

4

Web-графика в ImageReady

Кратко о файлах и форматах

- Документы создаются с помощью команды **File⇒New** (Файл⇒Создать) и сохраняются как файлы Photoshop (в формате PSD) с помощью команды **File⇒Save As...** (Файл⇒Сохранить как...).
- Изображения открываются с помощью команды **File⇒Open** (Файл⇒Открыть) и могут быть самых разных форматов, в том числе Photoshop, GIF, JPEG, PNG, TIFF, PDF, BMP и PICT; для того чтобы сохранить их (обычно в формате Photoshop), используется команда **File⇒Save As...** (Файл⇒Сохранить как...).
- Для экспорта оптимизированных файлов в формате GIF, PNG и JPEG применяется команда **File⇒Save Optimized As...** (Файл⇒Сохранить оптимизированный файл как...).
- Для создания ролловеров, карт и фрагментации изображений с помощью команды **File⇒Export...** (Файл⇒Экспортировать...) создаются изображения и HTML-файл объединяющей их Web-страницы.

Основные функции

- Стандартные команды меню **File** (Файл) и **Edit** (Редактирование) — с. 26
- Справочная система Adobe и Adobe Online — с. 31
- Комбинации клавиш и контекстные меню — с. 34
- Панель инструментов — с. 36

- Плавающие стыкуемые палитры — с. 39
- Панель параметров — с. 42
- Палитра предыстории — с. 45
- Палитра действий — с. 47
- Слои — с. 52
- Подбор цвета — с. 533
- Ввод текста, стандартные средства форматирования символов и абзацев — с. 537
- Кривые Безье — с. 543
- Оптимизация изображений для Web — с. 553

Еще сравнительно недавно большинство изображений, создаваемых на компьютере, предназначалось для печати, и это обстоятельство учитывалось при разработке графических редакторов. Рост популярности World Wide Web привел к резкому увеличению количества изображений, предназначенных для показа только на экране. Дизайнеры, работающие для Web, далеки от многих проблем печати: им не приходится беспокоиться о различии цветовых моделей в мониторах и принтерах, цветоделении, черно-белых экранах, зернистости и других особенностях процессов, в которых участвуют настоящие чернила и бумага. Однако, как и следовало ожидать, вместе с Web появились новые проблемы, связанные с низким разрешением мониторов, необходимостью передачи изображений по относительно медленным каналам связи и новыми способами использования изображений в качестве динамических элементов Web-страниц.

Такое изменение требований, предъявляемых к графическим редакторам, имело два вида последствий. Во-первых, в уже существующие программы были встроены новые функции, удовлетворяющие потребности не только старых средств печати, но и Web-графики. Во-вторых, появились новые программы, разработанные специально для производства Web-графики. Они освобождены от лишнего багажа, связанного с печатью, и удовлетворяют только требованиям, предъявляемым Web. Как раз к таким программам и относится ImageReady. Мы рассмотрим ImageReady 3.0, так как именно эта версия поставляется в комплекте с Photoshop 6.

ImageReady занимает не совсем обычное положение. Первоначально это был самостоятельный графический Web-редактор, созданный на базе упрощенной версии Photoshop, практически с таким же интерфейсом и набором функций. В настоящее время ImageReady предоставляется только как интегрированное приложение-компаньон Photoshop. Более того, некоторые Web-функции ImageReady снова вернулись в Photoshop. Однако, несмотря на то, что окончательное слияние ImageReady и Photoshop представляется практически неизбежным, ImageReady как самостоятельный продукт сохраняет свою привлекательность для Web-специалистов, которым вполне хватает сокращенного ядра Photoshop в сочетании с дополнительными Web-функциями, если все это объединено в одном приложении.

Для того чтобы получить полное представление об ImageReady, вам следует ознакомиться не только с этой главой, но также с главами 3 и 10. Если вы планируете использовать ImageReady для коррекции и создания композиций из отсканированных изображений, то сначала прочтите описание Photoshop в главе 3. Большинство инструментов, палитр и команд меню, описанных в этой главе, есть также и в ImageReady, где они используются для тех же целей. Некоторые из этих команд находятся на другом месте, однако такие различия минимальны. Несмотря на то что в ImageReady файлы сохраняются главным образом в Web-форматах, обрабатываются они в формате Photoshop. При

работе с одним и тем же изображением можно в случае необходимости переходить из одного приложения в другое, как описано в разделе “Интеграция” (с. 58).

В ImageReady есть несколько функций, которых нет в Photoshop, и наоборот, некоторые функции Photoshop отсутствуют в ImageReady. Например, в ImageReady нет инструментов для рисования многоугольника и пера для создания векторных объектов — можно создавать только эллипсы, линии и прямоугольники (в том числе со скругленными углами), которые затем применяются в качестве векторных фигур для кнопок. Кроме того, инструментарий для работы с масками в ImageReady значительно беднее. Корректирующие слои, к сожалению, тоже отсутствуют, а разрешение изображений ImageReady всегда равно 72 пикселя на дюйм, что соответствует номинальному разрешению экрана. Поэтому диалоговое окно **Image Size** (Размер изображения) в ImageReady гораздо проще, чем в Photoshop, и размеры изображения в нем задаются только в пикселях (рис. 4.1). Нет здесь и средств управления цветом, так как ImageReady поддерживает только модель RGB. Все эти ограничения довольно красноречиво говорят о назначении ImageReady, как и тот факт, что команда настраиваемой печати **Print...** (Печать...) в меню **File** (Файл) заменена на подменю **Preview In** (Просмотреть в), где перечисляются все браузеры, установленные в операционной системе, в которых можно открыть создаваемое изображение.

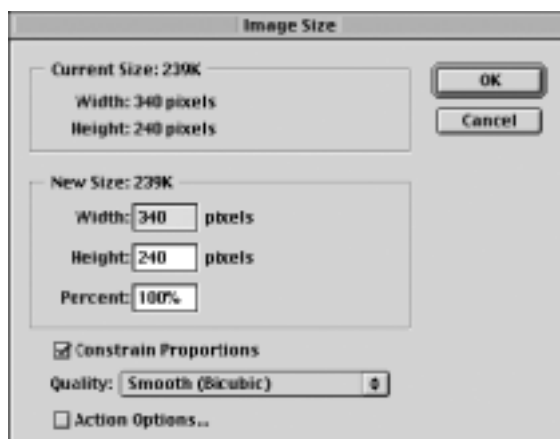


Рис. 4.1. Изменение размера изображения

Оптимизация изображений

Наиболее характерная операция, которую приходится выполнять с графическим файлом, прежде чем разместить его на Web-странице, — это *оптимизация*, т.е. преобразование файла в подходящий формат и задание различных параметров, определяющих степень сжатия и представление цветов. Цель оптимизации — уменьшить размер файла, чтобы он достаточно быстро загружался через Internet. Такая оптимизация выполняется в ImageReady, Photoshop и Illustrator примерно одинаково, несмотря на то, что лучше всего она реализована в главном приложении. Форматы Web-изображений, основные принципы и процедуры оптимизации для всех трех программ описаны в главе 10. В отличие от Illustrator и Photoshop, где для оптимизации используется специальное диалоговое окно **Save For Web** (Сохранить для Web), в ImageReady она интегрирована в другие функции. Соответствующие параметры находятся в палитре **Optimize** (Оптимизация), а окно документа разделено на четыре вкладки, на которых для сравнения ориги-

нала с оптимизированными версиями эти версии показаны отдельно, а также в окне, разделенном на две и четыре панели. Рисовать, выделять фрагменты, корректировать и выполнять другие операции можно только в исходной версии, хотя использовать для этого можно любую из вкладок. Например, при работе с оригиналом в двухпанельном окне можно отслеживать изменения в расположенной рядом оптимизированной версии. На рис. 4.2 показаны окно документа ImageReady и палитра Optimize (Оптимизация); сравните их с рис. 10.2.

☞ Откройте изображение в ImageReady и поэкспериментируйте с ним, используя палитру Optimize (Оптимизация). Более подробно свойства и упражнения по оптимизации изображений описаны в главе 10.

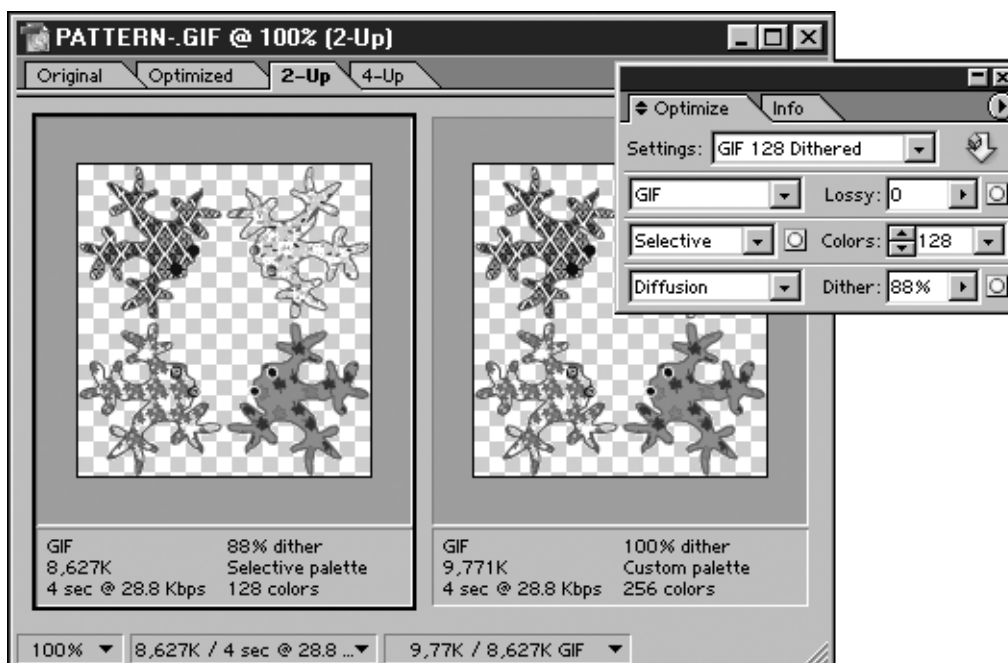


Рис. 4.2. Окно документа и палитра Optimize (Оптимизация)

Фрагментация изображений

Фрагменты изображения

В последнее время среди Web-дизайнеров получила распространение технология, которая заключается в разделении изображения на части, обычно называемые *фрагментами* (slices). Фрагменты хранятся в отдельных файлах и располагаются на Web-странице так, чтобы вновь составить единое изображение. Для этого их либо размещают в ячейках таблицы, либо используют позиционирование CSS. Благодаря фрагментации резко ускоряется загрузка, так как обычно с помощью команды поддерживает сразу несколько соединений. Однако этот прием срабатывает только в определенных условиях, в зависимости от доступной полосы пропускания, а также от загадочного, с трудом предсказуемо-

го поведения такой сложной сети, какой является Internet. Основные преимущества фрагментации проявляются несколькими способами.

Обычно Web-изображения хранятся в формате GIF или JPEG. (Третий формат, PNG, постепенно приобретает популярность, когда-нибудь заменит GIF.) Эти форматы описаны в главе 10, поэтому здесь мы только отметим, что в формате GIF хорошо и без потери качества сжимаются области однородного цвета. Однако в этом формате ограничено количество цветов (до 256), в то время как формат JPEG допускает использование миллионов цветов и обеспечивает эффективное сжатие изображений с плавными тональными переходами, таких как фотографии. Правда, при этом часть информации теряется, но это редко бывает заметно (за исключением сжатия с очень низким качеством для получения очень маленьких файлов). Однако четкость текста и резкость краев могут пострадать. У GIF-изображений есть еще несколько возможностей, отсутствующих у формата JPEG. В частности, GIF-изображения могут иметь прозрачные области, сквозь которые проступает фон Web-страницы или другое изображение; кроме того, из GIF-изображений можно создавать анимационные ролики (см. раздел “Анимация”, с. 188).

Благодаря фрагментации становится возможной различная степень сжатия фрагментов, а также сохранение одних фрагментов в формате GIF, а других — в JPEG. Можно также заменить некоторые фрагменты HTML-кодом. Таким образом, появляется возможность добавить к изображению текст, не создавая его в графическом редакторе, а также составить всю Web-страницу из одного фрагментированного изображения. При искусном дизайне одни и те же фрагменты можно использовать для составления нескольких изображений; время загрузки при этом сокращается, так как для повторного отображения используется уже загруженный файл. Фрагментированные изображения также являются альтернативой картам (см. раздел “Карты изображений”, с. 177), где изображение делится на функциональные зоны, реагирующие на действия пользователя.



Рис. 4.3. Фрагментация изображения

Многие фотографии состоят из объекта, размещенного на переднем плане, и фона. Пример такого изображения (старая борона на снегу) показан на рис. 4.3. В таких изображениях основной интерес обычно представляет центральный объект — в данном случае борона (и, возможно, ее тень). Изображение снега, служащее фоном, можно сжать сильнее, чем центральное, поскольку здесь потеря качества будет менее заметна. Разрезав изображение на части, чтобы отделить борону от того, что ее окружает (см. рис. 4.3), можно применить к полученным фрагментам разные уровни сжатия JPEG. В результате суммарный размер файлов с фрагментами будет меньше, чем у файла с целым изображением, так как для нефраgmentированного файла пришлось бы использовать сжатие с более

высоким качеством, чтобы сохранить важные детали бороны. Обратите внимание: фрагменты изображения должны быть прямоугольной формы. Именно поэтому центральный объект пришлось разделить на несколько фрагментов.

