

**Кетрин Ульрих**

# **Интерактивная Web-анимация во FLASH**

# ADOBE FLASH PROFESSIONAL

**Katherine Ulrich**

# ИНТЕРАКТИВНАЯ WEB-АНИМАЦИЯ ВО FLASH

**Кетрин Ульрих**



Москва, 2010

УДК 004.4'273  
ББК 32.973.26-018.2  
Э45

Ульрих К.

Э45 Интерактивная Web-анимация во Flash / Ульрих Кетрин ; Пер. с англ. Слинкина А. – М. : ДМК Пресс, 2010. – 568 с.: ил.

ISBN 978-5-94074-541-9

Применение технологии Flash позволяет быстро и эффективно создавать Internet-приложения, оформленные анимацией, музыкальными и видеофрагментами и красивыми кнопками, моментально реагирующими на действия пользователя. Также с помощью Flash можно создавать красочные и выразительные презентации.

Данное руководство посвящено работе с новейшей версией Adobe Flash CS3 Professional, в которой добавились новые графические инструменты, специализированные средства импорта графики, созданной в Adobe Photoshop и Adobe Illustrator, инструменты для импорта и воспроизведения видео, новые компоненты для разработки пользовательского интерфейса и привязки к данным, а также язык ActionScript 3.0 для программирования сложных интерактивных сценариев.

Все операции в книге описываются в виде последовательности шагов и иллюстрируются снимками с экрана. Даже читатель, никогда ранее не занимавшийся Flash-роliками, сможет быстро войти в курс дела и приступить к самостоятельной работе.

Данное издание рассчитано на широкую аудиторию читателей, занимающихся разработкой и оформлением сайтов, профессиональных Web-дизайнеров, программистов, бизнесменов, занятых в сфере Internet-рекламы, а также может быть полезно начинающих пользователям.

Authorized translation from the English language edition, entitled ADOBE FLASH PROFESSIONAL FOR WINDOWS AND MACINTOSH: VISUAL QUICKSTART GUIDE, by KATHERINE ULRICH, published by Pearson Education, Inc, publishing as Peachpit Press, Copyright ©.

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

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

Материал, изложенный в данной книге, многократно проверен. Но, поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

ISBN 0-321-50291-4 (англ.)  
ISBN 978-5-94074-541-9 (рус.)

© Peachpit Press  
© Перевод на русский язык, оформление  
ДМК Пресс, 2010



# СОДЕРЖАНИЕ

---

Благодарности .....	10
Введение .....	11
<b>Глава 1. Среда разработки Flash .....</b>	<b>19</b>
Работа с документами Flash .....	20
Работа с шаблонами .....	25
О среде разработки Flash .....	28
О свойствах документа .....	32
Привязка .....	41
Просмотр в разных масштабах .....	45
Панели .....	47
Объединение и стыковка панелей .....	51
Об инспекторе свойств .....	58
<b>Глава 2. Создание простых графических элементов .....</b>	<b>61</b>
Обзор инструментов .....	62
Сплошные цвета и градиенты .....	64
Образцы .....	69
Создание наборов цветов .....	70
Задание атрибутов заливки .....	72
Задание атрибутов обводки .....	74
Создание геометрических форм .....	78
Создание произвольных форм .....	84
Добавление обводок и заливок .....	94
<b>Глава 3. Работа с текстом .....</b>	<b>99</b>
Инструмент «Текст» .....	100
Задание атрибутов текста .....	102
Задание атрибутов абзаца .....	109
<b>Глава 4. Модификация простых графических элементов .....</b>	<b>113</b>
Задание параметров выбора .....	114
Выбор объектов .....	116
Работа с буфером обмена .....	122
Изменение размера графических элементов .....	124
Позиционирование графических элементов .....	128

Отражение, поворот и перекося	130
Искажение графических элементов	134
Модификация заливок и обводок	137
Модификация форм: инструменты естественного рисования	144
Модификация форм: кривые Безье	147
Модификация путей примитивных форм	155
Конвертирование форм из одного вида в другой	160
<b>Глава 5. Однослойная составная графика</b>	<b>163</b>
Работа с группами	166
Редактирование групп	169
Управление порядком в стопке	171
Объединение объектов-рисунков и примитивов	173
<b>Глава 6. Многослойная графика</b>	<b>175</b>
Знакомство с функциями временной шкалы для работы со слоями	176
Создание и удаление слоев и папок	177
Управление слоями и папками	180
Задание свойств слоев на панели временной шкалы	182
Организация слоев с помощью папок	184
Работа с графикой в нескольких слоях	187
Межслойные операции вырезания и вставки	189
Распределение элементов по слоям	192
Работа с направляющими слоями	193
Работа со слоями-масками	195
<b>Глава 7. Работа с символами</b>	<b>199</b>
Знакомство с библиотечной панелью	200
Иерархия библиотек	204
Преобразование графических элементов в символы	206
Создание символов с нуля	210
Использование экземпляров символа	213
Модификация экземпляров символа	214
Замена одного экземпляра символа другим	218
Редактирование главных символов	220
Дублирование главного символа	221
Удаление главных символов	222
Преобразование экземпляров символа в графические элементы	223
<b>Глава 8. Покадровая анимация</b>	<b>225</b>
Использование временной шкалы	226
Создание опорных кадров	230
Создание промежуточных кадров	234
Выбор кадров	236
Манипулирование кадрами в одном слое	239
Удаление кадров	243
Создание простой покадровой анимации	246
Предварительный просмотр действия	248
Сглаживание анимации за счет добавления опорных кадров	250
Использование восковок	252

Редактирование нескольких кадров .....	254
Задание частоты кадров .....	256
Анимация с переменной скоростью .....	257

## Глава 9. Анимация посредством интерполяции движения ..... 259

Создание прыгающего мяча с помощью интерполяции движения .....	260
Добавление опорных кадров в интерполированную последовательность .....	264
Эффекты анимации цвета .....	265
Анимация графики с изменяющимся размером .....	266
Вращение графических элементов .....	268
Перемещение графических элементов по прямой .....	271
Перемещение графических элементов вдоль пути .....	273
Ориентация графических элементов относительно пути движения .....	277
Изменение скорости анимации .....	279
Копирование интерполированного движения .....	285

## Глава 10. Анимация посредством интерполяции форм ..... 291

Создание прыгающего мяча с помощью интерполяции форм .....	292
Трансформация простых линий и заливок .....	295
Интерполяция нескольких форм .....	297
Трансформация простой формы в сложную .....	299
Создание форм, которые движутся при изменении .....	304

## Глава 11. Более сложные задачи анимации ..... 307

Сцены .....	308
Манипулирование кадрами в нескольких слоях .....	311
Анимация с помощью нескольких интерполяций движения .....	314
Анимация с помощью интерполяции нескольких форм .....	319
Изменение порядка кадров .....	321
Комбинирование интерполяции с покадровой анимацией .....	323
Сохранение анимаций в виде графических символов .....	325
Использование анимированных графических символов .....	329
Сохранение анимации в виде символа-клипа .....	332
Использование символов-клипов .....	336
Использование анимированных масок .....	339
Использование фильтров .....	345

## Глава 12. Добавление интерактивности с помощью кнопок ..... 353

Создание простейшего символа-кнопки .....	355
Создание символов-кнопок с изменяющейся формой .....	359
Создание анимированных символов-кнопок .....	362
Использование компонентов-кнопок .....	365
Модификация компонентов-кнопок .....	368
Создание кнопок-клипов .....	374

## Глава 13. Основы интерактивности ..... 379

Знакомство с панелью действий .....	380
Настройка панели действий .....	382
Организация действий в кадре .....	385

Добавление действий в кадр .....	386
Программирование кнопок с помощью сценария кадра .....	390
Тестирование действий .....	401
Усовершенствование сценариев для работы с кнопками .....	404
Выбор событий .....	408
Компоненты-кнопки .....	412
Использование одного обработчика для нескольких событий .....	415
Программирование клипов как кнопок .....	417
Ссылки на другие Web-страницы .....	425
Преобразование анимации на временной шкале в код .....	429
Использование кнопок для управления графическими объектами .....	432

## **Глава 14. Использование графики, созданной в других программах ..... 441**

Импорт растровых изображений .....	442
Импорт файлов, созданных в программе Photoshop .....	444
Импорт файлов, созданных в программе Adobe Illustrator .....	450
Импорт файлов, созданных в программе Fireworks .....	457
Импорт из программы FreeHand .....	459

## **Глава 15. Добавление звукового сопровождения ..... 461**

Импорт звука .....	462
Добавление звука в кадры .....	464
Добавление звука к кнопкам .....	468
Использование звуков, синхронизированных с событиями .....	470
Использование звуков, синхронизированных с началом .....	474
Использование потокового аудио .....	475
Прекращение звучания .....	478
Повтор звучания .....	480
Редактирование звука .....	482

## **Глава 16. Добавление видео ..... 487**

Импорт видео для прогрессивной или потоковой загрузки .....	489
Импорт внедряемого видео .....	494
Задание параметров кодирования .....	498
Работа с внедренным видео .....	506
Работа с компонентом FLVPlayback .....	508

## **Глава 17. Доставка ролика аудитории ..... 515**

Подготовка ролика для оптимального воспроизведения .....	516
Публикация .....	520
Настройки Flash-плеера .....	524
Публикация HTML-документа для просмотра файлов в Flash-плеере .....	534
Использование альтернативных графических форматов .....	544
Определение версии .....	548
Печать из Flash .....	550

## **Предметный указатель ..... 553**

Посвящается Перри Уиттлу, чья поддержка – моральная и техническая – помогает мне не уклоняться от цели. Он всегда преуменьшает мои неудачи и беззастенчиво восхваляет мои достижения.

## Благодарности

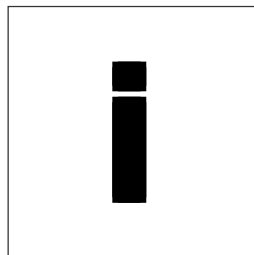
Особая благодарность редактору этой книги Вэнди Шарп (Wendy Sharp) за ценные советы, слова поддержки и настойчивость на протяжении всего времени работы. Огромное спасибо Андреасу Хейму (Andreas Heim), начальнику технического отдела компании Smashing Ideas, за то, что он написал главу 13 и придумал интерактивные задания, которые позволят приступить к работе с языком ActionScript даже тем, кто совсем ничего не знает о технологии Flash. Спасибо Марку Р. Ронкману (Mark R. Ronkman) за то, что он прочел рукопись, обращая внимание на точность формулировок и технические ошибки, а также за те советы и рекомендации, которыми он постоянно делился со мной. Аплодисменты литературному редактору Жаклин Кэн Аарон (Jacqueline Kan Aaron), от нее не укрылись никакие мои орфографические, грамматические, логические или стилистические ошибки. Слава и почет заведующему производством Конни Юнг-Миллс (Connie Jeung-Mills) и наборщику Оуэну Вулфсону (Owen Wollfson) за то, что эти страницы получились ясными и красивыми. Благодарю также Эмили Глоссбрэннер (Emily Glossbrenner) за составление указателя в очень сжатые сроки и Майкла Дж. Ульриха (Michael J. Ulrich) за помощь в тестировании импорта графических изображений из программы Adobe Illustrator. И еще спасибо Бекки Морган (Beckie Morgan), вклад которой был весьма заметен на заключительных этапах работы.

Нельзя обойти вниманием тех, кто помогал в работе над предыдущими изданиями, их труды не пропали даром: Брэд Бехтель (Brad Bechtel), Лайза Бразизель (Lisa Brazieal), Эрика Бэрбек (Erika Burback), Джереми Кларк (Jeremy Clark), Клифф Колби (Cliff Colby), Пэт Кристенсон (Pat Christenson), Питер Алан Дэйви (Peter Alan Davy), Джейн Дековен (Jane DeKoven), Джонатан Дюран (Jonathan Duran), Люп Эдгар (Lupe Edgar), Виктор Гавенда (Victor Gavenda), Сьюки Гир (Suki Gear), Бекки Морган (Beckie Morgan), Эрика Нортон (Erika Norton), Кристи Пэйн (Christie Payne), Джэнис Пирс (Janice Pearce), Нэнси Рейнхардт (Nancy Reinhardt), Шэрон Силден (Sharon Selden), Кэти Симпсон (Kathy Simpson), Джеймс Тэлбот (James Talbot), Тиффани Тэйлор (Tiffany Taylor), Бентли Вулф (Bentley Wolfe) и Лайза Янг (Lisa Young).

И, наконец, сердечная благодарность Марджори Бир (Marjorie Baer) за дружескую поддержку и за то, что она втянула меня в этот проект.

# ВВЕДЕНИЕ

---



Векторная графика в Web! Именно это обещала технология Flash во времена лавинообразного роста сети Интернет, когда все хотели обозначить свое присутствие в Web цветом и анимацией. Это и вызвало к ней такой интерес. Flash дала Web-дизайнерам эффективный способ передавать графику и анимацию по сетям с ограниченной пропускной способностью, которые имелись в распоряжении большинства посетителей. Плюс к этому полный набор инструментов рисования для создания и анимирования графического контента. Еще одно достоинство Flash — простые средства написания сценариев для реализации интерактивной работы.

В конечном итоге способность Flash эффективно доставлять графику через Интернет с учетом ограничений по скорости соединения привлекла внимание тех разработчиков Web-контента, которые стремились создавать сложные интерактивные управляемые данными сайты и разбирались в программировании сценариев. Язык ActionScript удовлетворил их потребности. По мере того как высокоскоростной Интернет распространялся все шире, разработчики стали включать более длительные и сложные анимации. Flash научили отображать видео. С каждой новой версией возможности продукта расширялись. Версия Adobe Flash CS3 Professional ориентирована не только на дизайнеров, желающих создавать красивую и не слишком ресурсоемкую графику и анимацию, но и на разработчиков, заинтересованных в надежных мультимедийных Интернет-приложениях.

## 0 технологии Flash

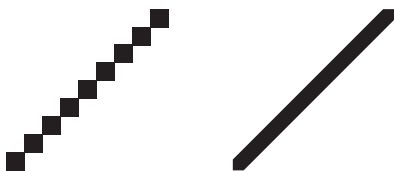
Технология Flash впервые появилась на свет в виде очень удачной небольшой программы FutureSplash Animator для создания и анимирования векторной графики. В 1997 году компания Macromedia приобрела эту программу, изменила название на Flash и позиционировала ее на рынке как инструмент создания графического контента для Web. В 2005 году компания Adobe купила Macromedia и приняла от нее эстафету.

Ранние версии Flash были прекрасными образчиками приложений для конструирования Web-сайтов, предоставляя все необходимое для создания визуально привлекательных, а не чисто текстовых сайтов: средства создания графических элементов, их анимирования, разработки интерактивного интерфейса и подготовки HTML-разметки, с помощью которой эти элементы

отображались в браузере. В версии Flash CS3 все это осталось, но добавились еще более изощренные графические инструменты, специализированные средства импорта графики, созданной в Adobe Photoshop и Adobe Illustrator, инструменты для импорта и воспроизведения видео, новые компоненты для разработки пользовательского интерфейса и привязки к данным, а также язык ActionScript 3.0 для программирования сложных интерактивных сценариев. Flash превратилась в набор инструментов для создания так называемых обогащенных Интернет-приложений (Rich Internet Application — RIA). В качестве такого приложения может выступать что угодно: онлайн-магазин, корпоративный образовательный модуль, сайт для просмотра видеоклипов, броская витрина для продвижения последней модели автомобиля, снабженная к тому же настраиваемым интерфейсом для виртуального тест-драйва.

### Векторная и растровая графика

Данные, лежащие в основе векторной и растровой графики, похожи в том смысле, что те и другие — это инструкции, следуя которым компьютер создает изображение на экране. Однако, растровые изображения занимают больше места и обладают меньшей гибкостью; векторная же графика компактнее и допускает масштабирование. В растровой графике изображение представлено в виде совокупности точек, тогда как в векторной оно математически описывается как последовательность прямых линий и дуг (рис. i.1). Возьмем, к примеру, черный горизонтальный отрезок длиной 1 дюйм на белом фоне. В растровом его представлении инструкции формулируются примерно так: поставить белую точку, поставить белую точку, поставить черную точку, поставить черную точку, поставить черную точку и повторять, пока черных точек не наберется на дюйм. Затем снова идут белые точки, пока ими не будет заполнен весь экран. В векторном же представлении будет лишь формула прямой линии плюс координаты ее начала на экране.



**Рис. i.1.** Для рисования растрового изображения компьютер должен получить инструкции для нанесения каждой составляющей его точки. Векторное изображение математически описывается прямыми и кривыми линиями. Растровый отрезок прямой (слева) выглядит гораздо грубее векторного (справа). Невозможно увеличить представленную растром линию без потери качества. А векторную — сколько угодно, при этом никакой зубчатости не появится.



## Что делает Flash особым инструментом Web-дизайна?

С самого начала Flash бросила вызов судьбе, предложив возможность доставлять через Web векторные изображения. А в чем же преимущества векторной графики? В том, что уменьшается размер файлов, а, значит, сокращается время загрузки. К тому же векторные изображения масштабируемы, а это означает, что пропорции сайта сохранятся, если пользователь изменит размеры окна браузера.

Еще одно достоинство Flash — способность к прогрессивной и потоковой загрузке. В результате некоторые элементы сайта становятся видны сразу же, пока остальная информация еще продолжает поступать. В совокупности это создает у пользователя благоприятное впечатление.

К числу прочих особенностей Flash следует отнести возможность создавать оригинальные изображения с помощью кривых Безье и естественных инструментов рисования; средства для импорта изображений, звука и видео, а также механизмы создания анимаций и интерактивности. Со временем инструменты Flash для поддержки интерактивности стали более надежными. Версия Flash CS3 включает полноценный объектно-ориентированный язык сценариев в двух вариантах ActionScript 2.0 и 3.0. Оба совместимы со спецификацией ECMA-262, лежащей в основе языка JavaScript, поэтому их легко освоит любой, кто знаком с этим языком.

## Как во Flash организуется анимация

В технологии Flash применяются стандартные методы анимации для создания иллюзии движения. Художник готовит последовательность статических изображений, в которой каждое последующее немного отличается от предыдущего. При быстрой смене этих изображений создается впечатление непрерывного движения. Инструменты, включенные во Flash, позволяют создавать, организовывать и синхронизировать анимацию нескольких графических элементов, звукового сопровождения и видеоклипов.

## Формат Flash-файлов

Flash предоставляет как среду для создания контента, так и систему воспроизведения, позволяющую визуализировать этот контент на локальном компьютере или в Web-браузере. Графика, анимация и описание интерактивного интерфейса хранятся в файлах с расширением .fla, которые часто называют FLA-файлами. Чтобы подготовить контент к просмотру через Web FLA-файлы преобразуются в формат Flash-плеера. Файлы для плеера имеют расширение .swf (произносится «свиф»).

## Как доставляется Flash-контент

Система публикации Flash создает HTML-код, необходимый для отображения Flash-контента в браузере. Можно выбрать и иные способы доставки Flash-контента, например, в виде анимированных GIF-файлов или видеоклипа для программы QuickTime. В процессе публикации Flash создает и эти альтернативные формы.

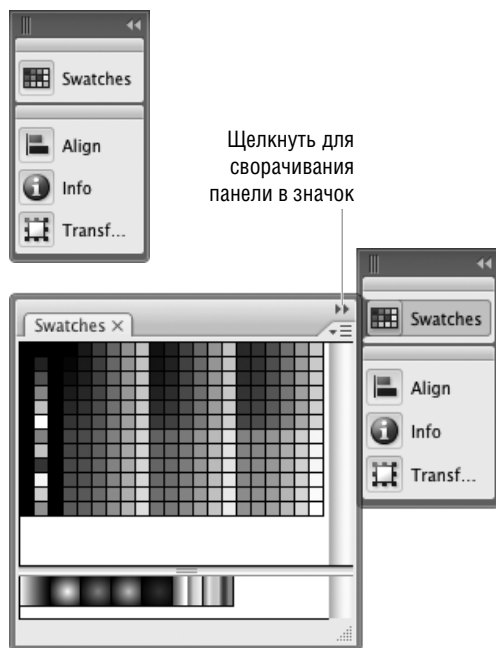
## Что нового появилось в версии Flash CS3

Компания Adobe немало потрудились, чтобы интегрировать продукты из линеек Adobe и Macromedia. В результате всех изменений и улучшений новый пользовательский интерфейс Flash стал больше похож на другие программы Adobe. Поскольку Flash входит в состав комплекта Adobe Creative Suite, то он легко позволяет загружать файлы, созданные, например, с помощью Photoshop или Illustrator. Кроме того, в эту версию Flash включены новые графические инструменты, средства для повторного использования анимаций и последняя версия языка ActionScript (3.0). Рассмотрим все это подробнее.

### Улучшенный интерфейс

**Все панели стыкуются.** Пользователи Windows давно привыкли к стыкуемым панелям; а теперь эта возможность оптимизировать использование экрана появилась и у пользователей Macintosh. Непростыкованные панели во время буксировки становятся прозрачными, что помогает точнее расположить их на экране.

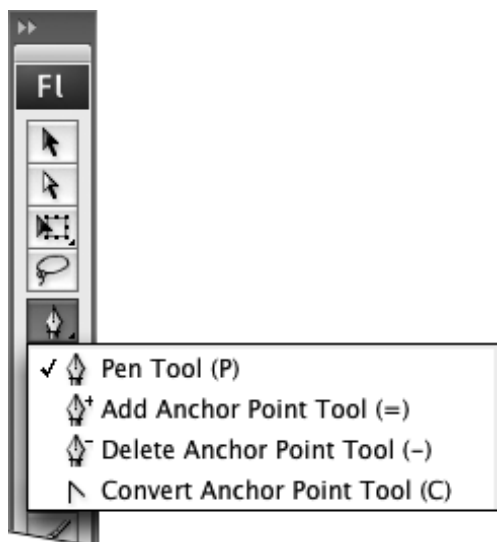
**Значки панелей.** Чтобы освободить место на рабочем столе, можно свернуть пристыкованную панель (или группу панелей) в значок (рис. i.2). Одиночный щелчок восстанавливает полноразмерную панель. Можно задать настройки так, чтобы панель автоматически возвращалась в свернутое состояние при щелчке вне нее либо оставалась открытой, пока не будет нажата специальная кнопка.



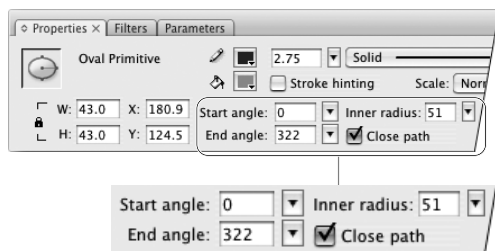
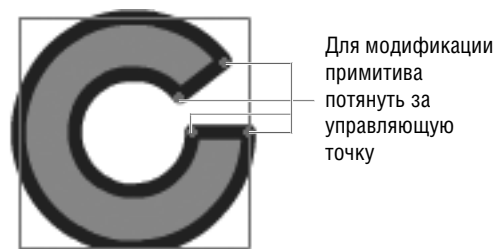
**Рис. i.2.** В версии Flash CS3 можно свернуть пристыкованные панели (или группы панелей) в значок (верхний рисунок). При щелчке по панели восстанавливается ее исходный размер (нижний рисунок). Настройки позволяют указать, должна ли панель возвращаться в свернутое состояние при щелчке вне нее или это нужно делать явно путем щелчка по кнопке с двумя стрелочками.

### О программе Adobe Flash Player

Когда Flash только появилась, необходимость использовать специальный плеер для просмотра Flash-контента в браузере считалась недостатком. Дизайнеры опасались, что пользователи не захотят тратить время на загрузку еще одной надстройки над своим браузером. Но с тех пор Flash стала стандартом де-факто для доставки обогащенного Интернет-контента — в особенности интерактивной векторной графики и анимации, и программа Flash Player получила широкое распространение. В Adobe считают, что примерно на 96 процентах машин, имеющих доступ к Интернету, уже установлен Flash Player версии 6 или выше, при этом более чем в 80% браузеров в США, Канаде, Великобритании, Германии, Франции и Японии стоит даже версия 9.



**Рис. i.3.** Инструменты Безье позволяют модифицировать пути, добавляя, удаляя или преобразуя точки. (Перо также сохранило возможность модифицировать расположенные на пути точки.)



Ввести новые значения свойств примитива

**Рис. i.4.** Свойства, описывающие примитивные формы, например овал, можно редактировать даже после создания формы. Значения можно изменять в инспекторе свойств или манипулируя управляющими точками на самом примитиве.

## Новые возможности конструирования

**Переработанные инструменты Безье.** Инструмент «Перо» во Flash CS3 больше напоминает перо, применяемое в Illustrator. Изменения, внесенные в математические алгоритмы, поддерживающие работу пера, привели к тому, что пути рисуются более точно и с меньшим числом точек. Для модификации существующих путей теперь есть специальные инструменты Безье (рис. i.3).

**Примитивные фигуры.** Во Flash CS3 появились два новых инструмента рисования фигур: овальный примитив и прямоугольный примитив (рис. i.4). Некоторые свойства овалов и прямоугольников и раньше можно было изменять после создания (например, размеры, насыщенность и цвет штриха), теперь же свойств, допускающих редактирование, стало больше. Например, степень скругленности углов прямоугольника можно изменять с помощью инспектора свойств или потянув за управляющую точку на самом прямоугольнике.

**Охватывающий прямоугольник напоминает о типе объекта.** Определить, объект какого типа выбран в данный момент (объект-рисунок, примитивная форма, символ, группа или что-то другое) можно по цвету охватывающего прямоугольника. Цвета можно настраивать.

**Неискажающее масштабирование.** Ранее неискажающее (9-дольное) масштабирование (9-slice scaling) было видно только в режиме редактирования символов или во Flash-плеере. В версии Flash CS3 реализован предварительный просмотр эффекта неискажающего масштабирования в среде конструирования и для символов видеороликов.

**Повторно используемые фильтры.** Фильтры и их настройки можно копировать из одного объекта в другой.

**Импорт данных из программ Photoshop и Illustrator.** В Flash CS3 имеются специальные инструменты для импорта данных из программ Photoshop и Illustrator. При выборе PSD или AI-файла можно указать, надо ли импортировать отдельные слои с сохранением структуры и возможности редактировать векторные элементы или следует преобразовать слои и/или элементы в растровые изображения (рис. 1.5).

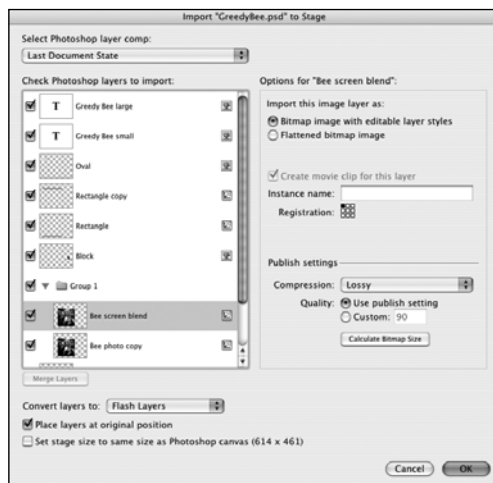
## Усовершенствованная анимация

**Повторное использование движения.** Теперь можно быстро воссоздать анимацию, построенную путем интерполяции движения (motion tween) различных объектов. Изменения, описывающие движение выбранного объекта, можно скопировать командой **Copy Motion**, а затем применить их к другому объекту с помощью команды **Paste Motion**.

**Преобразование движения в сценарий.** Новая команда **Copy Motion As ActionScript 3.0** также позволяет скопировать существующее движение и применить его к другому объекту. Однако эта команда преобразует анимацию с помощью временной шкалы в код, который затем вставляется в редактор ActionScript. Далее получившийся сценарий применяется к экземпляру объекта на арене.

## ActionScript 3.0 и прочее

Flash CS3 включает последнюю версию языка ActionScript 3.0, а также предыдущую версию ActionScript 2.0. Новый инструмент экспорта в формат QuickTime позволяет доставлять Flash-контент (как анимацию с помощью временной шкалы, так и представленную в виде сценария на языке ActionScript) в виде MOV-файлов для программы QuickTime. Шаблоны публикации Flash теперь создают небольшой JavaScript-файл, который активирует контент так, чтобы он сразу же стал доступен в таких программах просмотра, как Microsoft Internet Explorer.



**Рис. 1.5.** Инструмент PSD Importer предлагает вам решить, как поступать с каждым слоем файла, подготовленного в Photoshop. Параметры импорта позволяют соблюсти баланс между качеством изображения и возможностью редактировать его. Инструмент AI Importer делает то же самое для файлов, подготовленных в Illustrator.

## Как пользоваться этой книгой

Подобно всем прочим книгам из серии «Быстрый старт», эта побуждает вас отказаться от пассивного чтения и поскорее приступить к работе с программой. Приведенные упражнения помогут вам освоиться с возможностями Flash. Эта книга годится как для начинающих пользователей, только приступающих к работе с Flash, так и для дизайнеров среднего уровня. В первых главах излагаются основы создания графических элементов с помощью инструментария Flash. Затем вы научитесь преобразовывать графические элементы в анимации. Далее речь пойдет о создании базовых элементов пользовательского интерфейса, например, кнопок с эффектом наката. Чтобы сделать контент интерактивным, вы поработаете с панелью **Actions**, позволяющей писать сценарии на языке ActionScript. Кроме того, вы узнаете об импорте и работе с различными видами контента, не относящегося к Flash: графическим материалам, созданным в других программах, звуковому сопровождению и видео. И, наконец, вы научитесь пользоваться механизмом публикации Flash, который готовит HTML-разметку для размещения Flash-контента в Web.

### Что вы должны знать

Предполагается, что:

- на ваш компьютер установлен Adobe Flash CS3;
- вы знакомы с операционной системой;
- вы умеете выполнять простейшие действия: открывать, закрывать и сохранять документы; открывать, закрывать, изменять размеры, сворачивать и восстанавливать окна документов и диалогов; пользоваться иерархическими и контекстными меню, переключателями и флажками; выполнять такие стандартные команды, как копирование, вырезание, вставка, удаление и отмена.

## О кроссплатформенности

Рабочая среда на платформах Windows и Macintosh очень похожа. Однако есть и некоторые различия в интерфейсе. Там, где такие различия существенны, мы будем описывать процедуры для каждой платформы в отдельности. Снимки с экрана сделаны то на одной, то на другой платформе, и обычно не оговаривается, на какой именно. Если же некоторая функция заметно различается на разных платформах, различия иллюстрируются особо. Когда функция реализована только на какой-то одной платформе, это отмечается в тексте. Изначально компьютеры Macintosh комплектовались собственными клавиатурами, на которых некоторые клавиши имели нестандартные надписи (например, Return вместо Enter и Delete вместо Backspace). В этой книге указанные клавиши называются Enter и Delete соответственно.

### Выполнение команд с клавиатуры

У большинства команд из меню Flash есть клавиатурные эквиваленты. Соответствующая клавиша или комбинация клавиш указаны в меню справа от названия команды. При первом упоминании команды в этой книге мы приводим также ее клавиатурный эквивалент, а при последующих упоминаниях он опускается. Полный список команд можно найти на сопроводительном сайте книги по адресу <http://www.peachpit.com/title/0321502914>.

## Контекстные меню

На обеих платформах Windows и Macintosh имеются контекстные меню. Для доступа к такому меню нужно щелкнуть мышью по элементу Flash-ролика, удерживая клавишу **Control**, на платформе Mac или щелкнуть правой кнопкой мыши на платформе Windows. Меню будет содержать команды, применимые к данному элементу. По большей части эти команды присутствуют также в главном меню, поэтому обычно мы не будем говорить о них как об альтернативах описываемым командам. В этой книге мы не станем отмечать, когда пользоваться контекстным меню удобнее, или указывать, что в контекстном меню есть команда, отсутствующая в главном меню.

## Графика и сценарии

Flash-рисунки, рассматриваемые в этой книге, создать несложно. В большинстве случаев они состоят из простых геометрических фигур, а, значит, вы сэкономите время, сосредоточившись на работе Flash, а не на воспроизведении сложной графики. Чтобы еще облегчить вам жизнь, на сопроводительном сайте по адресу <http://www.peachpit.com/title/0321502914> имеются файлы с графическими элементами, необходимыми для выполнения всех упражнений. В главе 13 вы научитесь программировать сценарии, описывающие простые интерактивные действия. Полные тексты сценариев также есть на сайте.

