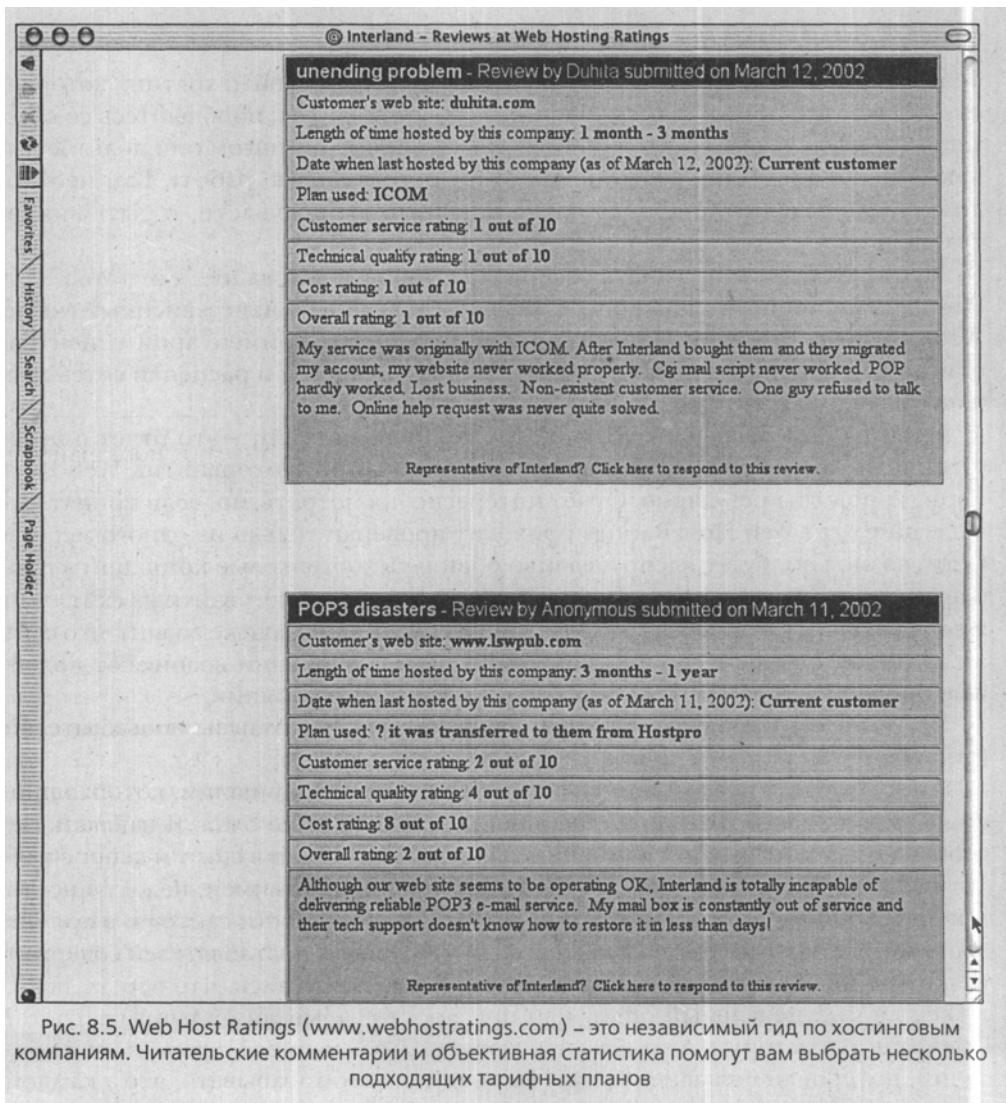
Место, откуда стоит начать беспристрастное исследование, - это Web Host Ratings (www.webhostratings.com). Сайт представляет собой независимый гид по Web-хостингу. Его база данных компаний содержит комментарии клиентов, описания планов с детальными обзорами возможностей и расценки сотен хостинговых компаний.