В рекламных плакатах и журналах часто встречается такой дизайнерский прием: центральные объекты, такие как продукция или модели, отделяют от фона путем маскирования и затем подкладывают под них однородный цвет. (Пример такого изображения показан на рис. 3.33.) Объекты обычно берутся с фотографий, поэтому лучше использовать для них сжатие JPEG. Обширные однотонные области, окружающие объект, лучше сжимать в формате GIF, чтобы сжатие с потерями JPEG затрагивало только ту часть изображения, которая в этом нуждается. GIF-области можно сделать прозрачными, однако, поскольку фрагменты могут иметь только прямоугольную форму, сделать весь фон прозрачным все равно не удастся — для этого необходимо, чтобы все изображение было представлено в формате GIF или PNG.

Фрагменты в формате GIF и JPEG полезно комбинировать и в том случае, если на изображение предполагается наложить текст. При JPEG-сжатии нарушается четкость букв, а наложить на изображение текст с помощью слоев CSS не всегда возможно (например, потому, что текст является частью исходного изображения). Однако можно фрагментировать изображение и сохранить фрагмент с текстом в формате GIF. (Разумеется, так стоит поступать только тогда, когда текст составляет небольшую часть изображения.)

Фрагментировать изображение в обычном графическом редакторе весьма утомительно. Это касается и последующей обработки фрагментов — например, переноса их границ и изменения степени сжатия. В приложениях Web-графики есть средства, упрощающие решение этой задачи, так как в них автоматизированы многие операции и есть удобный интерфейс для управления всем процессом.

Главной целью фрагментации является разная степень сжатия фрагментов изображения, но фрагменты, как и описанные ранее слои, оказались более разносторонней концепцией, которая получила в ImageReady другие способы применения. Фрагментированное изображение — это, в сущности, набор изображений, которые в одних случаях можно рассматривать как нечто целое, а в других — по отдельности. Например, фрагменты могут по-разному реагировать на события, такие как попадание на них курсора. С фрагментами можно связывать URL-адреса, и тогда, щелкнув на этих частях изображения, пользователь сможет открывать в браузере различные документы. Как мы покажем далее, сочетание этих эффектов приводит к некоторым известным методам Web-навигации. Наконец, отдельные фрагменты можно превратить в анимационные ролики, в то время как остальное изображение останется статичным.

Создание фрагментов



В ImageReady есть два способа разделения изображения на фрагменты. Более очевидный и интуитивно понятный заключается в том, чтобы начертить границы фрагментов с помощью инструмента **Slice** (Нож). Выбрав этот инструмент¹, переместите его в то место, где будет левый верхний угол фрагмента, нажмите кнопку мыши и перетащите курсор. На экране появится прямоугольник; его нижний правый угол будет создан в той точке, где вы отпустите кнопку мыши. Этот прямоугольник называется *пользовательским*, т.е. явно заданным фрагментом. В примере, показанном на рис. 4.3, можно было начать с создания большого

¹ Правда, на вид он больше напоминает кинжал, но нам кажется, что авторы хотели изобразить десертный нож.

прямоугольника, включающего в себя основную часть бороны. Для того чтобы на Web-странице фрагменты снова сложились в целое изображение, необходимо разделить на прямоугольные области и остальные части. (Фрагменты могут быть только прямоугольными, поскольку каждый из них представляет собой отдельное изображение.) В ImageReady это делается автоматически. В данном случае требуется создать три дополнительных фрагмента (рис. 4.4). Такие фрагменты, создаваемые в ImageReady программно, называются *автофрагментами*.

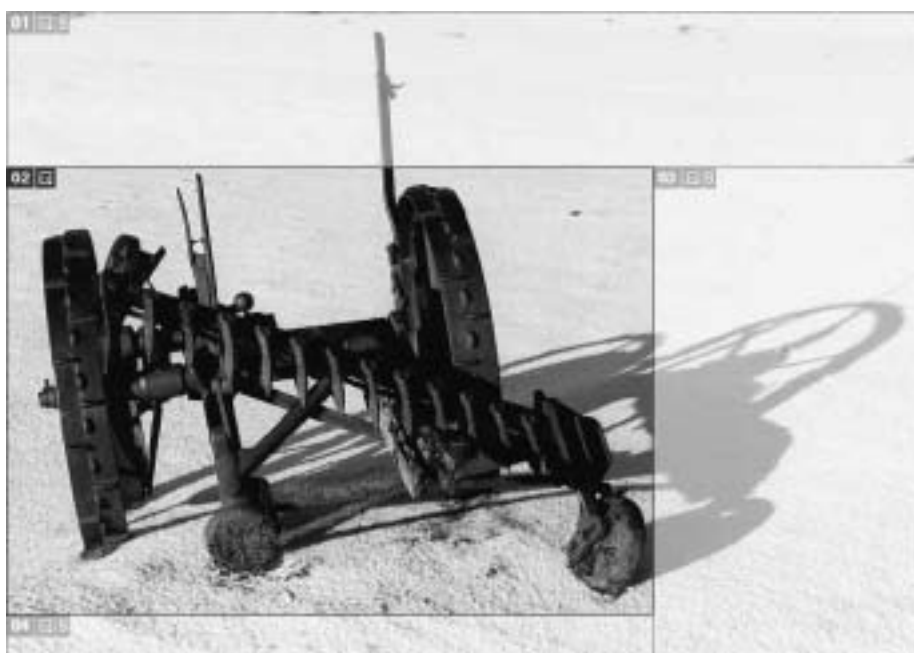


Рис. 4.4. Пользовательский фрагмент и автофрагменты

В изображении может быть несколько пользовательских фрагментов, так как там может быть несколько областей, интересующих пользователя, либо зон, которые требуется выделить для анимации, а также для повышенной или, наоборот, пониженной степени сжатия. Возможно также, что, как в нашем примере, для покрытия объекта сложной формы необходимо несколько фрагментов. Новые пользовательские фрагменты создаются с помощью мыши. При этом также создаются дополнительные и изменяются существующие автофрагменты, чтобы изображение по-прежнему состояло из точно подходящих друг к другу фрагментов. При создании перекрывающихся друг друга пользовательских фрагментов автоматически появляются более мелкие фрагменты, покрывающие область пересечения.

По умолчанию при фрагментации изображения фрагменты отображаются в окне документа так, как показано на рис. 4.4. Границами пользовательских фрагментов служат непрерывные линии, а автофрагментов — пунктирные. Каждый фрагмент имеет свой номер, который отображается в его верхнем левом углу, рядом с пиктограммой, обозначающей другие свойства фрагмента. Невыделенные фрагменты (о которых вы узнаете чуть позже) выглядят тускло, причем автофрагменты — более тускло, чем пользовательские. Общая картина получается несколько беспорядочной; воспользовавшись командой **Edit**⇒**Preferences**⇒**Slices...** (**Правка**⇒**Параметры**⇒**Фрагменты...**) и включив режим **Show**

Slices Only (Показывать только фрагменты), можно отключить отображение лишней информации и оставить видимыми только границы фрагментов.

☞ Откройте изображение объекта, четко выделяющегося на фоне, и разделите его на фрагменты, чтобы изолировать объект. Попробуйте добавить к изображению дополнительные пользовательские фрагменты. Как при этом изменилась структура автофрагментов?

При сохранении оптимизированной версии фрагментированного изображения для использования на Web-странице ImageReady создает отдельные файлы для каждого фрагмента и HTML-файл с кодом, необходимым для отображения всего изображения. Подробнее об этом читайте в главе 10.



При фрагментации изображения для разной степени сжатия отдельных частей необходимо выделить по очереди каждый фрагмент. Для этого используется инструмент выделения фрагментов, обычно скрытый на панели инструментов под инструментом **Slice** (Нож). Для того чтобы выделить фрагмент, щелкните в любой его точке. Для того чтобы выделить еще несколько фрагментов, щелкните на них, удерживая нажатой клавишу <Shift>. Выделив несколько фрагментов, перейдите к палитре **Optimize** (Оптимизация), выберите формат, в котором будут сохранены эти фрагменты, и настройте свойства сохраняемого изображения. О том, как это делается, рассказывается в главе 10. Если вы хотите, чтобы несколько фрагментов имело одинаковые свойства, их можно связать; для этого фрагменты выделяют и выбирают из меню **Slices** (Фрагменты) команду **Link Slices** (Связать фрагменты). После этого все параметры оптимизации, назначаемые одному из связанных фрагментов, будут распространяться и на остальные. Все автофрагменты изображения автоматически считаются связанными.

☞ Обращаясь по мере необходимости к главе 10, задайте требуемые параметры оптимизации для каждого из фрагментов изображения, созданных в предыдущем упражнении. Просмотрите оптимизированное изображение в Web-браузере.

Инструмент выделения фрагментов применяется также для их переноса и изменения размера. Выделенный фрагмент переносят обычным перетаскиванием. Для того чтобы изменить размер, нужно перетащить один из восьми манипуляторов, которые появляются при выделении фрагмента. Можно также разделить фрагмент на несколько частей по горизонтали или по вертикали, используя команду **Slices**⇒**Divide Slice...** (Фрагмент⇒Разделить фрагмент...). При этом откроется диалоговое окно, показанное на рис. 4.5. В этом окне указывается количество фрагментов, на которые делится данный фрагмент по горизонтали, по вертикали или в обоих направлениях, на равные части либо на части заданной высоты или ширины (при этом они не обязательно должны делиться нацело — остаток образует маленький дополнительный фрагмент, расположенный внизу или справа). Все эти операции определяют способ разбиения изображения на части, но никак не влияют на их содержимое — например, при переносе фрагмента не смещается ни один пиксель, изменяется лишь способ разбиения. Фрагменты можно также объединять друг с другом. Если выделить несколько фрагментов, то с помощью команды **Slices**⇒**Combine Slices** (Фрагменты⇒Объединить фрагменты) можно создать новый фрагмент, представляющий собой наименьший прямоугольник, заключающий в себе выделенные фрагменты. Все остальные фрагменты, попавшие в зону выделения, удаляются, а структура автофрагментов приводится в соответствие с внесенными изменениями.

☞ Потренируйтесь изменять фрагментацию изображения, изменяя размеры фрагментов, деля на части одни фрагменты и объединяя другие. Попробуйте создавать фрагменты заданной высоты и ширины.

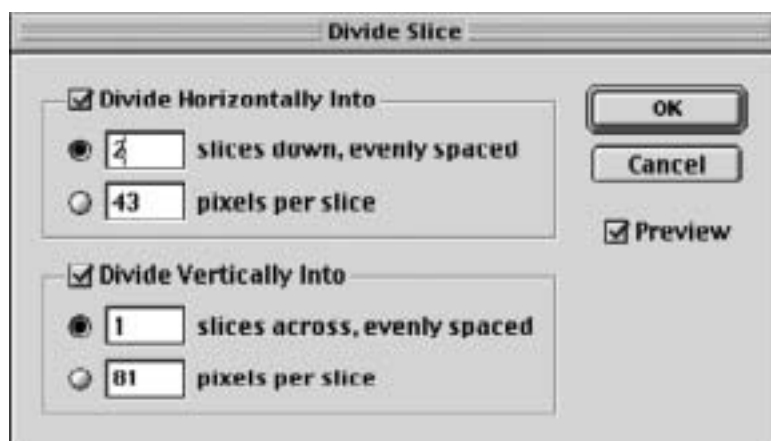


Рис. 4.5. Разбиение фрагмента

Альтернативой созданию пользовательских фрагментов с помощью инструмента **Slice** (Нож) является *фрагментация на базе слоев*. В некотором отношении с такими фрагментами, созданными на базе слоев, проще работать, особенно при создании из фрагментов ролловеров, как будет описано далее.

Фрагмент на базе слоя создается очень просто: нужно выделить слой в палитре **Layers** (Слои) и воспользоваться командой меню **Layer**⇒**New Layer Based Slice** (Слой⇒Новый фрагмент из слоя). Созданный фрагмент будет представлять собой наименьший прямоугольник, необходимый для покрытия всех непрозрачных пикселей данного слоя. При создании коллажа или композиции таким образом можно без труда превратить в фрагмент любой элемент изображения, под который был отведен отдельный слой.

Фрагменты, созданные на базе слоев, отличаются от пользовательских тем, что при изменении изображения, размещенного на слое, размер фрагмента автоматически изменится соответствующим образом. Вследствие этого нельзя вручную изменить размер фрагмента на базе слоя или разделить его. Для этого необходимо прежде преобразовать фрагмент в пользовательский с помощью команды **Slices**⇒**Promote to User-slice** (Фрагменты⇒Перевести в пользовательский). После этого фрагмент больше не будет изменяться автоматически в соответствии с изменениями слоя, зато вы сможете вручную изменять его размер, перемещать, делить на части и т.п., как любой пользовательский фрагмент.

☞ Откройте или создайте изображение, состоящее из нескольких слоев, и создайте фрагменты на базе каждого слоя. Внесите несколько изменений, затрагивающих один из слоев, — нарисуйте или удалите что-нибудь. Как изменились границы фрагментов?