# СРЕДА РАЗРАБОТКИ FLASH

---

## 1

Прежде чем приступать к созданию проектов в программе Adobe Flash CS3 Professional, будет полезно немного познакомиться со средой разработки, понять, из каких компонентов она состоит и как с ними обращаться. При первом открытии Flash появляется страница приветствия Flash Welcome. Она дает доступа ко многим документам и операциям. При открытии документа Flash вы оказываетесь в среде разработки. Каждый документ Flash состоит из четырех основных частей: временная шкала (Timeline), содержащая описание всех кадров, слоев и сцен в ролике; панель редактирования (Edit bar), где отображается текст и меню для выбора символов и сцен, с которыми вы собираетесь работать; арена (Stage) — та область, в которой отображается ролик; и монтажный стол (Pasteboard) — дополнительное рабочее пространство, окружающее арену. Арена и монтажный стол появляются, когда вы редактируете документ. Временную шкалу и панель редактирования можно скрыть. Можно открывать любые другие панели и инструменты, которые необходимы для работы.

Как выглядит среда разработки Flash? Как получить доступ к инструментам и различным представлениям документа? В этой главе мы дадим лишь краткий обзор, а в последующих подробно расскажем, как работать с каждым элементом.

## Работа с документами Flash

Простейшие операции над документами — открытие, закрытие и сохранение — знакомы любому сколько-нибудь опытному пользователю. Правда, создание нового документа производится несколько иначе, чем в других программах, так как Flash может создавать документы разных типов. Помощь в открытии и создании документов предложит вам окно приветствия.

Чтобы задать параметры запуска:

1. Из меню приложения Flash (Mac) или из меню **Edit** (Редактирование) (Windows) выберите пункт **Preferences** (Установки). Появится диалоговое окно Preferences, открытое на категории **General** (Общие).
2. Из раскрывающегося списка **On Launch** (При запуске) выберите один из следующих вариантов (рис. 1.1):
  - No Document (Нет документа). Полоса меню и панели появляются, но никакой документ не открывается.
  - New Document (Создать документ). При запуске открывается новый документ.
  - Last Documents Open (Последние открывавшиеся документы). Открываются те документы, которые были открыты в момент завершения предыдущего сеанса.
  - Welcome screen (Страница приветствия) (режим по умолчанию). При каждом запуске и после закрытия всех документов в ходе работы появляется страница приветствия.
3. Нажмите **OK**.

**C**

Чтобы быстро переключиться в режим создания нового документа при запуске, отметьте флажок **Don't show again** (Больше не показывать) в левом нижнем углу страницы приветствия. Появится диалоговое окно с напоминанием о том, что для того чтобы восстановить режим показа страницы приветствия при запуске, необходимо изменить параметры в категории **General** в окне Preferences.



**Рис. 1.1.** Список On Launch в категории **General** в диалоговом окне Preferences позволяет указать, какие документы открывать при запуске Flash и открывать ли вообще.

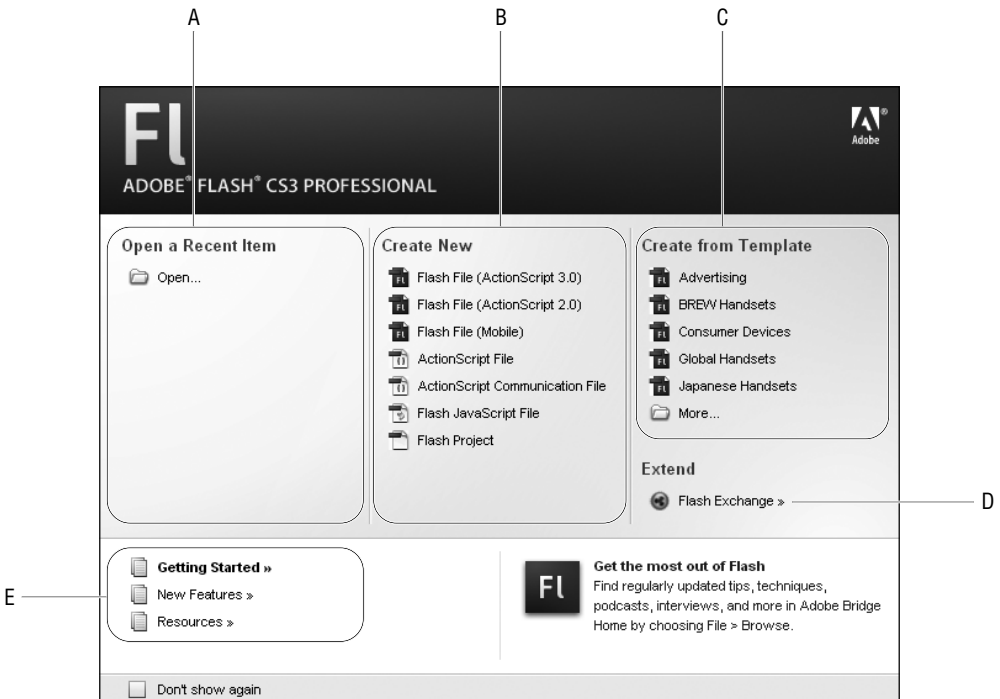
### О параметрах

Показывать или не показывать страницу приветствия — это лишь один из множества настроечных параметров Flash. Для задания параметров служит диалоговое окно Preferences. Из меню Flash (Mac) или меню **Edit** (Windows) выберите пункт **Preferences** и укажите категорию в списке слева. Задайте параметры, относящиеся к выбранной категории, в основной части окна. Всего есть девять категорий: General (Общие), ActionScript, Auto Format (Автоформат), Clipboard (Буфер обмена), Drawing (Рисунков), Text (Текст), Warning (Предупреждения), PSD File Importer (Импортер PSD-файлов) и AI File Importer (Импортер AI-файлов). Мы будем знакомиться с различными параметрами по мере возникновения потребности.



### Обзор страницы приветствия

По умолчанию при запуске Flash появляется страница приветствия. На ней расположены гиперссылки, позволяющие быстро открывать документы. Можно открыть новый документ, документ, с которым вы недавно работали, или шаблон. Можно заглянуть в учебные руководства по Flash или на сайт Adobe Exchange, с которого можно загрузить расширения, созданные сторонними фирмами, например, новые компоненты, эффекты временной шкалы (Timeline Effects) или поведения (Behaviors) (рис. 1.2).



**Рис. 1.2.** На странице приветствия присутствуют операции, которые часто выполняются в начале сеанса: открытие документа, с которым вы недавно работали (A), создание нового документа Flash (B) или создание документа по шаблону (C). Есть также ссылки для перехода на сайт Adobe Exchange, где можно найти расширения, созданные сторонними фирмами (D). Для новичков имеются ссылки на учебные руководства, поставляемые вместе с Flash, и на другие образовательные ресурсы (E).

### Чтобы создать новый документ Flash:

- В разделе **Create New** (Создать новый) на странице приветствия щелкните по ссылке **Flash File (ActionScript 3.0)**.

или

4. Выберите команду **File ⇒ New** (Файл ⇒ Создать) или нажмите комбинацию клавиш **⌘-N** (Mac) либо **Ctrl-N** (Windows).

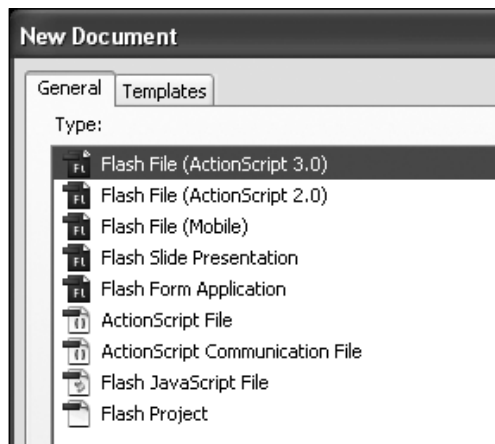
Появляется диалоговое окно **New Document** (рис. 1.3). В нем есть две вкладки: **General** (Общие) и **Templates** (Шаблоны). По умолчанию открыта вкладка **General**.

5. На вкладке **General** выберите пункт **Flash File (ActionScript 3.0)**.
6. Нажмите **OK**.

Открывается новый пустой документ.

**С**

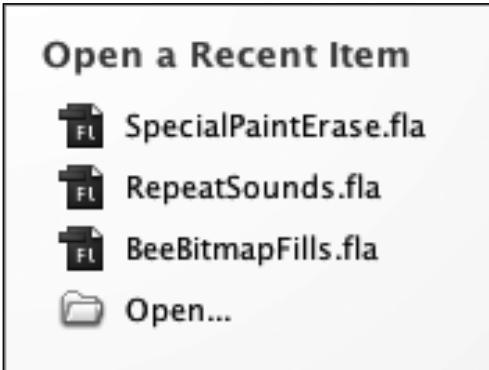
Для перехода на другую вкладку нажмите соответствующую ей кнопку (Mac) или щелкните по язычку в верхней части окна (Windows).



**Рис. 1.3.** Чтобы создать новый Flash-документ, выберите команду **File ⇒ New**. Диалоговое окно **New Document** откроется на вкладке **General**, где по умолчанию будет выбран пункт **Flash File (ActionScript 3.0)**. Нажмите **OK**.

### Какой тип Flash-файла выбрать?

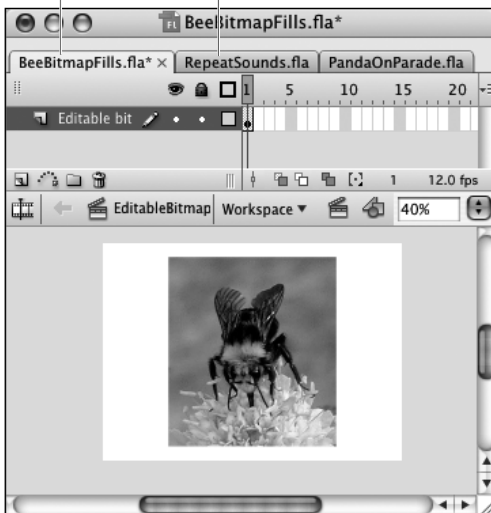
В разделе **Create New** на странице приветствия перечислены семь типов файлов, какой же из них вам нужен? В этой книге мы будем иметь дело исключительно с Flash-файлами, но все равно есть два варианта: **ActionScript 3.0** и **ActionScript 2.0**. Для большинства задач, связанных с созданием графики, текста или анимации, подойдет любой. Отличия проявляются, когда вы переходите к написанию сценариев (о них речь пойдет в главе 13). В разных версиях языка поддерживаются различные функции. В этой книге мы будем работать с версией **ActionScript 3.0**. Поэтому, когда в каком-то упражнении потребуется создать новый файл, выбирайте пункт **Flash File (ActionScript 3.0)**, если явно не оговорено противное. Если позже вы захотите воспользоваться другой версией **ActionScript**, то сможете изменить тип файла на **ActionScript 2.0** (или 1.0) в диалоговом окне **Publish Settings** (Параметры публикации). (Механизму публикации в Flash посвящена глава 17.)



**Рис. 1.4.** В разделе **Open a Recent Item** на странице приветствия может быть представлено до девяти ссылок на файлы, с которыми вы недавно работали. При щелчке по имени файла открывается соответствующий документ.

Щелкнуть для закрытия документа

Щелкнуть для просмотра документа



**Рис. 1.5.** По умолчанию во Flash CS3 документы открываются в отдельных вкладках. Чтобы сделать документ активным, щелкните по язычку его вкладки. Чтобы закрыть документ, щелкните по крестику справа от названия вкладки.

### Чтобы открыть существующий документ:

- В разделе **Open a Recent Item** (Открыть последний открытый документ) на странице приветствия щелкните по имени файла (рис. 1.4). Flash сразу откроет этот файл.

или

1. На странице приветствия в разделе **Open a Recent Item** щелкните по ссылке **Open** (Открыть).

или

Выберите из меню команду **File ⇨ Open** (Файл ⇨ Открыть).

Появится диалоговое окно **Open**.

2. Найдите в списке интересующий вас файл.
3. Выберите его.
4. Нажмите кнопку **Open** (Открыть).

**С**

По умолчанию в среде разработки Flash каждый документ открывается в отдельной вкладке в главном окне приложения. Вкладки неактивных документов выкрашены серым цветом. Чтобы сделать документ активным, щелкните по язычку вкладки (рис. 1.5). Чтобы каждый документ открывался в отдельном окне, выберите из меню **Window** (Окно) пункт **Tile** (Мозаика) или **Cascade** (Каскад). На платформе Mac можно также задать параметр, определяющий режим открытия документа: во вкладке или в отдельном окне. В меню Flash выберите пункт **Preferences**, а в категории **General** сбросьте флажок **Open Documents in Tabs** (Открывать документы во вкладках) и нажмите **OK**.

**С**

Чтобы изменить порядок следования вкладок, достаточно просто перетащить вкладку на новое место (рис. 1.6).

## C

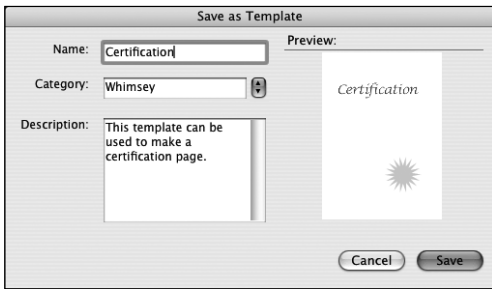
Когда по ходу работы с документом вы добавляете и удаляете графику, звук, видео и т.д., размер файла только растет, поскольку удаленные объекты физически в нем остаются. Чтобы уменьшить размер файла, выберите команду **File ⇒ Save and Compact** (Файл ⇒ Сохранить с уплотнением). По мере редактирования файла он снова будет быстро расти. Уплотнять его имеет смысл в самом конце работы или когда размер важен, например, перед отправкой по почте.



**Рис. 1.6.** Чтобы изменить порядок вкладок, щелкните по язычку вкладки и перетащите ее вправо или влево. Текущий документ останется активным, но его вкладка окажется в другом месте (нижний рисунок).



**Рис. 1.7.** Чтобы превратить файл Flash в шаблон, по которому впоследствии можно создавать другие документы, выполните команду **File ⇒ Save as Template**.



**Рис. 1.8.** Команда **File ⇒ Save as Template** позволяет сохранять документы Flash для повторного использования. Разрешается создавать свои категории шаблонов и снабжать шаблоны краткими описаниями.

## Работа с шаблонами

Если вы постоянно работаете с Flash-документами одного и того же типа, например создаете баннеры определенного размера с одним и тем же цветом фона и похожими элементами, то можете сохранить основу документа как шаблон.

### Чтобы создать шаблон:

1. Откройте документ, который хотите превратить в шаблон.
2. Выберите команду **File ⇒ Save as Template** (Файл ⇒ Сохранить как шаблон) (рис. 1.7).  
Появляется диалоговое окно **Save as Template**, в котором документ показан в режиме предварительного просмотра (рис. 1.8).
3. В поле **Name** (Имя) введите имя шаблона.
4. Категорию можно задать двумя способами:
  - выбрать существующую из раскрывающегося списка **Category** (Категория);
  - создать новую, введя ее имя в поле **Category**.
5. В поле **Description** (Описание) лаконично опишите назначение шаблона.  
Хотя вводить в это поле можно текст произвольной длины, при нажатии кнопки **Save** (Сохранить) описание будет урезано до 255 символов. Но все равно имеет смысл сообщить потенциальным пользователям шаблона, для чего он предназначен.
6. Нажмите кнопку **Save**.

Файл сохраняется как главный шаблон документа в папке с именем **Templates**, которая находится внутри папки **Configuration** (см. врезку «Папка Configuration» ниже в этой главе).



Закончив создание шаблона, закройте документ. Если вы думаете, что после сохранения документа как шаблона вы работаете с документом, созданным на его основе, то это не так. Это все еще главный шаблон и останется таковым, пока вы его не закроете.

**Чтобы создать новый документ по шаблону:**

1. На странице приветствия в разделе **Create from Template** (Создать по шаблону) щелкните по имени шаблона или по значку папки с надписью **More** (Дополнительно) (рис. 1.9).

Появится диалоговое окно New From Template (Создать из шаблона).

или

Выберите команду **File** ⇒ **New**.

Откроется диалоговое окно New from Template (рис. 1.10). Окна New Document и New from Template отличаются только названием. В обоих есть две вкладки: **General** и **Templates**. Название окна изменяется в зависимости от того, какая вкладка активна. При щелчке по папке шаблонов на странице приветствия окно открывается на вкладке **Templates**.

2. Выберите подходящую категорию из списка **Category**.
3. Из списка **Templates** выберите нужный шаблон.

В диалоговом окне вы увидите, как выглядит первый кадр шаблона, а также его краткое описание, если оно было введено при создании.

4. Нажмите кнопку **OK**.

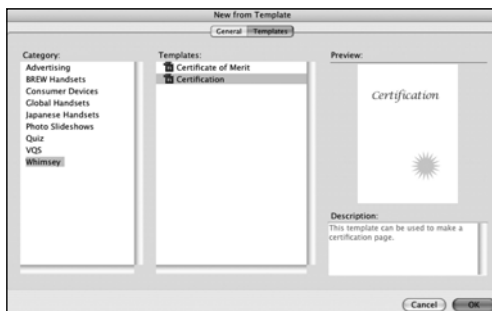
Flash открывает новый документ, в котором уже присутствует все содержимое шаблона.

**С**

При выполнении команды **File** ⇒ **New** диалоговое окно создания документа настроено так же, как при создании последнего документа. Если в последний раз вы создавали пустой документ, то оно откроется на вкладке **General**, а если документ по шаблону — то на вкладке **Templates**.



**Рис. 1.9.** Для открытия диалогового окна New from Template щелкните по имени шаблона на странице приветствия.



**Рис. 1.10.** На вкладке Templates выбранный с помощью списков Category и Templates шаблон показывается в области предварительного просмотра. Если для него имеется краткое описание, то вы увидите и его тоже.

**С**

На платформе Mac при открытии диалогового окна New from Template в списке **Templates** подсвечено имя последнего использовавшегося шаблона. Однако в области **Preview** (Просмотр) отображается шаблон, стоящий первым в списке. Если теперь нажать кнопку **OK**, то будет создан тот шаблон, который показан в области просмотра, а не тот, что подсвечен. Если вам нужен именно подсвеченный шаблон, щелкните по нему и дождитесь, пока он появится в области просмотра.

### Папка Configuration

Операционные системы, в которых работает Flash CS3, могут быть сконфигурированы в многопользовательском режиме, поэтому файлы и настройки разных пользователей должны храниться раздельно. Чтобы обеспечить это, применяются три папки Configuration. В папке уровня приложения хранятся настройки, общие для всех пользователей данного компьютера. Чтобы вносить в нее изменения, необходимы привилегии администратора. В папке уровня «всех пользователей» находятся главные файлы для всех пользователей, а в папке уровня пользователя — файлы с настройками отдельных пользователей.

Например, когда вы создаете шаблон, Flash помещает его в подпапку Folders папки Configuration уровня пользователя. Те же шаблоны, которые поставляются в составе дистрибутива Flash, находятся в папке уровня приложения.

Чтобы вручную добавить, удалить или переименовать файлы в любой папке Configuration, необходимо иметь соответствующие привилегии и спуститься по иерархии вложенных папок до нужной.

Наибольший интерес для отдельных пользователей Flash представляет папка Configuration уровня пользователя. В разных операционных системах пути к ней различны:

**Windows XP/2000 или Vista:** BootDrive\Documents and Settings\userName\Local Settings\Application Data\.

**Mac OS X:** HardDriveName:Users:username:Library:Application Support:

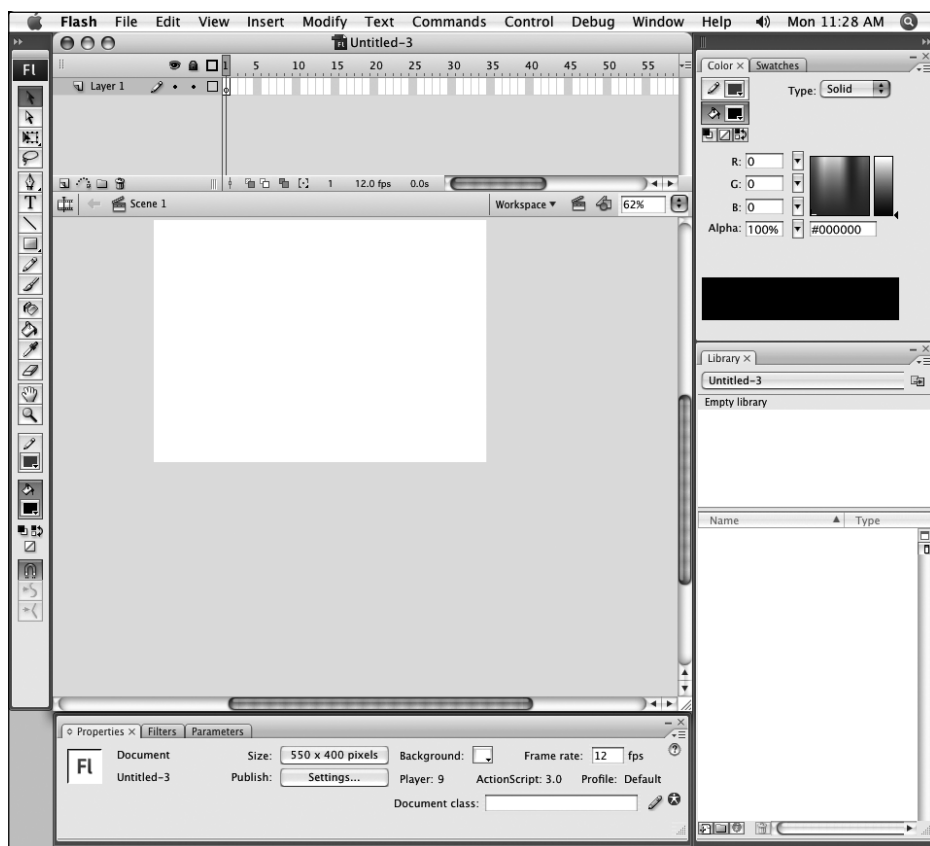
Начиная с этого места, путь одинаков. В английской версии Flash это Adobe\Flash CS3\en\Configuration. В локализованных версиях место en занимает код соответствующего языка.

## О среде разработки Flash

Запустив программу Flash, вы оказываетесь в среде разработки. Как и в большинстве приложений, полоса меню всегда находится сверху, но остальными компонентами вы можете управлять: открывать и закрывать документы и панели и располагать их по своему усмотрению. Это настраиваемая область экрана называется *рабочим пространством*.

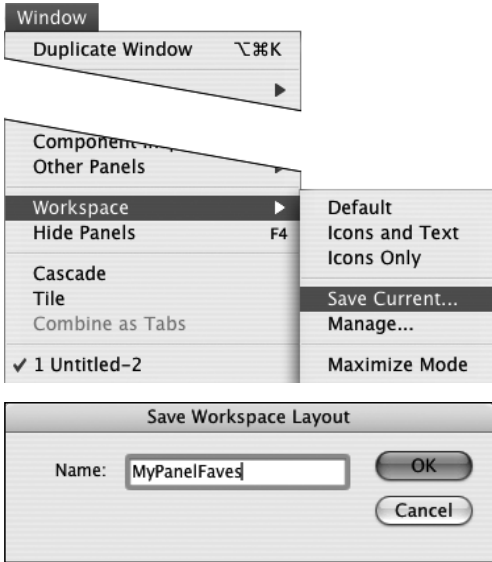
## Знакомимся с рабочим пространством

По умолчанию наиболее часто используемые панели пристыкованы к краям рабочего пространства (рис. 1.11). Рабочее пространство можно конфигурировать, открывая те или иные панели, изменяя их размеры и положение. (Чуть позже вы научитесь манипулировать панелями.) Новую конфигурацию панелей можно сохранить для последующего использования. Flash запоминает, какие панели были открыты, как они были сгруппированы или пристыкованы, а также размеры панелей и прочие параметры. В следующем сеансе запомненная конфигурация рабочего пространства будет восстановлена.

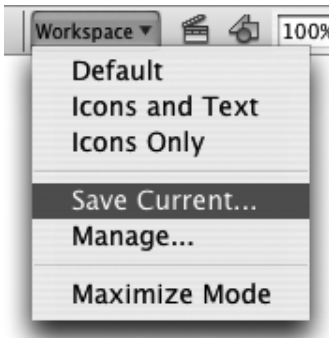


**Рис. 1.11.** По умолчанию панель Tools (Инструменты) пристыкована к левому краю, инспектор свойств — к нижнему краю, а панели Color (Цвет), Swatches (Образцы) и Library (Библиотека) — к правому краю.





**Рис. 1.12.** Чтобы получить доступ к диалоговому окну **Save Workspace Layout** (нижний рисунок), выполните команду **Window ⇒ Workspace ⇒ Save current** (верхний рисунок). Введите имя схемы рабочего пространства и нажмите **OK**.



**Рис. 1.13.** Выпадающее меню **Workspace** на панели редактирования документа дает быстрый доступ к командам управления рабочим пространством.

### Чтобы сохранить рабочее пространство:

1. Сконфигурируйте панели по своему усмотрению, как описано ниже в этой главе.
2. Выполните команду **Window ⇒ Workspace ⇒ Save current** (Окно ⇒ Рабочее пространство ⇒ Сохранить текущий). Появляется диалоговое окно **Save Workspace Layout** (Сохранить схему рабочего пространства) (рис. 1.12).
3. В поле **Name** (Имя) введите название схемы.
4. Нажмите **OK**.

Flash сохранит конфигурацию и добавит ее в меню **Workspace** (Рабочее пространство).

### Чтобы восстановить рабочее пространство по умолчанию:

- Выберите пункт меню **Window ⇒ Workspace ⇒ Default** (Окно ⇒ Рабочее пространство ⇒ По умолчанию).

Flash поместит панель **Tools** слева, инспектор свойств снизу, а к правому краю пристыкует панель **Library** и группу, содержащую панели **Color** и **Swatches**, представленные в виде вкладок.

**С**

Сохранить и выбрать схему рабочего пространства можно с помощью выпадающего меню **Workspace** (Рабочее пространство), которое находится на панели редактирования документа (рис. 1.13).

**С**

Чтобы восстановить нестандартную схему рабочего пространства, выберите ее имя из меню **Window ⇒ Workspace** или из выпадающего меню **Workspace** на панели редактирования документа.

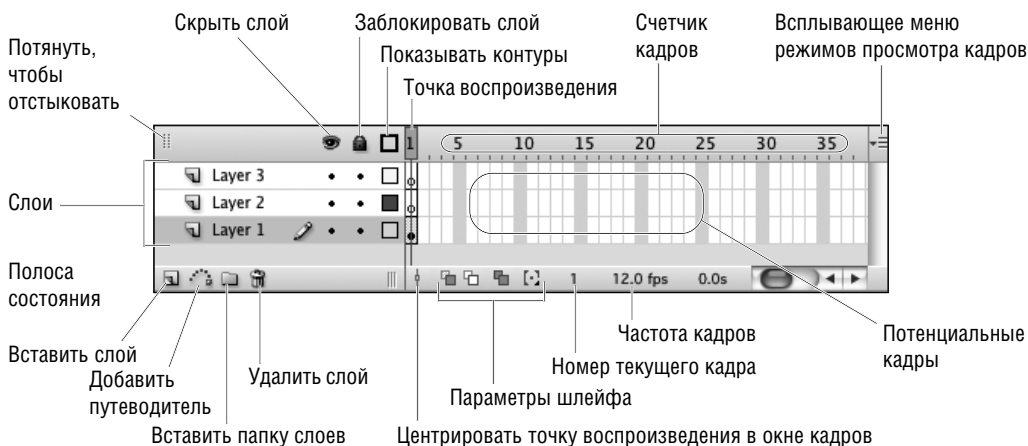
## Знакомимся с документом

Документ Flash состоит из временной шкалы (Timeline), в которой хранятся кадры, слои и сцены, составляющие ролик; арены (Stage), на которой находится графический контент ролика; панели редактирования, где отображается информация о том, что редактируется в данный момент, и имеются средства доступа к другим сценам и элементам; и монтажного стола (Pasteboard), обрамляющего арену со всех сторон (рис. 1.14). Помимо этого, среда разработки предоставляет различные панели, на которых расположены инструменты для работы с контентом (см. раздел «Использование панелей» ниже в этой главе).

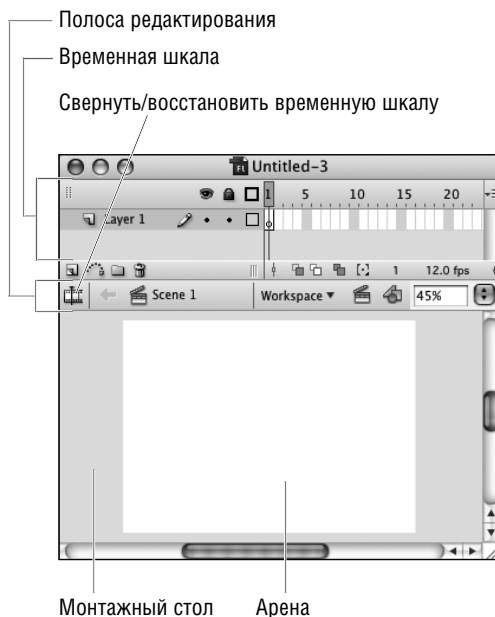
## Знакомимся с временной шкалой

Временная шкала содержит визуальные представления всех элементов ролика и является основой любого проекта. При создании анимаций вы будете постоянно ей пользоваться (см. главу 8).

На рис. 1.15 показаны основные элементы временной шкалы. Временную шкалу можно пристыковать к любому краю окна, сделать плавающим окном, свернуть или вовсе скрыть, чтобы освободить место для работы с элементами на арене.



**Рис. 1.15.** Временная шкала — это полная запись ролика. В ней представлены все сцены, кадры и слои, составляющие ролик. Кадры следуют в хронологическом порядке. Щелчок по любому кадру приводит к отображению его содержимого на арене.



**Рис. 1.14.** Документ Flash состоит из временной шкалы, панели редактирования и монтажного стола.

### Чтобы отстыковать окно временной шкалы:

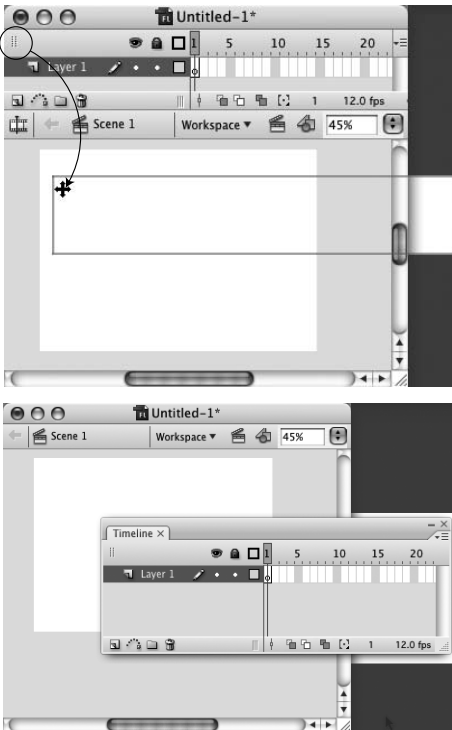
1. Наведите курсор на манипулятор (значок с двумя рядами точек), он расположен в левом углу полосы в верхней части окна Timeline. Курсор примет форму крестика.
2. Щелкните мышью и потяните окно (рис. 1.16). Серый контур покажет новое положение окна временной шкалы.
3. Отпустите кнопку мыши.

**С** Чтобы вновь пристыковать окно временной шкалы, выполните ту же процедуру в обратном порядке. Перетащите окно к верхнему или к любому другому краю документа. Если окно пристыковано к левому или правому краю арены, то временная шкала располагается вертикально.

**С** Во Flash CS3, если окно временной шкалы отстыковано, то для изменения его положения на рабочем столе необходимо потянуть плавающее окно за полосу заголовка. Если тянуть за манипулятор, то цели вы не достигнете; стоит отпустить кнопку мыши, как окно вернется в прежнее положение. Впрочем, пристыковать временную шкалу к краю окна можно и с помощью манипулятора.

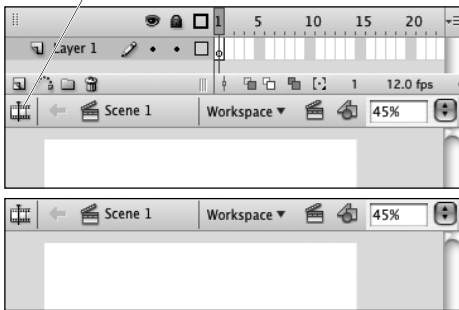
**С** На платформе Mac, если временная шкала пристыкована к нижнему или любому боковому краю, то двойной щелчок по манипулятору возвращает ее в начальное положение — у верхнего края.

**С** Чтобы свернуть окно временной шкалы, нажмите кнопку **Hide Timeline** (Скрыть временную шкалу) в левой части панели редактирования (рис. 1.17). Чтобы восстановить скрытую временную шкалу, нажмите ту же кнопку еще раз.