Один из интересных отчетов, которые проводит сайт, - это отчет о недоступности сайтов на хостах хостинговых компаний, который гид Web Host Ratings проводит регулярно. Отчет интересно посмотреть, но, если принять во внимание, что Web Host Ratings проводит проверки только из одного места в Сети, возможно, будет несправедливо обвинять хостинговые компании в проблемах, в которых они не виноваты. Только то, что я не могу зайти на сайт, если я подключена к Сети через коммутируемое соединение, еще не значит, что сайт не работает. Очень часто при коммутируемом соединении возникают подобные проблемы. А сам сайт может быть в отличном состоянии.

На Web Host Ratings размещены очень интересные отзывы пользователей (рис. 8.5).

Я просмотрела несколько отзывов по хостинговой компании, которая прислала мне счет на \$1141,20, и обнаружила, что я была не одна. Я нашла массу сообщений, в которых эту компанию обвиняли не только в снятии денег с кредитных карт без разрешения, но и в закрытии доменных имен, недоступности технической поддержки, невероятно долгой недоступности сайтов и в прочих дорогостоящих проблемах. Имейте в виду, что отзывы пользователей содержат точки зрения совершенно разных людей; поинтересуйтесь, что это за люди. У всех ли был печальный опыт, или они просто забывали оплачивать счета? Прочитав эти истории, я поняла, что люди, пользовавшиеся услугами этой компании, потерпели большие убытки. Впрочем, не стоит забывать, что у каждой истории есть две стороны.

Когда вы будете проводить исследования, можете сделать что-нибудь вроде табл. 8.2. В верхней графе напишите ваши потребности и начните заполнять остальные графы по мере проведения исследования. Когда начнете работать,



будьте очень внимательны к дополнительным расходам, таким как стоимость установки и расходы при превышении ограничений.

Г Примечание

Немного о периоде работоспособности. Когда хостинговая компания дает любую информацию о периоде работоспособности (uptime) - неважно, гарантия это или констатация факта, - проводите подсчеты. Сайт, находящийся в рабочем состоянии 99,9% от всего времени, в сумме за весь год будет находиться в состоянии простоя примерно треть суток. Многие сайты могут простаивать по десять раз в час, а то и больше, когда проводится работа по перемещению серверов, обновлению» или выполняются прочие необходимые действия по поддержке сайта. Это нормально и обычно планируется хостинговой компанией заранее. Ваша хостинговая компания должна сообщать вам, когда сайт будет недоступен, и совершать необходимые работы не в самые популярные среди посетителей часы. Сложно добиться работоспособности в 99,99999%; это получается дважды на один час за пять лет.

Подберите несколько компаний, чьи услуги вам по карману, и постарайтесь выбрать одну, которая максимально будет соответствовать вашим потребностям. *Низкая цена не должна быть единственным фактором, влияющим на принятие решения.* Экономия в \$5 или \$10 - ничто по сравнению с плохим хостингом. Помните, что небольшой бюджет не значит, что нужно идти отовариваться на барахолке.

Просто дайте мне дешевый хостинг

Исследование - дело прекрасное, но, возможно, вам придется сразу же обратиться к самым дешевым услугам. Несмотря ни на что, помните, что, даже если это и звучит слишком хорошо, чтобы быть правдой, возможно, это все-таки правда. Некоторые компании исчезают без следа, потому что допустили ошибку: стараясь предложить дешевые услуги, они не позаботились о прибыли, необходимой для того, чтобы оставаться в деле. Возможно, самое важное качество хостинговой компании - его долгое существование.