Если же в конце концов вы решите, что фрагментировать изображение не стоило, то для удаления всех фрагментов и восстановления изображения в исходном виде воспользуйтесь командой **Slices**⇒**Delete All** (Фрагменты⇒Удалить все).

С помощью палитры **Slices** (Фрагменты), показанной на рис. 4.6, можно выполнить некоторые дополнительные действия. Самое полезное из них — привязка к фрагменту URL-адреса, чтобы, когда пользователь щелкнет на этом фрагменте, открывался определенный документ. URL вводится в текстовом поле **URL** полностью, начиная с `http://` или другого подобного префикса. Возможность поиска файла по относительному адресу URL не предусмотрена, однако все URL-адреса, использованные в ImageReady, заносятся в раскрывающийся список, связанный с полем **URL**, так что повторно вводить адрес не обязательно — его можно выбрать из этого списка. В поле **Target** (Целевой объект) указывается фрейм или окно, в котором браузер откроет документ, находящийся по заданному адресу. В раскрывающемся списке, связанном с этим полем, предлагаются стандартные варианты целевых объектов, используемые при работе с фреймовыми структурами: `_blank` — открытие документа в новом окне; `_self` — отображение документа в том же окне или фрейме, где содержится ссылка; `_parent` — замена фреймовой структуры, содержащей фрейм со ссылкой, окном с документом; `_top` — замена окном с документом всего окна, в котором содержалась ссылка. В этом текстовом поле можно также ввести имя фрейма, если фрагментированное изображение входит в состав фреймовой структуры с именованными фреймами. Подобные ситуации возникают тогда, когда изображение содержит навигационные ссылки, помогающие пользователям перемещаться по сайту.

☞ Преобразуйте одно из изображений, фрагментированных вами в предыдущих упражнениях, в карту сайта, назначив каждому фрагменту свой URL. Протестируйте карту в Web-браузере.

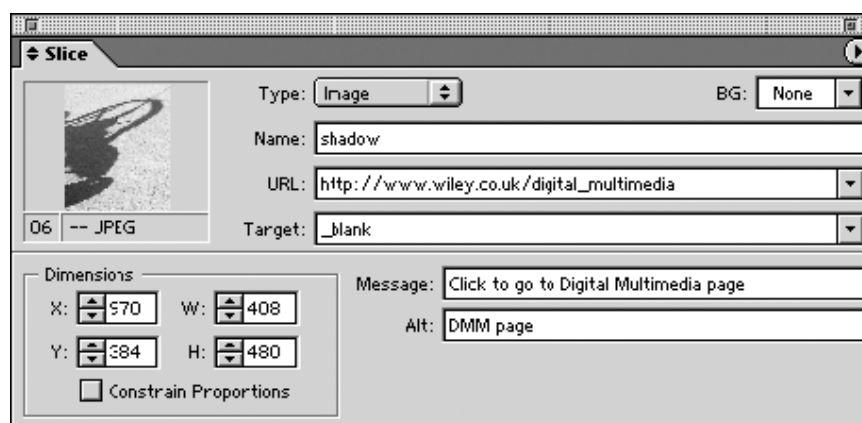


Рис. 4.6. Палитра *Slice* (Фрагмент)

С помощью палитры **Slice** (Фрагмент) также задают фоновый цвет для отдельных фрагментов, используя раскрывающийся список **BG**, расположенный в верхнем правом углу палитры. Цвет фона в ImageReady не отображается, так как устанавливается средствами браузера. Каждому фрагменту можно присвоить имя, соответствующее его содержанию. Это полезно, когда в изображении есть несколько ключевых фрагментов. С помощью полей, расположенных в нижней части палитры, можно точно задать размер и положение пользовательского фрагмента, введя цифровые значения соответствующих

координат². (Для фрагмента на базе слоя это сделать нельзя — необходимо предварительно преобразовать его в пользовательский.) Можно также ввести сообщение, которое будет отображаться в строке состояния браузера, когда курсор будет попадать на данный фрагмент, а также альтернативный текст (в терминах HTML — значение атрибута alt), отображаемый на месте фрагмента теми браузерами, которые не воспроизводят графику или в которых такая возможность отключена. При разумном использовании альтернативный текст для изображений значительно повышает доступность Web-страниц.

Наконец, с помощью палитры **Slice** (Фрагмент) можно заменить изображение, находящееся в пределах фрагмента, на HTML-код. Если в раскрывающемся списке **Type** (Тип), расположенном вверху палитры, выбрать тип фрагмента **No Image** (Без изображения), то верхняя часть палитры изменится так, как показано на рис. 4.7. В открывшемся текстовом поле можно ввести текст с дескрипторами HTML. При воспроизведении фрагментированного изображения этот текст появится на месте выбранного фрагмента: он разместится в соответствующей ячейке таблицы вместо графического файла. Программа никак не помогает при вводе HTML-кода, так что все приходится вводить вручную; кроме того, здесь нет никакой возможности проверить введенный код и убедиться, что там нет ошибок. (Возможно, имеет смысл ввести в ImageReady лишь некоторый заменитель настоящего текста, а затем отредактировать HTML-файл, созданный вместе с изображениями, в HTML-редакторе или в программе Web-дизайна.)

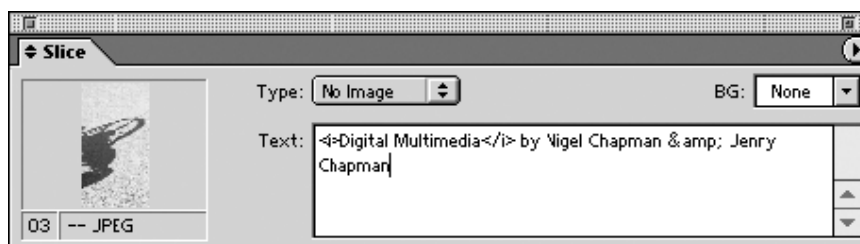


Рис. 4.7. Создание текстового фрагмента

Если изображение не фрагментировано, то с помощью палитры **Slice** (Фрагмент) можно задать цвет фона, URL, сообщение в строке состояния и альтернативный текст для всего изображения.

☞ Потренируйтесь использовать палитру **Slice** (Фрагмент) для ввода сообщения в строке состояния и альтернативного текста к фрагментам, задания цвета фона для прозрачных фрагментов и создания в изображении текстовых HTML-фрагментов.

Карты изображений

Web-страницам присущ уникальный способ использования графики — создание *карт изображений*. Карты изображений — это структуры, содержащие активные области, называемые *активными зонами* (hot spot), с которыми связаны URL-адреса. Если щелкнуть в такой активной зоне, то браузер загрузит новую страницу, расположенную по указанному адресу. Классический пример карты изображения — настоящая географическая

² Для отображения всех полей иногда приходится использовать стрелки, расположенные на вкладке палитры.

карта, активные зоны которой соответствуют достопримечательностям для туристов: если щелкнуть в такой точке, браузер откроет страницу с более подробным описанием данной достопримечательности. Менее буквальные примеры карт изображений представляют собой абстрактные структуры или более конкретные изображения и часто применяются для общей навигации по Web-сайту.

Благодаря резкому увеличению количества приложений для обработки Web-графики с удобными инструментами фрагментации стало довольно распространенным явлением создание карт изображений путем назначения HTML-ссылок некоторым или всем фрагментам. (Для этого дескриптор `img`, используемый для размещения фрагмента изображения на Web-странице, помещается внутри дескриптора `a` с атрибутом `href`.) Если же вы не хотите фрагментировать изображение или желаете использовать в качестве активных зон более сложные фигуры, чем те, которые можно получить с помощью фрагментации, воспользуйтесь дескрипторами и атрибутами HTML, предназначенными для создания карт изображений.

Если не углубляться в подробности написания HTML-кода, для обработки карты изображения браузером (а в настоящее время это обычно происходит именно так) нужно создать карту, описав с помощью координат множество активных зон — прямоугольников, многоугольников и кругов — и назначив каждому URL. Затем с помощью специального атрибута карта ассоциируется с изображением. Все это вполне можно сделать вручную, введя HTML-код в тестовом редакторе. Однако даже если вы умеете быстро и правильно вводить HTML-дескрипторы и их атрибуты, вам все равно необходимо откуда-то узнать точные геометрические координаты всех активных зон. Будет гораздо быстрее нарисовать активные зоны на изображении с помощью соответствующих векторных инструментов рисования, а затем программно преобразовать эти фигуры в нужный HTML-код. Изменять положение и форму активных зон тоже удобнее геометрически. Утилиты для создания подобных карт изображений появились практически одновременно с соответствующими средствами HTML. Такие инструменты все чаще встраиваются в программы обработки изображений и считаются обязательным элементом любого приложения для работы с Web-графикой.

Поскольку карта изображения — это изображение с активными зонами, при обращении к которым в Web-браузер загружается новая страница, фрагментированное изображение с назначенными некоторым или всем его фрагментам URL-адресами тоже является картой. Однако обычно, как мы уже объясняли, под картой изображения понимается другое. Чаще всего это выражение означает единое, нефраgmentированное изображение, с которым связан HTML-элемент `map`, описывающий геометрию активных зон.



В ImageReady применяются отдельные инструменты для создания каждой из трех форм активных зон. Все три инструмента занимают общее место на панели, вместе с инструментом выделения активной зоны (рис. 4.8). Для того чтобы создать активную зону, нужно выбрать соответствующий инструмент и нарисовать фигуру на изображении. Инструменты создания круглой и прямоугольной зон перетаскиваются по диагонали, подобно инструменту **Slice** (Нож) для прямоугольных фрагментов. Инструмент создания активной зоны в форме многоугольника используется иначе: курсор помещают в точку, где должна находиться одна из вершин (углов), щелкают кнопкой мыши и переходят к следующей вершине. При перемещении курсора за ним протягивается прямая линия, соответствующая стороне многоугольника. Щелкнув в точке, где должна находиться следующая вершина, курсор снова перемещают, создавая новую сторону фигуры. Для того чтобы замкнуть многоугольник, нужно начертить его последнюю сторону

так, чтобы она заканчивалась в исходной вершине, или дважды щелкнуть, чтобы многоугольник замкнулся автоматически. Как видно из этого описания, многоугольники могут быть неправильными: длина их сторон не всегда одинакова.



Рис. 4.8. Инструменты создания карты изображения



После того как на изображении будет создано несколько активных зон, можно назначить им URL-адреса с помощью палитры **Image Map** (Карта изображения), показанной на рис. 4.9. Значения, вводимые в этой палитре, относятся к активной зоне, выделенной в текущий момент; выделяются активные зоны, как обычно, с помощью инструмента выделения карты изображения. Как видим, палитра карты изображения похожа на палитру фрагментации, только меньше размером. URL и целевой объект вводятся точно так же; то же касается альтернативного текста для Web-браузеров, не отображающих графику. В этой палитре также задаются точный размер и положение прямоугольных и круглых активных зон путем ввода цифровых значений в левой части палитры. Однако изменить таким образом параметры активной зоны, имеющей форму многоугольника, нельзя.

☞ Откройте любое изображение и начертите на нем активные зоны всех форм. Потренируйтесь создавать многоугольные зоны с разным количеством сторон. Назначьте каждой активной зоне URL и альтернативный текст. Протестируйте карту изображения в Web-браузере.

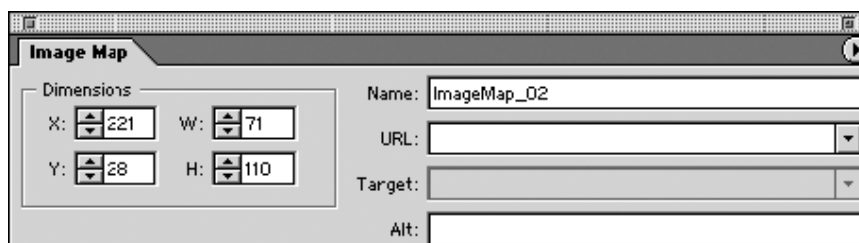


Рис. 4.9. Палитра **Image Map** (Карта изображения)

Сходство между активными зонами карты изображений и фрагментами гораздо больше, чем подобие соответствующих палитр. Как и фрагменты, активные зоны можно создавать на базе слоев. Для этого нужно выделить слой в палитре **Layers** (Слой) и воспользоваться командой **Layer ⇒ New Layer Based Image Map Area** (Слой ⇒ Создать на базе слоя активную область карты изображения). Будет создана прямоугольная активная зона, включающая в себя все непрозрачные пиксели выделенного слоя. Если выделить в палитре карты изображения активную зону, созданную таким образом, появится возможность изменить ее форму. Как показано на рис. 4.10, в левой части палитры появляется раскрывающийся список с вариантами формы активной зоны: **Rectangle** (Прямоугольник), **Circle** (Круг) и **Polygon** (Многоугольник). Для многоугольника задается допуск (максимальное количество сторон), определяющий, насколько данная активная зо-

на будет близка к контурам изображения, расположенного на слое. Чем больше допуск, тем точнее активная зона повторяет контур изображения.

☞ Откройте изображение, состоящее из нескольких слоев, и постройте по ним карту изображения. Потренируйтесь использовать разные значения допуска, чтобы активные зоны соответствовали содержимому каждого слоя. Назначьте каждой активной зоне, созданной на базе слоя, URL и альтернативный текст.