**Рис. 1.16.** Потяните временную шкалу за манипулятор — значок с двумя рядами точек в полосе заголовка (верхний рисунок) — и отпустите кнопку мыши. Окно временной шкалы станет плавающим (нижний рисунок).

Кнопка сокрытия временной шкалы

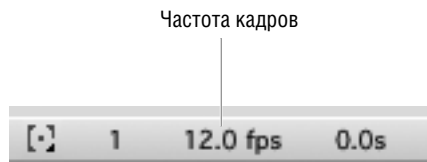


**Рис. 1.17.** Если окно временной шкалы открыто, то на панели редактирования присутствует кнопка Hide Timeline. Если временная шкала видима, то нажатие этой кнопки скрывает ее (верхний рисунок), а если невидима — восстанавливает (нижний рисунок).

## О свойствах документа

В диалоговом окне Document Properties (Свойства документа) можно задать параметры арены (размеры, цвет фона и единицы измерения для линеек и сеток), а также частоту кадров при воспроизведении ролика (рис. 1.18). Кадры — это живая кровь анимации, а частота определяет пульс биения сердца, которое гонит эту кровь. По умолчанию принята частота 12 кадров в секунду, подходящая для просмотра ролика через Web. (Для сравнения отметим, что стандартная частота кадров в обычном фильме вдвое больше). Подробнее о том, как частота влияет на анимацию, вы узнаете в главе 8.

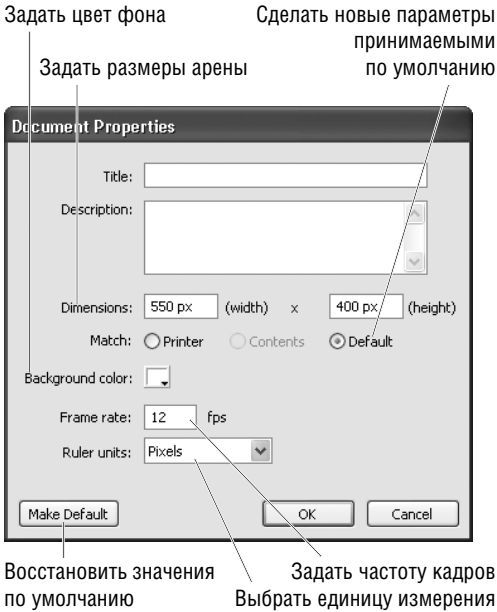
Во Flash CS3 в окне свойств документа появилась также возможность задавать метаданные для SWF-файлов. С помощью метаданных поисковые машины смогут найти ваши творения, опубликованные в Web, по заголовку или ключевым словам. Для задания некоторых свойств документа можно воспользоваться также вкладкой Document Properties в инспекторе свойств. Подробнее о работе с инспектором свойств мы еще поговорим ниже.



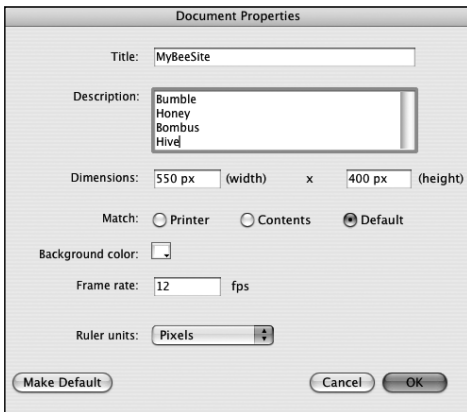
**Рис. 1.18.** Чтобы быстро открыть диалоговое окно Document Properties, дважды щелкните по области полосы состояния, где отображается частота кадров.

### Метаданные SWF-файла

В окне свойств документа есть поля Title (Заголовок) и Description (Описание). В момент публикации документа (см. главу 17) текст, введенный в эти поля, преобразуется в метаданные, присоединенные к SWF-файлу. Поисковые машины используют эти метаданные, чтобы помочь потенциальным зрителям найти ваше творение. Хотя версии Flash ниже 8 не создавали метаданных SWF, при публикации из Flash CS3 роликов, предназначенных для старых версий, метаданные все же генерируются. Отметим, что заголовок и ключевые слова, введенные в поле описания, не включаются в состав метаданных HTML-файла, содержащего ролик; это вы должны сделать самостоятельно.



**Рис. 1.19.** В окне свойств документа задаются все параметры арены. Задание единицы измерения для линеек приводит к изменению единиц измерения всех прочих характеристик арены. При щелчке по элементу управления цветом появляется текущий набор цветов, из которого вы можете выбрать цвет фона. Кнопка **Make Default** (Использовать по умолчанию) служит для того, чтобы сделать выбранные параметры принимаемыми по умолчанию для всех вновь создаваемых документов.



**Рис. 1.20.** Если ввести заголовок в поле Title и ключевые слова в поле Description, то в публикуемый ролик (SWF-файл) будут добавлены метаданные, доступные поисковым машинам.

## Чтобы открыть окно свойств документа:

Существует два способа:

- Выберите команду **Modify ⇨ Document** (Изменить ⇨ Документ) или нажмите комбинацию клавиш **⌘-J** (Mac) либо **Ctrl-J** (Windows).
- В полосе состояния временной шкалы дважды щелкните по частоте кадров (рис. 1.18). Появляется окно Document Properties (рис. 1.19).

**С** На вкладке Document Properties в инспекторе свойств также есть способ открыть окно свойств документа — кнопка **Size** (Размер). Подробнее об инспекторе свойств речь пойдет ниже в этой главе.

## Чтобы создать метаданные SWF-файла:

1. Откройте окно свойств документа.
2. В поле Title введите заголовок документа.
3. В поле Description введите описание или ключевые слова (рис. 1.20).

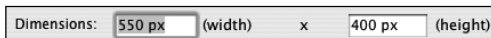
При публикации ролика SWF-файл будет содержать метаданные из полей Title и Description. Дополнительная информация о публикации роликов приводится в главе 17.

### Чтобы задать размеры арены:

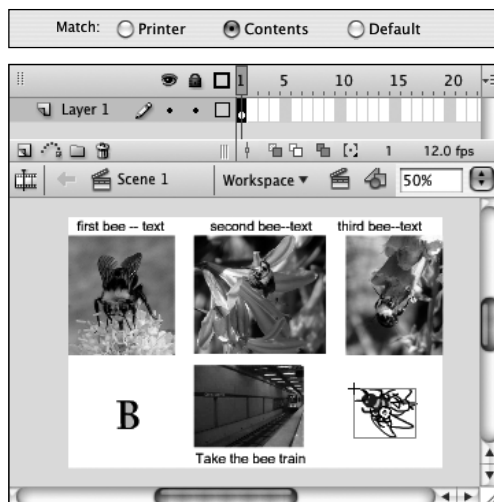
1. Откройте окно свойств документа и выполните одно из следующих действий:

- чтобы задать размеры арены, введите ширину и высоту в соответствующие поля (рис. 1.21). Подразумевается, что они выражены в единицах, указанных в списке **Ruler Units** (Единицы измерения линейки);
- чтобы создать арену, достаточно большую для размещения всех элементов ролика, выберите значение **Contents** (Содержимое) переключателя **Match** (Подобрать) (рис. 1.22). Flash автоматически вычислит минимально необходимые размеры арены и поместит соответствующие значения в поля ширины и высоты;
- чтобы размер арены соответствовал текущему размеру области для печати, установите переключатель **Match** в положение **Printer** (Принтер) (рис. 1.23). Flash возьмет размер листа бумаги из диалога **Page Setup** (Параметры страницы), вычитет величину полей и поместит вычисленные значения в поля ширины и высоты.

2. Нажмите **ОК**.



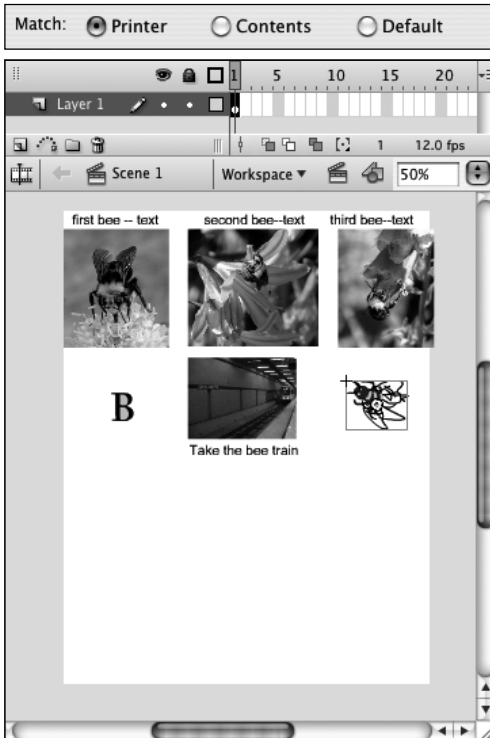
**Рис. 1.21.** Чтобы изменить пропорции арены, введите новые значения ширины и высоты в соответствующие поля в окне свойств документа.



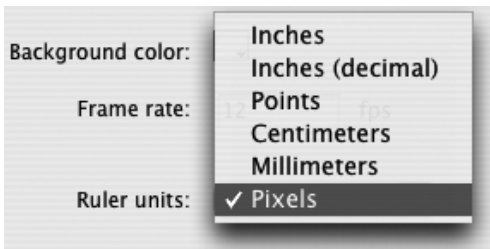
**Рис. 1.22.** Чтобы сделать арену достаточно большой для размещения всех элементов, установите переключатель **Match** в положение **Contents** (верхний рисунок). Flash создаст арену, на которой помещаются все созданные к этому моменту объекты (нижний рисунок). (Объекты, находящиеся на монтажном столе, учитываются при вычислении размера арены.)

### О монтажном столе

Монтажный стол в Flash CS3 расширяется с учетом потребностей в дополнительном пространстве. При перетаскивании элементов с арены на монтажный стол Flash увеличивает размеры последнего. Если вы поместите на монтажный стол большое изображение, которое не умещается целиком в окне открытого документа, то увидите, что ползунок в полосе прокрутки смещается к центру, так что какая-то часть стола окажется скрытой. Чтобы увидеть свой элемент, воспользуйтесь полосой прокрутки. (О том, как создавать графические элементы, вы узнаете в главе 2, а о том, как изменять их положение, — в главе 4.)



**Рис. 1.23.** Если переключатель Match в окне свойств документа установлен в положение Printer, то ширина и высота арены выбираются так, чтобы она поместилась на листе бумаги заданного в диалоге Page Setup размера за вычетом полей.



**Рис. 1.24.** Чтобы установить единицу измерения для документа, выберите ее из списка Ruler Units в окне свойств документа.

C

Если вы изменили размеры арены вручную или путем установки переключателя **Match** в положение Printer или Contents, то вернуться к размеру, подразумеваемому по умолчанию, можно, выбрав значение переключателя **Default** (По умолчанию).

C

Чтобы установить единицу измерения для документа, выберите ее из списка **Ruler Units** (Единицы измерения линейки) в окне свойств документа (рис. 1.24). Измерять линейные размеры можно в дюймах, десятичных долях дюйма, пунктах, сантиметрах, миллиметрах и пикселях. Выбранная единица используется для вычисления всех линейных размеров на арене: в линейках, сетках и размерах самой арены.

C

Если вы хотите сделать баннер высотой 1 дюйм и шириной 5 дюймов, но не знаете, сколько это будет в пикселях (стандартная единица измерения, принятая в Web), то за вас может произвести вычисления окно свойств документа. Сначала выберите в качестве единицы измерения дюймы. Введите 1 в поле Height и 5 в поле Width. Теперь измените единицу измерения на пиксели. Flash выполнит вычисления и установит размеры арены. (Отметим, что при вычислении используются экранные пиксели, то есть длина дюйма в ролике может отличаться от реальной длины. Зависит это от разрешающей способности монитора, на котором просматривается ролик.)

### Чтобы задать цвет фона:

1. Откройте окно свойств документа.
2. Щелкните по элементу управления **Background Color** (Цвет фона). Курсор принимает форму пипетки и появляется окно с образцами (рис. 1.25).
3. Выбрать цвет фона можно двумя способами:
  - щелкнуть по образцу;
  - ввести шестнадцатеричное значение цвета в поле сверху и нажать **Enter**.

Выбранный цвет отображается в элементе управления **Background Color**.

4. Нажмите **ОК**.

Арена перекрашивается выбранным цветом.

**С**

На вкладке Document Properties в окне инспектора свойств тоже есть элемент управления **Background Color** для задания цвета фона в ролике. О работе с инспектором свойств будет рассказано ниже, а об использовании элементов управления цветом — в главе 2.

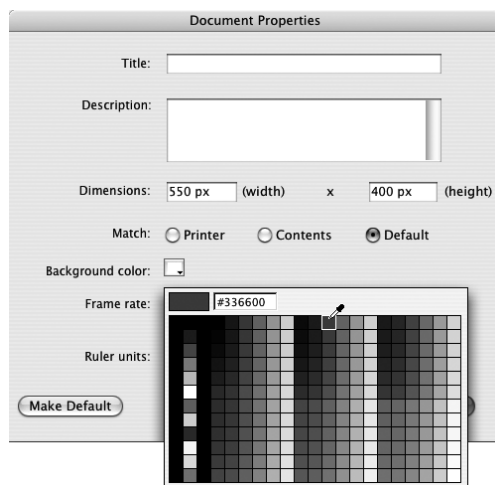
### Чтобы задать частоту кадров:

1. Откройте окно свойств документа.
2. В поле Frame Rate (Частота кадров) введите желаемое число кадров в секунду (рис. 1.26).
3. Нажмите **ОК**.

Новое значение появляется в полосе состояния текущего документа.

### Чтобы сохранить новые параметры как принимаемые по умолчанию:

В окне свойств документа нажмите кнопку **Make Default** (рис. 1.27). Текущие значения параметров будут в дальнейшем применяться по умолчанию.



**Рис. 1.25.** Чтобы перекрасить арену, выберите новый цвет с помощью элемента управления Background Color в окне свойств документа.



**Рис. 1.26.** Чтобы установить число кадров, показываемых в секунду, введите значение в поле Frame Rate (Частота кадров) в окне свойств документа.



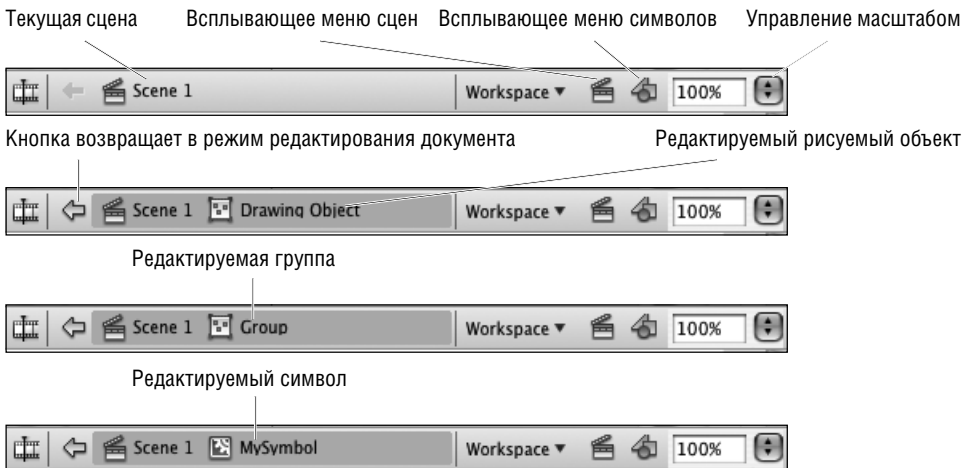
**Рис. 1.27.** Чтобы текущие параметры в окне свойств документа применялись по умолчанию ко всем вновь создаваемым документам, нажмите кнопку Make Default в левом нижнем углу окна.



### Знакомимся с панелью редактирования

В окне Flash-документа имеется еще один элемент — панель редактирования. По умолчанию она расположена под окном временной шкалы, если последнее пристыковано к верхнему краю арены. Но можно расположить панель редактирования и над пристыкованной временной шкалой. Если окно временной шкалы плавают или пристыковано к нижнему или одному из боковых краев арены, то панель редактирования остается в верхней части арены. При желании панель редактирования можно вовсе убрать. Этот элемент показывает, в каком режиме вы сейчас работаете (редактирование документа, объекта-рисунка, группы или символа). Всплывающие меню, связанные с панелью редактирования, позволяют переключаться между сценами, выбирать символ для редактирования или сразу же переходить в режим редактирования символа, а также изменять увеличение при просмотре арены. Кроме того, можно выбрать новое рабочее пространство и управлять наборами рабочих пространств (см. раздел «Знакомимся с рабочим пространством» выше). При редактировании объекта-рисунка, группы или символа на панели редактирования видно, какой элемент редактируется (рис. 1.28). Об объектах-рисунках и группах рассказывается в главе 5, о символах — в главе 7, о сценах — в главе 11.

Чтобы изменить положение панели редактирования, дважды щелкните по любой свободной точке в ней, удерживая клавиши **⌘-Shift** (Mac) или **Alt-Shift** (Windows). Если панель редактирования находилась под временной шкалой, то в результате такого щелчка она окажется над ней и наоборот. Чтобы скрыть панель редактирования, выберите команду **Window ⇒ Toolbars ⇒ Edit Bar** (Окно ⇒ Панели инструментов ⇒ Панель редактирования) или нажмите кнопку **Hide Edit Bar** (Скрыть панель редактирования), которая находится слева в ней же. Но, если панель редактирования не видна, то вам труднее будет понять, с каким именно объектом-рисунком, группой или символом вы сейчас работаете на арене.



**Рис. 1.28.** Панель редактирования расположена над ареной. С ее помощью вы получаете доступ к символам и сценам, а также изменяете увеличение. При редактировании объекта-рисунка, группы или символа на панели редактирования видно, с чем именно вы работаете. При нажатии кнопки **Back** (Назад) выполняется переход на один уровень вверх по иерархии редактируемых элементов; в конце концов вы вернетесь в режим редактирования документа.

## Линейки, сетки и направляющие

Для более точного позиционирования графических элементов на арене Flash предлагает линейки, сетки и направляющие. Включить эти визуальные средства можно с помощью меню **View** (Просмотр) (рис. 1.29) — отмеченный флажок означает, что соответствующая функция включена. В готовом ролике ни одно из этих вспомогательных приспособлений не присутствует.

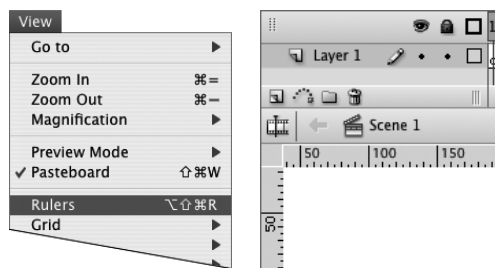
### Чтобы показать или скрыть линейки, сетки и направляющие:

С помощью меню **View** или эквивалентных комбинаций клавиш выполните одно из следующих действий:

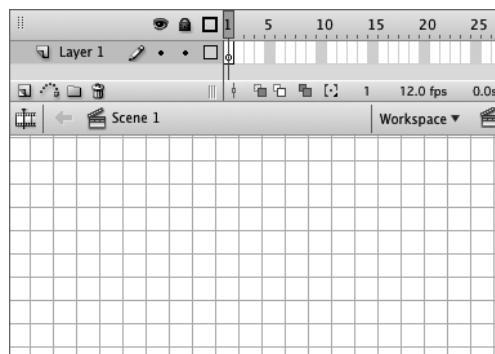
- Выберите пункт **Rulers** (Линейки) или нажмите комбинацию клавиш **Option-Shift-⌘-R** (Mac) либо **Ctrl-Alt-Shift-R** (Windows). Вдоль левого и верхнего краев арены появляются линейки (рис. 1.30). Единицы измерения устанавливаются в окне свойств документа.
- Выберите пункт **Grid** ⇨ **Show Grid** (Сетка ⇨ Показать сетку) или нажмите комбинацию клавиш **⌘-апостроф (')** (Mac) либо **Ctrl-апостроф (')** (Windows). На арену накладывается сетка (рис. 1.31). Сетка помогает точнее рисовать и позиционировать элементы, то есть выполняет ту же функцию, что миллиметровая бумага. Кроме того, в режиме Snap To Grid (Привязка к сетке) элементы располагаются точно вдоль линий сетки.
- Выберите пункт **Guides** ⇨ **Show Guides** (Направляющие ⇨ Показать направляющие) или нажмите комбинацию клавиш **⌘-точка с запятой (;)** (Mac) либо **Ctrl-точка с запятой (;)** (Windows). В этом режиме любая добавленная направляющая становится видимой. О том, как добавить направляющую, рассказывается в разделе «Работа с направляющими» ниже.
- Для быстрого доступа к средствам управления линейками, направляющими, привязки к сетке и другим параметрам документа вы в любой момент можете щелкнуть правой кнопкой мыши в свободной точке арены или монтажного стола и выбрать соответствующее подменю.



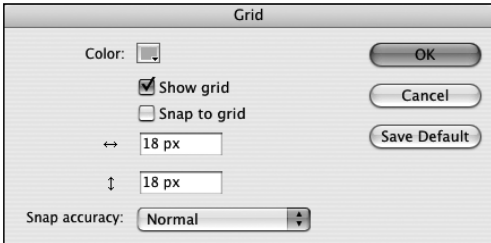
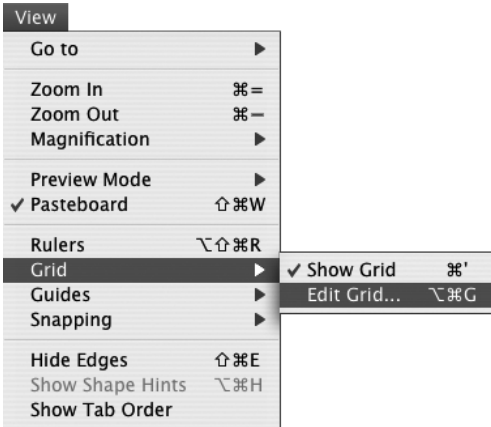
**Рис. 1.29.** Flash предлагает три вида вспомогательных визуальных средств: линейки, сетки и направляющие. Чтобы показать или скрыть их, воспользуйтесь меню View. На этом рисунке показано, как включить сетку.



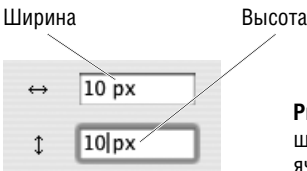
**Рис. 1.30.** При выборе пункта меню View [ Rulers (левый рисунок) на арене появляются линейки (правый рисунок).



**Рис. 1.31.** Наличие сетки помогает точнее позиционировать элементы при работе над документом.

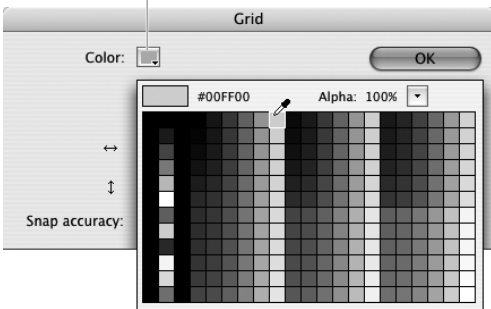


**Рис. 1.32.** Команда **View ⇒ Grid ⇒ Edit Grid** открывает диалоговое окно **Grid**, в котором можно задавать параметры сетки.



**Рис. 1.33.** Введите ширину и высоту ячейки сетки.

Управление цветом



**Рис. 1.34.** Чтобы задать цвет сетки, щелкните курсором-пипеткой по образцу цвета. Выбранный цвет отображается в элементе управления цветом.

## Чтобы задать параметры сетки:

1. Выполните команду **View ⇒ Grid ⇒ Edit Grid** (Просмотр ⇒ Сетка ⇒ Изменить сетку) или нажмите комбинацию клавиш **Option-⌘-G** (Mac) либо **Ctrl-Alt-G** (Windows). Открывается диалоговое окно **Grid** (рис. 1.32).
2. Чтобы задать шаг сетки:
  - введите ширину;
  - введите высоту (рис. 1.33).
3. Чтобы задать цвет сетки:
  - щелкните по элементу управления цветом **Color**. Курсор принимает форму пипетки и появляется окно с образцами;
  - щелкните по образцу цвета (рис. 1.34).

Новый цвет отображается в элементе управления **Color**.
4. Чтобы задать, на каком минимальном расстоянии от линии сетки должен находиться элемент, чтобы он автоматически был совмещен с сеткой, выберите из раскрывающегося списка **Snap Accuracy** (Точность привязки) одно из возможных значений.
5. Нажмите **OK**.

**C**

Сетка необязательно должна быть квадратной.

**C**

Можно задать параметры сетки, которые по умолчанию будут применяться ко всем вновь создаваемым документам. Для этого после ввода всех значений в окне **Grid** нажмите кнопку **Save Default** (Сохранить значение по умолчанию), а затем **OK**.

### Работа с направляющими:

1. Включив режим показа линеек, наведите курсор на вертикальную или горизонтальную линейку. Если в данный момент используется инструмент, отличный от «Стрелка», то курсор примет форму стрелки.
2. Щелкните мышью и отбуксируйте курсор на арену. В момент щелчка появится стрелочка, указывающая в каком направлении тянуть (рис. 1.35).
3. Отпустите кнопку мыши.

На арене появится вертикальная или горизонтальная линия.

**С**

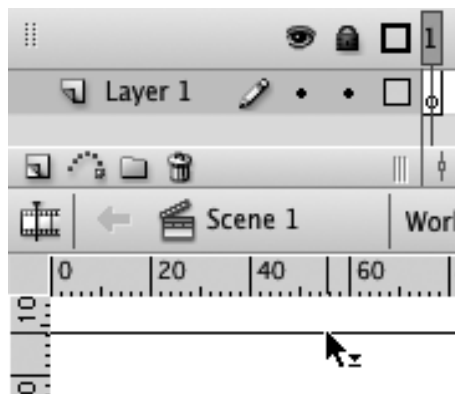
Чтобы переместить направляющую, наведите на нее курсор в режиме выбора. Рядом с курсором появится стрелочка, указывающая, в каком направлении тянуть направляющую (рис. 1.36). Перетащите направляющую в новое положение и отпустите кнопку мыши. Чтобы убрать направляющую, перетащите ее за пределы окна открытого документа.

**С**

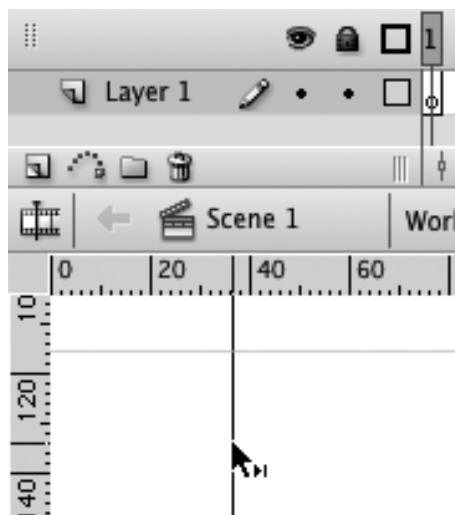
Чтобы избежать случайного перемещения направляющих, выполните команду **View ⇒ Guides ⇒ Lock Guides** (Просмотр ⇒ Направляющие ⇒ Заблокировать направляющие) или нажмите комбинацию клавиш **Option-точка с запятой (;)** (Mac) или **Ctrl-Alt-точка с запятой (;)** (Windows). Теперь при наведении инструмента «Стрелка» на направляющую стрелка-указатель не появляется. Чтобы разблокировать направляющие, снова выполните команду **View ⇒ Guides ⇒ Lock Guides** или нажмите ту же самую комбинацию клавиш.

**С**

Если вы вставили в документ много направляющих, то перетаскивать их за пределы арены по отдельности утомительно. Чтобы убрать все сразу, выполните команду **View ⇒ Guides ⇒ Clear Guides** (Просмотр ⇒ Направляющие ⇒ Очистить направляющие) или откройте диалоговое окно Guides и нажмите в нем кнопку **Clear All** (Очистить все).



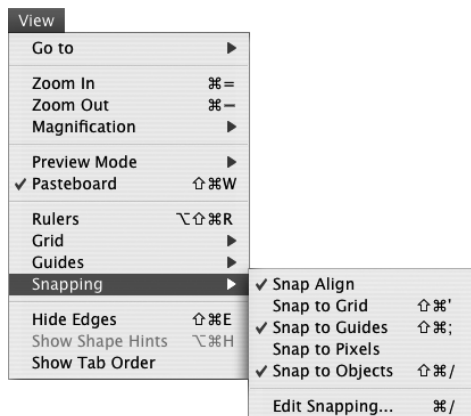
**Рис. 1.35.** При перетаскивании направляющей с линейки рядом с курсором появляется стрелка-указатель.



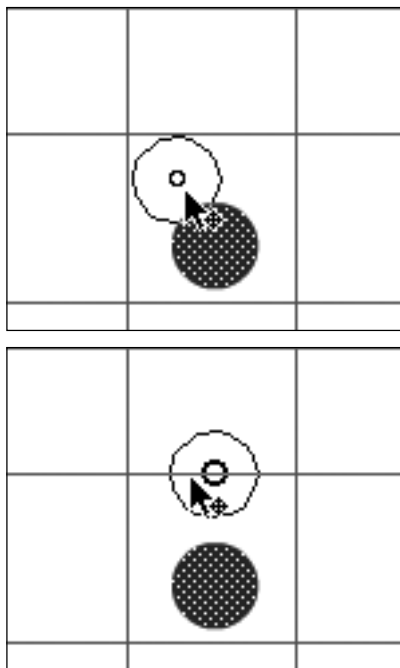
**Рис. 1.36.** Вертикальные и горизонтальные направляющие можно поместить в любое место на арене.

**С**

Чтобы задать параметры направляющих, откройте диалоговое окно Guides, выполнив команду **View ⇒ Guides ⇒ Edit Guides** (Просмотр ⇒ Направляющие ⇒ Изменить направляющие) или нажав комбинацию клавиш **Option-Shift-⌘-G** (Mac) или **Ctrl-Alt-Shift-G** (Windows).



**Рис. 1.37.** Выберите пункт View ⇒ Snapping, чтобы открыть меню режимов привязки. Щелчок по неотмеченному пункту активирует соответствующий режим, а щелчок по отмеченному пункту деактивирует.



**Рис. 1.38.** В процессе буксировки объекта рядом с кончиком курсора-стрелки появляется кружок — индикатор совмещения (верхний рисунок). При пересечении элемента, с которым может произойти совмещение, например линии сетки, направляющей или края либо центра другого объекта, этот кружок увеличивается.

## Привязка

Во Flash есть пять команд привязки, которые помогают точнее позиционировать элементы на арене. Команда **Snap to Grid** (Привязка к сетке) точно совмещает край или центр элемента с линией сетки. Команда **Snap to Guide** (Привязка к направляющим) делает то же самое, только с направляющими, а не сеткой. Команда **Snap to Object** (Привязка к объектам) позиционирует один элемент относительно другого. Команда **Snap to Pixels** (Привязка к пикселям) позволяет перемещать объекты на расстояние, кратное одному пикселю; при масштабе 400 процентов и выше в этом режиме отображается сетка 1x1 пиксель. Команда **Snap Align** (Выравнивание с привязкой) помогает выравнивать элементы, если в процессе перетаскивания они приблизились друг к другу или к краю арены на заданное пользователем расстояние.

### Чтобы включить или выключить режим привязки:

1. Выберите пункт меню **View ⇒ Snapping** (Просмотр ⇒ Привязка).

Режимы, отмеченные галочками, в данный момент активны (рис. 1.37). Если выбрать отмеченный пункт, соответствующий режим становится неактивным и наоборот.

2. Чтобы активировать/деактивировать режим привязки в подменю **Snapping**, выполните одно из следующих действий:

- Для привязки элементов к сетке выберите пункт **Snap to Grid** или нажмите комбинацию клавиш **Shift+⌘-апостроф (')** (Mac) либо **Ctrl+Shift-апостроф (')** (Windows).

Если режим привязки к сетке активен, то в процессе буксировки элемента рядом с кончиком курсора-стрелки появляется кружок, который называется индикатором совмещения. Когда элемент приближается к линии сетки, Flash выделяет точку, с которой может произойти совмещение, увеличивая радиус индикатора (рис. 1.38).

- Для привязки элементов к направляющим выберите пункт **Snap to Guides** или нажмите комбинацию клавиш **Shift-⌘-точка с запятой (;)** (Mac) либо **Ctrl-Shift-точка с запятой (;)** (Windows).
- Для привязки элементов к элементам выберите пункт **Snap to Objects** или нажмите комбинацию клавиш **Shift-⌘-/** (Mac) либо **Ctrl-Shift-/** (Windows).
- Для привязки элементов к пикселям выберите пункт **Snap to Pixels**.