Помните также, что новые хостинговые компании иногда специально предлагают низкие цены, чтобы получить клиентов. По мере роста своей деятельности компания может:

- повысить расценки и изменить условия контракта, ничего вам об этом не сказав;
- увеличить дополнительную плату за трафик;
- предоставить хост сотням сайтов на одном общем сервере за счет уменьшения скорости вашего сайта и повышения его уязвимости к аварии и простоя;

Таблица 8.2. Рабочая таблица для исследования хостинговых компаний

План	Оплата в месяц	ОС	Место на диске	Ограничения на передачу данных	Возможности сервера	Базы данных	Установка	Количество почтовых ящиков	Отсутствие недопустимости	Гарантия возврата денег	Техническая поддержка	Отзывы пользователей
Fancy Pants Hosting	\$16	Windows 2000	200 Мб	10 Гб Стоимость дополнительной трафика: \$1 за Гб	Perl, PHP, ASP	MS Access	\$15	Без ограничений	99,996%	30 дней	Пн-пт, 9-17 часов	Неприятные отзывы на Web Host Ratings
Smarty Pants Hosting	\$30	Linux	Неограниченное	20 Гб Стоимость дополнительной трафика: \$20 за Гб	Perl, PHP	MySQL	Нет	35	Неизвестно	Нет	Нет	Никого не смогла найти
Dirty Pants Hosting	\$10	Unix	50 Мб	Нет ограничений	Perl, PHP	MySQL	Нет	10	99%	Нет	24x7	Кэтрин пользуется их услугами
Hot Pants Hosting	\$30	Unix или Windows 2000	100 Мб	Нет ограничений	CFM, ASP	Нет	Нет данных	Без ограничений	99,999%, 90 дней	90 дней	24x7	Тому, какая это плохая компания, посвящен целый сайт

- не получить достаточного количества денег для того, чтобы остаться в деле, и затем тихо исчезнуть;
- быть купленной кем-то еще, кто поднимет цены и, возможно, понизит уровень обслуживания.

Теперь, когда было сказано все о том, что вы должны получить то, за что платили, посетите эти сайты, где вы сможете найти адреса компаний, предлагающих дешевый хостинг:

- список дешевого хостинга на Web Host Ratings (www.webhostingratings.com/cheap.html) - это список, состоящий из пяти самых дешевых хостинговых тарифных планов, который регулярно обновляется и основан на данных из разных источников, включая комментарии пользователей;
- бесплатные решения по Web-хостингу на About.com (<http://html.about.com/cs/freewebhosting>). Дженнифер Кирнин (Jennifer Kyrnin) долгие годы занималась хостингом в разделе XML/HTML на сайте About.com. Несмотря на то что сам сайт About.com напоминает увядшую розу, работа Дженнифер от этого хуже не стала. Ее список бесплатного Web-хостинга соответствует времени и хорошо исследован;
- бесплатный Web-хостинг Spoke and Axle (<http://spoke-and-axle.com>). Если у вас некоммерческий сайт, посмотрите «услуги сводницы» сайта Келли Эбботт (Kelly Abbott) Spoke and Axle (рис. 8.6). Этот сайт сводит тех, кто нуждается в хостинге, с теми, кто может его предоставить. Spoke and Axle - это реакция Эбботт в ответ на кражу трафика на сайтах вроде Nosepilot.

Будучи профессионалом по дизайну или содержанию, вам, возможно, никогда не приходилось сталкиваться с Web-хостингом, вы никогда не сравнивали различные предложения, разве что для личных дел. Однако выполнение неожиданных заданий - это часть работы Web-дизайнеров, работающих при небольшом бюджете, и хорошее выполнение этих заданий поможет вам почувствовать разницу между провалом и успехом.

Любой дизайн - это вызов, а дизайн при небольшом бюджете - это вызов среди вызовов, потому что он требует от нас сделать больше с меньшими возможностями. Он побуждает нас мощнее использовать наши навыки (и учиться новым) и действительно фокусироваться на том, что нужно пользователям и что именно должны представлять собой наши сайты.