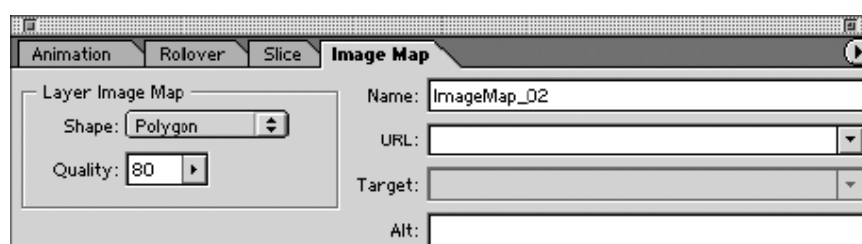


Рис. 4.10. Изменение формы активной зоны, созданной на базе слоя

Учитывая сходство активных зон и фрагментов, вряд ли стоит удивляться, что с помощью инструмента выделения карты изображения можно перемещать активные зоны, перетаскивая их, и изменять их размеры, перемещая манипуляторы.

Активные зоны могут перекрывать друг друга, но если щелкнуть в области перекрытия, то на действие мыши отреагирует только самая верхняя из зон, т.е. та, которая была создана последней.

Обычно применяют либо фрагментацию, либо карты изображений, однако в действительности ничто не мешает использовать и то и другое одновременно. Активные зоны могут даже перекрывать границы фрагментов. ImageReady — достаточно интеллектуальная программа, чтобы ассоциировать карту изображения с каждым фрагментом, на который попадает активная зона.

Ролловеры

После внедрения JavaScript в Netscape Navigator 2.0, а затем более широкой поддержки сценариев и DOM (Document Object Model — объектная модель документа), разработанной Microsoft под названием *динамического HTML*, которая затем была формализована консорциумом World Wide Web в виде DOM Recommendations, роль изображений на Web-страницах изменилась: они стали меняться динамически. Другими словами, если на Web-странице размещено изображение, то сценарий, активизируемый определенным событием, может поставить на его место другое изображение. Это обстоятельство легло в основу Web-технологии, применяемой уже практически повсеместно, а именно *ролловеров* (rollovers).

Ролловер представляет собой изображение, изменяющееся при наступлении некоторого события. Самым распространенным из таких событий является наведение на изображение курсора, после чего оно “переворачивается”, сообщая таким образом пользователю о том, что, если он щелкнет в этом месте, то что-то произойдет. Такие ролловеры состоят из двух отдельных изображений — для исходного и “перевернутого” состояния — и некоего кода на JavaScript, меняющего изображения местами, когда на них наводят курсор, и возвращающего все на место, когда курсор перемещают вне области изображе-

ния. Более сложные ролловеры меняются в ответ на большее число событий, состоят из нескольких изображений и имеют более сложный сценарий, однако основной принцип сохраняется. Создать ролловер нетрудно — нужно лишь немного знать JavaScript (однако заставить его работать во всех браузерах и на всех платформах несколько сложнее). Однако, как и фрагментация изображений, этот процесс может оказаться утомительным и чреватым ошибками. Если же вы совсем не знаете и не желаете знать JavaScript — это ваше право (и многие им пользуются), то для создания ролловера потребуются некий метод, скрывающий написание сценария от глаз пользователя. Поскольку сценарии ролловеров представляют собой стереотипные последовательности команд, они хорошо генерируются автоматически компьютерными программами. Те же программы выполняют вычисления, необходимые для того, чтобы изображения, связанные сценарием, образовали ролловер.

Принцип, лежащий в основе ролловеров, легко применим и для получения эффекта, когда при перемещении курсора по изображению одно изображение заменяется другим. Такой эффект называется *вторичным*, или *удаленным*, ролловером. Другой вариант того же приема — *анимационный* ролловер, когда при наведении курсора на изображение оно начинает двигаться. Это делается путем замены исходного изображения анимационным GIF (подробнее об этом — в следующем разделе) или с помощью дополнительного сценария, превращающего ролловер в анимационный ролик путем замены исходного изображения последовательно сменяющимися друг друга анимационными кадрами.

Вероятно, чаще всего ролловеры применяются для идентификации навигационных кнопок, используемых для перехода к разным частям Web-сайта. Такие кнопки часто объединяются в *панель навигации*. Один из простых способов организации навигационной панели с красиво расположенными кнопками заключается в фрагментации изображения и создании кнопок из отдельных фрагментов. На основе фрагментированного изображения сравнительно легко создать вторичный ролловер, когда одновременно с изменением одного фрагмента появляется другой, соответствующий событию ролловера. Например, при попадании курсора на фрагмент с именем главного раздела сайта может появиться дополнительный фрагмент, содержащий меню с перечнем его подразделов.

В основе ролловеров лежит довольно общая идея: в момент определенного события одно изображение заменяется другим. Однако для большинства практических применений эта слишком общая идея. Обычно ролловеры используются более систематично. Заменяющее изображение чаще всего является лишь слегка измененным вариантом исходного. Например, предположим, что ролловер используется для кнопки, с помощью которой пользователи переходят на Web-страницу загрузки программного обеспечения. Такой кнопке достаточно подписи, например **Downloads**. При наведении на нее курсора надпись может менять цвет или выделяться другим способом, давая понять пользователю, что в этот момент кнопка отреагирует на щелчок. Вряд ли пользователь поймет подобный намек, если надпись превратится во что-то совсем постороннее, вроде изображения рыбки³.

В ImageReady можно создавать ролловеры практически с любыми изменениями изображений, однако в программе есть удобные и сравнительно простые способы создания более структурированных ролловеров того типа, который чаще всего встречается на Web-страницах.

³ Конечно, человек привыкает ко всему, но если вы хотите, чтобы пользователи сразу понимали назначение элементов навигации сайта, не нуждаясь в дополнительных инструкциях, избегайте подобной “рыбной” экзотики.

Мгновенные ролловеры

Ролловеры в ImageReady представляют собой фрагменты или активные зоны карт изображений, реагирующие на события мыши. (Упрощенный случай — когда ролловер применяется ко всему нефрагментированному изображению.) В любой момент ролловер находится в одном из *состояний*, определяемом последним происшедшим событием. Основную часть времени ролловер находится в *нормальном* состоянии, когда ничего не происходит. Когда курсор попадает на изображение, ролловер переходит в *активное* состояние, а при нажатии кнопки мыши — в состояние *нажатия*. Каждому состоянию соответствует свое изображение кнопки, определяемое в палитре **Layers** (Слой): какой слой виден, какие эффекты к нему применяются, и т.п. (Возможно, вам стоит перечитать описание слоев и стилей слоев в Photoshop в главе 3. Палитра **Layers** (Слой) в ImageReady такая же, как и в Photoshop, за исключением незначительных различий в меню палитры.) В действительности каждое состояние ролловера — это абстракция, представляющая собой набор параметров палитры **Layers** (Слой). Для того чтобы продемонстрировать, что означает это довольно загадочное утверждение, мы опишем простую процедуру создания типичного ролловера.

Предположим, что мы хотим разместить на Web-странице слово **Downloads** и превратить его в ролловер, чтобы оно отбрасывало тень, когда на него попадает курсор, и меняло цвет при нажатии кнопки мыши (рис. 4.11). Для этого мы поступим следующим образом. Прежде всего создадим текстовый слой с надписью **Downloads**, выполненной подходящим шрифтом. Затем создадим на базе этого слоя фрагмент (см. раздел “Фрагментация изображений”, с. 170). Фрагменты на базе слоя удобны при создании ролловеров, так как изменения при добавлении новых состояний не влияют на размер изображения, содержащегося во фрагменте. Для того чтобы впоследствии было удобно обращаться к фрагменту, лучше присвоить ему имя, например **Download Button**. Выделив фрагмент, превратим его в ролловер с помощью палитры **Rollover** (рис. 4.12).



Рис. 4.11. Простой ролловер

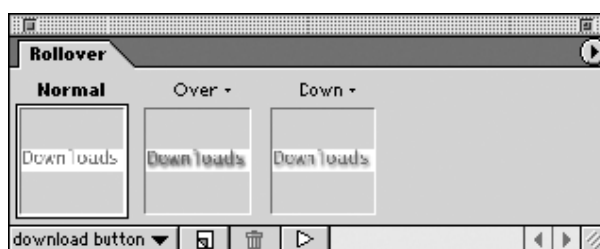


Рис. 4.12. Палитра Rollover




Вначале палитра **Rollover** содержит один мини-слайд изображения — с исходным текстом, из которого мы только что сделали фрагмент. Это нормальное состояние ролловера. Для того чтобы добавить новые состояния, нужно щелкнуть на кнопке **New** (Создать), расположенной внизу палитры. Первое из созданных таким образом состояний называется **Over**, а второе — **Down**. Эти состояния образуют простой ролловер, графически реагирующий на наведение курсора на изображение и на нажатие кнопки мыши. Для того чтобы выделить состояние, нужно щелкнуть

на соответствующем ему мини-слайде, и выделенный мини-слайд будет заключен в рамку. Все изменения в палитре **Layers** (Слои) при выделенном состоянии отражаются на изображении, когда роlover переходит в это состояние, но в других состояниях не видны. Например, если при выделенном состоянии **Over** применить стиль к слою, на основе которого был создан фрагмент роloverа, то этот стиль слоя будет применяться тогда, когда курсор находится на фрагменте.

В нашем примере, для того чтобы надпись отбрасывала тень, можно выделить состояние **Over** и воспользоваться командой **Layer⇒Layer Style⇒Drop Shadow** (Стиль⇒Стиль слоя⇒Отбросить тень). Затем выделим состояние **Down** и применим перекрытие цветов с помощью команды **Layer⇒Layer Style⇒Color Overlay** (Стиль⇒Стиль слоя⇒Перекрытие цветов), чтобы изменить цвет фрагмента. Вот и все, что нужно для создания эффекта роloverа. Однако обычно роlover должен быть не только красивым, но и информативным — сообщать, что если щелкнуть в этом месте, то что-то произойдет. Для того чтобы это “что-то” произошло, фрагменту назначают URL с помощью палитры **Slice** (Фрагмент).



Для предварительного просмотра роloverа нужно щелкнуть на кнопке **Play** (Просмотр) в палитре **Rollover** и передвинуть курсор на изображение в окне документа. После того как мы добавили к фрагменту **Download Button** новые состояния, при наведении курсора на надпись при просмотре роloverа у надписи появится тень — та самая, которую мы добавили в состоянии **Over**, — а если щелкнуть кнопкой мыши, то цвет надписи изменится в соответствии с перекрытием цветов, заданным в состоянии **Down**. Для просмотра новой кнопки необходимо воспользоваться Web-браузером. Если щелкнуть на кнопке предварительного просмотра, которая находится на панели инструментов, то фрагменты изображения вместе с HTML-файлом, содержащим объединяющие их дескрипторы, и кодом JavaScript, описывающим работу роloverа, сохранятся во временной папке и откроются в браузере, используемом по умолчанию. Роlover должен работать так же, как и при просмотре в ImageReady; кроме того, если щелкнуть на созданной кнопке, должен открыться документ, расположенный по URL-адресу, ассоциированному с фрагментом. Лучше всего просмотреть роlover в как можно большем количестве браузеров, так как не все они одинаково интерпретируют код JavaScript (у каждого браузера — свои ошибки). В раскрываемом меню, связанном с кнопкой предварительного просмотра, перечислены все браузеры, установленные в операционной системе, так что для просмотра можно выбрать любой из них.

 Потренируйтесь создавать кнопки с роloverами, применяя к текстовым слоям различные эффекты. Какие эффекты слоев лучше всего отражают реакцию кнопки? Просмотрите полученные роloverы во всех доступных для вас браузерах.



Панель навигации можно создать из расположенных соответствующим образом фрагментов одного документа, каждый из которых является отдельным роloverом. Когда в роли навигационной панели выступает несколько роloverов, для них обычно используют одни и те же визуальные эффекты. Если при наведении курсора на кнопку **Downloads** у нее появляется тень, то было бы логично ожидать от соседней кнопки **Contact Us** такого же поведения. Вместо того чтобы в точности повторять процедуру создания роloverа каждый раз, когда вы хотите, чтобы очередной фрагмент вел себя, как первый, в ImageReady создают *стили роloverов*, объединяющие в себе все аспекты роloverа — его состояния и эффекты, применяемые в каждом из них. Для этого нужно выделить фрагмент

роллера и выбрать из меню палитры **Style (Стиль)** команду **New Style...** (Создать стиль...). Откроется диалоговое окно, показанное на рис. 4.13. Если включить в нем все режимы, будет создан стиль роллера, который затем появится в палитре **Style (Стиль)** в виде образца. О том, что это именно стиль роллера, свидетельствует черный треугольник в верхнем левом углу пиктограммы.