Flash создает сетку с ячейками размером 1 x 1 пиксель. Чтобы эта сетка стала видна, масштаб арены должно быть не менее 400 процентов (рис. 1.39). Подробнее о масштабировании вы узнаете из раздела «Просмотр в разных масштабах» ниже.

- Чтобы в процессе позиционирования элементов видеть направляющие линии, выберите пункт **Snap Align**.

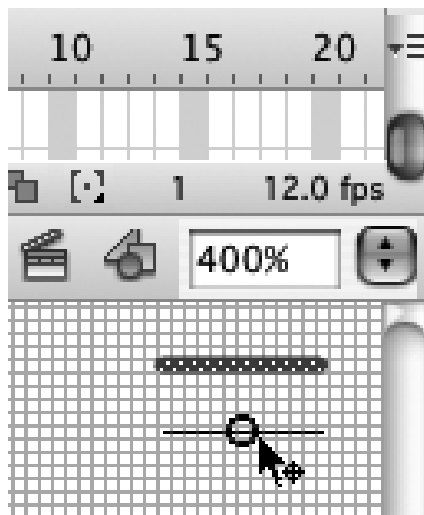
Если режим выравнивания с привязкой активен, то при буксировке элементов на арене отображаются пунктирные линии-направляющие в момент, когда какой-то край или центр элемента приближается к уровню края или центра какого-либо другого элемента или когда элемент оказывается в пределах заданного пользователем расстояния от края либо центра другого элемента либо края арены (рис. 1.40).

C

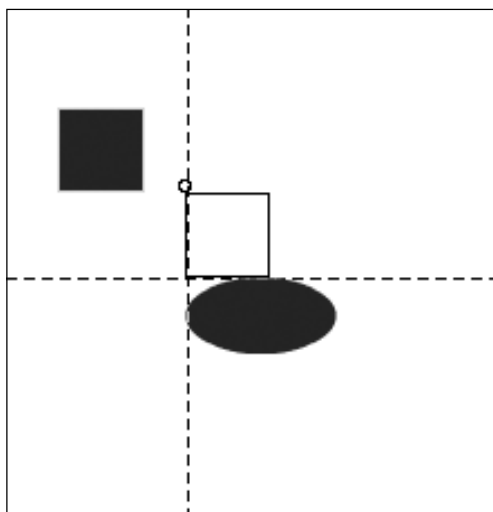
Отключить привязку к сетке и направляющим можно также и в тех диалоговых окнах, где вы включали соответствующий режим. Выберите пункт меню **View ⇒ Grid ⇒ Edit Grid** (или **View ⇒ Guides ⇒ Edit Guides**), и в появляющемся окне сбросьте флажок **Snap to Grids** (или **Snap to Guides**).

C

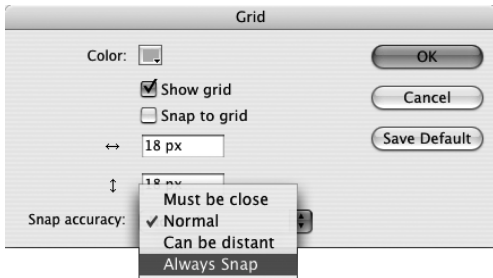
Когда вы одновременно работаете с большим числом инструментов рисования, в левой нижней части панели инструментов появляется значок магнита. Щелчок по нему активирует или деактивирует режим совмещения с объектами.



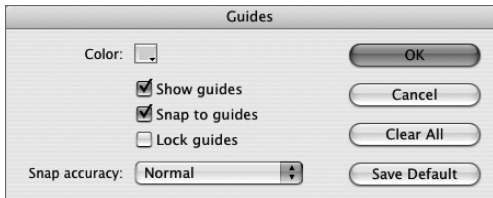
**Рис. 1.39.** Если активирован режим привязки к пикселям, то при масштабе 400 процентов и выше становится видна сетка с ячейками, соответствующими 1 пикселю. В этом режиме вы можете точно позиционировать графические элементы.



**Рис. 1.40.** В режиме выравнивания с привязкой Flash отображает направляющие пунктирные линии, когда буксируемый элемент отвечает определенным условиям. Например, такая линия появится, если какой-то край буксируемого объекта совмещается с краем другого объекта или оказывается достаточно близко к краю арены.



**Рис. 1.41.** Из списка Snap Accuracy выберите значение, определяющее, насколько близко элемент должен подойти к линии сетки, чтобы произошло совмещение. Значение Always Snap (Всегда) принудительно помещает край или центр элемента на линию сетки.



**Рис. 1.42.** В диалоговом окне Guides можно указать, насколько близко элемент должен подойти к направляющей, чтобы произошло совмещение с ней. Можно также выбрать цвет направляющих, задать их видимость и заблокировать или разблокировать их.

**С** Индикатор совмещения появляется примерно в том месте, где курсор-стрелка касается буксируемого элемента. Однако Flash может совмещать только с центром элемента или с какой-нибудь точкой на его контуре. Если индикатор совмещения не появляется, попробуйте подтянуть элемент ближе к центру, краю или к угловой точке.

**С** Хотя в режиме привязки к пикселям используется сетка с 1-пиксельными ячейками, перемещаемые графические элементы не совмещаются с узлами сетки, если на панели информации (Info) или на вкладке **Properties** (Свойства) в инспекторе свойств не выставлены целочисленные значения *x* и *y*. (О том, как задавать координаты графических элементов, вы узнаете в главе 4.)

### Чтобы задать параметры привязки к сетке:

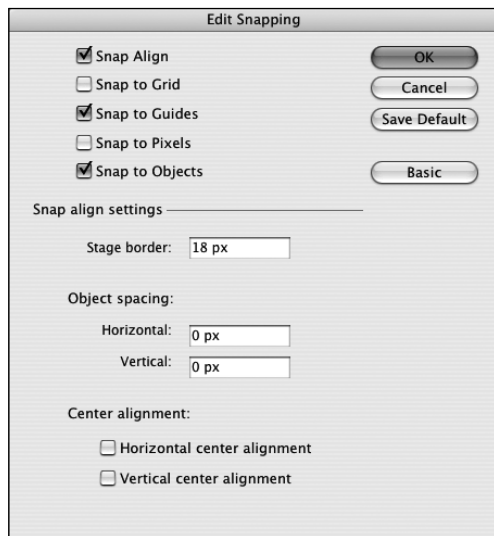
1. Выберите пункт меню **View** ⇒ **Grid** ⇒ **Edit Grid** или нажмите комбинацию клавиш **Option-⌘-G** (Mac) или **Ctrl-Alt-G** (Windows).
2. В диалоговом окне Grid выберите подходящее значение из списка **Snap Accuracy** (рис. 1.41).
3. Нажмите **OK**.

### Чтобы задать параметры привязки к направляющим:

4. Выберите пункт меню **View** ⇒ **Grid** ⇒ **Edit Guides** или нажмите комбинацию клавиш **Option-Shift-⌘-G** (Mac) или **Ctrl-Alt-Shift-G** (Windows).
5. В диалоговом окне Guides выберите подходящее значение из списка **Snap Accuracy** (рис. 1.42).
6. Нажмите **OK**.

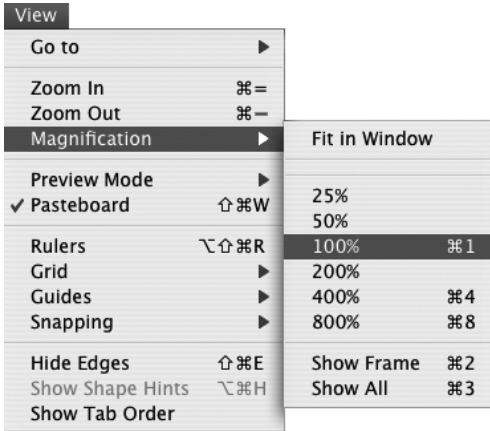
## Чтобы задать параметры выравнивания с привязкой:

1. Выполните команду **View ⇒ Snapping ⇒ Edit Snapping** (Просмотр ⇒ Привязка ⇒ Изменить привязку). Откроется диалоговое окно Edit Snapping. Если раздел Snap Align Settings (Параметры выравнивания с привязкой) не виден, нажмите кнопку **Advanced** (Дополнительно) (рис. 1.43).
2. Чтобы изменить способ взаимодействия с элементами и ареной, выполните одно из следующих действий:
  - Чтобы направляющие выравнивания появлялись, когда перетаскиваемый объект оказывается в пределах указанного расстояния от края арены, введите значение в поле Stage Border (Рамка рабочей области).
  - Чтобы направляющие выравнивания появлялись, когда верхний или нижний край перетаскиваемого объекта оказывается в пределах указанного расстояния от верхнего или нижнего края другого объекта, введите значение в поле Vertical (По вертикали) в группе Object Spacing (Интервал между объектами).
  - Чтобы направляющие выравнивания появлялись, когда одна из боковых сторон перетаскиваемого объекта оказывается в пределах указанного расстояния от боковой стороны другого объекта, введите значение в поле Horizontal (По горизонтали) в группе Object Spacing.
  - Чтобы направляющие выравнивания появлялись, когда центр перетаскиваемого объекта совмещается по горизонтали или по вертикали с центром другого объекта, отметьте флажок Horizontal или Vertical в группе Center Alignment (Выравнивание по центру).

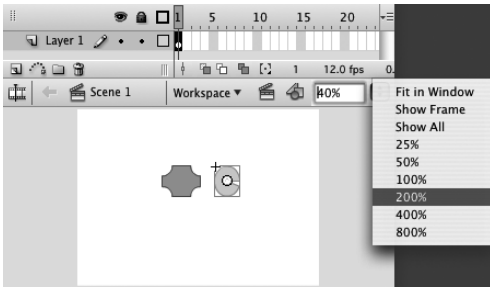


**Рис. 1.43.** Допуски выравнивания с привязкой задаются в разделе Advanced диалогового окна Edit Snapping. На рисунке показаны принятые по умолчанию значения.





**Рис. 1.44.** В масштабе 100% графические элементы имеют такой же размер, как в готовом ролике.



**Рис. 1.45.** Если значение масштаба больше 100, то размеры элементов на арене увеличиваются. В раскрывающийся список уже включены несколько наиболее употребительных масштабов.

## Просмотр в разных масштабах

Flash предусматривает несколько способов изменить масштаб элементов на арене.

### Чтобы просматривать элементы в естественном масштабе:

- Выберите пункт меню **View ⇒ Magnification ⇒ 100%** (Просмотр ⇒ Увеличение ⇒ 100%) (рис. 1.44) или нажмите комбинацию клавиш **⌘-1** (Mac) либо **Ctrl-1** (Windows).

или

- В поле управления масштабом, которое находится в правой части панели редактирования, введите значение 100%. Нажмите **Enter**.

В масштабе 100% Flash выбирает размер элементов, максимально приближенный к тому, который будет виден в готовом ролике. (Для некоторых мониторов и видеокарт элементы могут отображаться чуть иначе, чем в системе, на которой создавались.)

### Чтобы увеличить или уменьшить масштаб арены:

1. В поле управления масштабом введите степень увеличения в процентах (рис. 1.45).
2. Нажмите **Enter**.

**С**

Если щелкнуть по стрелочке справа от поля управления масштабом, выпадет такой же список, как в подменю **View ⇒ Magnification**. Можно сразу выбрать нужный масштаб из этого списка. Среди прочих, в нем имеются пункты **Fit in Window** (По размерам окна) (при этом арена отображается в текущем окне целиком без полос прокрутки), **Show Frame** (Показать кадр) (при этом арена отображается в текущем окне целиком с полосами прокрутки) и **Show All** (Показать все) (масштаб выбирается так, чтобы все элементы, находящиеся на арене и монтажном столе, были видны в текущем окне).

### Чтобы увеличить или уменьшить отдельные области:

1. На панели инструментов выберите инструмент Zoom (Масштаб) (или нажмите любую из клавиш **M** либо **Z**). Курсор примет форму лупы.
2. Чтобы увеличить масштаб некоторой области или элемента, нарисуйте с помощью буксировки мышью охватывающий прямоугольник. В результате все окно будет заполнено увеличенным содержимым выбранной области (рис. 1.46). Этот способ годится и для увеличения, и для уменьшения.

### Чтобы увеличить или уменьшить масштаб:

3. Выбрав инструмент «Масштаб», в нижней части панели инструментов выберите еще один из модификаторов:

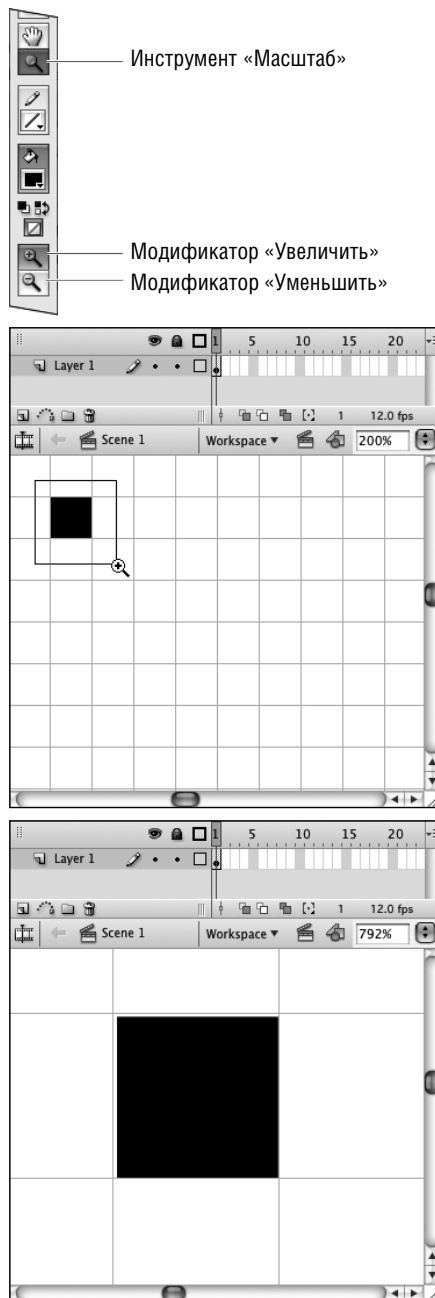
- увеличить (лупа со знаком плюс);
- уменьшить (лупа со знаком минус)

4. Щелкните по той области или элементу на арене, который хотите увеличить или уменьшить.

Та точка, по которой вы щелкнули, окажется в центре окна просмотра, при этом в поле управления масштабом будет представлен новый масштаб. Если инструмент «Масштаб» работает в режиме увеличения, то Flash удваивает текущий масштаб, а если в режиме уменьшения — сокращает вдвое.

**C**

Чтобы при работающем инструменте «Масштаб» временно переключиться из режима увеличения в режим уменьшения или наоборот, удерживайте нажатой клавишу **Option** (Mac) или **Alt** (Windows).



**Рис. 1.46.** С помощью инструмента «Масштаб» (верхний рисунок) нарисуйте прямоугольник, охватывающий нужный элемент (средний рисунок). Flash увеличит выбранную область так, чтобы она заполнила все текущее окно (нижний рисунок).

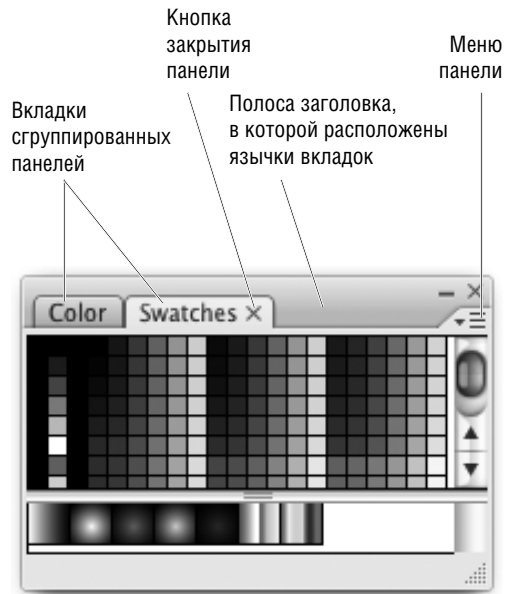
## Панели

Во Flash все инструменты разработки распределены по нескольким панелям, отображаемым *внутри* окон. Выше мы уже видели, как можно быстро открывать различные панели. На одних панелях находятся инструменты для создания и изменения графических элементов, на других — для организации Flash-документа и навигации по его компонентам. Можно сделать окно панели плавающим и расположить его в любом месте рабочего пространства; можно *сгруппировать* несколько панелей в одном окне или расположить окна панелей одно под другим; можно *пристыковать* панель к любому краю рабочего пространства. В последующих главах вы научитесь работать с конкретными панелями, а пока познакомимся с характеристиками, общими для всех панелей, и посмотрим, как организовать их в рабочем пространстве.

Открытые панели выглядят как вкладки внутри окна. Окно может содержать одну или несколько таких вкладок. В любом окне панелей есть следующие компоненты (рис. 1.47): полоса заголовка (серая область в верхней части окна сразу под списком вкладок), одна или несколько вкладок, соответствующих отдельным панелям, кнопка свернуть/восстановить и кнопка закрытия окна. (Если окна панелей расположены друг под другом, то последние два компонента присутствуют только в самом верхнем окне и управляют всей стопкой.) У каждой панели есть кнопка закрытия и меню.



Окно пристыкованной панели



Окно плавающей панели

**Рис. 1.47.** Общие для всех панелей элементы интерфейса позволяют организовать панели так, чтобы сэкономить место на экране. Кнопки, относящиеся ко всему окну панелей, находятся в полосе заголовка окна, а те, что относятся к конкретной панели, — на языке вкладки с именем этой панели.

### Чтобы открыть (или закрыть) окно панели с помощью меню:

- Выберите из меню **Window** (Окно) нужную панель, например, **Color** (Цвет) (рис. 1.48). Результат может быть таким:
- Если выбранная панель была закрыта, то открывается окно, содержащее эту панель. Окно может быть пристыкованным или плавающим в зависимости от конфигурации текущего рабочего пространства.
- Если окно выбранной панели было свернуто, оно восстанавливается.
- Если выбранная панель была перекрыта другими вкладками или каким-либо окном плавающей панели, то она перемещается на передний план.
- Если выбранная панель — единственная или текущая вкладка в уже открытом, не свернутом и находящемся на переднем плане окне, то это окно (вместе со всеми своими вкладками) закрывается.

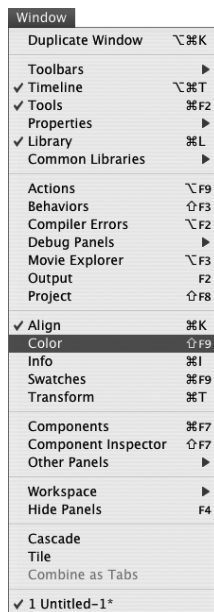


Рис. 1.48. В меню **Window** имеется список всех панелей Flash.

### Чтобы закрыть окно панели мышью:

- Нажмите кнопку закрытия в правом углу полосы заголовка (рис. 1.49).  
Окно панели закроется. Если окно содержит сгруппированные панели, то нажатие этой кнопки закрывает все вкладки. Если окно содержит вертикальную стопку панелей, то кнопка закрытия присутствует лишь в самом верхнем окне, и ее нажатие закрывает всю стопку.

Закреть окно

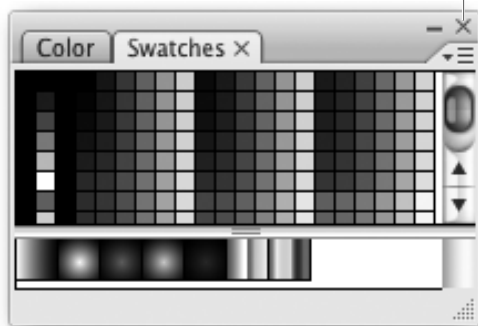
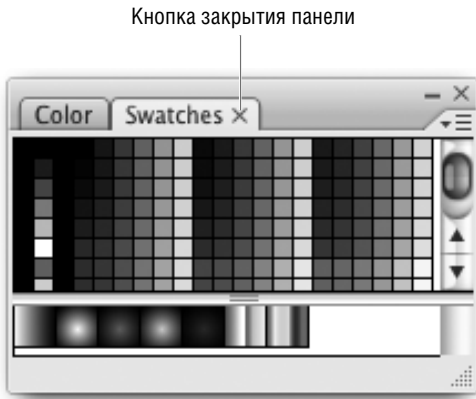


Рис. 1.49. Нажатие кнопки закрытия в полосе заголовка окна приводит к закрытию окна.

**С** Закрыть открытое окно панели можно также, щелкнув мышью с нажатой клавишей **Click** (Mac) или правой кнопкой мыши (Windows) по полосе заголовка и выбрав из контекстного меню пункт **Close Group(s)** (Закреть группу).

**С** Чтобы скрыть все открытые панели (включая панель инструментов, инспектор свойств и все открытые панели библиотек), нажмите клавишу **F4**. Повторное нажатие той же клавиши приводит к показу всех скрытых панелей.



**Рис. 1.50.** Нажатие кнопки закрытия на язычке вкладки панели приводит к закрытию только этой панели; остальные панели в той же группе или стопке остаются открытыми.

### Чтобы закрыть отдельную панель внутри группы или стопки:

- Нажмите кнопку закрытия на язычке вкладки панели (рис. 1.50). Панель закрывается, то есть пропадает из группы или стопки.

**С**

Когда закрывается панель, входящая в состав группы или стопки, она не перестает быть частью последней. В следующий раз, когда вы выберете эту панель из меню **Window**, она будет находиться в окне, содержащем ту же группу или стопку.

### Чтобы свернуть или восстановить окно панели:

- Нажмите кнопку сворачивания (или восстановления) (рис. 1.51).

**С**

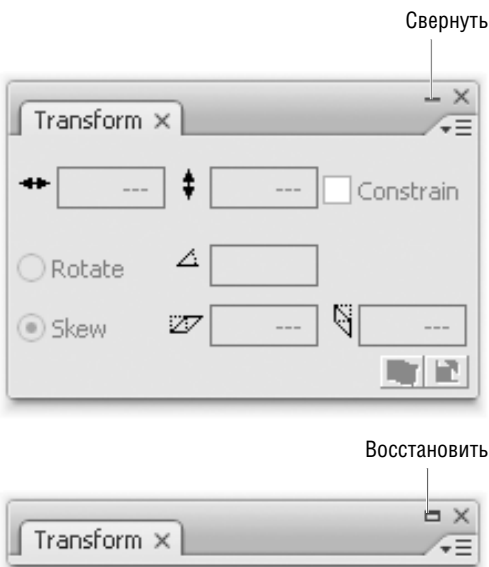
Если нажать кнопку сворачивания в окне стопки, то Flash свернет все панели в этой стопке. В результате в стопке останутся только полосы заголовков.

**С**

Свернуть или восстановить открытое окно панели можно также, щелкнув мышью с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows) по полосе заголовка и выбрав из контекстного меню пункт **Minimize Group(s)/Restore Group(s)** (Свернуть группу / Восстановить группу).

**С**

Для переключения между свернутым и восстановленным состоянием можно также один раз щелкнуть мышью в полосе заголовка группы панелей или дважды щелкнуть по язычку вкладки панели.



**Рис. 1.51.** В развернутом окне панели присутствует кнопка сворачивания — короткая горизонтальная линия; при нажатии на нее окно сворачивается в полосу заголовка. Если окно свернуто, то в нем присутствует кнопка восстановления — небольшой прямоугольник; при нажатии на нее окно восстанавливается.

**Чтобы изменить положение окна панели:**

- Щелкните по полосе заголовка и перетащите окно в другое место (рис. 1.52). В процессе перетаскивания окно панели становится полупрозрачным, чтобы было видно, куда его поместить.

**С**

Стыковка, группировка и складывание панелей в стопку — это лишь разные способы изменения положения панелей в рабочем пространстве (см. раздел «Объединение и стыковка панелей» ниже в этой главе).

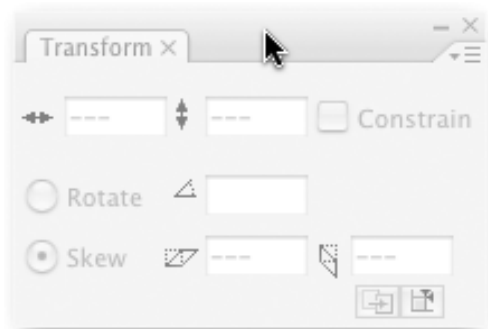
**Чтобы изменить размер плавающего окна панели:**

Выполните одно из следующих действий:

- Чтобы изменить ширину или высоту окна панели, потяните за манипулятор, расположенный в правом нижнем углу окна (рис. 1.53).
- Чтобы изменить только высоту окна, наведите курсор на верхний или нижний край панели. Когда курсор примет форму двунаправленной стрелки, потяните окно, — будет изменяться вертикальный размер.
- Чтобы изменить только ширину окна, наведите курсор на левый или правый край панели. Когда курсор примет форму двунаправленной стрелки, потяните окно, — будет изменяться горизонтальный размер.

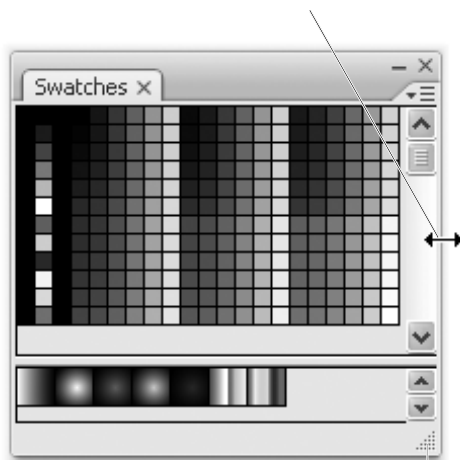
**С**

Для некоторых панелей изменять размер запрещено. Если в окне панели нет значка изменения размера, значит, ее размер фиксирован. Если в одном окне сгруппированы панели с изменяемым и фиксированным размером, то значок изменения размера появляется лишь тогда, когда активная изменяемая панель.



**Рис. 1.52.** Потяните окно панели за полосу заголовка, чтобы поместить ее в другое место в рабочем пространстве. В процессе перетаскивания панель становится полупрозрачной.

Курсор готов к изменению ширины



Манипулятор изменения размера

**Рис. 1.53.** В правом нижнем углу окна, содержащего панель изменяемого размера, имеется составленный из точек манипулятор; за него можно потянуть точно так же, как при изменении размера окна документа. Можно также менять размер в каком-то одном направлении, для чего следует навести курсор на любую сторону окна и, когда он примет форму двунаправленной стрелки, потянуть за эту сторону.

## Объединение и стыковка панелей

Для более рационального использования места на экране панели можно объединять в группы (несколько вкладок в одном окне) или стопки (несколько окон панелей, расположенных друг под другом). Отдельные панели, группы и стопки можно пристыковывать к любому краю окна. Чтобы сэкономить еще больше места, пристыкованные панели можно свернуть в значки и восстанавливать по мере необходимости (см. ниже врезку «Экономия места с помощью сворачивания панелей»).

### Чтобы сгруппировать панели в виде вкладок в одном окне:

1. Открыв одну или несколько панелей, щелкните по язычку какой-нибудь из них.
2. Перетащите эту панель на любую другую открытую панель — конечную. Перетаскиваемая панель становится полупрозрачной (рис. 1.54). Когда курсор попадает в те или иные части конечной панели, Flash подсвечивает *зоны бросания* голубым цветом, показывая, как именно панели будут объединены.
3. Подведите курсор к полосе заголовка конечной панели. Flash подсветит границы и полосу заголовка конечной панели; это означает, что при бросании панели будут объединены в группу.
4. Отпустите кнопку мыши. Панель, которую вы перетаскивали, теперь представлена вкладкой в окне конечной панели.



**Рис. 1.54.** Когда вы перетаскиваете панель на полосу заголовка другого окна панелей (верхний рисунок), границы конечного окна подсвечиваются (средний рисунок). Если в этот момент отпустить кнопку мыши, то будет создана группа панелей, представленных вкладками в конечном окне (нижний рисунок).

### Чтобы расположить однопанельные окна друг под другом:

1. Открыв одно или несколько однопанельных окон, щелкните по полосе заголовка какой-нибудь панели. Соответствующая вкладка станет серой.
2. Чтобы поместить первую панель над второй, не отпуская кнопку мыши, тащите вкладку, пока курсор не совместится с верхним краем конечного окна панели.

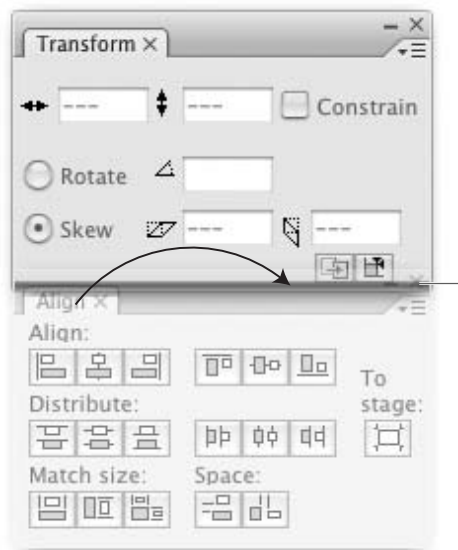
*или*

Чтобы поместить первую панель под второй, подведите курсор к нижнему краю конечного окна панели.

Flash отмечает зону бросания в конечном окне голубой линией (рис. 1.55).

3. Отпустите кнопку мыши.

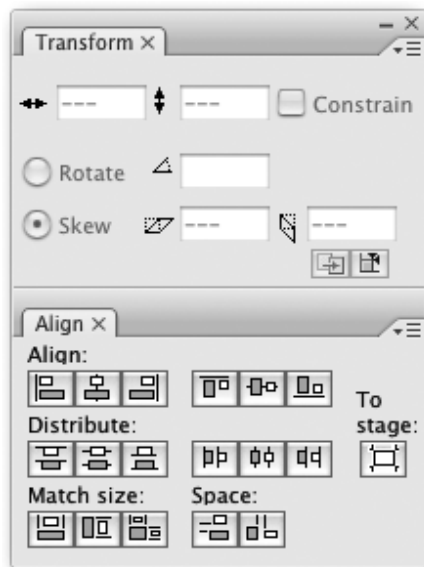
Теперь первая панель оказывается над или под целевым окном.



Зона бросания в зоне стыковки

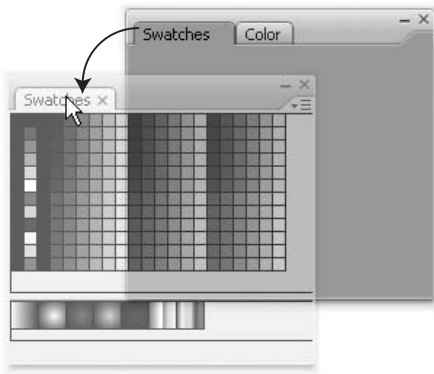
**С**

Можно также организовать стопку из окон, содержащих группы панелей. Вместо того, чтобы перетаскивать отдельную вкладку, тащите полосу заголовка всего окна. Но при перетаскивании полосы заголовка число зон бросания меньше. Зона бросания в верхней части конечного окна не активируется. Аналогично, если вы перетаскиваете полосу заголовка в уже существующую стопку, то не активируется зона бросания над верхним окном.

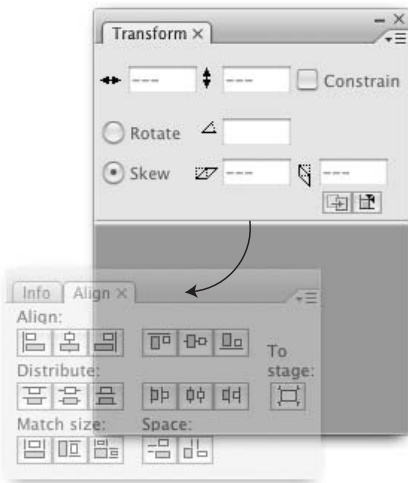


**Рис. 1.55.** Чтобы создать стопку, перетащите одну панель на верхний или нижний край другой. Если отпустить кнопку мыши в зоне бросания у нижнего края конечного окна панели (верхний рисунок), то создается стопка с конечным окном сверху (нижний рисунок).





**Рис. 1.56.** Потянув панель за язычок вкладки, вы сможете исключить ее из группы.



**Рис. 1.57.** Чтобы исключить окно панели из стопки, достаточно перетащить его. Чтобы исключить одну панель, потяните за язычок, а чтобы исключить группу панелей — за полосу заголовка.

### Чтобы разделить сгруппированные панели:

1. Чтобы исключить панель из группы, щелкните по язычку соответствующей ей вкладки. На платформе Мас вкладка подсвечивается серым цветом.
2. Перетащите вкладку за пределы окна панели. Перетаскиваемая панель становится полупрозрачной, а ее место в исходном окне перекрашивается в серый цвет (рис. 1.56).
3. Отпустите кнопку мыши.