Малобюджетный дизайн не для богатых и знаменитых, хотя дизайнеры, работающие с небольшим бюджетом, могут получить золото из песка и привести малобюджетный сайт к славе и богатству. Он также не для тех, кто легко

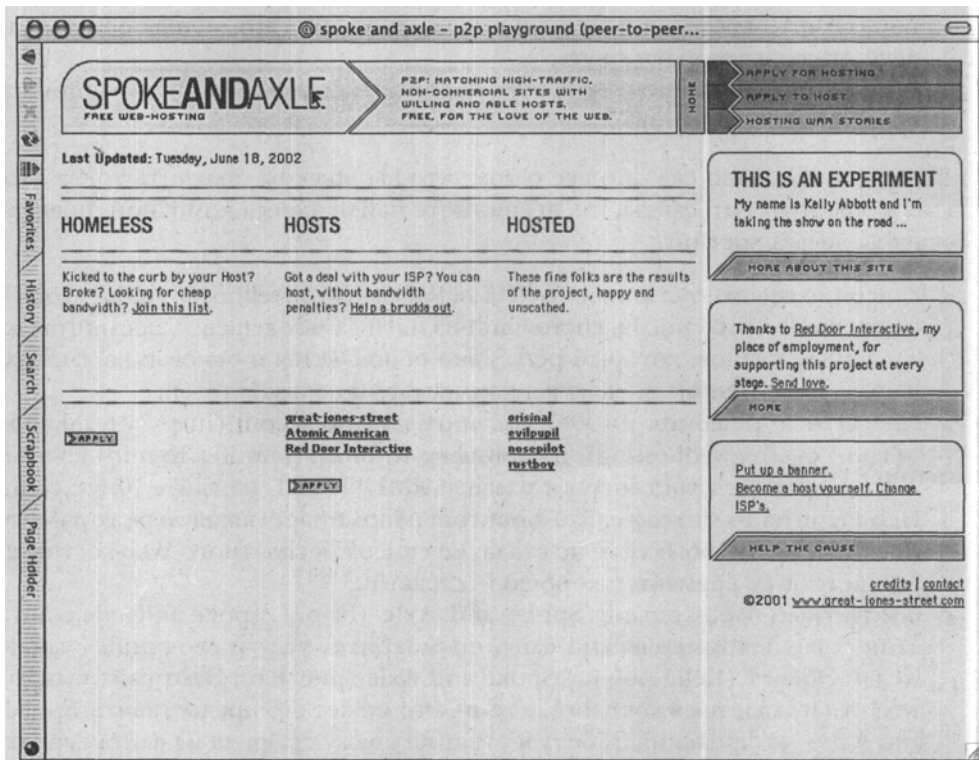


Рис. 8.6. Spoke and Axle (<http://spoke-and-axle.com>) - это сайт, созданный Келли Эбботт, который сводит бедных на трафик с богатыми на него. Бесплатный хостинг может быть чем-то, что люди предлагают для сайтов по искусству, таких как Nosepilot, или для сайтов, которые просто кому-то нравятся. Не ждите, что такое великодушие коснется коммерческого сайта

утрачивает мужество. Если вы не утратите мужества, то невероятно вырастаете творчески и профессионально. А это то, что нельзя купить ни за какие деньги. Увидимся у прилавков со скидками!