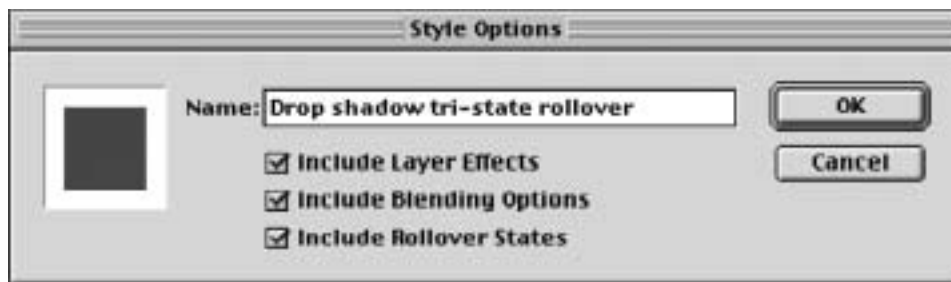


Рис. 4.13. Создание нового стиля роллера

Предположим, что после создания роллера для кнопки **Downloads**, как описано выше, мы создали на его основе стиль роллера и назвали его, скажем, **Drop Shadow tri-state** (Трехступенчатая тень). Для того чтобы создать другую кнопку с таким же поведением, нужно создать новый фрагмент из соответствующим образом расположенного текстового слоя, содержащего нужную нам надпись, такую как **Contact Us**. Затем, выделив этот фрагмент, нужно щелкнуть на образце, представляющем роллер **Drop Shadow tri-state** в палитре **Style (Стиль)**. (Когда курсор наведен на образец стиля, рядом с ним появляется экранная подсказка с именем этого стиля.) Вот и все — остается только назначить фрагменту URL, чтобы превратить роллер в гиперссылку.

☞ Выберите один из эффектов роллера, созданных вами в предыдущих упражнениях, сохраните его как стиль и примените еще к нескольким текстовым слоям, чтобы получилась навигационная панель.

Библиотеки стилей, поставляемые в комплекте с ImageReady, включают в себя несколько стилей роллеров. Некоторые из них весьма полезны, так что самый простой способ создать роллер — это применить один из них к фрагменту, содержащему текст или пиктограмму. Следующий по простоте способ заключается в том, чтобы применить стиль из библиотеки, внося в него некоторые изменения. Например, одна из библиотек стилей ImageReady представляет собой набор очень элегантных стилей “стеклянных кнопок”. Если применить один из них к какому-либо тексту, надпись становится полупрозрачной и приобретает один из нескольких цветов; при нажатии кнопки мыши изменяется “освещение” стекла. Если вам нравится такой эффект, но вы хотели бы, чтобы поведение кнопки изменялось при наведении на нее курсора, а не при нажатии кнопки мыши, нужно только заменить состояние **Down**, которое было создано при применении стиля, на **Over**. Если вы внимательно рассмотрели рис. 4.12 или саму палитру **Rollover**, то заметили, что, за исключением нормального состояния, рядом с надписями над каждым мини-слайдом есть маленький треугольник. Как и следовало ожидать, на самом деле это раскрывающийся список, с помощью которого можно изменить состояние данного мини-слайда, — другими словами, изменить событие, по которому роллер будет переходить в это состояние. В данном случае достаточно заменить состояние **Down** на **Over**. На базе

полученной кнопки можно создать новый стиль ролловера и использовать его для создания других подобных кнопок.

☞ С помощью библиотечных стилей ролловеров создайте навигационные панели. Внесите в библиотечный стиль мелкие изменения, подобные описанному выше, сохраните измененный стиль и примените его к нескольким ролловерам.

Более сложные ролловеры

С помощью описанного метода можно создавать ролловеры, чтобы при перемещении курсора изменялись только некоторые свойства изображения. Если же требуется создать ролловер, где изменяется само изображение, необходимо задействовать дополнительные слои. Следует помнить, что состояние ролловера отражает состояние всех слоев изображения, т.е. содержимого палитры **Layers** (Слои). Таким образом, видимость, координаты и эффекты каждого состояния ролловера соответствуют всем слоям, а не отдельному слою. Какое бы состояние вы ни выбрали, если нарисовать что-либо на слое, изменения отразятся на всех состояниях ролловера. Соответственно, для того, чтобы некоторые части изображения были видны, например, в состоянии **Over** и невидимы в нормальном состоянии, необходимо поместить их на отдельный слой и воспользоваться элементами управления видимостью в палитре **Layers** (Слои), чтобы данный слой появлялся и исчезал при изменении состояния ролловера.

Для демонстрации этого приема приведем простой пример. Предположим, что мы хотим создать собственные экранные подсказки для навигационных кнопок Web-страницы. Другими словами, чтобы при наведении курсора на кнопку появлялся дополнительный текст, сообщающий пользователю о назначении этой кнопки. Пример такой кнопки показан на рис. 4.14. Эта кнопка была создана следующим образом. Вначале был создан текстовый слой, содержащий только слово **Downloads**. Затем был создан второй слой, с экранной подсказкой, заключенной в характерное для комиксов облачко. На основе обоих слоев были созданы фрагменты.



Рис. 4.14. Экранные подсказки

Потом был выделен слой с надписью **Downloads**; в палитре **Rollover** было выделено его единственное (нормальное) состояние. В палитре **Layers** (Слои) слой с экранной подсказкой был сделан невидимым. После этого в палитре **Rollover** было создано второе состояние — **Over**. Оно было выделено, а слой с экранной подсказкой в палитре **Layers** (Слои) — снова активизирован. В результате при просмотре документа в Web-браузере экранная подсказка появляется только тогда, когда курсор наводит на надпись **Download**, что и требовалось.

☞ Попробуйте повторить ролlover, показанный на рис. 4.14, выполнив вышеописанные действия. Создайте несколько подобных ролloverов с раскрывающимися списками собственного дизайна.

Возможно, все это не совсем интуитивно понятно, однако такой метод работы с ролloverами и слоями позволяет довольно легко создавать ролloverы со сложными эффектами, с превращением многослойных изображений, знакомых большинству дизайнеров, работающих в Photoshop и других приложениях, в структуру кадров. Таким методом можно также создавать весьма эффективные ролloverы, обычно без дублирования изображения в нескольких файлах. В более сложных случаях возможны некоторые затруднения, так что имеет смысл заранее составить план действий. Следует определить, сколько слоев вам нужно, что будет находиться на каждом из них, при каких обстоятельствах каждый из элементов должен быть видимым, невидимым или отображаться с неким эффектом слоя. Для того чтобы у вас сложилось некоторое представление о том, что мы имеем в виду, приведем более сложный пример.

Если Web-сайт достаточно большой, то будет удобнее для пользователей, если мы снабдим некоторые или все навигационные кнопки всплывающими меню. Назначение этих меню заключается в следующем: каждая кнопка, постоянно отображаемая на странице, соответствует одному из основных разделов Web-сайта; когда на нее наводят курсор, открывается меню со списком подразделов. Пример такой системы навигации показан на рис. 4.15. Для того чтобы показать, каким образом был достигнут такой эффект, мы опишем реализацию всего одной кнопки, показанной на рисунке; остальная часть навигационной панели реализуется аналогично. Следует отметить, что это, конечно, только один из возможных способов решения данной задачи.

Наш первый шаг — решить, как именно будет распределяться изображение по слоям. Очевидно, что каждая надпись — **Downloads**, **Demos** и **Updates** — должна располагаться на отдельном слое, поскольку каждая из них появляется в ответ на определенное событие и, следовательно, является ролloverом. Менее очевидно то, что область вокруг линий, соединяющих подзаголовки с основной кнопкой, также должна реагировать на определенные события, и, как мы увидим впоследствии, ее тоже следует разместить на отдельном слое.



Рис. 4.15. Ролlover со всплывающим меню

Поэтому в первую очередь мы создадим четыре слоя и разместим на них текст и графику. На этом этапе мы не будем применять к слоям эффекты. Мы только создадим фрагменты на базе каждого слоя. Обычно для создания ролloverов видимость и эффекты каждого слоя определяются по очереди, в соответствии с назначенным ему состоянием (табл. 4.1). Пока что мы ограничились ролloverами с двумя состояниями. Слой **Lines** реагирует на перемещение по нему курсора, т.е. пока пользователь перемещает мышью от надписи **Downloads** к одному из подзаголовков и курсор находится между двумя словами, меню исчезнет. Как и в предыдущих примерах, мы решили, что при наведении курсора на кнопку она будет отбрасывать тень. (Мы использовали эффект тени, так как она хорошо видна на печатных страницах. Однако мы не настаиваем на ней, как на обязательном элементе хорошего ролloverа.)

Таблица 4.1. Изменение состояний ролловера для всплывающего меню

Фрагмент	Состояние ролловера	Слой			
		Downloads	Lines	Demos	Updates
Downloads	Normal	Видимый	Невидимый	Невидимый	Невидимый
	Over	Видимый + тень	Видимый	Видимый	Видимый
Lines	Normal	Видимый	Невидимый	Невидимый	Невидимый
	Over	Видимый + тень	Видимый	Видимый	Видимый
Demos	Normal	Видимый	Невидимый	Невидимый	Невидимый
	Over	Видимый + тень	Видимый	Видимый + тень	Видимый
Updates	Normal	Видимый	Невидимый	Невидимый	Невидимый
	Over	Видимый + тень	Видимый	Видимый	Видимый + тень

Вооружившись этой таблицей, легко создать необходимые нам ролловеры. Вначале выделим фрагмент **Downloads** (его нормальное состояние видно в палитре **Rollover**). Затем воспользуемся палитрой **Layers** (Слои) и убедимся, что слой **Downloads** является видимым, а остальные слои невидимы. После этого добавим состояние **Over** и, снова, воспользовавшись палитрой **Layers** (Слои), добавим к слою **Downloads** эффект тени и сделаем видимыми остальные три слоя. Таким образом, при отображении ролловера в Web-браузере меню будет появляться на экране при наведении курсора на слово **Downloads**, как нам и хотелось.

Аналогично создадим ролловеры из оставшихся трех фрагментов, настраивая в палитре **Layers** (Слои) параметры каждого состояния каждого ролловера в соответствии с таблицей.

При просмотре результата этой операции становятся очевидными несколько недостатков. Прежде всего, фрагмент, созданный на базе слоя **Lines**, слишком мал: в него вошли только линии. Поэтому при перемещении курсора по горизонтали от слова **Demos** к слову **Updates** меню исчезает, сбивая пользователя с толку. Для того чтобы устранить эту проблему, мы преобразовали фрагмент **Lines** в пользовательский и растянули его с помощью инструмента выделения фрагмента так, что он заполнил всю область, по которой, скорее всего, будет перемещаться курсор при обращении пользователя к всплывающему меню.

Вторая проблема заключается в том, что, согласно нашему первоначальному плану, если пользователь щелкнет на слове **Downloads**, ничего не произойдет. Это логично, но не всегда оправдывает ожидания пользователей. Мы привыкли, щелкнув на кнопке, ожидать какого-то действия. В данном случае, после некоторых экспериментов, мы решили, что адекватной реакцией на щелчок на кнопке **Downloads** будет исчезновение всплывающего меню. Для реализации этой функции мы воспользовались некоторыми свойствами палитры ролловера, еще не описанными в этой книге.

Прежде всего мы добавили еще одно состояние и назначили ему с помощью раскрывающегося списка событие **Click**, так как хотели, чтобы ролловер переходил в это состояние именно в ответ на щелчок, а не на простое нажатие или отпускание кнопки мыши. Зная, что после щелчка ролловер должен возвращаться в нормальное состояние, мы не стали восстанавливать это состояние явно — в ImageReady его можно скопировать. Для этого мы просто выделили нормальное состояние ролловера и выбрали из меню палитры команду **Copy State** (Копировать состояние). Для того чтобы превратить результат выполнения этого упражнения в настоящую Web-страницу, остается только выделить по

очереди фрагменты **Demos** и **Updates** и с помощью палитры **Slice** (Фрагмент) назначить каждому из них URL.

☞ Попробуйте создать собственную версию описанного здесь примера. Основываясь на той же методике, придумайте собственный вариант меню — например, единого меню для управления подменю, размещенными по всей странице.

Анимация

Формат GIF допускает анимацию. В одном GIF-файле может храниться несколько изображений. Большинство Web-браузеров, обнаружив в GIF-файле больше одного изображения, показывают их по очереди. Когда изображения сменяют друг друга достаточно быстро, благодаря инерции нашего зрения создается эффект движения, или анимация. Формат GIF является простым способом снабжать Web-страницы анимационными элементами, не прибегая к помощи внешних модулей браузера. Программы для работы с Web-графикой обычно имеют те или иные средства создания GIF-анимации, т.е. GIF-файлов с несколькими изображениями.

В GIF-анимации может быть указано, сколько раз должен воспроизводиться анимационный ролик (вплоть до бесконечности) и какую задержку браузер должен выдерживать между кадрами. Задержка определяет частоту смены кадров анимации, однако нет никакой гарантии, что любой браузер будет постоянно поддерживать эту скорость. Нулевая задержка означает, что кадры должны сменять друг друга настолько быстро, насколько это возможно.

В случае изображений с прозрачными областями возникает дополнительная сложность. Обычно при отображении кадра желательно, чтобы сквозь прозрачные области был виден фон страницы. Для этого перед отображением текущего кадра необходимо убрать предыдущий. Однако иногда предпочтительнее, чтобы предыдущий кадр был виден сквозь следующий. Так сознательно поступают при создании анимации, где изображение формируется путем последовательного наложения нескольких кадров. Если же не углубляться в специфику анимации, такое частичное перекрытие кадров позволяет уменьшить размер GIF-файлов. Если вместо полного кадра каждое из хранящихся в файле изображений содержит только различия между предыдущим и текущим кадрами, то при последовательном наложении таких изображений (размер которых обычно меньше — и, вероятно, намного, — чем размер полного кадра) получится такой же анимационный ролик, как и при последовательном показе полных кадров, когда перед отображением следующего удаляется предыдущий. Такая оптимизация GIF-файлов является обычной функцией приложений для работы с Web-графикой.