Панель оказывается в отдельном окне.

### Чтобы разделить стопку панелей:

- Чтобы вынести однопанельное окно из стопки, щелкните по ее язычку и перетащите за пределы стопки.

или

1. Чтобы вынести группу панелей из стопки, щелкните по полосе заголовка окна, содержащего эту группу. Перетаскиваемое окно станет полупрозрачным, а его место в исходном окне перекрашивается в серый цвет (рис. 1.57).
2. Отпустите кнопку мыши.

Группа панелей оказывается в отдельном окне.

**С**

Панель, которая выносится из стопки в отдельное окно путем перетаскивания язычка, может занимать любое место в стопке. Однако при перетаскивании полосы заголовка есть ограничения. Потянув за полосу заголовка верхнего окна в стопке, вы переместите всю стопку. Единственный способ вынести из стопки групповое окно состоит в том, чтобы предварительно разобрать группу, перетаскивая все вложенные панели по одной, пока в верхнем окне не останется всего одна панель<sup>1</sup>.

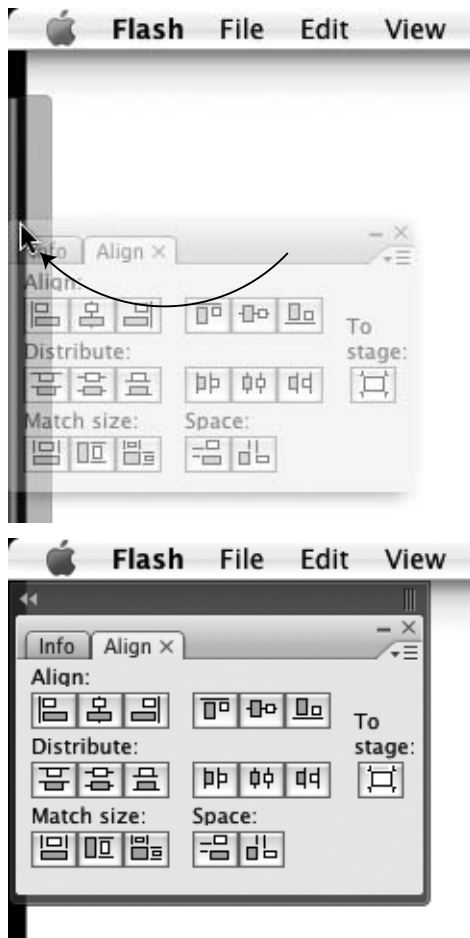
<sup>1</sup> На платформе Windows такого ограничения нет (Прим. перев.).

**Чтобы создать новую зону стыковки:**

1. Щелкните по полосе заголовка плавающего окна панелей. Это окно может содержать как единственную панель, так и группу или стопку панелей.
2. Перетащите окно панелей к любому краю окна приложения. Появится серый прямоугольник, по краю которого расположена голубая линия (рис. 1.58). Этот прямоугольник является границей зоны стыковки, а голубая линия — это зона бросания. Если зона стыковки ничем не занята, она остается скрытой, пока в ней не окажется перетаскиваемая панель.
3. Отпустите кнопку мыши.

**С**

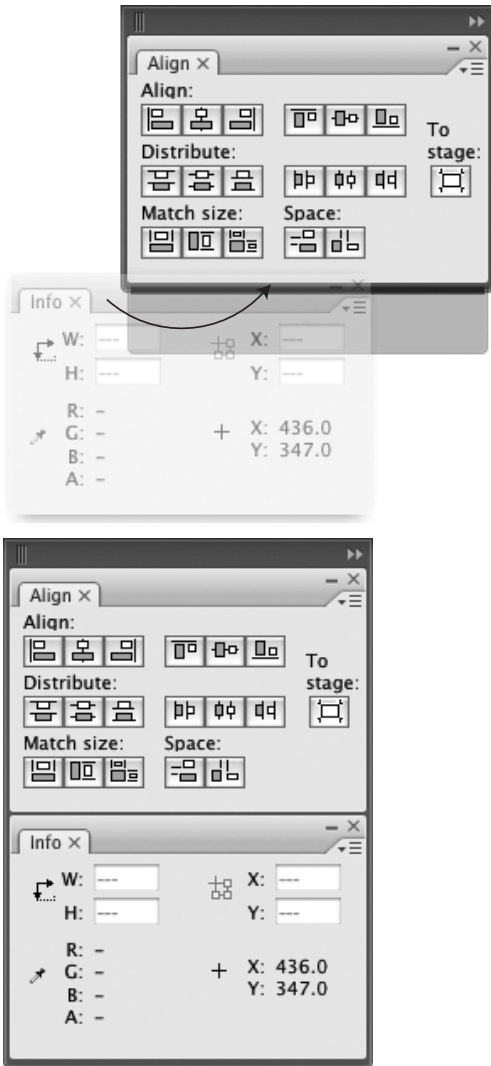
Размеры почти всех зон стыковки можно изменять. Подведите курсор к краю зоны стыковки, который ближе всего к арене. Когда курсор примет форму двунаправленной стрелки, потяните за край, чтобы увеличить или уменьшить ширину зоны стыковки. Но размер некоторых пристыкованных панелей, например панели инструментов, изменять нельзя. В таком случае форма курсора не меняется при наведении на край пристыкованной панели.



**Рис. 1.58.** Перетаскивание окна панелей к краю окна приложения активирует зону стыковки. Когда появляется голубая линия, обозначающая зону бросания (верхний рисунок), отпустите кнопку мыши; Flash поместит окно панелей в зону стыковки (нижний рисунок).

### Панели инструментов и справки отличаются от всех остальных

На *панели инструментов* находятся инструменты рисования Flash. В отличие от других панелей, ее нельзя ни свернуть, ни изменить размер, ни сгруппировать с другими панелями. Инструменты можно расположить в одну или две колонки, а саму панель пристыковать к левому или правому краю, но не к верхнему и не к нижнему. Подробнее о работе с панелью инструментов пойдет речь в главе 2. *Панель справки* похожа на все остальные панели Flash, но вы не найдете ее в меню **Window**. Чтобы получить к ней доступ, выберите пункт меню **Help** ⇒ **Flash Help** (Справка ⇒ Справка Flash) или нажмите клавишу **F1**.



**Рис. 1.59.** При перетаскивании новой панели к верхнему или нижнему краю уже пристыкованного окна панелей, зона бросания окрашивается в голубой цвет. Если отпустить кнопку мыши в момент, когда появилась нужная вам зона бросания (верхний рисунок), то Flash добавит окно панели в зону стыковки (нижний рисунок).

### Чтобы добавить панель в существующую зону стыковки:

1. Щелкните по полосе заголовка плавающего окна панелей. Это окно может содержать как единственную панель, так и группу или стопку панелей.
2. Перетащите окно панелей в существующую зону стыковки. Перетаскиваемое окно становится полупрозрачным. Когда указатель оказывается в различных областях зоны стыковки, появляется голубая линия, обозначающая зону бросания.
3. Чтобы добавить окно панелей в зону стыковки, выполните следующие действия:

- Чтобы новое окно оказалось над уже пристыкованным, наведите курсор на верхний край пристыкованного окна панелей. Чтобы новое окно оказалось под уже пристыкованным, наведите курсор на нижний край пристыкованного окна панелей (рис. 1.59).

- Отпустите кнопку мыши.

Flash создаст отдельное окно для панелей внутри уже существующей зоны стыковки.

*или*

Чтобы создать новую группу панелей в зоне стыковки (или добавить панель к существующей группе), выполните следующие действия:

- Наведите курсор на полосу заголовка пристыкованного окна панелей; весь периметр окна окрасится в голубой цвет, обозначив зону бросания.

- Отпустите кнопку мыши.

Flash добавит вкладку панели в конечное окно внутри зоны стыковки (рис. 1.60).

C

Стопку панелей можно перетащить в зону стыковки, но при этом окна панелей теряют связь со стопкой. Отстыковать их можно только по отдельности.

C

Можно создавать удвоенные зоны стыковки. Если окно панелей перетащить к краю существующей зоны стыковки, то вдоль этого края активируется зона бросания. Если в этот момент отпустить кнопку мыши, то рядом с существующей зоной стыковки появляется еще одна (рис. 1.61).

### Чтобы отстыковать панель:

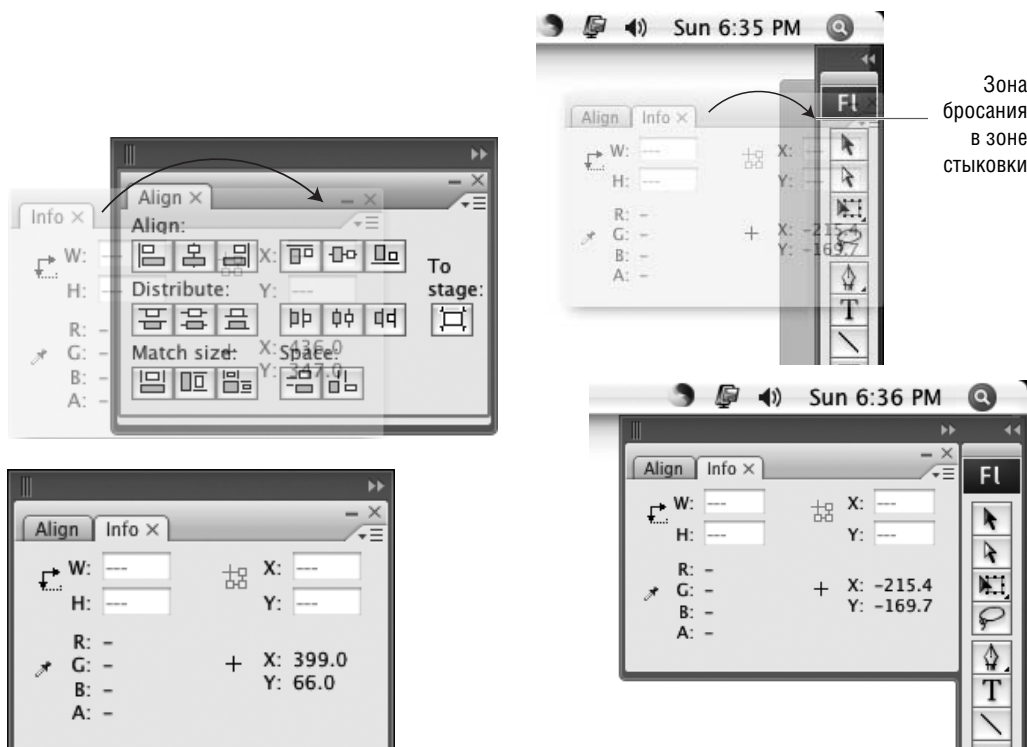
1. Чтобы убрать окно панелей из зоны стыковки, выполните одно из следующих действий:

- Если окно однопанельное, щелкните по язычку панели и перетащите ее за пределы зоны стыковки.
- Если окно содержит группу панелей, щелкните по полосе заголовка и перетащите окно за пределы зоны стыковки.

Тащить надо до тех пор, пока область, ранее занятая окном, не перекрасится в серый цвет.

2. Отпустите кнопку мыши.

Окно панелей станет плавающим.

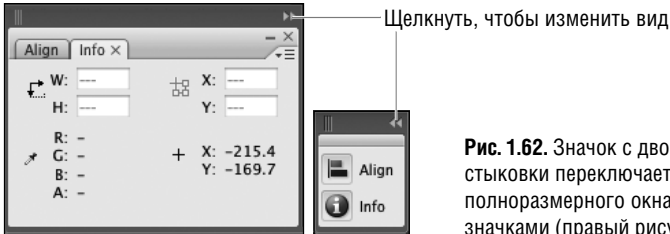


**Рис. 1.60.** При перетаскивании окна панели на полосу заголовка в зоне стыковки подсвечивается зона бросания для создания группы (верхний рисунок). Если в этот момент отпустить кнопку мыши, то в окно будет добавлена новая вкладка.

**Рис. 1.61.** Иногда бывает полезна удвоенная зона стыковки. Тащите окно панели к краю существующей зоны стыковки, пока не появится зона бросания (верхний рисунок). Если в этот момент отпустить кнопку мыши, рядом добавится еще одна зона стыковки (нижний рисунок).

### Экономия места с помощью сворачивания панелей

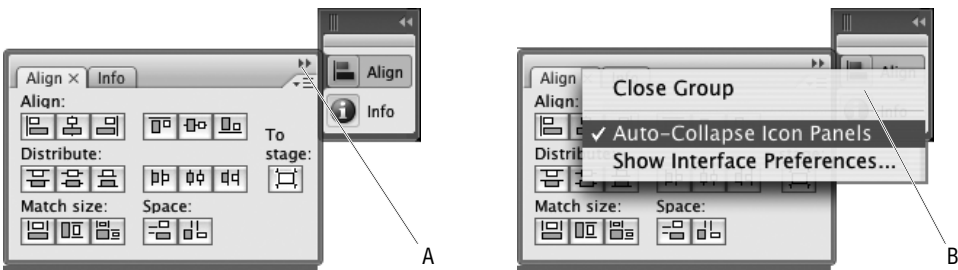
Чтобы максимально расчистить место для работы, можно свернуть пристыкованные окна панелей по левую или правую сторону от рабочего пространства в небольшие значки. (Панели, пристыкованные к нижнему и верхнему краю, в значки не сворачиваются, но можно оставить от них только полосы заголовков.) Для переключения между полным и свернутым видом панели, щелкните по двойному треугольничку в верхнем углу зоны стыковки (рис. 1.62). Если в зоне стыковки отображаются значки, то при щелчке по значку полностью раскрывается соответствующее окно панелей; если окно содержит сгруппированные панели, то раскрывается вся группа.



**Рис. 1.62.** Значок с двойным треугольничком в зоне стыковки переключает вид между отображением полноразмерного окна панелей (левый рисунок) и значками (правый рисунок).

Существует два варианта сворачивания окна панелей в значок. По умолчанию Flash автоматически сворачивает окно панелей, когда вы им не пользуетесь; достаточно просто щелкнуть в любой точке вне открытого окна. Для отмены режима автоматического сворачивания, нужно задать соответствующий параметр. Чтобы сделать это быстро, щелкните мышью, удерживая клавишу **Control** (Mac), или правой кнопкой мыши (Windows) по любому значку в области стыковки и сбросьте флажок **Auto-Collapse Icon Panels** (Автоматически сворачивать панели со значками) в контекстном меню. Попутно изменится одноименный параметр в категории **General** в диалоговом окне Preferences. (Чтобы самостоятельно открыть это окно, выберите пункт **Show Interface Preferences** (Показать настройки интерфейса) из того же контекстного меню или из меню **Flash** (Mac) либо **Edit** (Windows) ⇨ **Preferences**.) Если режим автоматического сворачивания выключен, то для закрытия любого открытого окна панели придется щелкнуть по двойному треугольничку в его левом верхнем углу (рис. 1.63).

У свернутых окон панелей нет кнопки закрытия. Чтобы убрать свернутую панель из рабочего пространства, придется сначала раскрыть ее, а уже потом нажать кнопку закрытия. Можно также перетащить значок из зоны стыковки, тогда окно панелей раскроется и можно будет нажать кнопку закрытия.



**Рис. 1.63.** Если окна панелей свернуты в значки и режим автоматического сворачивания выключен, то для закрытия открытого окна панелей нужно щелкнуть по двойному треугольничку (А). Чтобы быстро изменить настройки автосворачивания, щелкните мышью, удерживая клавишу **Control** (Mac), или правой кнопкой мыши (Windows) по любому значку, чтобы появилось контекстное меню (В).

## Об инспекторе свойств

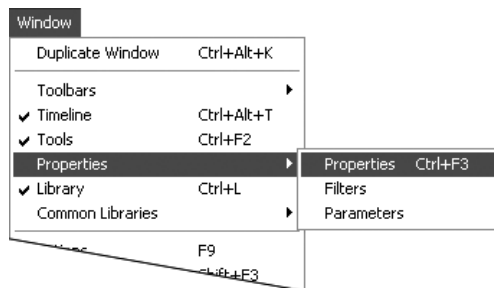
По умолчанию Flash CS3 создает группу панелей, известную под названием «инспектор свойств». В ней есть три вкладки. На вкладке **Properties** (Свойства) отображается информация о свойствах и атрибутах инструментов и графических элементов (например, цвет, стиль и шрифт в случае инструмента «Текст»). На вкладке **Parameters** (Параметры) отображается информация о компонентах (об одном типе компонентов — кнопке — вы узнаете в главе 12). Вкладка **Filters** (Фильтры) позволяет добавлять специальные эффекты к тексту и некоторым символам (символам посвящена глава 7). Описанные выше манипуляции позволяют изменить стандартную группировку панелей в окне инспектора свойств. Но пока не делайте этого, хотя в этой книге мы будем работать главным образом с панелью **Properties**.

Вкладка **Properties** в инспекторе свойств является контекстно-зависимой, на ней отражается состояние инструмента или графического элемента, выбранного в настоящий момент. О том, что может показываться на этой и других вкладках в окне инспектора свойств, вы узнаете подробнее в последующих главах. А пока рассмотрим некоторые общие операции.

### Чтобы обратиться к вкладке Properties в окне инспектора свойств:

- Выберите из меню пункт **Window ⇒ Properties ⇒ Properties** (Окно ⇒ Свойства ⇒ Свойства) или нажмите комбинацию клавиш **⌘-F3** (Mac) либо **Ctrl-F3** (Windows) (рис. 1.64).

Flash откроет инспектор свойств на вкладке **Properties**. На панели отображается информация о том объекте документа Flash, который выбран в настоящий момент (см. врезку «Возможности инспектора свойств»).



**Рис. 1.64.** Для доступа к вкладке **Properties** инспектора свойств выберите из меню пункт **Window ⇒ Properties**, а в подменю укажите ту вкладку, которая должна стать активной.

## C

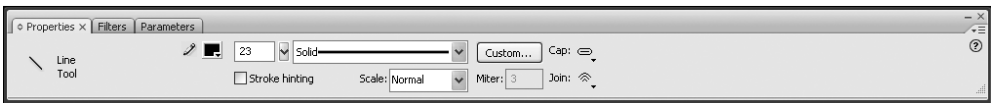
В зависимости от выбранного инструмента или элемента на вкладке **Properties** отображается разная информация. Так, у инструмента «Линия» меньше свойств, чем у инструмента «Текст». Можно уменьшить высоту инспектора свойств, когда число свойств на вкладке невелико. А при смене инструмента наоборот увеличить, чтобы видеть всю информацию.

### Возможности инспектора свойств

Можно считать, что инспектор свойств — это контекстно-зависимая суперпанель, то есть панель, которая изменяет свой вид в зависимости от того, что выбрано в данный момент. На вкладке **Properties** отображается информация об активном документе Flash либо о выбранном инструменте, графическом элементе (комбинированная форма, объект-рисунок, примитивная форма, сгруппированная фигура, символ, текстовое поле, растровое изображение или видеоролик) или кадре. На вкладке **Parameters** отображается информация о выбранном компоненте, а на вкладке **Filters** — о специальных эффектах, примененных к тексту, ролику или кнопке.

Вкладка **Properties** — это место, где задаются различные параметры инструментов и изменяются атрибуты выбранных элементов.

Если выбран, к примеру, инструмент «Линия», то на вкладке **Properties** будут показаны свойства этого инструмента (рис. 1.65): цвет, толщина и стиль. Но стоит выбрать линию, нарисованную на арене, как на вкладке **Properties** появятся свойства формы. Если выбранной формой является линия, то показываются примерно такие же атрибуты, как на вкладке свойств линии. Если изменить значения параметров на вкладке **Properties**, то внешний вид линии изменится соответствующим образом.



**Рис. 1.65.** Инспектор свойств отображает информацию о выбранных объектах и позволяет ее изменять. Например, если выбран инструмент «Линия», то с помощью инспектора свойств можно будет изменить цвет, толщину и стиль линий, рисуемых этим инструментом.

Щелкните по незанятой области на арене, и вы увидите на вкладке **Properties** свойства документа и сможете изменить их. А если выбрать экземпляр символа, присутствующего на арене, то на вкладке **Properties** будет информация о наследовании этого экземпляра (от какого главного символа он произошел), а также ширина, высота и положение на арене. Как только вы измените эти параметры в инспекторе свойств, изменения тут же будут внесены в сам выбранный символ.

Во многих режимах инспектор свойств требует ввести в поле какое-то значение, чтобы изменить параметр. Вы всегда можете ввести новое значение. Обычно при модификации выбранных объектов необходимо нажать клавишу **Enter**, чтобы изменения вступили в силу.

Небольшой треугольник справа от поля ввода говорит о наличии всплывающего ползунка, облегчающего ввод значения. Часто ползунок позволяет предварительно посмотреть, каков будет результат. К большинству ползунков применимы следующие методы:

- Щелкнуть и потянуть. Щелкните мышью по треугольничку и не отпускайте кнопку; можно начинать перемещение ползунка. Как только вы отпустите кнопку, Flash поместит текущее значение ползунка в поле и в большинстве случаев автоматически применит его к выбранным элементам.
- Щелкнуть и щелкнуть. Щелкните мышью по треугольничку и сразу же отпустите кнопку мыши; ползунок появится и останется на экране. Теперь можно перемещать ползунок или щелкать в разных его точках для выбора нового значения. Flash поместит новое значение в поле. Чтобы применить его к выбранным элементам, нужно щелкнуть где-то вне ползунка.





# СОЗДАНИЕ ПРОСТЫХ ГРАФИЧЕСКИХ ЭЛЕМЕНТОВ

## 2

### Об обводках и заливках

Что такое *обводка* и *заливка*? Обводка — это контур, а заливка — область, закрашенная одним цветом. Представьте себе книжку-раскраску, где картинки нарисованы черными линиями; вот эти линии и называются обводками. Если закрасить цветным карандашом область, ограниченную обводками, получится заливка. В книжке-раскраске контур уже есть и требуется создать внутри него заливку. Во Flash можно начать с другого конца — создать закрашенную форму, а затем контур как отдельный элемент.

Инструменты «Прямоугольник», «Овал», «Прямоугольный примитив», «Овальный примитив» и «Многоугольник» позволяют создавать как обводки, так и заливки. Помимо этого, с их помощью можно создать элемент, являющийся обводкой и заливкой одновременно. Инструмент «Линия», как нетрудно догадаться, создает только обводки. Инструмент «Перо» создает и обводки, и заливки.

В применении к инструменту «Кисть» идея обводок и заливок не так очевидна. Этот инструмент создает заливки. Заливки могут выглядеть как линии или мазки, но по сути являются формами, которые можно обвести. Во Flash имеются специальные инструменты для добавления, редактирования и удаления обводок и заливок: «Чернильница», «Ведро с краской» и «Ластик с краном». Дополнительную информацию см. в главе 4.

В этой главе вы научитесь пользоваться инструментами рисования, имеющимися в программе Adobe Flash CS3 Professional, для создания простых фигур, составленных из линий и закрашенных областей. В терминологии Flash они называются соответственно *обводками* и *заливками*. Flash предлагает естественные инструменты рисования, имитирующие работу с листом бумаги и карандашом или кистью; инструменты, соответствующие геометрическим фигурам, и инструмент «Перо» для точного рисования с помощью кривых Безье.

В совокупности эти инструменты позволяют создавать формы трех видов. Инструменты, работающие в режиме рисования комбинированных форм (Merge Drawing), создают простые формы, которые могут взаимодействовать с другими простыми формами в том же слое (о взаимодействии форм рассказывается в главе 5). Инструменты, работающие в режиме рисования объектов (Object Drawing), создают редактируемые формы, которые не взаимодействуют с другими формами. В версии Flash CS3 появился новый тип форм — *примитивные*. Два новых инструмента — прямоугольный примитив и овальный примитив — создают неинтерактивные формы, описываемые рядом параметров. Параметры примитивной формы (например, степень скругления углов прямоугольника) можно редактировать, но произвольным образом менять ее контур (скажем, превратить овал в бесформенную кляксу) запрещено.

Созданную форму можно в дальнейшем изменять (см. главу 4). Можно также импортировать графику из других программ (см. главу 14).

## Обзор инструментов

На панели инструментов находятся все инструменты, необходимые для создания и модификации графических элементов (рис. 2.1). Во Flash CS3 имеются инструменты для создания графики, прокрутки арены, изменения масштаба и задания цветов создаваемых элементов.

Чтобы выбрать инструмент, достаточно щелкнуть по нему. Если у выбранного инструмента есть дополнительные модификаторы, то в нижней части панели инструментов появятся соответствующие кнопки и/или меню для их конфигурирования.

**C**

По умолчанию инструменты расположены на панели в одну колонку. На рис. 2.1 показана панель с двумя колонками — для экономии места. Для переключения между этими видами щелкните по двойному треугольнику в левом верхнем углу панели (рис. 2.2).

Переключение между видами



**Рис. 2.2.** Для переключения между видами с одной и двумя колонками щелкните по двойному треугольнику в левом верхнем углу панели.



**Рис. 2.1.** На панели инструментов находятся инструменты для рисования, редактирования и манипулирования графическими элементами. Щелкните по значку текущего инструмента преобразования, чтобы увидеть инструменты «Свободное преобразование» и «Преобразование градиента». Щелкните по значку текущего инструмента Безье, чтобы увидеть инструмент «Перо» и инструменты редактирования пути. Щелкните по значку текущего инструмента-формы и выберите один из инструментов «Прямоугольник», «Овал», «Прямоугольный примитив», «Овальный примитив» или «Многоугольник».



**Рис. 2.3.** При выборе инструмента в нижней части панели появляются относящиеся к нему модификаторы. На этом рисунке показаны модификаторы кисти. Нажмите на кнопку (например, «Рисование объектов»), чтобы изменить режим на противоположный. Или выберите пункт из выпадающего меню, например, «Режим кисти».

### Чтобы получить доступ к инструментам и модификаторам:

1. Открыв панель инструментов, щелкните по какому-нибудь инструменту, например, «Кисть».  
В нижней части панели появляются модификаторы, относящиеся к выбранному инструменту.
2. Щелкните по значку или выберите из меню команду, чтобы модифицировать способ работы инструмента (рис. 2.3).

**С**

По умолчанию для инструментов включены всплывающие подсказки (если задержать курсор над инструментом, то появится его краткое описание). Настройки подсказок можно изменить в диалоговом окне Preferences: выберите категорию **General** (Общие), а в ней отметьте (или сбросьте) флажок **Show Tooltips** (Показывать подсказки) и нажмите **OK**, чтобы закрыть окно. (Подробнее о том, как открывать окно Preferences, см. в главе 1.)

**С**

Помимо названий инструментов, во всплывающей подсказке приведены еще комбинации клавиш для быстрого вызова. Когда вы ближе познакомитесь с инструментами, то станете активировать их клавишами — это значительно ускорит работу.

**С**

Чтобы закрыть панель инструментов (или, наоборот, открыть, если она уже закрыта), выберите из меню команду **Window ⇨ Tools** (Окно ⇨ Инструменты).

**С**

Если инструменты расположены в одну колонку, то панель очень узкая, поэтому к ней вполне можно пристыковать еще одну. Для этого перетащите какую-нибудь другую панель к правому краю панели инструментов (подробнее о стыковке см. главу 1).

## Сплошные цвета и градиенты

Цвет заливки и обводки позволяют задать большинство элементов управления цветом (см. врезку «Элементы управления цветом»), но наиболее широкие возможности все же предоставляет панель Color (Цвет). Цвет можно выбрать визуально — щелчком по графическому представлению пространства цветов, в котором есть все цвета, доступные в данной системе, или введя числовой код цвета. Можно также задать коэффициент прозрачности цвета. А чтобы определить градиент цвета, без панели цветов просто не обойтись.

Прежде чем определять цвет или градиент, необходимо указать, применяется он к заливке или к обводке. Для этого нужно воспользоваться одним из двух элементов: **Fill Color** (Цвет заливки) или **Stroke Color** (Цвет обводки). После определения нового цвета Flash обновляет все связанные элементы управления цветом. Например, если определяется новый цвет заливки, то этот цвет становится подразумеваемым по умолчанию во всех инструментах, имеющих дело с заливками.

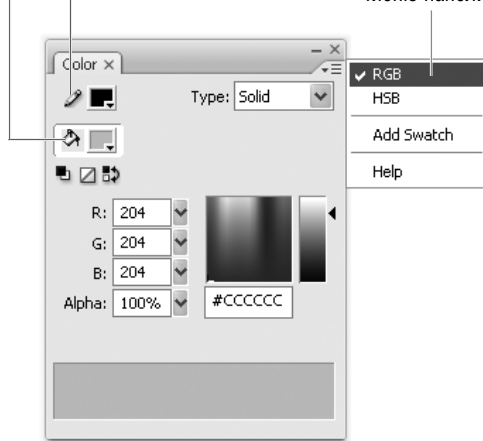
### Чтобы задать сплошной цвет на панели цветов:

1. Откройте панель цветов (если она еще не открыта, выберите из меню пункт **Window ⇒ Color** (Окно ⇒ Цвет)).
2. В списке **Type** (Тип) выберите пункт **Solid** (Сплошной).
3. Выбрать пространство цветов позволяет меню панели:
  - для задания цвета в виде комбинации красного, зеленого и синего выберите схему RGB;
  - если хотите определять цвет путем задания насыщенности, яркости и оттенка, выберите схему HSB (рис. 2.4).

Элемент управления цветом заливки

Элемент управления цветом обводки

Меню панели



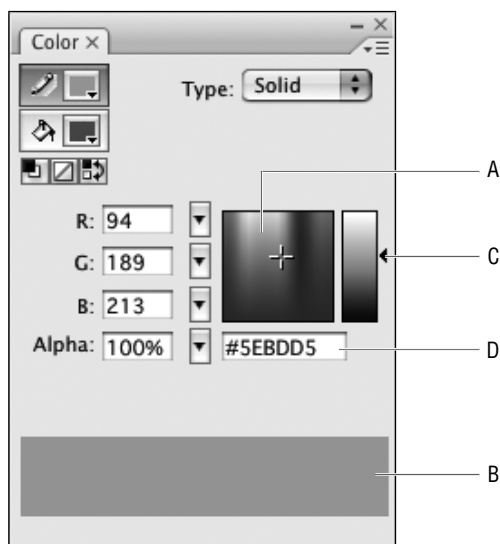
**Рис. 2.4.** С помощью панели цветов можно выбрать цвет из пространства цветов или ввести значение напрямую, выбрав одну из схем RGB или HSB. Желаемая схема задается с помощью меню панели.

### Что такое шестнадцатеричные цвета?

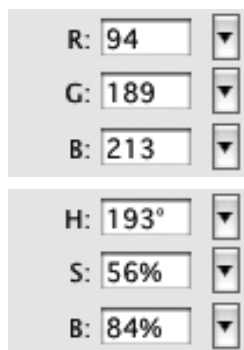
Выражение *шестнадцатеричный цвет* — это просто сокращение от «цвет, заданный числом в шестнадцатеричной системе счисления». В шестнадцатеричном виде представляются биты и байты, которыми оперирует компьютер.

Именно в таком виде цвета задаются в HTML-разметке. Во Flash ввести одно шестнадцатеричное значение проще, чем три значения для красной, зеленой и синей компонент (RGB).

Если знания о системах счисления еще не выветрились из вашей памяти, то вы, наверное, вспомните, что десятичной называется система по основанию 10, в которой числа представляются цифрами от 0 до 9. В шестнадцатеричной системе есть шесть дополнительных цифр, обозначаемых буквами от A до F.



**Рис. 2.5.** На панели цветов присутствуют: область пространства цветов (А), область предварительного просмотра нового цвета (В), ползунок для задания яркости цвета (С) и текстовое поле для ввода точного шестнадцатеричного значения цвета (D). Чтобы задать цвет визуально, щелкните по соответствующей ему точке в области пространства цветов.



**Рис. 2.6.** Введите числовые значения красной, зеленой и синей компонент цвета. Или значения оттенка, насыщенности и яркости. Новый цвет отображается в выбранном элементе управления цветом.

4. Чтобы указать, к чему применять новый цвет, выполните одно из следующих действий:

- если нужно задать цвет обводки, выберите элемент управления цветом обводки, щелкнув по значку, изображающему карандаш;
- если нужно задать цвет заливки, выберите элемент управления цветом заливки, щелкнув по значку, изображающему ведро с краской;

### Чтобы визуально задать новый цвет на панели цветов:

1. Открыв панель цветов, выберите пространство цветов (цветовую схему).
2. Наведите курсор на нужный цвет в области пространства цветов.
3. Щелкните мышью.

Когда курсор находится над областью пространства цветов, он принимает форму перекрестья, центр которого отмечает выбранный цвет (рис. 2.5).