Предметный указатель

Байты 234

Биты 234

Бюджет

высокий 32

изменение 34

малый 32

В

ВалидаторШ3С 192

Внешний вид 177

ХНТМL 189

доступность 188

отделение от структуры 181

совместимость с пользовательскими

агентами 185

Гигабайт 234

Гиперссылки 80

Д

Доменные имена 221

выбор регистратора 224

скрытые расходы 224

советы 223

Заголовки 68

Закрытые теги 191

Затраты 32

И

Идея 29

Изменение дизайна с таблицами

стилей 213

Изображения 107

без оплаты роялти 116

плохие 121

с управляемыми правами 116

Иллюстрации 26

К

Картинки 98

разделительныйGIF 100

Килобайты 234

Клипарт 108

библиотеки 109

в Сети 109

использование 111

Л

Лэй Бейкер-Фoley 24

концепция 25

М

Мегабайт 234

Место на диске

стоимость 229

Метаданные 137

Н

Набор знаков 199

Написание тегов 190

Области для подготовки 140

Области для производства, CMS 140

Организация труда 33

Оформление текста

отображение шрифтов в браузерах 96

Очистка тела документа 199

Ошибка 404 82

П

- Палитра цветов 104
- Период работоспособности 237
- Персонал
 - количество 28
- План
 - составление 32, 33
- Подпись в электронных письмах 81
- Пользовательский агент 176
 - на основе текста 186
- Представление 177
- Презентационная разметка 179
- Проект 34
- Пропускная способность
 - стоимость 229

- Разметка
 - заголовка 196
 - таблицы 202
 - тела документа 199
 - чистка 194
- Регистраторы доменных имен 221
 - выбор 224
- Рецензенты 15

- Сайт
 - задачи 34
 - переделка 102
- Селекторы класса 209
- Совместимость с пользовательскими агентами 185
- Содержание сайта 26
- Сравнение хостинговых компаний 237
- Средства для создания сайта 28
- Стандарты 26, 175
 - презентационная разметка 179
 - хорошая разметка 178
- Стиль 88
 - дешевый 89

- Страница 76
 - ссылки 80
 - ter title 77

- Таблицы 202
- Таблицы стилей 203
 - изменение текста 208
 - компоновка страницы 208
 - края 207
 - междустрочные пробелы 207
 - оформление текста 206
 - свойства цвета 205
 - цвет 206
- Текст
 - выравнивание 90
 - заголовок 68
 - изображение 97
 - атрибут title 81
 - оформление 88
 - ошибки 75
 - перегруженный 67
 - представление 96
 - простота 73
 - размещение 95
 - слова 62
 - составляющие 63
 - цвет 91
 - цель 71
 - шрифт 89
 - традиционный 92
- Теги
 - закрытие 191
 - метатеги 191
 - написание 190

- Управление последовательностью выполняемых действий 139

- Хостинг 27
- Хостинговые компании 222

Ц

Цвет 104

Ч

Чистка разметки 194
 заголовка 196
 таблицы 202
 тела документа 199

Ш

Шаблоны 131
 Шрифты 89, 91
 без засечек 92
 декоративные графические 111
 декоративный и рукописный 93
 контроль благодаря изображениям 97
 последовательное использование 93
 с засечками 92
 символов 111

А

Access 167
 Aegir 151
 AssetNow 166
 g

Broad Vision 165

Q

cli part

вСети 109

CMS (Система управления сайтом) 128
 Contribute 168 •

возможности для написания и

редактирования 136

возможность соединения с базой

данных '140

^\/rc ice

коммерческие CMS 165

AssetNow 166

Easy Publish 167

WebWorld 165

контроль версий 129, 139

лучшие CMS для вашего бюджета 145

области для подготовки 140
 области для производства 140
 открытые CMS 148

Greymatter 159

Midgard 150

Movable Type 159

PHPmole 151

Textpattern 157

поддержка CSS 138

поддержка XHTML 138

стоимость 145

угрозы бюджету 130

управление последовательностью

выполняемых действий 139

шаблоны 131

Contribute 168

Easy Publish 167

G

Greymatter 159

"

HTML, чистка 201

,

Ю-селекторы 209

Ingeniux 165

Intranet 167

!•

Linux 167

IV

Macromedia Contribute 168

*MediaSurface 1165

Merant 165

-

Microsoft Access 228

Midgard 150

O

урениуре 14У

Paperthin 165

PHPmole 151

RSS (Rich Site Summary) 142

S

SSI (Server-Side Include) 135

T

Textpattern 157

The Web Standards Project 186

И

UNIX 167

w

W3C (World Wide Web Consortium) 176

Web-хостинг 221

WebWorld 165

X

XHTML

отделение структуры

от внешнего вида 189

типы документов 194