ImageReady можно использовать для создания анимационных роликов, сохраняемых в формате GIF или в виде фильмов QuickTime. Анимационный ролик в ImageReady представляет собой последовательность кадров, управляемых с помощью палитры **Animation** (Анимация). К анимации применяется примерно тот же подход, что и к ролloverам. Подобно состоянию ролloverа, каждому кадру анимационного ролика соответствует определенная конфигурация палитры **Layers** (Слой). При помощи такого подхода удастся поддерживать множество анимационных стилей и технологий, однако лучше всего он совместим с технологией, основанной на слоях, эффектах и композиции — базисных концепциях обработки изображений в Photoshop (и, следовательно, в ImageReady).

Покадровая анимация

Одна из традиционных анимационных технологий — вероятно, самая простая для понимания — заключается в рисовании каждого кадра на отдельном листе бумаги, фотографировании таких листов кинокамерой в нужной последовательности и воспроизведении полученной последовательности кадров. В ImageReady эта технология имитируется путем размещения каждого кадра на отдельном слое, так что только этот слой виден в кадре. Кадр создается так, как будто он нанесен на отдельный лист бумаги при помощи рисования, раскрашивания, импортирования изображений со сканера и других способов создания изображения на слое. Механизм создания такой анимации в ImageReady довольно прост, однако следует признать, что сам процесс весьма трудоемок. Позже мы опишем технологии анимации, при которых программа выполняет за пользователя значительную часть работы.


 Кадры создаются с помощью палитры **Animation** (Анимация), показанной на рис. 4.16. В нижней части этой палитры находится стандартная кнопка **New** (Создать) — именно с ее помощью создаются новые кадры. Каждый новый кадр, созданный таким образом, первоначально является копией предыдущего. Таким образом, для создания анимационного ролика способом, описанным в предыдущем абзаце, нужно сначала создать новый слой в первом кадре и на нем — изображение первого кадра, а затем — новый кадр. Сразу после создания этот кадр будет выделен. В палитре **Layers** (Слои) следует добавить к нему новый слой и сделать невидимым слой с изображением предыдущего кадра. После этого остается только создать на новом слое изображение второго кадра. Дальше все продолжается аналогично: новые кадры создаются на новых слоях, при этом все остальные слои, кроме того, который содержит изображение рабочего кадра, делаются невидимыми. Как видно на рис. 4.16, в палитре **Layers** (Слой) в каждом кадре виден только один слой, а остальные скрыты.



Рис. 4.16. Палитры анимации и соответствующие слои

При создании таким методом анимационных роликов в ImageReady можно автоматизировать некоторые операции. Если выбрать из меню палитры **Animation** (Анимация) команду **Add A Layer To New Frames** (Добавлять слой к новым кадрам), то при создании кадра к изображению каждый раз будет добавляться новый слой, который по умолчанию

будет невидимым во всех предыдущих кадрах. Тогда при создании нового кадра будет достаточно сделать слой предыдущего кадра невидимым и создать изображение нового кадра на новом слое. На практике эти операции удобнее выполнять в обратной последовательности — другими словами, сначала нарисовать новый кадр поверх старого и только потом, когда все будет готово, сделать слой предыдущего кадра невидимым. При этом лучше видно перемещение элементов анимации в соседних кадрах и проще создать убедительную иллюзию движения. (Обратите внимание: этот прием еще полезнее, если перед тем, как рисовать следующий слой, сделать предыдущий слой полупрозрачным. Конечно, на вид предыдущего кадра это не повлияет.)

С помощью палитры **Animation** (Анимация) настраиваются различные параметры анимационного ролика. Например, можно задать длительность задержки между сменой кадров. Для этого нужно выбрать значение из раскрывающегося списка, расположенного в палитре под соответствующим мини-слайдом: у каждого кадра может быть своя задержка, хотя обычно предпочитают воспроизводить ролик с постоянной скоростью. Самый простой способ сделать это — воспользоваться командой **Select All Frames** (Выбрать все кадры) из меню палитры; если затем задать задержку для любого кадра, то она будет распространяться на весь ролик, включая и те кадры, которые будут добавлены впоследствии.

Количество воспроизведений ролика в браузере (циклов) задается в раскрывающемся списке, расположенном в нижнем левом углу палитры **Animation** (Анимация). Здесь есть три варианта: **Forever** (Бесконечно), **Once** (Один раз) и **Other...** (Другое...). Если выбрать последний вариант, откроется диалоговое окно для задания количества воспроизведений.

Внизу палитры **Animation** (Анимация) расположен ряд удобных элементов управления воспроизведением, похожих на кнопки видеомэгафона и предназначенных для предварительного просмотра ролика в ImageReady. Однако обычно при таком воспроизведении не соблюдается частота смены кадров, так что лучше просматривать ролики в браузере. Обратите внимание, что, поскольку GIF является единственным Web-форматом, поддерживающим анимацию, именно этот формат приходится выбирать в палитре **Optimize** (Оптимизация) при создании анимационных роликов (несмотря на то, что их можно сохранять и в виде фильмов QuickTime с помощью команды **File⇒Export Original...** (Файл⇒Экспортировать оригинал...)). После этого для создания GIF-анимации достаточно выбрать из меню **File** (Файл) команду **Save Optimized** (Сохранить оптимизированное изображение).

☞ Потренируйтесь создавать простые ролики с покадровой анимацией. Начните с нескольких кадров и простого рисунка, чтобы освоить методику. Затем, если вам понравится, можете перейти к чему-нибудь более сложному, только следите, чтобы размеры готового файла держались в разумных пределах.

Другой способ создания кадров анимации заключается в том, чтобы вначале поместить каждое изображение на отдельном слое в ImageReady или Photoshop, как на обычном статическом многослойном изображении. Затем, если понадобится, файл открывают в ImageReady и выбирают из меню палитры **Animation** (Анимация) команду **Make Frames From Layers** (Создать кадры из слоев). Каждый слой помещается в отдельный кадр. После этого анимационный ролик и его свойства редактируют так же, как если бы кадры создавались по одному с помощью палитры **Animation** (Анимация), как описывалось выше.

Как уже отмечалось в этой главе, анимационные GIF-ролики можно оптимизировать так, чтобы в файле сохранялись только различия между соседними кадрами. Для этого используется команда **Optimize Animation...** (Оптимизировать анимацию...) из меню палитры. (В открывшемся диалоговом окне следует оставить включенными оба предлагаемых режима.) Как уже объяснялось выше, эффективность оптимизации зависит от того, насколько соседние кадры отличаются друг от друга. Обычно единственным средством задать метод разделения кадров является контекстное меню; оно открывается, если щелкнуть на кадре правой кнопкой мыши (или удерживать нажатой клавишу <Ctrl>), и содержит команды **Restore To Background** (Восстановить до фона), **Do Not Dispose** (Не разделять) и **Automatic** (Автоматически).

Свойства слоев анимации

Больше всего возможностей для художника-мультипликатора предоставляет покадровая анимация. Однако это тяжелая работа, и если вы предпочитаете такую технику, то вам, возможно, лучше воспользоваться более мощным инструментом. В ImageReady проще создавать анимационные ролики, постепенно меняя атрибуты слоев и не пытаясь изменить само изображение. В отличие от покадровой анимации такие ролики создаются гораздо легче.

Самые простые анимационные ролики, создаваемые в ImageReady, получаются путем переноса содержимого слоя на следующий кадр. Для этого создавать новый слой для каждого кадра не обязательно, поэтому режим **Add A Layer To New Frames** (Добавлять слой к новым кадрам) в меню палитры **Animation** (Анимация) можно сразу отключить. Затем создают один слой и помещают на него какое-либо изображение. В случае необходимости можно воспользоваться инструментом перемещения и сдвинуть слой, чтобы позиция изображения соответствовала первому кадру ролика. После этого можно добавлять в ролик новые кадры, каждый раз изменяя положение слоя с помощью инструмента перемещения. Рассмотрим простой пример: если разместить на слое только один объект — скажем, чашку, — так, чтобы она находилась на левом крае первого кадра, и каждый раз при создании нового кадра немного сдвигать ее вправо, то при просмотре ролика будет казаться, что чашка перемещается по экрану слева направо. Учтите, что в ImageReady можно двигать только слой, и если вы попытаетесь сдвинуть само изображение чашки, перерисовав ее, то ее положение изменится во всех кадрах ролика.

Разумеется, при создании анимации таким способом вы не ограничены одним слоем. Можно разместить несколько изображений на разных слоях и двигать слои независимо друг от друга, добиваясь таким образом сложных движений и взаимодействий.

☞ Попробуйте создать анимационный ролик, последовательно сдвигая содержимое одного слоя при добавлении кадров. Для начала создайте на слое простую абстрактную фигуру и поэкспериментируйте, передвигая ее в разных направлениях. Попробуйте использовать более характерные элементы и добиться большей естественности в их перемещении.

Кроме положения, можно менять и другие атрибуты слоев. Один из самых распространенных эффектов, достигаемых таким образом, — постепенное растворение и концентрация. Путем постепенного увеличения прозрачности слоя на протяжении нескольких кадров изображение становится все более тусклым, пока наконец не исчезает совсем. И наоборот, путем постепенного уменьшения прозрачности сначала появляется тусклое изображение, которое затем становится все ярче. Для создания простейшего эффекта растворения достаточно создать новый слой (на фоновом слое эффект не сработает) и поместить на него изображение; затем с помощью палитры **Animation** (Анимация)

вставить второй кадр и уменьшить его плотность до 90%; добавить третий кадр и уменьшить его плотность до 80%, и т.д. Для такой анимации достаточно одного слоя, который по-разному выглядит в разных кадрах вследствие изменения его свойств. Подобный метод применяется и для анимации эффектов слоя путем изменения их параметров для каждого кадра.

☞ Создайте анимационный ролик, состоящий из геометрической фигуры или какого-нибудь теста, отбрасывающих тень, направление и интенсивность которой изменяются, как будто изображение освещено солнцем, движущимся по небу от восхода до заката.

Анимация атрибутов слоя опирается на тот факт, что все изменения, вносимые в палитре **Layers** (Слои), применяются только к тому кадру, который в данный момент выделен в палитре **Animation** (Анимация). Если изменения должны затронуть все кадры или если вы хотите распространить некоторые свойства одного кадра на весь ролик, необходимо воспользоваться командой **Match Layer Across Frames** (Распространить слой на все кадры) из меню палитры **Animation** (Анимация).

Выполнять вручную простое линейное изменение параметров, подобное описанному выше растворению, довольно утомительно. Кроме того, подобные операции хорошо автоматизируются. В Web-анимации автоматическая интерполяция кадров называется *раскадровкой* (tweening). Возможности ImageReady для раскадровки слоев весьма примитивны: по начальному и конечному значениям автоматически вычисляются промежуточные положения слоя, его прозрачность и свойства примененных к слою эффектов.

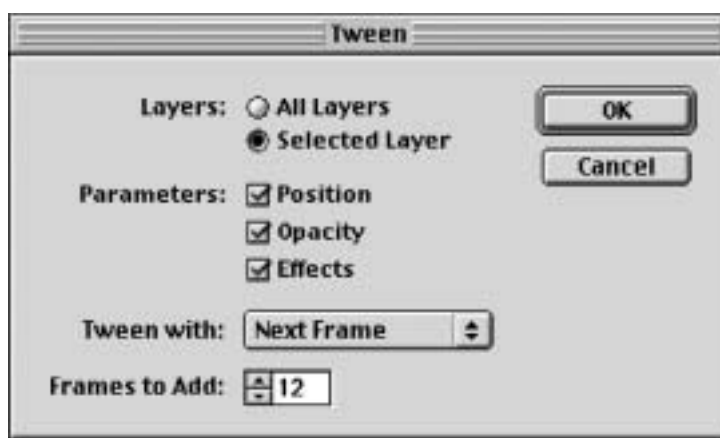


Рис. 4.17. Диалоговое окно для раскадровки



Вернемся к нашему примеру растворяющегося изображения. Проще всего создать такую анимацию с помощью раскадровки. Мы начнем, как и раньше, с создания слоя и размещения на нем изображения, которое впоследствии должно “раствориться”. Однако затем, перейдя к палитре **Animation** (Анимация), мы сделаем только одну копию этого исходного кадра. Затем выделим в палитре **Animation** (Анимация) скопированный кадр — другими словами, второй кадр ролика — и в палитре **Layers** (Слои) уменьшим его плотность до нуля. Затем выделим первый кадр и щелкнем на кнопке раскадровки, расположенной внизу палитры. Откроется диалоговое окно, показанное на рис. 4.17. Два самых важных его параметра находятся внизу. В раскрывающемся списке **Tween with**

(Раскадровка с) выбирается конечный кадр создаваемой последовательности. В нашем случае это следующий (второй) кадр ролика. (Если бы мы сочли это более удобным, мы могли бы с самого начала выделить второй кадр и выбрать из этого списка режим **Previous Frame** (Предыдущий кадр).) Внизу диалогового окна находится поле, в котором указывается, сколько кадров нужно вставить между начальным и конечным кадрами последовательности. В сущности, именно это количество кадров в сочетании с межкадровой задержкой, заданной для всего ролика, определяет скорость растворения.