**С** Если не удастся точно щелкнуть по нужному цвету, поведите мышью с нажатой кнопкой в области пространства цветов. В окне предварительного просмотра будет отображаться как старый, так и новый цвет. Когда появится нужный вам цвет, отпустите кнопку мыши. Числовые значения, определяющие выбранный цвет, будут занесены в соответствующие поля.

### Чтобы численно задать цвет на панели цветов:

1. Открыв панель цветов, выберите пространство цветов (цветовую схему).
2. Чтобы задать новый цвет, выполните одно из следующих действий:
  - если выбрана схема RGB, введите значения от 0 до 255 в поля R, G и B (К, З и С) (рис. 2.6);
  - если выбрана схема HSB, введите значения оттенка, насыщенности и яркости в поля H, S и B (О, Н и Я) (рис. 2.6).

### Чтобы задать коэффициент прозрачности цвета:

1. Открыв панель цветов, задайте цвет.
2. Введите значение в поле Alpha (Альфа) (рис. 2.7). Величина 100 (100 процентов) соответствует непрозрачному цвету, величина 0 — полностью прозрачному.

### Чтобы задать линейный градиент:

1. Откройте панель цветов.
2. В списке **Тип** выберите пункт **Linear** (Линейный) (рис. 2.8). Появляются инструменты для определения градиента (рис. 2.9). По умолчанию градиент начинается с черного цвета (указатель слева) и заканчивается белым (указатель справа).
3. В списке **Overflow** (Переполнение) оставьте значение по умолчанию **Extend** (на платформе Mac этот список состоит из текстовых строк, одна из которых называется **Extend**; а Windows список графический и значению **Extend** соответствует градиент от черного к белому). Параметр «переполнение» определяет, что делать, если градиент оказывается уже формы, которая им заливается (об изменении протяженности градиента см. главу 4).
4. Чтобы включить в градиент еще один цвет, выполните следующие действия:
  - наведите курсор на шкалу определения градиента или область под ней; рядом с концом стрелки появится знак +, говорящий о том, что можно добавить новую точку градиента;
  - щелкните в любой точке шкалы определения градиента. Flash добавит новый указатель.
5. Чтобы изменить цвет указателя градиента, щелкните по нему и определите новый цвет любым из описанных выше способов.

Alpha: 100%

Рис. 2.7. Введите в поле Alpha значение коэффициента прозрачности, меньшее 100.

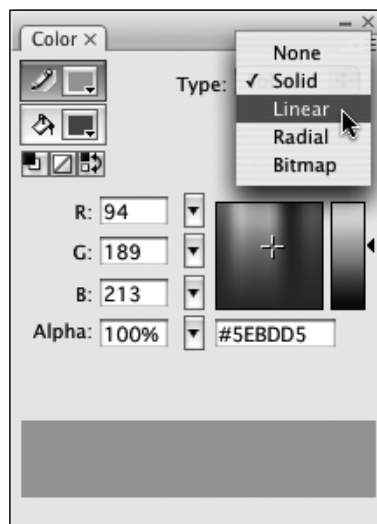


Рис. 2.8. Выберите в списке **Тип** пункт **Linear**, чтобы появились инструменты для задания линейного градиента.

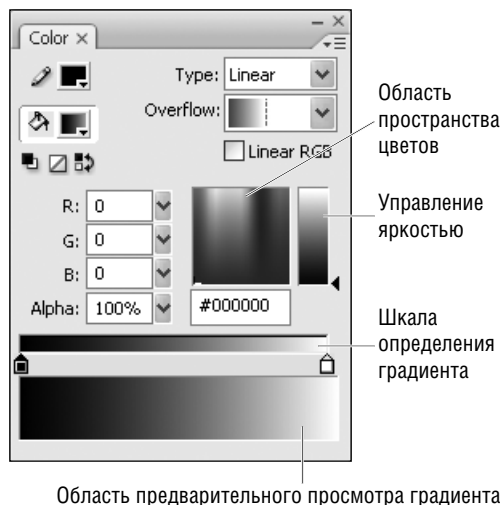
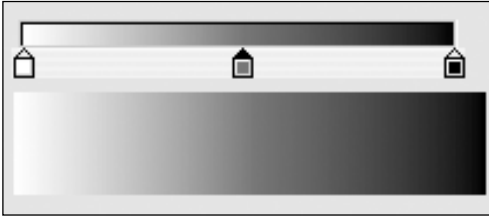
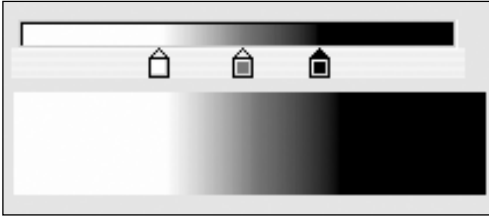


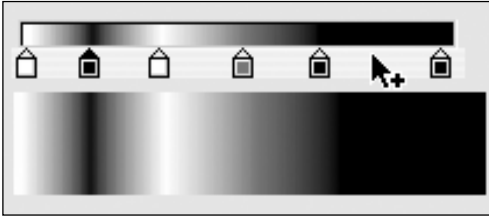
Рис. 2.9. Если выбрать из списка **Тип** пункт **Linear** или **Radial**, то на панели цветов отображается область предварительного просмотра градиента, шкала определения градиента и область пространства цветов.



Градиент начинается с белого цвета и через серый переходит к черному



Переместите указатели, чтобы расширить внешние полосы



Щелкните мышью для добавления новых указателей

**Рис. 2.10.** Задайте цвет для каждого указателя градиента. Цвета и положения указателей на шкале определяют переходы цветов. Если указатели расположены близко друг к другу, то переход цветов будет резким; если далеко — то более плавным.

или

дважды щелкните по указателю градиента; откроется окно образцов, в котором можно выполнить одно из следующих действий:

- щелкнуть по образцу, чтобы скопировать его цвет;
- щелкнуть по любому объекту в рабочей области, чтобы скопировать его цвет;
- ввести шестнадцатеричный код цвета;
- щелкнуть по значку элемента выбора цвета в правом верхнем углу окна образцов; откроется системное окно выбора цвета, в котором можно указать новый цвет.

6. Чтобы убрать цвет из градиента, перетащите соответствующий ему указатель со шкалы вниз. При этом указатель исчезнет, а в области предварительного просмотра произойдут изменения, отражающие новый состав опорных точек градиента.
7. Повторите шаги 4-6 для создания нужного вам набора цветов в градиенте. Можно добавить до 13 указателей (так что с учетом начального и конечного получится 15).
8. Расположите указатели на шкале определенного градиента так, как вам нужно (рис. 2.10). После определения нового градиента изменения отразятся во всех элементах управления цветом, относящихся к обводкам или заливкам (в зависимости от того, что вы выбрали в самом начале).

### Об использовании системного диалога для выбора цвета

Задавать цвета можно с помощью одного из системных диалогов для выбора цвета. В них применяется метод, который, возможно, понравится вам больше, чем предлагаемый панелью цветов, встроенной во Flash CS3. В системном диалоге цвет задается с помощью оттенка, контраста и яркости. В ОС Macintosh есть пять таких системных диалогов, в том числе и основанный на схеме CMYK (голубой-пурпурный-желтый-черный). Чтобы открыть системный диалог, дважды щелкните мышью, удерживая клавишу **Option** (Mac) или **Alt** (Windows), по элементу управления цветом заливки или обводки на панели цветов, панели инструментов или на вкладке **Properties** в инспекторе свойств. Можно также один раз щелкнуть по элементу управления цветом, чтобы открыть окно образцов, а затем — по значку, который расположен в правом верхнем углу этого окна.

**Чтобы создать радиальный градиент:**

1. Откройте панель цветов.
2. В списке **Типе** выберите пункт Radial (Радиальный). Появятся инструменты для определения радиального градиента. Шкала определения градиента выглядит точно так же, как для линейного градиента, а в области предварительного просмотра отображается ряд концентрических эллипсов (рис. 2.11). Самый левый указатель соответствует внутреннему кольцу, а самый правый — внешнему.
3. Для определения цветовых переходов выполните шаги 3-8 из предыдущего упражнения.

**С**

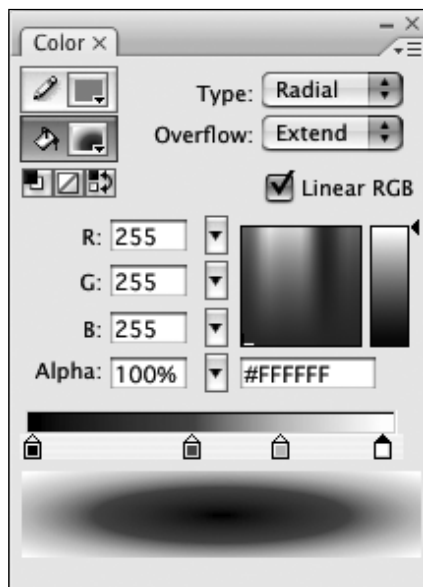
Чтобы изменить существующий градиент, выберите его на панели образцов. Flash переведет панель цветов в режим задания градиента и покажет выбранный градиент. Теперь можно вносить изменения.

**С**

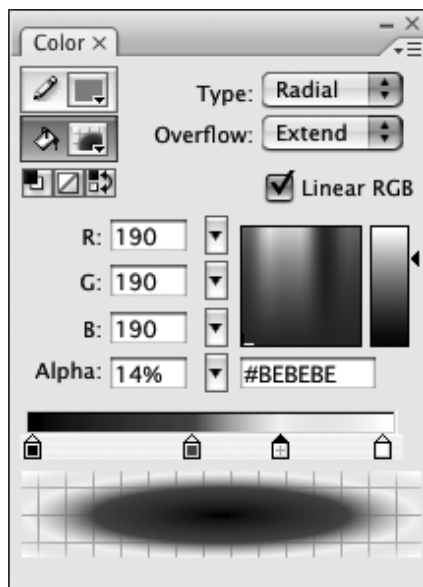
Градиенты могут быть прозрачными. Для этого достаточно выбрать прозрачный цвет при определении одного или нескольких указателей градиента (см. раздел «Чтобы задать коэффициент прозрачности цвета» выше). Если в составе градиента есть прозрачные цвета, то в элементе управления цветом обводки или заливки, а также в области предварительного просмотра градиента появится решетка (рис. 2.12).

**С**

Для разных указателей градиента коэффициент прозрачности может быть разным. Чтобы создать эффект затухания цвета, попробуйте градиент с переходом от полностью непрозрачного цвета к прозрачному.



**Рис. 2.11.** Выберите пункт Radial из списка Type. В области предварительного просмотра содержимое шкалы определения градиента преобразуется в круговые переходы цветов.



**Рис. 2.12.** Если в составе градиента есть прозрачные цвета, то в элементе управления цветом обводки или заливки, а также в области предварительного просмотра градиента появляется решетка.



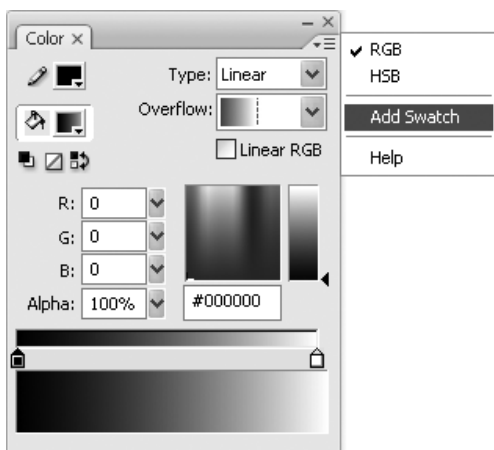


Рис. 2.13. В меню панели цветов есть команда добавления текущего цвета на панель образцов.

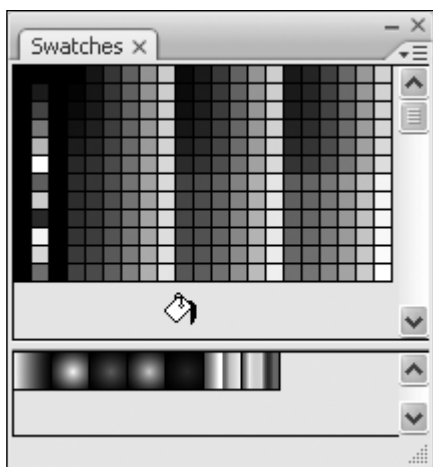


Рис. 2.14. Если навести курсор на пустое место на панели образцов, он примет форму ведра с краской. Щелкните мышью, чтобы добавить текущий цвет с панели цветов в состав набора образцов. Сплошные цвета добавляются в верхнюю часть панели образцов, градиенты — в нижнюю.

## Образцы

Новый цвет или градиент можно сохранить на все время сеанса работы, добавив цвет, отображаемый на панели цветов, на панель образцов (Swatches). Быстрый доступ к текущему набору образцов дают элементы управления цветом заливки и обводки, которые находятся на панели инструментов, на вкладке **Properties** инспектора свойств и на панели цветов.

### Чтобы добавить цвет или градиент на панель образцов:

1. Создайте новый цвет или градиент, как описано в предыдущих разделах.
2. Находясь на панели цветов, выполните одно из следующих действий:
  - в меню панели выберите команду **Add Swatch** (Добавить образец) (рис. 2.13);
  - укажите курсором на пустое место на панели образцов; когда курсор примет форму ведра с краской, щелкните мышью. Flash добавит новый сплошной цвет или градиент в соответствующую область панели образцов (рис. 2.14).

**С** Добавлять новые цвета на панель образцов можно даже тогда, когда она закрыта. Но если вы хотите видеть результат своих действий, откройте эту панель. Измените размер окна панели, так чтобы под существующими образцами осталось немного пустого места.

**С** Чтобы удалить образец с панели, выберите его, а затем выполните команду **Delete Swatch** (Удалить образец) из меню панели.

**С** Если образцы на панели образцов слишком малы, измените ее размер. По мере увеличения размера панели образцы становятся шире.

## Создание наборов цветов

Flash сохраняет подразумеваемый по умолчанию набор цветов и градиентов в системном файле цветов, а цвета и градиенты, используемые в конкретном документе, — вместе с этим документом. Вы можете сами определить, какие цвета и градиенты входят в набор, подразумеваемый по умолчанию. Кроме того, можно создать и сохранить другие наборы цветов и загрузить их в панель образцов. При таком подходе становится проще поддерживать согласованную цветовую палитру для создания нескольких документов, используемых в одном ролике или сайте.

### Чтобы определить новый набор цветов:

1. Задайте цвета и градиенты, которые должны войти в набор цветов (см. раздел «Сплошные цвета и градиенты» выше). Необязательно задавать все сразу. Когда будете готовы сохранить набор, переходите к шагу 2.
  2. Перейдите на панель образцов.
  3. В меню панели выберите команду **Save Colors** (Сохранить цвета) (рис. 2.15). Появится диалоговое окно Export Color Swatch (Экспорт образца цвета) (рис. 2.16).
  4. Перейдите в папку, в которой хотите сохранить свой набор цветов.
  5. Введите имя набора в поле Save As (Сохранить как) (Mac) или File Name (Имя файла) (Windows).
  6. В раскрывающемся списке **Format** (Формат) (Mac) или **Save As Type** (Тип файла) (Windows) выберите один из двух форматов:
    - чтобы сохранить цвета и градиенты во внутреннем для Flash формате Flash Color Set (CLR), выберите пункт Flash Color Set (Палитра Flash);
    - для сохранения в формате Color Table (ACT) выберите пункт Color Table (Таблица цветов).
- В формате АСТ сохраняются только цвета (не градиенты), зато он позволяет использовать наборы цветов и в других программах, например, Adobe Fireworks.
7. Нажмите кнопку **Save** (Сохранить).

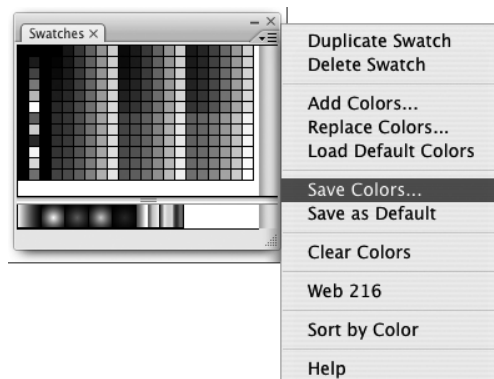


Рис. 2.15. В меню панели образцов есть команды для работы с наборами цветов.

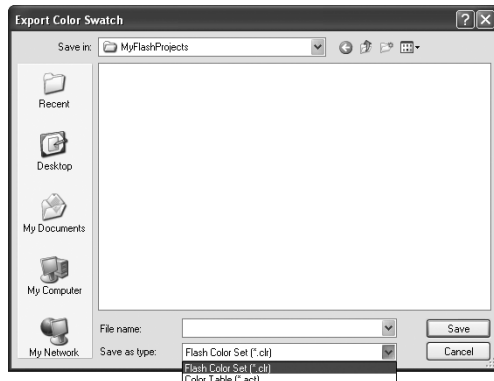
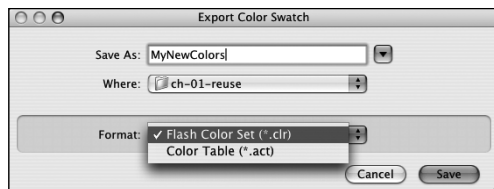


Рис. 2.16. Чтобы сохранить набор цветов для повторного использования, в диалоговом окне Export Color Swatch выберите формат Flash Color Set из списка, который на платформе Mac называется Format (верхний рисунок), а в Windows — Save As Type (нижний рисунок).



**Рис. 2.17.** Чтобы загрузить сохраненный набор цветов, в диалоговом окне **Import Color Swatch** выберите пункт **Flash Color Set** из списка, который на платформе Mac называется **Enable** (верхний рисунок), а в Windows — **Files of Type** (нижний рисунок).

## Чтобы загрузить набор цветов:

1. С помощью меню панели образцов выполните одно из следующих действий:

- чтобы добавить новые цвета в набор, который отображается на панели образцов в данный момент, выберите команду **Add Colors** (Добавить цвета);
- чтобы заменить весь набор, выберите команду **Replace Colors** (Заменить цвета).

Открывается диалоговое окно **Import Color Swatch** (Импорт образца цвета) (рис. 2.17).

2. Чтобы указать, файлы каких типов показывать, выберите из раскрывающегося списка **Enable** (Включить) (Mac) или **Files of Type** (Тип файлов) (Windows) один из следующих пунктов:

- **All Formats** (Все форматы) — в этом случае показываются файлы типа CLR, ACT и GIF;
- **Flash Color Set** (Палитра Flash) — показываются только файлы типа CLR;
- **Color Table** (Таблица цветов) — показываются только файлы типа ACT;
- **GIF** — показываются только файлы типа GIF;
- **All Files** (Все файлы) — показываются файлы любого типа.

Отметим, что форматы **Color Table** и **GIF** пригодны только для импорта цветов, градиенты в них не поддерживаются. Формат **Flash Color Set** поддерживает как цвета, так и градиенты.

3. Укажите файл, который хотите импортировать.
4. Нажмите кнопку **Open** (Открыть).

## С

В меню панели образцов есть также команды для быстрого выполнения операций с наборами цветов. Чтобы загрузить стандартный набор цветов, выполните команду **Load Default Colors** (Загрузить цвета по умолчанию). Чтобы удалить все образцы цветов с текущей панели, выполните команду **Clear Colors** (Очистить цвета). Чтобы загрузить стандартный набор цветов, считающихся безопасными при работе в Web, выполните команду **Web 216**. Для упорядочения цветов по оттенку служит команда **Sort by Color** (Сортировать по цвету). (Отметим, что отменить сортировку по цвету невозможно. Поэтому сохраняйте текущий набор цветов, если полагаете, что неотсортированный набор еще может понадобиться.)

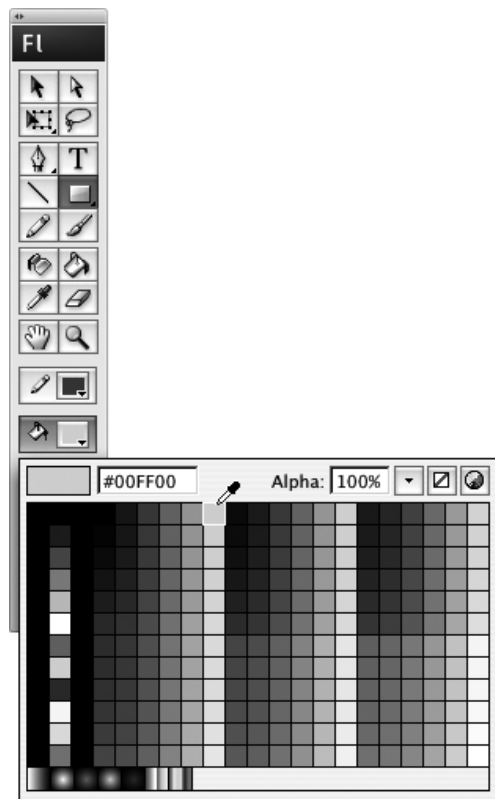
## Задание атрибутов заливки

Во Flash есть заливки пяти типов: пустая, сплошная, линейно-градиентная, радиально-градиентная и растровая. Новые цвета и градиенты можно создавать с помощью панели цветов (см. раздел «Сплошные цвета и градиенты» выше). Чтобы назначить цвета или градиенты выбранным инструментам или графическим элементам, можно воспользоваться панелью цветов, панелью инструментов или инспекторами свойств, относящимися к заливкам (например, инспектором, ассоциированным с инструментом «Прямоугольник»).

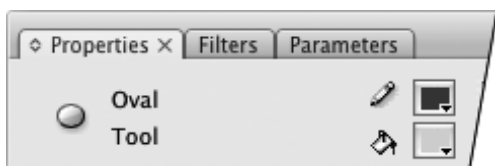
### Чтобы назначить цвета заливки, пользуясь панелью инструментов:

1. На панели инструментов щелкните по элементу управления цветом заливки (рядом с ним находится значок ведра с краской). Этот элемент подсвечивается, курсор принимает форму пипетки и появляется всплывающее окно образцов (рис. 2.18).
2. Для назначения нового цвета или градиента выполните одно из следующих действий:
  - чтобы назначить градиент, выберите один из образцов линейного или радиального градиента;
  - чтобы назначить сплошной цвет, выберите его образец или щелкните по какому-нибудь объекту на арене; цвет пикселя, находящегося под кончиком пипетки, будет назначен заливке;
  - чтобы определить новый цвет, введите его шестнадцатеричный код в поле, расположенное над образцами;
  - чтобы задать коэффициент прозрачности для текущего цвета заливки, введите число, меньшее 100, в поле Alpha и нажмите клавишу **Enter**.

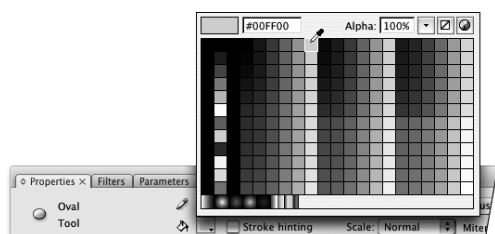
Отметим, что этот элемент управления не позволяет задавать прозрачность градиентных заливок, так как прозрачность является частью определения самого градиента.



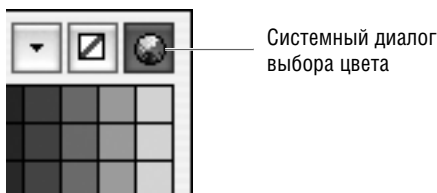
**Рис. 2.18.** Чтобы задать цвет заливки, пользуясь панелью инструментов, щелкните по элементу управления цветом заливки. Открывается окно с набором образцов, из которого можно выбрать цвет заливки.



**Рис. 2.19.** Если на арене не выбран ни один объект и выбирается инструмент, создающий заливки, например овал, то на вкладке **Properties** инспектора свойств появляется элемент управления цветом заливки.



**Рис. 2.20.** Элементы управления цветом в инспекторе свойств работают точно так же, как на панелях инструментов и цветов. Щелчок по такому элементу открывает окно образцов.



**Рис. 2.21.** Щелкните по значку системного диалога выбора цвета, чтобы с его помощью задать цвет, отсутствующий в текущем наборе образцов.

Новый цвет отображается в элементе управления цветом заливки (на панели инструментов, на вкладке **Properties** инспектора свойств или на панели цветов) и будет использоваться всеми инструментами, которые создают заливки.

### Чтобы задать цвет заливки, пользуясь вкладкой **Properties** инспектора свойств:

1. Перейдите на вкладку **Properties** инспектора свойств.
2. На панели инструментов выберите какой-нибудь инструмент, создающий заливки: прямоугольник, овал, прямоугольный примитив, овальный примитив, многоугольник, кисть или ведро с краской. В результате на вкладке **Properties** появится элемент управления цветом заливки (рис. 2.19).
3. На вкладке **Properties** инспектора свойств щелкните по элементу управления цветом заливки (рядом с ним находится значок ведра с краской). Появляется текущий набор образцов (рис. 2.20).
4. Чтобы задать новый цвет заливки, поступайте так же, как описано в шаге 2 предыдущего упражнения.

**С** Чтобы получить доступ к системному диалогу выбора цвета, щелкните по элементу управления цветом заливки, а затем по значку, обозначающему системный диалог во всплывающем окне образцов (рис. 2.21).

### Элементы управления цветом

Элементы управления цветом — цветной квадратик с треугольничком в правом нижнем углу — встречаются в самых разных местах Flash. Если щелкнуть по нему, откроется всплывающее окно образцов цвета, а курсор примет форму пипетки. Чтобы назначить новый цвет, щелкните мышью по соответствующему образцу или введите шестнадцатеричный код в поле Hex Color, введите коэффициент прозрачности в поле Alpha, щелкните по значку No Color или по значку системного диалога выбора цвета. Многие элементы управления цветом позволяют указать пипеткой на произвольный объект на арене и выбрать его цвет. Для назначения цвета годится любой элемент управления цветом заливки или обводки. Назначенный цвет появится на панели цветов и во всех остальных элементах управления цветом.

## Задание атрибутов обводки

У линии есть три основных атрибута: цвет, толщина (ее называют также плотностью, а во Flash еще и высотой обводки) и стиль. Все они задаются на вкладке **Properties** инспектора свойств для любого инструмента, создающего обводки. Там же можно определить, как должны выглядеть окончания и точки пересечения линий.

### Чтобы задать свойства обводки:

1. Открыв вкладку **Properties** инспектора свойств, выберите на панели инструментов какой-нибудь инструмент, создающий обводку (перо, линия, прямоугольник, овал, прямоугольный примитив, овальный примитив, многоугольник, карандаш или чернильницу).

На вкладке **Properties** появляются свойства выбранного инструмента, в том числе и текущие параметры обводки.

2. Чтобы выбрать цвет обводки, выполните следующие действия:

- на вкладке **Properties** инспектора свойств (или на панели инструментов) щелкните по элементу управления цветом обводки;

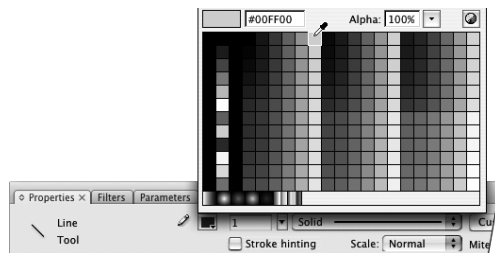
курсор принимает форму пипетки и появляется окно с набором образцов (рис. 2.22);

- щелкните по образцу нужного цвета.

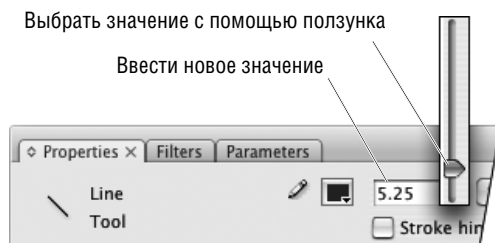
3. Чтобы задать толщину обводки, введите число от 0.25 до 200 в соответствующее поле или воспользуйтесь расположенным рядом с ним ползунком (рис. 2.23).

4. Выберите стиль обводки из раскрывающегося списка стилей (рис. 2.24).

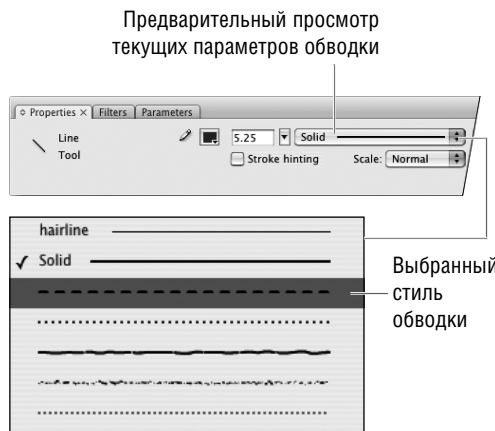
Всего существует семь стилей: волосная линия, сплошная, пунктирная, точечная, зубчатая, гравировка и штриховка. В списке отображаются их графические представления.



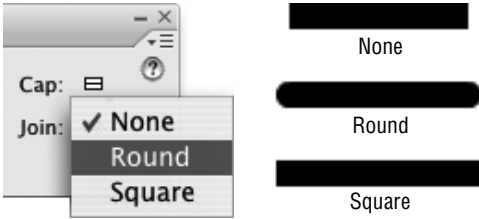
**Рис. 2.22.** При щелчке по элементу управления цветом обводки на вкладке **Properties** инспектора свойств открывается окно с набором образцов цвета, а курсор принимает форму пипетки.



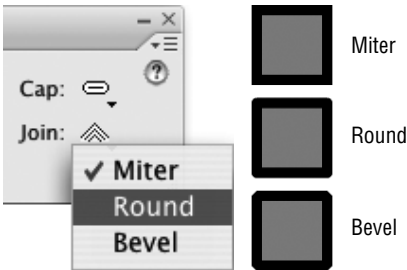
**Рис. 2.23.** Значение в поле высоты обводки определяет толщину или плотность обводок, создаваемых соответствующими инструментами.



**Рис. 2.24.** Выберите стиль обводки из раскрывающегося списка на вкладке **Properties** инспектора свойств.



**Рис. 2.25.** Чтобы задать способ рисования окончаний линий, выберите стиль из списка Cap на вкладке Properties инспектора свойств. Если выбран вариант None, то исходная длина линии сохраняется, а ее окончание будет плоским. В случае варианта Round линия продлевается, а окончание будет скругленным. Наконец, вариант Square означает, что линия продлевается, а окончание будет квадратным.



**Рис. 2.26.** Чтобы задать способ соединения линий, выберите стиль из списка Join на вкладке Properties инспектора свойств. Если выбран вариант Miter, то рисуется резкий угол, если Round — то закругленный, а если Bevel — то скошенный. Списки Cap и Join доступны только, если выбрана плотность обводки Solid или Hairline.

5. Чтобы определить, как заканчивается волосная или сплошная обводка, воспользуйтесь раскрывающимся списком **Cap** (Окончания).

- None (Нет) означает, что обводка заканчивается точно там, где вы прекратили рисование.
- Round (Скругленное) — линия продолжается еще на половину текущей толщины, и ее конец будет скругленным.
- Square (Квадратное) — линия продолжается еще на половину текущей толщины, и ее конец будет квадратным (рис. 2.25).

6. Чтобы определить способ рисования точки стыка волосных или сплошных обводок, воспользуйтесь списком **Join** (Форма стыка).

- Miter (Без сглаживания) — рисуется резкий угол.
- Round (Скругленный) — рисуется слегка скругленный угол.
- Bevel (Скос) — рисуется слегка скошенный угол (рис. 2.26).

## C

На панели инструментов есть три кнопки для быстрого выполнения часто используемых операций с цветами обводок (и заливок). Если щелкнуть по паре пересекающихся квадратиков — черному и белому, — то элемент управления цветом обводки закрасивается черным цветом, а элемент управления цветом заливки — белым. Щелчок по паре квадратиков, рядом с которыми изображена стрелочка, меняет местами цвета обводки и заливки. Щелчок по квадратiku, перечеркнутому красной линией, переводит выбранный в данный момент элемент управления цветом (обводки или заливки) в состояние No Color (Без цвета).

## C

Во Flash параметр «волосная линия» считается стилем, а не толщиной обводки. (Для задания этого стиля используется раскрывающийся список стилей.) Толщина таких линий не меняется при изменении размера содержащего их символа. Все остальные линии при масштабировании символа становятся тоньше или толще. (Символам посвящена глава 7.)

C

Чтобы ограничить масштабирование обводок в готовом ролике, выберите интересующую вас обводку (или создающую обводки инструмент) и воспользуйтесь раскрывающимся списком **Scale** (Масштаб) на вкладке **Properties** инспектора свойств. В режиме по умолчанию (Normal — Обычный) обводка толщиной 1 пиксель рисуется как двухпиксельная, если ролик масштабируется до 200 процентов. Чтобы предотвратить масштабирование обводки, выберите пункт None (Нет). Чтобы разрешить масштабирование обводки только в одном направлении, выберите пункт Horizontal (По горизонтали) или Vertical (По вертикали).

C

Отметьте флажок **Stroke Hinting** (Хинтинг кривой) на вкладке **Properties** инспектора свойств, если в окончательном ролике линии должны быть резкими. В противном случае на некоторых мониторах линии иногда будут казаться слегка размытыми.

C

Стили обводок можно модифицировать. Например, в пунктирной линии можно увеличить диаметр точек, а в штрих-пунктирной — расстояние между черточками. Выберите какой-нибудь инструмент создания обводок, на вкладке **Properties** инспектора свойств выберите стиль, который собираетесь модифицировать, и нажмите кнопку **Custom** (Заказной). Появится диалоговое окно **Stroke Style**, в котором можно задать новые параметры. Чтобы подтвердить изменения и закрыть окно, нажмите **OK**. Новые параметры будут действовать для данного стиля до конца сеанса или пока вы их явно не измените.

C

Цвет обводки можно задать также с помощью любого элемента управления цветом обводки (в инспекторе свойств, на панели инструментов или цветов). Щелкните по элементу управления цветом обводки и выберите цвет, как описано выше в разделе «Задание атрибутов заливки».

C

Свойство **Join** можно задавать и для прямоугольников, созданных с помощью инструмента «Прямоугольный примитив», но соединения будут видны только в резких углах (с радиусом скругления равным 0).

### Комбинированные формы, объекты-рисунки и примитивные формы

Во Flash CS3 можно создавать графические объекты трех типов: *комбинированные формы* (их еще называют элементарными формами), *объекты-рисунки* и *примитивные формы*. Комбинированные формы и объекты-рисунки можно создавать с помощью большинства инструментов рисования, достаточно просто выбрать подходящий режим рисования. Что касается примитивов, то они создаются только инструментами «Прямоугольный примитив» и «Овальный примитив». Обводки и заливки, созданные в режиме **Merge Drawing** (Комбинированное рисование), можно редактировать прямо на арене; элементарные формы, находящиеся в одном и том же слое, взаимодействуют друг с другом, то есть разбивают и заменяют те обводки и заливки, которые перекрывают или пересекают (о взаимодействии форм мы еще будем говорить в главе 5).



Обводки и заливки, созданные в режиме **Object Drawing** (Рисование объектов), тоже можно редактировать на арене, но они не взаимодействуют с другими формами в том же слое. Формы, созданные в режиме рисования объектов, ведут себя так, как будто помещены в отдельный слой или защищены командой **Group** (о работе со сгруппированными формами см. главу 5, а о формах в разных слоях — главу 6). Однако формы, созданные в режиме рисования объектов, можно редактировать прямо на арене (см. главу 4), а сгруппированные объекты — вообще говоря, нельзя.

Примитивные формы — созданные инструментами «Прямоугольный примитив» и «Овальный примитив» — не взаимодействуют с формами в том же слое. Такие формы Flash создает в режиме рисования объектов, но при этом блокирует их и налагает такие ограничения, чтобы у них всегда были определенные характеристики. Эти характеристики можно редактировать, и в инспекторе свойств, который появляется, когда выбран примитивный инструмент на панели инструментов или примитивная форма на арене, они выглядят как свойства. Эти свойства примитива можно редактировать в любой момент, но изменить примитив произвольным образом, как форму, созданную в режиме комбинированного рисования или рисования объектов, не удастся. Например, можно изменить внутренний радиус овального примитива, в результате чего сплошной овал трансформируется в бублик, но нельзя менять его контур так, чтобы концы стали заостренными, как у мяча для игры в регби.

По умолчанию все инструменты рисования работают в режиме комбинированного рисования. Чтобы переключиться в режим рисования объектов, выберите инструмент для создания обводок или заливок и нажмите кнопку **Object Drawing** (Рисование объектов) на панели инструментов (рис. 2.27). Если активирован режим рисования объектов хотя бы для одного инструмента, то и все инструменты создания обводок и заливок переходят в тот же режим. Для возврата в режим комбинированного рисования нажмите кнопку **Object Drawing** еще раз (режимы переключает также клавиша J). Режим, который был активен в момент завершения сеанса, будет действовать и при следующем открытии Flash. При выборе любого примитивного инструмента кнопка **Object Drawing** на панели инструментов отсутствует (рис. 2.28).

В большинстве случаев различия между тремя типами форм незаметны, но иногда оказываются очень важными. Во всех упражнениях в этой книге, если явно не оговорено противное, можно рисовать в любом режиме или пользоваться примитивными инструментами.

Режим рисования объектов



**Рис. 2.27.** Можно настроить инструменты рисования так, что создаваемые формы не будут взаимодействовать с другими формами в том же слое. Для этого задайте на панели инструментов режим рисования объектов.



Выбран инструмент «Овальный примитив»

Кнопка выбора режима рисования отсутствует

**Рис. 2.28.** Кнопка, определяющая режим рисования, отсутствует на панели инструментов, если выбран инструмент «Прямоугольный примитив» или «Овальный примитив».

## Создание геометрических форм

Во Flash имеются инструменты для рисования овалов, прямоугольников и многоугольников. Все они работают по одному принципу и позволяют рисовать форму в виде контура (только обводку) или сплошного объекта (заливка). Можно также создавать геометрические формы, содержащие и обводки, и заливки. Для прямоугольников и овалов существуют специальные свойства, точные числовые значения которых можно задать в инспекторе свойств.

### Чтобы создать контур:

1. Для выбора инструмента создания геометрической формы выполните следующие действия:
  - на панели инструментов щелкните по значку активного инструмента;
  - из выпадающего меню выберите конкретный инструмент, например «Овал» (рис. 2.29).

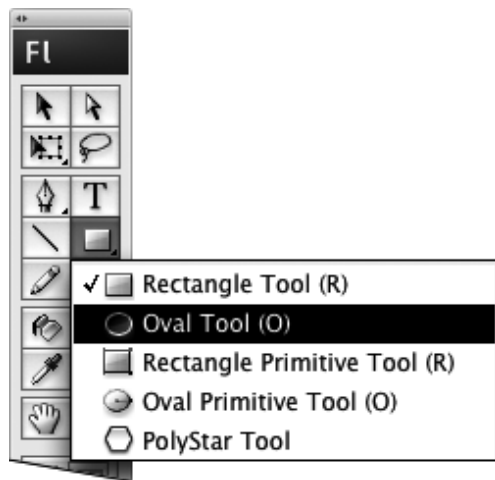
2. В инспекторе свойств задайте значения интересующих вас свойств.

При создании контура формы можно задавать цвет, толщину и стиль обводки (см. раздел «Задание атрибутов обводки» выше). Для овала и прямоугольника есть также ряд специальных свойств, которые появляются в инспекторе, если выбран один из инструментов «Прямоугольник», «Прямоугольный примитив», «Овал» или «Овальный примитив».

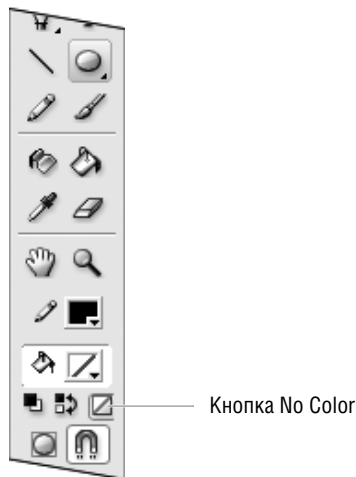
3. Чтобы выбранный инструмент создавал только контур без заливки, выполните следующие действия:

- щелкните по значку ведра с краской при элементе управления цветом заливки; этот элемент станет активным;
- нажмите кнопку **No Color** (Без цвета) (рис. 2.30).

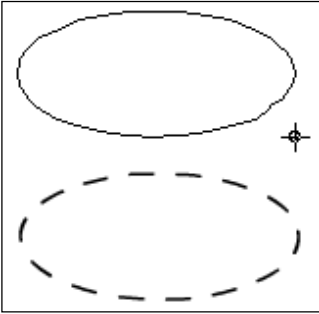
4. Переместите курсор на арену. Он примет форму перекрестия.



**Рис. 2.29.** На панели инструментов все инструменты для рисования геометрических форм сведены в одно выпадающее меню. Чтобы увидеть полный набор, щелкните по значку активного инструмента, а затем щелкните по нему еще раз.



**Рис. 2.30.** Выбрав на панели инструментов элемент управления цветом заливки, нажмите кнопку **No Color**. В результате текущий инструмент будет создавать форму без заливки.



**Рис. 2.31.** Выбрав какой-либо инструмент рисования неометрических форм, щелкните по арене и, буксируя мышью, нарисуйте предварительный контур формы (верхний рисунок). Как только вы отпустите кнопку мыши, Flash создаст форму с текущими параметрами обводки и заливки. В данном случае был выбран инструмент «Овал», с черной обводкой, без заливки, со штрих-пунктирным стилем обводки толщиной 1 пункт (нижний рисунок).

5. Буксируя курсор, нарисуйте геометрическую форму (рис. 2.31). В процессе буксировки Flash показывает, как будет выглядеть форма.
6. Отпустите кнопку мыши. Flash нарисует окончательный контур.

**C**

Чтобы нарисовать идеальную окружность (или квадрат), удерживайте клавишу **Shift** во время рисования прямоугольника, овала, прямоугольного примитива или овального примитива.

**C**

Чтобы овал (или прямоугольник) в процессе рисования расширялся из центра, укажите курсором на предполагаемую центральную точку, а при буксировке удерживайте нажатой клавишу **Option** (Mac) или **Alt** (Windows). Формы, которые рисуются инструментом «Многоугольник», всегда расширяются из центра.

### Особенности примитивных форм

Свойства геометрических форм не зависят от того, выбран ли режим комбинированного рисования, рисования объектов или примитивный инструмент. На самом деле, внутри Flash пользуется главным шаблоном, в котором определены все свойства овалов и прямоугольников. Инструменты «Овал» и «Овальный примитив» создают овалы по одному и тому же шаблону. Однако, если овал был создан в режиме комбинированного рисования или в режиме рисования объектов, его связь с главным шаблоном разрывается. «Овальная форма» становится путем, который можно модифицировать с помощью инструментов выбора и спецвыделения. Что касается овального примитива, то он сохраняет связь со своим главным шаблоном. Некоторыми свойствами главного шаблона можно манипулировать с помощью инспектора свойств и управляющих точек примитива. Инспектор свойств в какой-то степени раскрывает механизм работы связи с главным шаблоном (и утраты этой связи). Например, если выбран инструмент «Овал», то инспектор свойств покажет точно такие же свойства и элементы управления, как для овального примитива. Если выбрать овальный примитив на арене, то в инспекторе свойств будут присутствовать все свойства и элементы управления, присущие овалу; при изменении элементов, которые определяют «овальность» формы (начальный угол, конечный угол, внутренний радиус), форма синхронно изменяется, но только в рамках параметров, заданных в главном шаблоне. Так, можно изменить внутренний радиус овального примитива, но нельзя произвольно трансформировать его в кляксу. Если на арене выбран овал, созданный в режиме комбинированного рисования или в режиме рисования объектов, то инспектор покажет некоторые свойства и элементы управления этой формы (например, элементы управления цветом заливки и обводки), но среди них не будет элементов, определяющих «овальность». Путь, представляющий такой овал, можно произвольно модифицировать, но изменить внутренний радиус не получится. Примитивные формы можно преобразовать в комбинированные или в объекты-рисунки. Дополнительную информацию о модификации форм см. в главе 4.

**Чтобы создать заливку:**

1. Для выбора инструмента выполните те же действия, что описаны в шагах 1 и 2 предыдущего упражнения. Но на этот раз выберите инструмент «Прямоугольник».
2. Чтобы настроить инструмент на создание заливки без контура, выполните следующие действия:
  - щелкните по значку карандаша при элементе управления цветом обводки; этот элемент станет активным;
  - нажмите кнопку **No Color** (рис. 2.32).
3. Повторите шаги 4-6 из предыдущего упражнения.

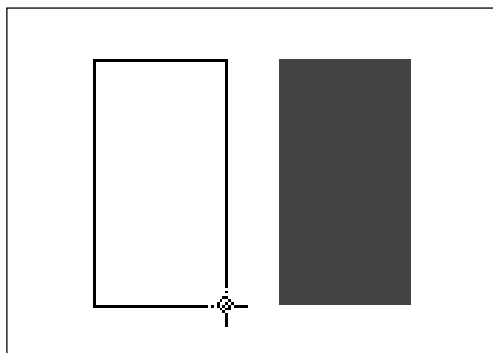
Flash нарисует геометрическую форму, залитую выбранным цветом (рис. 2.33).



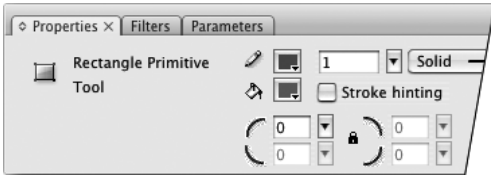
**Рис. 2.32.** Для создания геометрической формы, являющейся заливкой без обведенного контура, выберите на панели инструментов элемент управления цветом обводки и нажмите кнопку No Color.

**С**

Инструмент «Многоугольник» позволяет создавать многоугольники и звездообразные формы (выберите его из меню геометрических форм на панели инструментов). Для задания числа сторон многоугольника и для перехода от многоугольника к звезде выберите инструмент «Многоугольник» и нажмите кнопку **Options** (Параметры) на вкладке **Properties** инспектора свойств. Появится диалоговое окно Tool Settings (Параметры инструмента). Введите число сторон и меру остроты концов звезды, после чего нажмите **OK**.

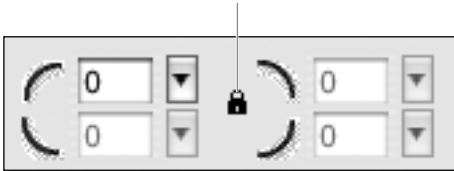


**Рис. 2.33.** При буксировке инструмента «Прямоугольник» Flash рисует эластичный контур прямоугольника (левый рисунок). Чтобы завершить рисование, отпустите кнопку мыши (правый рисунок).

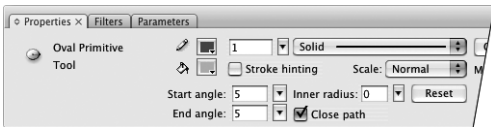


**Рис. 2.34.** При выборе инструмента «Прямоугольник» или «Прямоугольный примитив» в инспекторе свойств отображаются свойства, присущие прямоугольнику, — одинаковые в обоих случаях.

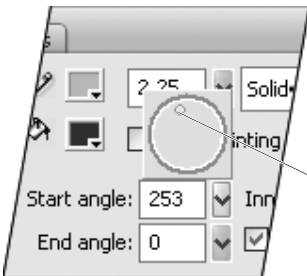
Ограничить радиус угла



**Рис. 2.35.** По умолчанию инструменты «Прямоугольник» или «Прямоугольный примитив» создают прямоугольники, все углы которых имеют один и тот же радиус. Если значок изображает запертый замок, то доступно лишь одно поле для ввода значения, то есть на параметры налагаются ограничения.



**Рис. 2.36.** При выборе инструмента «Овал» или «Овальный примитив» в инспекторе свойств отображаются свойства, присущие овалу, — одинаковые в обоих случаях.



Потянуть ползунок для перемещения управляющей точки

**Рис. 2.37.** Если ввести значения в поля Start Angle и End Angle, то будет создан сектор (такая фигура часто используется в секторных диаграммах). Щелкните по треугольничку справа от поля, чтобы открыть круговой ползунок, помогающий ввести значение.

## Чтобы задать свойства прямоугольника:

1. На панели инструментов выберите инструмент «Прямоугольник» или «Прямоугольный примитив». На вкладке **Properties** инспектора свойств появятся свойства выбранного инструмента (рис. 2.34).
2. Чтобы создать прямоугольник с четырьмя одинаковыми углами, установите значок **Constrain Corner Radius** (Ограничить радиус угла) в состояние блокировки (рис. 2.35). Щелчок по этому значку переключает состояние из «не заблокировано» (открытый замок) в «заблокировано» (запертый замок) и обратно.
3. Чтобы создать прямоугольник со скругленными углами, введите положительное число в поле Rectangle Corner Radius (Радиус закругления прямоугольника).

или

Чтобы создать прямоугольник с вырезанными углами, введите отрицательное число в поле Rectangle Corner Radius.

## Чтобы задать свойства овала:

1. На панели инструментов выберите инструмент «Овал» или «Овальный примитив». На вкладке **Properties** инспектора свойств появятся свойства выбранного инструмента (рис. 2.36). Они одинаковы в обоих случаях.
2. Для создания разновидностей овала (в том числе примитивного) выполните одно из следующих действий:
  - чтобы создать сектор, введите число от 0 до 360 в поле Start Angle (Начальный угол) и/или End Angle (Конечный угол) (рис. 2.37).

- Для создания незамкнутой дуги сбросьте флажок **Close Path** (Замкнуть путь). Flash удалит заливку овала (если она была) и сотрет отрезок, соединяющий ее концы (рис. 2.38).
- Для создания овала с вырезанной внутренней областью, введите число от 0 до 99 в поле Inner Radius (Внутренний радиус). Flash нарисует пустой овал внутри внешнего (рис. 2.39). Параметр задает, какой процент внутренний овал занимает от внешнего. (Если для внешнего овала задана заливка «Без цвета», то внутренний овал выглядит как контур.)

С

Начальный и конечный углы овального примитива можно изменять интерактивно. Выберите на арене какую-нибудь форму. В инспекторе свойств щелкните по треугольничку справа от поля Start Angle или End Angle, появится круговой ползунок. Щелкните в какой-нибудь точке ползунка или буксируйте управляющую точку, чтобы изменить значение параметра. При этом форма на арене тоже будет изменяться. Нажмите **Enter** или щелкните мышью в любой точке вне ползунка, чтобы зафиксировать новое значение.

Форма с замкнутым путем



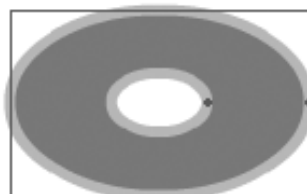
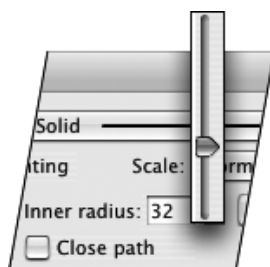
Start angle:	36	▼	Inner radius:	0	▼
End angle:	149	▼	<input checked="" type="checkbox"/> Close path		

Форма с незамкнутым путем



Start angle:	36	▼	Inner radius:	0	▼
End angle:	149	▼	<input type="checkbox"/> Close path		

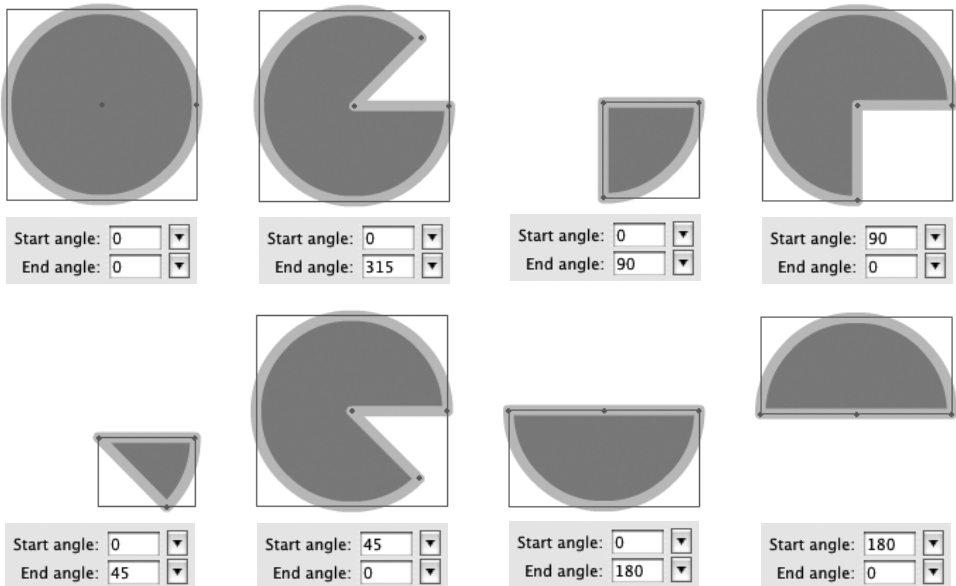
**Рис. 2.38.** Если сбросить флажок Close Path, то можно будет создавать незамкнутые дуги. Если внутренний радиус овала больше 0 и при этом флажок сброшен, рисуются две дуги.



**Рис. 2.39.** Если ввести положительное значение в поле Inner Radius, то будет нарисован овал с «дыркой».

### Углы овала

На вкладке инспектора свойств, соответствующей инструменту «Овал» (и «Овальный примитив») присутствуют элементы управления для задания начального и конечного угла. Задание этих углов позволяет создавать секторные формы, но поначалу трудно понять, как интерпретируются их значения. Представьте себе бумажный круг поверх циферблата аналоговых часов. Начальный угол — это точка, в которой вы начинаете рисовать карандашом, конечный угол — точка, в которой вы отрываете карандаш от бумаги. (Когда на арене создается овальный примитив, его начальный и конечный углы становятся управляющими точками, за которые можно тянуть для изменения формы сектора.) В поля Start Angle и End Angle вводятся градусные величины углов. Так, значение 0 соответствует трем часам, 90 — шести часам, 180 — девяти часам и так далее. При создании овальных примитивов углы отсчитываются в направлении против часовой стрелки. Если начальный и конечный угол совпадают, обе управляющие точки совмещаются, и рисуется полный круг. Если же значения различны, получается круг с вырезанным сектором. Если для овального примитива флажок замкнутости пути отмечен, то проводятся отрезки между крайними точками дуги и центром овала. Если внутренний радиус овала больше 0, то эти отрезки доходят только до внутреннего овала, в результате получается С-образная форма или ее часть (рис. 2.40). Чтобы почувствовать, как все это работает, создайте на арене овальный примитив, выберите его и поэкспериментируйте, задавая различные параметры в инспекторе свойств. Чтобы увидеть аналогию углов с часами, воспользуйтесь управляющими точками, которые позволяют менять начальный и конечный углы. Заодно поэкспериментируйте с внутренним радиусом и флажком замкнутости пути.



**Рис. 2.40.** За счет изменения значений в полях Start Angle и End Angle на вкладке инспектора свойств, соответствующей овальному примитиву, можно настроить инструмент на создание различных секторов.

## Создание произвольных форм

Для создания произвольных форм Flash предлагает три инструмента: карандаш, перо и кисть. Карандаш и перо создают контуры-обводки, а кисть — заливки без обводок. Карандашом и пером можно создать только контур, даже если рисуется замкнутая форма. После того как форма создана, ее можно модифицировать, например, залить пустой контур или добавить контур к имеющейся заливке.

Карандаш позволяет рисовать линии (обводки) естественным образом, как мы обычно рисуем на бумаге, только с помощью мыши или графического планшета. Flash скрывает информацию об узлах кривых, из которых состоит обводка, нарисованная карандашом. При работе с пером вы размещаете узлы, а между ними проводятся кривые Безье. С помощью пера можно не только с самого начала создавать кривые Безье, но и модифицировать уже имеющиеся на арене объекты. Для модификации кривой Безье имеются три инструмента: добавление узла, удаление узла и преобразование узла. В этой главе мы будем пользоваться пером для создания произвольных форм, а в главе 4 вы научитесь с помощью пера и инструментов работы с узлами модифицировать формы.

При выполнении следующих упражнений наложите на арену сетку (см. главу 1) и установите следующие параметры рисования: откройте диалоговое окно Preferences (Установки), выбрав пункт меню **Flash ⇨ Preferences** (Mac) или **Edit ⇨ Preferences** (Windows), перейдите в категорию **Drawing** (Рисунок) и отметьте флажки **Show Pen Preview** (Предварительный просмотр пера) и **Show Solid Points** (Показывать сплошные точки). Для всех остальных параметров оставьте значения по умолчанию.

### О средствах поддержки рисования

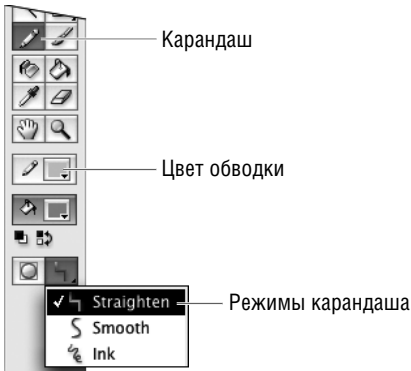
У инструмента «Карандаш» есть два вспомогательных режима проведения линий: **Straighten** (Выпрямление) и **Smooth** (Сглаживание). Чтобы обеспечить полную свободу рисования, есть еще режим **Ink** (Краска), в котором проведенные линии не подвергаются никаким модификациям.

В **режиме выпрямления** устраняются все разрывы и неровности в проведенной от руки линии, она преобразуется в последовательность отрезков прямых и дуг. В этом режиме производится также *распознавание формы*. Flash оценивает каждую нарисованную форму и, если она достаточно близка к тому, что Flash определяет как прямоугольник или овал, то ваше грубое приближение преобразуется в форму, которая удовлетворила бы школьного учителя геометрии.

В **режиме сглаживания** грубый набросок преобразуется в совокупность отрезков гладких кривых. Отметим, что в этом режиме формы не распознаются, проведенные от руки линии просто сглаживаются. При сглаживании уменьшается число точек на кривой, а, следовательно, и размер файла, что, в свою очередь, повышает скорость воспроизведения конечного результата.

Невозможно переоценить важность **параметров допуска**, особенно в режиме выпрямления. Можно настроить Flash так, чтобы практически любой оvoid преобразовывался в круг. Чтобы настроить объем помощи при рисовании, в окне Preferences перейдите в категорию **Drawing** и задайте допуски с помощью раскрывающихся списков **Connect Lines** (Соединить линии), **Smooth Curves** (Сгладить кривые), **Recognize Lines** (Распознавать линии) и **Recognize Shapes** (Распознавать формы). Чтобы закрыть окно, нажмите кнопку **OK**.





**Рис. 2.41.** При выборе инструмента «Карандаш» на панели инструментов появляется меню режимов карандаша.



**Рис. 2.42.** Когда карандаш работает в режиме выпрямления, Flash предварительно проводит кривую, точно отражая ваши движения. Как только вы отпустите кнопку мыши, производится выпрямление, в результате чего грубая «загогулина» (верхний рисунок) преобразуется в набор прямолинейных и криволинейных сегментов (нижний рисунок).



**Рис. 2.43.** Выберите инструмент «Перо» для создания путей.

### Чтобы нарисовать произвольную обводку карандашом:

1. Выберите на панели инструментов карандаш или нажмите **Y**.
2. Из меню **Pencil Mode** (Режим карандаша) (рис. 2.41) выберите один из следующих вспомогательных режимов:

*Straighten* (Выпрямление) — мелкие неточности преобразуются в отрезки прямых.

*Smooth* (Сглаживание) — мелкие неточности преобразуются в гладкие кривые.

*Ink* (Чернила) — практически никакой поддержки, мелкие неточности остаются.

3. Переместите курсор на арену. Он примет форму карандаша.
4. Нажмите кнопку мыши и, не отпуская ее, проведите загогулину. Flash будет рисовать ее на экране.
5. Отпустите кнопку мыши.

Нарисованная линия преобразуется в соответствии с режимом, выбранным на шаге 2, и будет заменена последовательностью прямолинейных и криволинейных сегментов (рис. 2.42).

### С

Сглаживание и выпрямление (даже распознавание форм) можно применить и после того, как контур или форма уже нарисованы. Для этого нужно выбрать объект на арене, а затем щелкнуть по модификатору *Straighten* (Выпрямление) или *Smooth* (Сглаживание) инструмента «Стрелка» (дополнительную информацию о том, как модифицировать параметры выбора, см. в главе 4).

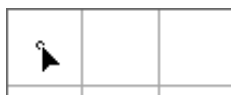
### Чтобы нарисовать произвольную обводку пером:

1. На панели инструментов выберите перо или нажмите одну из клавиш **P**, **=**, **-** или **C** (рис. 2.43).
2. Задайте атрибуты обводки.

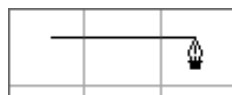
3. Переместите курсор на арену. Он примет форму пера, рядом с которым отображается маленькая буква х (рис. 2.44). Это означает, что все готово к размещению первой точки пути.
4. Щелкните мышью в том месте, где должен начинаться первый сегмент. Курсор примет форму сплошной стрелки, а кружочек обозначает положение якорной точки на арене.
5. Подведите перо к точке, где должен заканчиваться сегмент. По мере того, как вы буксируете мышью, Flash рисует прямолинейный сегмент, соединяющий начальную точку с текущим положением кончика пера.
6. Отпустите кнопку мыши. Flash нарисует окончательное положение сегмента с учетом установленных атрибутов обводки. Узлы представляются квадратами (рис. 2.45).
7. Чтобы добавить к пути еще один прямолинейный сегмент, щелкните в той точке арены, где он должен заканчиваться (рис. 2.46).
8. Чтобы добавить криволинейный сегмент, щелкните в той точке арены, где должна заканчиваться кривая, а затем буксируйте мышью. Пока вы буксируете, курсор принимает форму сплошной стрелки и рисуется кривая Безье вместе с манипуляторами (иногда они называются касательными манипуляторами) (рис. 2.47).



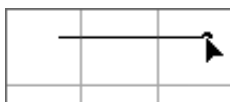
**Рис. 2.44.** Буква х рядом с кончиком пера означает готовность к рисованию нового пути. Щелкните мышью, чтобы поместить первую точку.



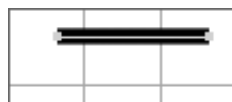
Первая Точка



Предварительный отрезок прямой

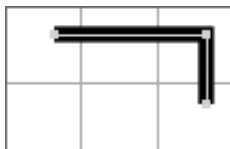
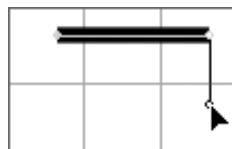
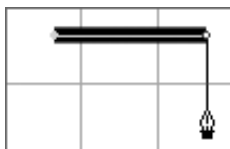


Щелкните, чтобы поместить вторую точку

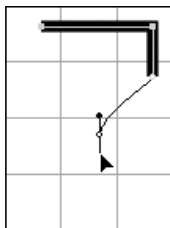


Окончательный отрезок прямой

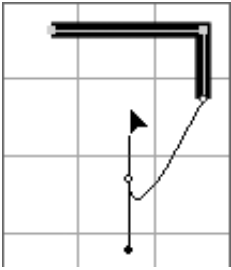
**Рис. 2.45.** Flash рисует точки, которые вы ставите (верхний рисунок) и добавляет в путь линию, когда вы заканчиваете сегмент (нижний рисунок).



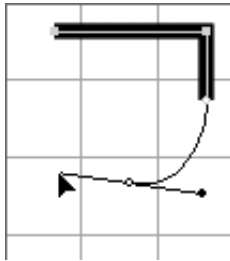
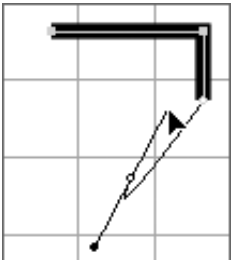
**Рис. 2.46.** Продолжайте добавлять сегменты в произвольную форму. Одиночный щелчок (верхний рисунок) добавляет прямолинейный сегмент (нижний рисунок).



**Рис. 2.47.** Нажмите кнопку мыши и буксируйте, чтобы создать криволинейный сегмент, при этом появляются манипуляторы Безье. Горб кривой выгибается в сторону, противоположную направлению буксировки. Поэтому, если хотите, чтобы кривая выгибалась вверх, буксируйте вниз.



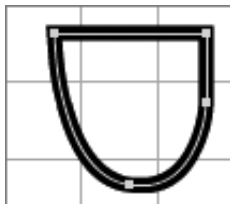
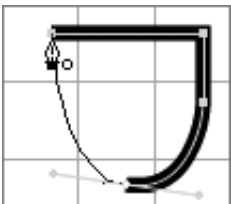
**Рис. 2.48.** Чтобы кривая выгибалась вверх, буксируйте мышью вниз. Чем дальше манипулятор от узла, тем круче кривая.



**Рис. 2.49.** Чтобы изменить форму кривой, перемещайте мышью вокруг узла по или против часовой стрелки (верхний рисунок). Чтобы сделать кривую более крутой или более пологой, отдайте или приблизьте манипуляторы к узлу (нижний рисунок).