Два параметра, представленные вверху диалогового окна, позволяют выбрать для интерполяции только выделенный слой (при этом в процессе раскадровки остальные слои, за исключением статичного фонового, становятся невидимыми) или все слои. Группа режимов в центре диалогового окна позволяет выбрать, какие атрибуты слоя подлежат раскадровке — это те же атрибуты, которые можно изменять и при создании анимации вручную.

☞ Выполните предыдущее упражнение, используя раскадровку вместо того, чтобы создавать каждый кадр вручную. Потренируйтесь создавать анимационные ролики с помощью раскадровки.

Раскадровка в ImageReady сводится к вставке в анимационный ролик новых кадров, на которых заданные пользователем атрибуты изменяются от значения начального кадра до значения конечного. (Это вставка именно новых промежуточных кадров, а не инструкций об их вычислении, как во Flash.) Интервалы между промежуточными значениями одинаковы; для более сложного изменения параметров необходимо вставить дополнительные кадры вручную, разделив всю последовательность на ряд участков с линейно изменяющимися параметрами.

☞ Используя раскадровку, создайте анимационный ролик, в котором изображение быстро концентрируется и плавно растворяется.

Результатом кадрирования в ImageReady является такая же последовательность кадров, которую бы вы создали, если бы делали то же самое вручную. Эти кадры редактируются точно так же, как и обычные. Но если изменить значение уже раскадрованного параметра в начальном или конечном кадре последовательности, промежуточные значения не будут пересчитаны. Предположим, что, создав эффект растворения путем раскадровки, как описано выше, вы решите вместо полного растворения объекта уменьшить его плотность только наполовину. Если затем вы выделите последний кадр и присвоите ему значение плотности, равное 50%, то вместо неполного растворения получите анимационный ролик, в котором изображение растворяется почти полностью, а затем снова резко концентрируется. Для того чтобы действительно изменить начальные и конечные значения последовательности, необходимо выполнить всю раскадровку заново. Однако удалять кадры старой раскадровки при этом не обязательно: достаточно выделить их, прежде чем щелкнуть на кнопке раскадровки. Тогда в раскрывающемся списке **Tween with** (Раскадровка с) будет предложен вариант **Selection** (Выделение), а остальные варианты будут недоступны, как и поле для ввода количества кадров. Вместо выделенных кадров в ролик будут вставлены кадры, полученные в результате интерполяции.

☞ Поэкспериментируйте, изменяя значения параметров раскадровки в предыдущем упражнении, в том числе и меняя уже готовую раскадровку.

В процессе раскадровки появилось одно серьезное улучшение. Если выделить последний кадр анимационного ролика, то в открывшемся диалоговом окне появится новый режим раскадровки — до первого кадра. В результате за последним кадром вставляются новые, с интерполяцией выбранного параметра до значения, назначенного ему в первом кадре. Таким образом, если ролик воспроизводится в непрерывном цикле, то различия между первым и последним кадрами сглаживаются, и цикл замыкается сам на себя.

✎ Создайте циклический анимационный ролик, в котором изменялись бы один или несколько параметров эффекта слоя, примененного к тексту, от исходного значения до некоторого максимума, а затем опять до исходного значения. Сделайте анимационный цикл бесконечным и просмотрите ролик в Web-браузере. Используя раскадровку, сделайте так, чтобы цикл завершился плавно.

Анимированные ролловеры

Анимированные ролловеры представляют собой сочетание анимации и ролловеров, однако при их создании необходимо придерживаться здравого смысла. Основная идея заключается в том, что если выбрать состояние ролловера и создать анимационный ролик с помощью палитры **Animation** (Анимация), то этот ролик будет воспроизводиться при наступлении выбранного состояния — другими словами, после того, как произойдет событие, на которое настроен ролловер. Для того чтобы продемонстрировать, как это работает, и объединить предыдущие описания анимации и ролловеров, мы рассмотрим пример простого анимированного ролловера.

Если вас привлекает интерфейс Mac OS X, то вам должны понравиться пульсирующие полупрозрачные кнопки на Web-страницах. Одним из способов сделать это является ролловер с анимацией состояния **Over**, имитирующей пульсацию. Это можно сделать следующим образом.

Прежде всего воспользуемся инструментом создания прямоугольников со скругленными углами и создадим на отдельном слое кнопку, а из этого слоя — фрагмент. Выделим этот фрагмент и применим к нему один из стилей “стеклянных кнопок” — ее цвет выберите по своему усмотрению. (Если в палитре **Styles** (Стили) нет ролловеров “стеклянных кнопок”, загрузите их с помощью команды **Glass Button Rollovers** (Ролловеры “стеклянных кнопок”) из меню палитры.) После того как к фигуре, изображающей кнопку, будут применены эффекты слоя, образующие данный стиль, она будет казаться блестящей и полупрозрачной, а к ролловеру добавится состояние **Down** с несколько иной подсветкой кнопки. Именно это состояние мы хотим заменить на пульсирующую анимацию. Однако прежде мы изменим событие, вызывающее это состояние, выбрав из раскрывающегося списка, расположенного под мини-слайдом в палитре **Rollover**, вариант **Over**. После этого выделим состояние **Over** и перейдем к палитре **Animation** (Анимация).

Есть несколько способов имитировать пульсацию кнопки. Нам кажется, что самый убедительный из них — уменьшить плотность эффекта перекрывающегося градиента слоя, который является частью стиля “стеклянной кнопки”. Разумеется, мы облегчим себе задачу, используя раскадровку.

Вначале в палитре **Animation** (Анимация) есть только один кадр, повторяющий статичное состояние **Over** для ролловера кнопки. Мы добавим еще один кадр, выделим его и эффект перекрывающегося градиента в палитре **Layers** (Слой) — он перечислен там среди прочих эффектов, примененных к слою кнопки, с которого мы начинали. Когда

эффект будет выделен, как обычно, откроется палитра с его параметрами (см. главу 3). В этой палитре мы изменим только плотность — до 50% (рис. 4.18).

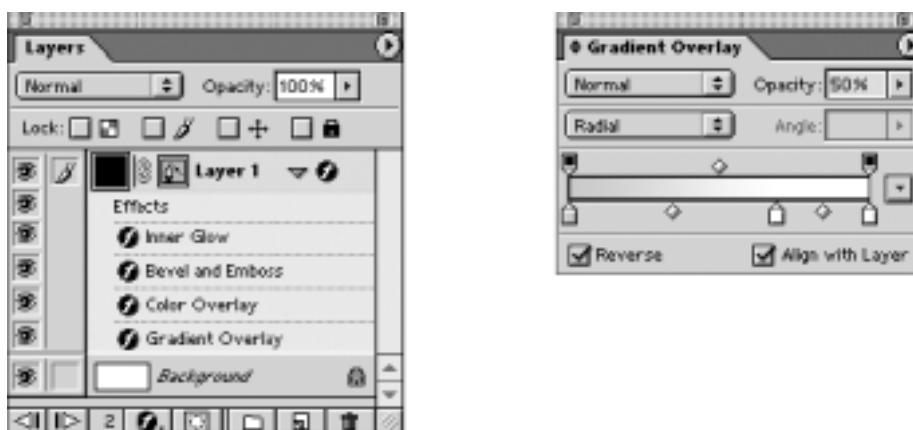


Рис. 4.18. Эффекты слоя для пульсирующей кнопки

Сохраняя выделение нового кадра, откроем диалоговое окно раскадровки, выберем режим раскадровки до предыдущего кадра и введем подходящее количество промежуточных кадров — хотя бы с полдюжины. Закрыв диалоговое окно и получив промежуточные кадры, снова выделим конечный кадр последовательности и откроем диалоговое окно раскадровки. На этот раз мы выберем режим раскадровки до первого кадра, плавно завершая цикл пульсации. Количество промежуточных кадров мы зададим такое же, как и в прошлый раз. Таким образом, получим анимационный ролик, в котором эффект перекрывающегося градиента вначале затухает, а затем снова усиливается.

В заключение работы над анимационным роликом нам остается задать подходящую межкадровую задержку — не больше 1/5 секунды, если вы хотите, чтобы пульсация была плавной, — и включить режим бесконечного цикла. Затем, убедившись, что в палитре **Optimize** (Оптимизация) выбран формат GIF, можно воспользоваться командой **File** ⇒ **Save Optimized As...** (Файл ⇒ Сохранить оптимизированный файл как...) из меню **File** (Файл) и сохранить все созданные изображения, а также объединяющий их HTML-файл. При открытии его в Web-браузере, когда пользователь наведет курсор на кнопку, она начнет мягко переливаться.

☞ Выполните описанный выше пример, создав пульсирующую кнопку. Потренируйтесь создавать подобные анимированные ролики, применяя к слоям с кнопками и текстом различные эффекты слоя, изменяющиеся во времени. Проследите, чтобы переходы цикла были плавными.

Для того чтобы кнопка была полезной, необходимо назначить фрагменту, на основании которого был создан ролик, URL-адрес. Кроме того, обычно на кнопку накладывается текст, описывающий ее назначение. К сожалению, сохранить стиль анимированного ролика нельзя. Если вы попытаетесь это сделать, анимация будет утрачена и останется обычный стиль ролика. Для того чтобы создать пульсирующие кнопки разной формы, придется повторить весь процесс сначала. (Однако его можно автоматизировать, записав в виде макроса, как описано в главе 9.)

Новое в ImageReady CS

В ImageReady CS мало что изменилось по сравнению с предыдущими версиями, но благодаря новым элементам интерфейса программы некоторые операции стали выполняться по-другому. Самой важной из новых функций ImageReady CS стало использование переменных и наборов данных для создания управляемой графики — так же, как в Illustrator (подробнее об этом читайте в главе 6). Из остального единственная новая возможность — экспортирование анимации в формат SWF. (Для этого нужно выбрать команду **File**⇒**Export**⇒**Macromedia® Flash™ SWF...** (Файл⇒Экспорт⇒Macromedia® Flash™ SWF...)) и настроить несколько параметров в открывшемся диалоговом окне (рис. 4.19). Обратите внимание: при экспортировании в формат SWF кадры анимации остаются растровыми, а не превращаются в векторную анимацию наподобие Flash. Но при экспортировании в формат SWF раскадрованной анимации растровые изображения преобразуются в символы, к которым в SWF-фильме применяется раскадровка движения.

Палитры

Палитры **Slice** (Фрагмент) и **Image Map** (Карта изображения) теперь выглядят по-другому. Новая палитра **Slice** (Фрагмент) показана на рис. 4.20: как мы видим, она разбита на разделы, благодаря чему занимает на экране меньше места. Для того чтобы раскрыть раздел, нужно щелкнуть на соответствующий пиктограмме в виде треугольной стрелки. Но содержание палитры и способ ее использования остались прежними. Аналогичные преобразования коснулись и палитры **Image Map** (Карта изображения).



Рис. 4.19. Параметры экспортирования в формат SWF

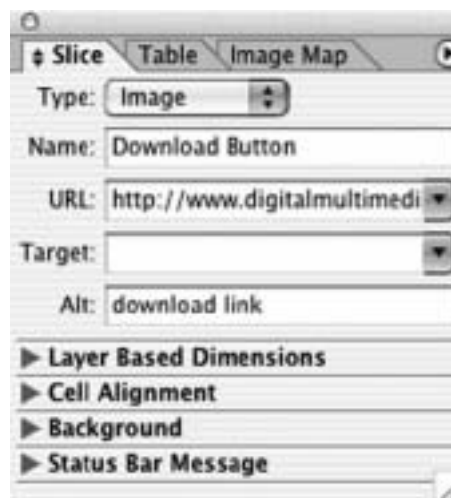


Рис. 4.20. Измененная палитра Slice (Фрагмент)

Палитра Web Content (Web-содержимое)

Палитра **Rollover** из предыдущих версий ImageReady теперь включена в состав новой палитры **Web Content** (Web-содержимое), позволяющей не только создавать ролловеры, но также манипулировать фрагментами, активными областями карт изображений и кадрами анимации. Как видно на рис. 4.21, эта палитра делится на три области: верхняя

предназначена для управления ролловерами и анимацией всего изображения; средняя, открываемая при помощи пиктограммы в виде треугольной стрелки, — для отображения всех областей карт данного изображения; нижняя — для отображения всех фрагментов в изображении, аналогично предыдущей. Как видно на рисунке, при создании ролловеров для фрагмента состояния ролловера отображаются в палитре **Web Content** (Web-содержимое) как часть этого фрагмента. Для их сокрытия и отображения используются пиктограммы в виде треугольных стрелок в разделе палитры, посвященном фрагментам (обратите внимание, например, на фрагмент **Download Button**).



Рис. 4.21. Палитра **Web Content** (Web-содержимое)

С помощью палитры **Web Content** (Web-содержимое) можно выбирать фрагменты и области карт изображений и, щелкнув на элементе палитры, выделять соответствующие фрагменты и области на самом изображении. Иногда это удобнее, чем применять инструменты выделения непосредственно к изображению. Но чаще всего эта палитра используется для создания ролловеров. Это делается практически так же, как и с помощью старой палитры **Rollover** (см. раздел “Мгновенные ролловеры”, с. 182). Для того чтобы добавить к фрагментам на базе слоя новые состояния ролловера, нужно воспользоваться кнопкой **New** (Создать), расположенной внизу палитры **Web Content** (Web-содержимое). Для создания простых эффектов ролловера часто достаточно стилей слоев. Несколько изменился и способ создания удаленных ролловеров, когда событие мыши, происходящее на одном фрагменте, влияет на один или несколько других фрагментов. Общий метод остался прежним: выделяем состояние ролловера и изменяем видимость и другие атрибуты слоя, чтобы получить желаемый вид для этого состояния. Но теперь требуется также выбрать, на какие фрагменты воздействует ролловер. Для того чтобы восстановить прежнее поведение панели, когда состояние любого фрагмента изменялось в соответствии с изменениями палитры **Layers** (Слои) для данного ролловера, нужно выбрать состояние ролловера и воспользоваться командой **Find Remote Slices for State...** (Найти удаленные фрагменты для состояния...) из меню палитры **Web Content** (Web-содержимое). Для того чтобы выбрать изменяющиеся удаленные фрагменты вручную, нужно щелкнуть напротив имени соответствующего фрагмента во втором столбце палитры **Web Content** (Web-содержимое). На этом месте появится пиктограмма, означающая, что данный фрагмент является удаленным ролловером.

- ☞ Поэкспериментируйте, выделяя различные фрагменты с помощью палитры **Web Content** (Web-содержимое) и настраивая их атрибуты в палитре **Slice** (Фрагмент).
- ☞ Выполните упражнения с ролловерами из разделов “Мгновенные ролловеры” (с. 182) и “Более сложные ролловеры” (с. 185), используя новую палитру **Web Content** (Web-содержимое).

Управляемая графика

В главе 6 описывается реализация управляемой графики в Illustrator 10. Аналогичные возможности есть в ImageReady CS. Здесь мы рассмотрим их отличия от функций Illustrator.

Принципиальное — и неизбежное — отличие реализации управляемой графики в этих двух приложениях заключается в том, что переменные ImageReady связаны со слоями, а не с объектами, как в Illustrator. Но типы переменных в обоих случаях одинаковые: текст, видимость и замещение (пикселей, а не изображений, как в Illustrator). Переменные назначаются слоям в диалоговом окне **Variables** (Переменные), которое открывается после выбора команды **Image**⇒**Variables**⇒**Define...** (Изображение⇒Переменные⇒Определить...).

Проиллюстрируем применение переменных в ImageReady на упрощенном примере, подобном тому, который описывается в разделе “Переменные, наборы данных и шаблоны” (с. 359). Предположим, мы хотим создать несколько пар изображений, каждая из которых состоит из цифры, которая в первом изображении замещается на однородный красный прямоугольник со скругленными углами, а во втором — на такой же скругленный зеленый прямоугольник с красным контуром. Создадим такие пары для разных цифр (рис. 4.22).



Рис. 4.22. Семейство управляемых изображений

Вначале создадим элементы первой пары: красный скругленный прямоугольник на нижнем уровне и поверх него несколько меньший зеленый скругленный прямоугольник, чтобы при наложении одного на другой получалась красная рамка; поверх всего этого создадим текстовый слой, на котором разместим цифру 1 соответствующего размера. Как видим, мы можем создать пару изображений для цифры 1, включая и отключая видимость среднего слоя. Чтобы получить семейство таких пар, достаточно изменить цифру на верхнем, текстовом слое. Используя переменные, мы можем автоматизировать эту процедуру и сохранить источник всего семейства изображений в одном файле.

На рис. 4.23 показано диалоговое окно **Variables** (Переменные), в котором слоям назначаются переменные. Слой выбирается из раскрывающегося списка, расположенного сверху окна. Вид нижней части окна зависит от содержимого слоя: если это растровое изображение, то окно выглядит так, как показано на рис. 4.23, *слева*; если же это текст, то окно выглядит как на рис. 4.23, *справа*. Для того чтобы назначить слою переменную одного из допустимых типов, нужно установить переключатель и ввести имя переменной. У переменных замещения пикселей (которые применяются для замены пикселей слоя

изображением из другого файла) есть несколько дополнительных параметров, определяющих, как замещающее изображение должно располагаться на слое. Для этого используется дополнительное диалоговое окно, которое открывается, если щелкнуть на кнопке **Pixel Replacement Options** (Параметры замещения пикселей). Как пользоваться этим диалоговым окном, понятно благодаря диаграммам, на которых отображается результат каждого изменения.



Рис. 4.23. Определение переменных

После того как слоям будут назначены переменные, можно определить наборы данных, которые будут служить значениями этих переменных. Каждый набор данных описывает изображение, создаваемое, когда переменные принимают эти значения. Для того чтобы определить набор данных, применяется команда **Image⇒Variables⇒Data Sets...** (Изображение⇒Переменные⇒Наборы данных...), с помощью которой открывается диалоговое окно, показанное на рис. 4.24. (В действительности это уже знакомое нам диалоговое окно **Variables** (Переменные)). Для переключения между режимами определения наборов данных и переменных используется раскрывающийся список, расположенный вверху окна.) Для того чтобы создать новый набор данных, нужно щелкнуть на пиктограмме **New** (Создать), расположенной справа от раскрывающегося списка **Data Set** (Набор данных). Для того чтобы присвоить значения переменным этого набора, нужно щелкнуть по очереди на именах переменных, представленных в списке, расположенном в нижней части диалогового окна, и назначить желаемые значения с помощью элементов управления, расположенных в центре окна.

Кнопки со стрелками, расположенные справа от раскрывающегося списка **Data Set** (Набор данных), используются для просмотра последовательности наборов данных, а кнопка **Apply** (Применить) — для применения выбранного набора данных, т.е. для присвоения его значений переменным и соответствующего изменения изображения. Таким образом, применяя наборы данных по очереди, можно получить семейство изображений.

Часто возникает необходимость в создании отдельных изображений на основе каждого набора данных после того, как его структура будет окончательно сформирована. Для этого используется простая операция, выполняемая с помощью команды **File⇒Export⇒Data Sets as Files...** (Файл⇒Экспортировать⇒Наборы данных в файл...). Таким образом, автоматизируется назначение наборов данных и сохранение файлов для каждого из них. (У этой команды есть несколько очевидных параметров, касающихся формата файлов и способов их именования.)

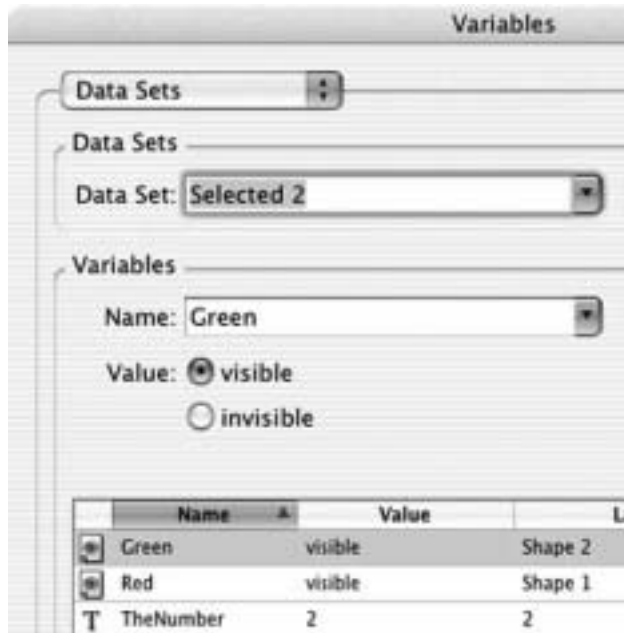


Рис. 4.24. Определение наборов данных

Для больших наборов данных иногда проще ввести все значения в файл или генерировать такой файл с помощью какого-нибудь сценария. В Illustrator такие наборы данных хранятся в формате XML, но в ImageReady для этого используется более простой формат с разделением значений запятыми (*Comma-Separated Values, CSV*). Файл начинается строкой, в которой через запятую перечисляются имена всех переменных. Затем вводится столько строк, сколько наборов данных нужно определить. Каждая строка представляет собой множество значений, разделенных запятыми. В нашем примере такой файл выглядел бы так:

```
Green, Red, TheNumber
false, true, 1
true, true, 1
false, true, 2
true, true, 2
false, true, 3
true, true, 3
false, true, 4
true, true, 4
...
```

Обратите внимание, что переменные видимости принимают значения `true` и `false`, соответствующие состоянию видимости и невидимости. Наборы данных импортируются с помощью кнопки **Import** (Импортировать), расположенной в диалоговом окне **Variables** (Переменные) (рис. 4.24). По умолчанию импортируемые наборы данных заменяют уже определенные, но этот режим можно изменить в диалоговом окне импортирования наборов данных.

☞ Создайте документ, состоящий из цветного фона, мини-слайда фотографии, расположенного в верхнем левом углу, и текстовой подписи этой фотографии, идущей вдоль нижнего края. Назначьте слоям переменные, чтобы можно было создать семейство изображений для разных фотографий, в котором каждой фотографии соответствовал бы набор изображений с подписями и тремя разными фоновыми цветами. Создайте наборы данных для двух фотографий, используя диалоговое окно **Data Sets** (Наборы данных). Создайте текстовый файл с шестью наборами данных и используйте его для создания полного набора графических файлов для разных фотографий.

Дополнительные упражнения для освоения ImageReady

1. Импортируйте фотографию своего знакомого или родственника. Фрагментируйте изображение так, чтобы эффективно оптимизировать его для Web, и просмотрите результат в браузере.
2. Импортируйте изображение из интересующей вас области — музыки, спорта, филателии и т.п., — которое было бы удобно превратить в карту. Создайте в соответствующих частях изображения активные зоны и назначьте им URL-адреса Web-сайтов, посвященных этой теме.
3. Разработайте дизайн и создайте несколько кнопок, которые меняются, когда на них наводят курсор.
4. Создайте ролlover с разными изображениями для каждого из состояний. Протестируйте этот ролlover в браузере так, чтобы проявились все состояния.
5. Создайте анимированный ролlover в виде кнопки, которая, когда на нее наводят курсор, начинает переливаться всеми цветами радуги.
6. Создайте навигационную панель для небольшого Web-сайта, не менее чем с тремя кнопками, имеющими надписи для основных разделов сайта, каждая из которых снабжена всплывающим меню. Например, на типичном сайте небольшой фирмы по разработке программного обеспечения могут быть разделы **Downloads** с подразделами **Demos** и **Updates**; **Contacts** с подразделами **Enquiries**, **Complaints** и **Brochure Requests**; **Information** с подразделами **FAQs**, **Company Profile** и **Products**. Проследите, чтобы каждое всплывающее меню открывалось только тогда, когда курсор попадает на соответствующий заголовок основного меню.
7. Разместите на странице семь невидимых кнопок произвольным образом и добавьте к ним такие ролloverы, при которых, когда курсор попадает на кнопку, в другом месте страницы появляется одно из семи связанных с кнопками изображений.
8. Создайте Web-страницу для учебного или коммерческого использования, с логичным дизайном, удобную в использовании и информативную. Страница должна иметь ненавязчивое фоновое изображение и набор кнопок, вид которых изменяется в режиме ролloverа, так что пользователь получает обратную связь; после щелчка на кнопке должно появляться изображение и краткое описание. Например, у коммерческого сайта могут быть кнопки для каждой группы продукции, после щелчка на которых появляются изображения этой продукции с кратким описанием.

9. Создайте анимационный ролик примерно из двадцати кадров путем последовательной настройки или изменения одного изображения. Сделайте это сначала вручную, а затем с помощью раскадровки.
10. Создайте небольшой баннер с характеристиками коммерческих рекламных объявлений, принятых в Web. Баннер должен иметь наиболее распространенный размер (468×60 пикселей) и передавать короткое емкое сообщение не более чем в девяти кадрах.
11. Вырежьте из иллюстрированного журнала несколько фигур или других объектов, отсканируйте их и импортируйте в ImageReady. Откорректируйте их и вставьте на отдельные слои изображения. Создайте анимационный ролик путем перемещения отдельных слоев, чтобы создать иллюзию движения фигур.
12. Возьмите подходящее изображение, разделите его на фрагменты и создайте анимационный ролик на основе одного из них. Например, фрагментируйте изображение динозавра и сделайте так, чтобы двигалась только голова.
13. Возьмите восемь небольших квадратных изображений одного размера, поместите на отдельных слоях и выровняйте так, чтобы они образовали таблицу 3×3, с пустой ячейкой в центре, имеющей тот же размер, что и изображения.
 - Создайте такие ролловеры, чтобы при наведении курсора на изображение в центре таблицы появлялась его копия.
 - Создайте такие ролловеры, чтобы после щелчка на изображении оно перемещалось в свободную ячейку, оставляя за собой пустую рамку. При следующем щелчке изображение должно перемещаться на освободившееся место и т.д. (Возможно, вначале лучше поэкспериментировать с тремя изображениями, расположенными в ряд, с пробелом между ними.)
14. Импортируйте или создайте изображение, некоторые элементы которого можно анимировать. (Изображение может быть как абстрактным, так и конкретным.) Фрагментируйте изображение должным образом и превратите два или три его фрагмента в анимированные ролловеры.