**Рис. 2.50.** Выбрав подходящее положение манипуляторов, отпустите кнопку мыши. Flash добавит новый сегмент в путь.



**Рис. 2.51.** Чтобы создать замкнутую форму, укажите курсором на первый узел. Когда появится кружочек, щелкните по этой точке (левый рисунок). Flash добавит последний сегмент, и путь замкнется (правый рисунок).

9. Тяните курсор в сторону, противоположную той, куда должна выгибаться кривая. Манипуляторы Безье исходят из узла и в процессе буксировки движутся в противоположные стороны. Пока вы буксируете мышью, Flash рисует предварительную кривую (рис. 2.48).

10. Не отпуская кнопку мыши, потяните ее, чтобы сместить манипулятор Безье. Перемещение вокруг узла изменяет направление горба; при отдалении от узла кривая становится круче (рис. 2.49).

11. Когда кривая примет нужную вам форму, отпустите кнопку мыши. Flash нарисует кривую в окончательном виде и добавит ее в путь (рис. 2.50).

12. Чтобы создать незамкнутый путь, щелкните мышью по арене, удерживая клавишу **⌘** (Mac) или **Ctrl** (Windows).

или

Чтобы создать замкнутый путь, выполните следующие действия:

- укажите курсором на первый узел;
- Flash нарисует предварительный замыкающий сегмент. Рядом с кончиком пера появится кружочек (рис. 2.51);
- щелкните по первому узлу;
- Flash замкнет форму, добавив последний сегмент в путь.

Как только путь будет полностью нарисован, рядом с пером вновь возникнет буква **x**, означающая, что инструмент готов к помещению первого узла нового пути.

С

Есть и другие способы закончить рисование незамкнутых путей. Выполните команду **Edit** ⇒ **Deselect All** (Редактирование ⇒ Отменить выбор всего) или нажмите комбинацию клавиш **⌘-Shift-A** (Mac) либо **Ctrl-Shift-A** (Windows). На панели инструментов щелкните по какому-нибудь другому инструменту. Можно также дважды щелкнуть по последней добавленной точке. Чтобы воспользоваться этим приемом, путь должен завершаться прямолинейным сегментом, не содержащим манипуляторов Безье.

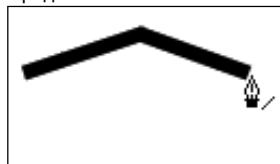
С

В прежних версиях Flash автоматически заливал нарисованную пером форму текущим цветом заливки. Во Flash CS3 для этого необходимо выбрать инструмент «Ведро с краской».

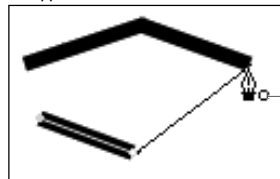
С

С помощью пера можно добавлять сегменты к ранее нарисованной линии (рис. 2.52). Чтобы продолжить исходную линию, подведите курсор к ее последней точке (*оконечному узлу*). Рядом с кончиком пера появится модификатор — косая черта. Если теперь щелкнуть по оконечному узлу, то перо привяжется к нему, как если бы узел была только что поставлен. Продолжайте добавлять сегменты, как в предыдущем упражнении. Можно также соединить новую линию с уже существующей. Ставьте узлы новой линии одиночным щелчком, но не щелкайте дважды в конце линии. Вместо этого укажите курсором на один из оконечных узлов той линии, с которой хотите соединить новую. Рядом с пером появится модификатор. Если старая линия была нарисована в режиме комбинирования форм, то модификатор выглядит как кружок, а если в режиме рисования объектов — как звено цепи. После щелчка по старому оконечному узлу Flash соединит две линии.

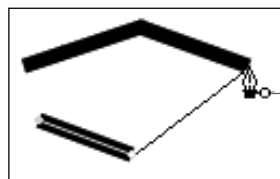
Продолжить линию



Соединить линии



Модификатор при соединении с линией, созданной в режиме рисования объектов



Модификатор при соединении с линией, созданной в режиме комбинирования форм

**Рис. 2.52.** С помощью пера можно добавлять сегменты к уже нарисованной линии. Укажите курсором на оконечный узел существующей линии и щелкните мышью (верхний рисунок), затем продолжайте щелкать для добавления новых узлов. А можете сначала разместить все новые узлы, а потом указать курсором на оконечный узел старой линии (нижний рисунок). Когда рядом с пером появится модификатор, щелкните мышью, чтобы соединить линии.

### О математике путей во Flash

Путь — последовательность точек и соединяющих их линий — это основа графического объекта. При работе с большинством инструментов математические операции, необходимые для создания путей, производятся за кулисами. Вы рисуете линию или форму, а Flash размещает точки (не показывая), соединяет их и добавляет обводку. При работе с пером вы сами ставите опорные точки (они называются *узлами*) и выстраиваете соединительные сегменты с помощью управляющих *манипуляторов Безье* (или *касательных манипуляторов*). Когда все точки будут поставлены, Flash материализует путь, добавляя к нему обводку.

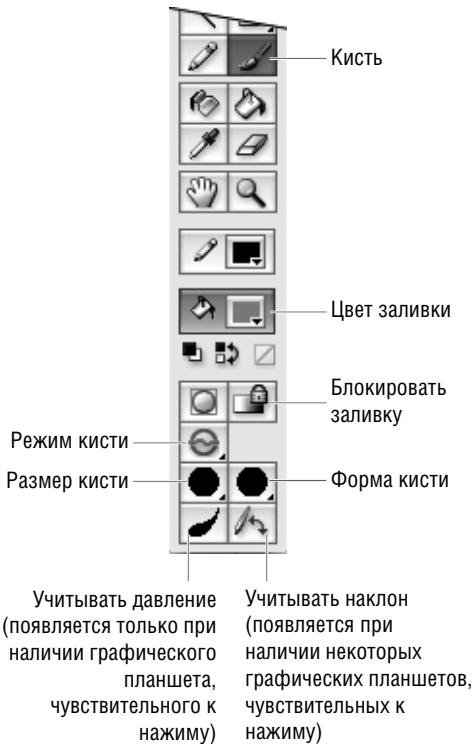


Рис. 2.53. Кисть и ее модификаторы.

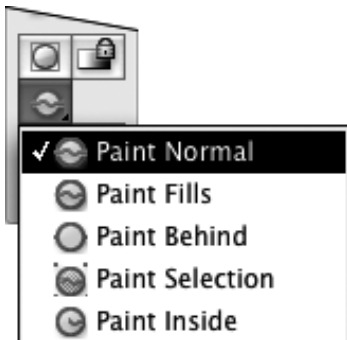


Рис. 2.54. Меню Brush Mode позволяет определить, как кисть должна взаимодействовать с другими формами в том же слое. Если выбран пункт Paint Normal, то заливки, создаваемые в режиме комбинирования форм, ведут себя при перекрытии с другими формами так же, как все остальные заливки.

### Чтобы создать произвольную сплошную заливку с помощью кисти:

1. На панели инструментов выберите кисть или нажмите клавишу **B** (рис. 2.53).
2. Чтобы оптимизировать работу кисти для решения конкретной задачи, настройте следующие параметры в нижней части панели инструментов:

- в меню **Brush Size** (Размер кисти) выберите толщину кончика кисти;
- в меню **Brush Shape** (Форма кисти) выберите форму кончика кисти;
- в меню **Brush Mode** (Режим кисти) выберите режим закрашивания; в данном упражнении укажите режим Paint Normal (Нормальная закразка) (рис. 2.54). Режим закрашивания определяет, что происходит, когда кисть проходит над другими формами.

Перекрывающиеся формы взаимодействуют по-разному в зависимости от того, созданы они в режиме комбинирования форм, рисования объектов или являются примитивами (дополнительную информацию о взаимодействии перекрывающихся форм см. в главе 5);

- чтобы при окрашивании заливки толщина кисти изменялась, отметьте флажок **Use Pressure** (Учитывать давление). Он появляется только в том случае, когда к компьютеру подключен графический планшет.

3. Любым из рассмотренных выше способов выберите или определите цвет сплошной заливки.
4. Переместите курсор на арену. Он примет форму, отражающую текущий размер и форму кисти.

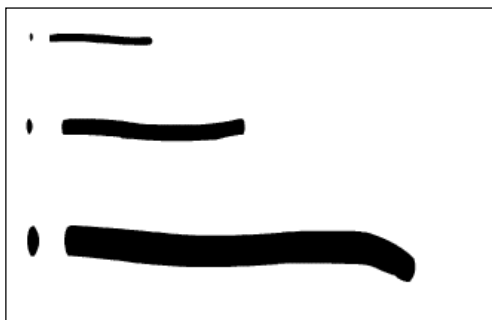
5. Щелкните мышью и начинайте рисовать на арене. Flash отображает предварительный результат с учетом выбранных параметров заливки (рис. 2.55).
6. Закончив рисовать форму, отпустите кнопку мыши. Flash создает окончательную форму, выполняя сглаживание в соответствии с параметрами, заданными в инспекторе свойств (см. врезку «Параметры сглаживания кисти» ниже).

## С

Толщину мазка можно изменить, выбрав другой масштаб арены. Чтобы создать толстый мазок, не изменяя настроек самой кисти, выберите мелкий масштаб. Для перехода к тонкому мазку увеличьте масштаб (рис. 2.56). Обязательно посмотрите, как выглядит результат в масштабе 100%.



**Рис. 2.55.** При рисовании кистью показывается предварительный вид формы (левый рисунок). По завершении Flash строит окончательное векторное изображение с учетом выбранного цвета заливки и параметров сглаживания (правый рисунок).



**Рис. 2.56.** Эти три мазка созданы при одном и том же размере кисти, но разным масштабе арены.

### Об инструменте «Кисть»

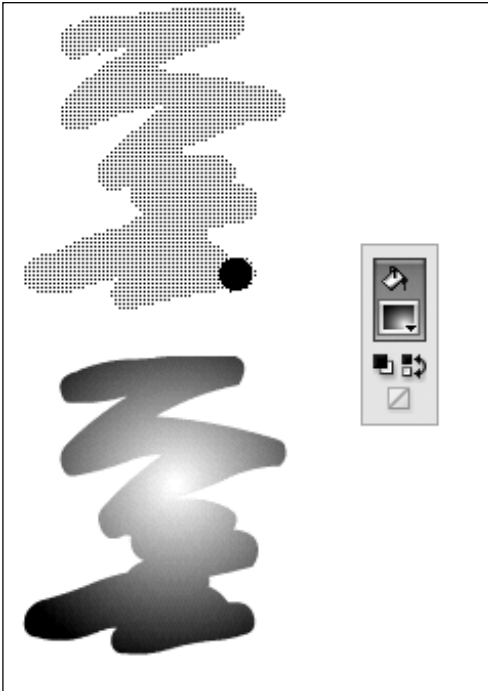
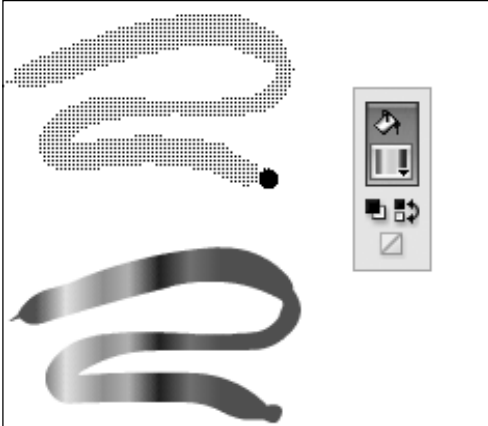
Во Flash с помощью кисти можно наносить цветные мазки. Получающаяся форма представляет собой произвольную заливку без обводки. Этот инструмент имитирует обычную малярную кисть или маркер. Разнообразие размеров и форм кисти позволяет придавать рисункам более профессиональный вид. При наличии графического планшета, чувствительного к нажиму, кисть может создавать линии переменной толщины, имитируя действия реального художника. Чем сильнее вы давите на планшет, тем толще получается линия (рис. 2.57).



**Рис. 2.57.** Задание модификатора Use Pressure активирует механизм восприятия давления, встроенный в графический планшет. Для изменения толщины рисуемой линии давите на планшет сильнее или слабее. Все линии на изображении котенка проведены кистью одного и того же размера и формы.



**Рис. 2.58.** Для окрашивания с разблокированным градиентом сделайте кнопку **Lock Fill** неактивной.



**Рис. 2.59.** Форма, нарисованная кистью с разблокированной линейно-градиентной заливкой (верхний рисунок) и радиально-градиентной заливкой (нижний рисунок).

### Чтобы окрашивать с градиентом:

1. Выполните шаги 1 и 2 из предыдущего упражнения.
2. Любым из описанных выше способов выберите или определите линейно или радиально-градиентную заливку.
3. Чтобы заблокировать или разблокировать градиент, выполните одно из следующих действий:

- для создания формы, содержащей полный спектр цветов градиента, сделайте неактивной кнопку **Lock Fill** (Заблокировать заливку) на панели инструментов (рис. 2.58);
- для создания формы, в которой присутствует лишь часть цветов градиента (как если бы она была частью арены, целиком покрашенной выбранным градиентом), сделайте кнопку **Lock Fill** активной.

4. Красьте кистью, как описано в шагах 4-6 предыдущего упражнения.

Flash не может показать предварительный вид формы, окрашенной градиентной заливкой, поэтому она рисуется черно-белой. По завершении форма перерисовывается с учетом текущего режима блокировки и цвета заливки (см. врезку «Градиенты и режимы рисования во Flash» ниже).

Если заливка не заблокирована, то в форме представлены все цвета градиента (рис. 2.59).

Если заливка заблокирована, то цвета градиента представлены в форме лишь частично (рис. 2.60).

**С**

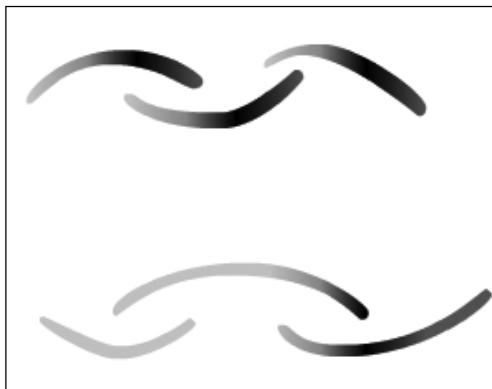
Попробуйте нанести мазки в разных режимах рисования; воспользуйтесь заблокированными и разблокированными градиентами; рисуйте в разных местах арены; создайте формы разной длины. Обратите внимание, как в каждой форме выглядит градиент.

**С**

Если для первого градиента, которым вы красили в текущем сеансе, заливка была заблокирована, то Flash совместит центр градиента с левым краем арены. Следовательно, половина градиента окажется в области монтажного стола, а не арены. Чтобы выбрать другое положение центра, сначала воспользуйтесь кистью с разблокированным градиентом, который центрируется так, как вы захотите. Если теперь переключиться в режим блокировки, то заблокированный градиент будет выровнен с последним использованным разблокированным градиентом. После этого ненужную форму, нарисованную с разблокированным градиентом, можно удалить или с помощью инструмента «Ведро с краской» преобразовать градиент в заблокированный (см. главу 4).

**С**

После того как форма с градиентной заливкой создана, положение центра градиента можно изменить, воспользовавшись инструментом преобразования градиента (см. главу 4).



**Рис. 2. 60.** Если кнопка Lock Fill неактивна, то каждый мазок кистью содержит весь спектр цветов текущего градиента (верхний рисунок); если же она активна, то в мазке присутствует лишь часть цветов градиента в предположении, что им закрашена вся область, занятая ареной и монтажным столом (нижний рисунок).



### Градиенты и режимы рисования во Flash

Когда форма создается кистью с разблокированной градиентной заливкой, не имеет значения, какой режим рисования установлен: комбинирования форм или рисования объектов. Flash размещает центр градиента в центре минимального охватывающего прямоугольника формы (невидимого), и в форме представлен весь спектр цветов градиента.

Но если точно такая же форма рисуется с заблокированной заливкой, то формы, созданные в разных режимах рисования, ведут себя по-разному.

Когда заливка заблокирована, Flash создает виртуальный градиент, распространяющийся на арену и монтажный стол. (По умолчанию центр заблокированного градиента находится на левом краю арены.) В форме, созданной с заблокированной заливкой, видна лишь часть градиента, попадающая в занятую формой часть арены. Если заблокировать заливку и нарисовать несколько форм в режиме комбинирования, то всем формам сопоставляется один и тот же виртуальный градиент; инструмент преобразования градиента будет сдвигать его сразу во всех таких формах. У каждой формы, созданной в режиме рисования объектов, может быть собственный виртуальный градиент. Все эти виртуальные градиенты центрируются одинаково, поэтому вначале складывается впечатление, что градиент общий, как и в режиме комбинирования форм. Однако инструмент преобразования градиента сдвигает градиент для каждого объекта-рисунка индивидуально.

### Параметры сглаживания кисти

Flash позволяет вам управлять тем, как мазки преобразуются в векторные формы. Для этого на вкладке свойств кисти в инспекторе свойств введите значение в поле Smoothing (Сглаживание). Этот параметр определяет, насколько точно Flash должен транслировать каждое движение кистью в отдельные векторы. Диапазон значений параметра — от 0 до 100, по умолчанию принимается 50. Чем меньше значение, тем скрупулезнее Flash воспроизводит рисуемую форму (для этого требуется больше векторов, а, стало быть, растет размер файла и снижается скорость анимации). Если значение велико, то у Flash остается больше свободы для аппроксимации, и число векторов оказывается меньше.

Чтобы почувствовать разницу, назначьте кисти сглаживание 1 и нарисуйте кривую на арене. Теперь измените значение на 30 и нарисуйте вторую кривую. С помощью инструмента «Спецвыделение» (см. главу 4) выберите сначала одну, а потом другую форму. Вы увидите, что в линии, нарисованной со сглаживанием 1, точек гораздо больше, то есть она состоит из большего числа векторных сегментов.

## Добавление обводок и заливок

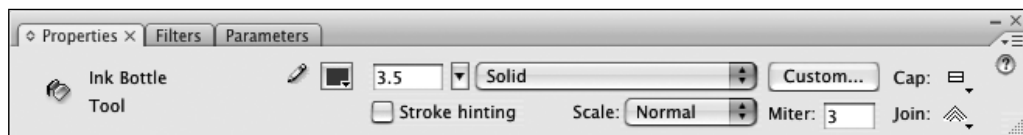
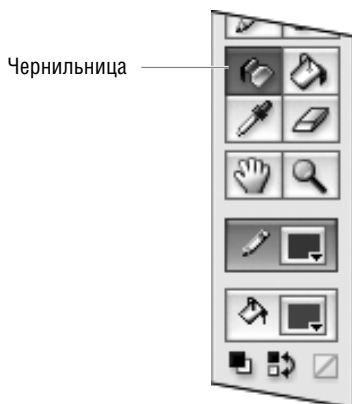
В этой главе вы научились пользоваться инструментами «Линия», «Карандаш», «Перо» и различными геометрическими фигурами для создания контуров — обводок без заливки. Для создания залитых форм без обводящего контура вы применяли кисть, перо и геометрические фигуры. В любой момент к созданной форме можно добавить недостающий элемент. Инструмент «Чернильница» добавляет обводку к чистой заливке, а инструмент «Ведро с краской» вставляет заливку внутрь обводки. (Эти же инструменты можно применять для модификации обводок и заливок; см. главу 4.)

### Чтобы добавить обводку к заливке:

1. Нарисуйте на арене заливку без обводки или возьмите уже существующую не обведенную форму.

Если установлен режим комбинирования форм, разблокируйте заливку; в режиме рисования объектов это несущественно.

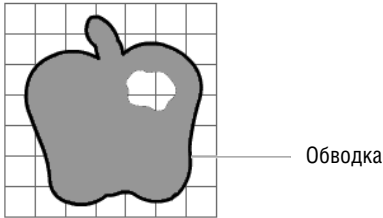
2. На панели инструментов выберите чернильницу или нажмите клавишу S (рис. 2.61).
3. На вкладке свойств чернильницы в инспекторе свойств задайте атрибуты обводки (см. раздел «Задание атрибутов обводки» выше).
4. Наведите курсор на какую-нибудь заливку без обводки, он примет форму бутылочки, из которой льются чернила.



**Рис. 2.61.** Инструмент «Чернильница» применяет атрибуты обводки, заданные на соответствующей ему вкладке в инспекторе свойств.

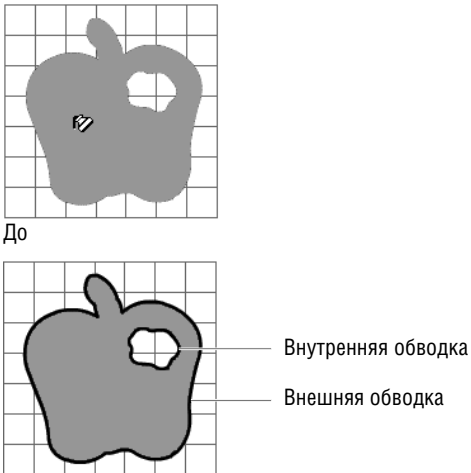


До

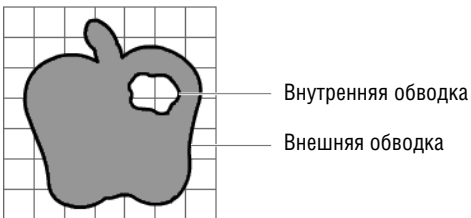


После

**Рис. 2.62.** Когда чернильница находится над заливой формой, ее активная точка выглядит как белое пятно на кончике выливающейся капли. Чтобы добавить обводку вдоль внешнего края заливки, подведите активную точку к этому краю (верхний рисунок) и щелкните мышью. Flash добавит обводку с учетом атрибутов, заданных на вкладке свойств чернильницы в инспекторе свойств (нижний рисунок).



До



После

**Рис. 2.63.** Подведите активную точку к середине заливой формы (верхний рисунок) и щелкните мышью. Flash добавит обводки вдоль внешнего и внутреннего края с учетом текущих атрибутов.

5. Кончиком чернильницы щелкните по форме-заливке одним из следующих способов:

- чтобы добавить обводку вдоль внешнего края формы, щелкните вблизи внешнего края (рис. 2.62);
- чтобы добавить обводку вдоль внутреннего края формы с «дыркой», щелкните вблизи внутреннего края;
- чтобы обвести и внешний край, и «дырку», щелкните в середине формы (рис. 2.63);

Flash добавит заказанные обводки с учетом текущих цвета, толщины и стиля, заданных в инспекторе свойств.

**С** Flash допускает градиентные обводки. На шаге 3 предыдущего упражнения задайте на вкладке **Properties** инспектора свойств линейный или радиальный градиент, выбрав его из набора образцов, ассоциированного с элементом управления цветом обводки. Зачем может понадобиться градиентная обводка? Например, добавив толстую обводку к овальной форме, можно создать иллюзию трехмерности или впечатление, что форма переливается.

**С** Есть и другой способ «добавить» недостающую обводку к объектам-рисункам и примитивам — модифицировать их с помощью инспектора свойств. Сначала выберите объект-рисунок или примитив на арене. Затем в инспекторе свойств измените свойства обводки, в том числе цвет. Этот прием можно применять для добавления обводок к нескольким объектам-рисункам или примитивам (но для форм, созданных в режиме комбинирования, он не годится). Дополнительную информацию о модификации графических объектов см. в главе 4.

**Чтобы залить контур сплошным цветом:**

1. Нарисуйте на арене контур-обводку без заливки или выберите уже существующую форму. Контур может быть выделен или не выделен.
2. На панели инструментов выберите ведро с краской или нажмите клавишу **K** (рис. 2.64).
3. В меню **Gap Size** (Размер промежутка) укажите, какой объем помощи вы хотели бы получить от Flash (рис. 2.65).

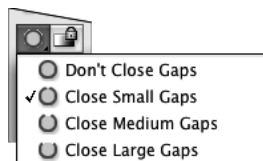
Если вы рисуете точно, то лучше выбрать режим заполнения малых или средних лакун; вряд ли вам нужно, чтобы автоматически заливались области, которые не должны быть частью формы. Если точно рисовать не получается, выберите режим **Close Large Gaps** (Закрывать крупные промежутки). Этот параметр позволяет Flash распознавать недорисованные формы.

4. С помощью любого элемента управления цветом заливки выберите сплошной цвет.
5. Укажите активной точкой ведра с краской (капелька на конце струи) куда-нибудь внутрь контура (рис. 2.66).
6. Щелкните мышью.

Форма будет закрашена текущим цветом заливки (рис. 2.67).



**Рис. 2.64.** Инструмент «Ведро с краской» и его модификаторы.



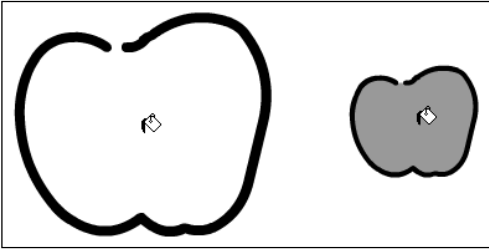
**Рис. 2.65.** Меню Gap Size управляет механизмом достраивания плохо прорисованных форм.



**Рис. 2.66.** Активной точкой ведра с краской является капелька на конце выливающейся струи. Она становится белой, когда ведро проходит над темными областями.



**Рис. 2.67.** Щелчок инструментом «Ведро с краской» внутри контура (верхний рисунок) закрашивает форму текущим цветом заливки (нижний рисунок).



**Рис. 2.68.** Ведро с краской не может залить это яблоко, когда выбран режим **Close Large Gaps**, а масштаб равен 100% (левый рисунок). Но при масштабе 50% и том же режиме заполнения лакун форма считается замкнутой и заливается.



Сделать кнопку Lock Fill неактивной

**Рис. 2.69.** Чтобы залить форму незаблокированным градиентом, сделайте неактивной кнопку Lock Fill.

С

Вы можете и не подозревать о том, что в нарисованной форме есть лакуны. Если при щелчке инструментом «Ведро с краской» внутри формы ничего не происходит, попробуйте изменить параметр Gap Size, как описано в шаге 3.

С

Значение параметра, определяющего режим заполнения лакун, вычисляется относительно масштаба арены. Если при текущем масштабе лакуны не заполняются даже при наибольшем значении параметра, попробуйте уменьшить масштаб (рис. 2.68).

С

Еще один способ «добавить» недостающую заливку к объектам-рисункам и примитивам — модифицировать их с помощью элемента управления цветом заливки. Сначала выберите объект-рисунок или примитив на арене. Затем в инспекторе свойств либо на панели инструментов или цветов выберите элемент управления цветом заливки. Этот прием можно применять для добавления заливок к нескольким объектам-рисункам или примитивам (но для форм, созданных в режиме комбинирования, он не годится). Дополнительную информацию о модификации графических объектов см. в главе 4.

### Чтобы залить контур разблокированным градиентом:

1. На панели инструментов выберите ведро с краской.
2. На панели цветов определите новый градиент (см. раздел «Сплошные цвета и градиенты» выше).

или

Пользуясь любым элементом управления цветом заливки, выберите существующий линейный или радиальный градиент (см. раздел «Задание атрибутов заливки» выше).

3. Убедитесь, что кнопка **Lock Fill** на панели инструментов неактивна (рис. 2.69).

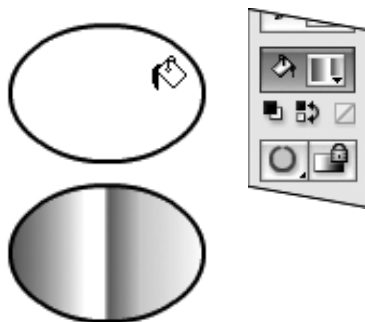
4. Повторите шаги 5 и 6 из предыдущего упражнения.

Любой контур, внутри которого вы щелкаете, заливается градиентом, выбранным с помощью элемента управления цветом заливки. Если на шаге 2 был выбран линейный градиент, то Flash располагает его центр внутри контура (рис. 2.70). В случае радиального градиента центр будет находиться там, где была активная точка ведра с краской в момент щелчка (рис. 2.71).

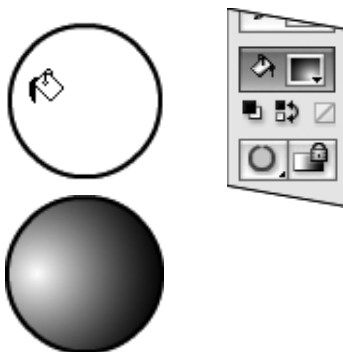
### Чтобы залить контур заблокированным градиентом:

1. Выполните шаги 1 и 2 из предыдущего упражнения.
2. На панели инструментов активируйте кнопку **Lock Fill**.
3. Повторите шаги 5 и 6 из упражнения «Чтобы залить контур сплошным цветом».

Любой контур, внутри которого вы щелкаете, заливается частью градиента, выбранного с помощью элемента управления цветом заливки. (См. врезку «Градиенты и режимы рисования во Flash» выше в этой главе).



**Рис. 2.70.** Для заливки линейным градиентом можно воспользоваться ведром с краской. Центр разблокированного градиента помещается внутри прямоугольника, охватывающего контур.



**Рис. 2.71.** Ведро с краской годится и для заливки радиальным градиентом. Щелкните в том месте, где должен находиться центр градиента.

### Издержки градиентов

Градиенты придают графике блеск, но при этом увеличивается размер файла, а, значит, замедляется загрузка ролика. Для каждой области градиентной заливки требуется 50 байтов, не нужных в случае сплошной заливки.

Кроме того, для визуализации градиента расходуется процессорное время. Если градиентов слишком много, то частота смены кадров уменьшится, а это приведет к замедлению анимации в готовом ролике.

Инструмент «Текст» в Adobe Flash CS3 Professional создает не только графические объекты в виде литер, но и текстовые поля, причем в среде разработки их содержимое можно редактировать без ограничений. При создании текстового элемента необходимо решить, как он будет использоваться в готовом ролике, и назначить ему тип. Если вы хотите, чтобы конечный пользователь мог взаимодействовать с полем (например, вводить в него персональную информацию), то присвойте тип *ввод*. Если нужно обновлять текст во время выполнения программы (например, с помощью ActionScript загрузить и показать счет баскетбольного матча на спортивном сайте) присвойте тип *динамическое*. Если нужно просто показать текст, присвойте тип *статическое*.

Во Flash CS3 используется механизм рендеринга текста Flash Type, который делает шрифты более легкими для восприятия, особенно при небольших размерах. Это видно и в среде разработки на этапе создания ролика, и конечно-му пользователю, просматривающему ролик в проигрывателе Adobe Flash Player версии 8 или более поздней (см. главу 17).

В настоящей главе вы научитесь работать со статическим текстом. Для манипуляций с динамическими полями и полями ввода необходимо владеть языком ActionScript лучше, чем предполагается в этой книге. Если вам это интересно, обратите внимание на издание *Adobe Flash CS3 Professional; Visual QuickPro Guide* (Peachpit Press).

## Инструмент «Текст»

Инструмент «Текст» позволяет создавать блоки редактируемого текста. Текст можно располагать как горизонтально, так и вертикально. К тексту применимы разнообразные атрибуты, в том числе стили текста и абзацев.

### Чтобы создать однострочный текст как графический элемент:

1. На панели инструментов выберите инструмент «Текст» или нажмите клавишу T (рис. 3.1).

Для этого упражнения не трогайте текущие параметры шрифта и стилей. О том, как их изменять, будет рассказано в следующих упражнениях.

2. Переместите курсор на арену. Он примет форму перекрестья с буквой T в правом нижнем углу (рис. 3.2).

3. Щелкните в том месте арены, где должен начинаться текст. Flash создает охватывающий прямоугольник переменного размера, в котором точка вставки очередного символа обозначается мигающей чертой (рис. 3.3).

В каждом углу прямоугольника имеется манипулятор, за который можно потянуть, чтобы изменить размер. Круглый манипулятор в правом верхнем углу означает, что для данного поля режим переноса слов не включен.

4. Начните вводить текст. Охватывающий прямоугольник будет расширяться, подстраиваясь под длину текста (рис. 3.4).

5. Закончив печатать, щелкните в какой-нибудь точке арены (или выберите другой инструмент), чтобы отменить выбор текста.

Flash убирает охватывающий прямоугольник и оставляет на экране только текст. Если щелкнуть по нему инструментом «Стрелка», то текст станет выбранным объектом, для которого доступны операции перемещения, изменения размера и атрибутов.



Рис. 3.1. Выберите инструмент «Текст», чтобы приступить к созданию текстовых полей на арене.



Рис. 3.2. Курсор, соответствующий инструменту «Текст».



Рис. 3.3. Для создания текстового поля щелкните инструментом «Текст» по арене. Круглый манипулятор означает, что для данного поля режим переноса слов не включен.

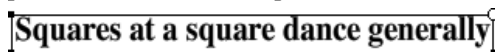
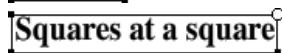
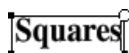
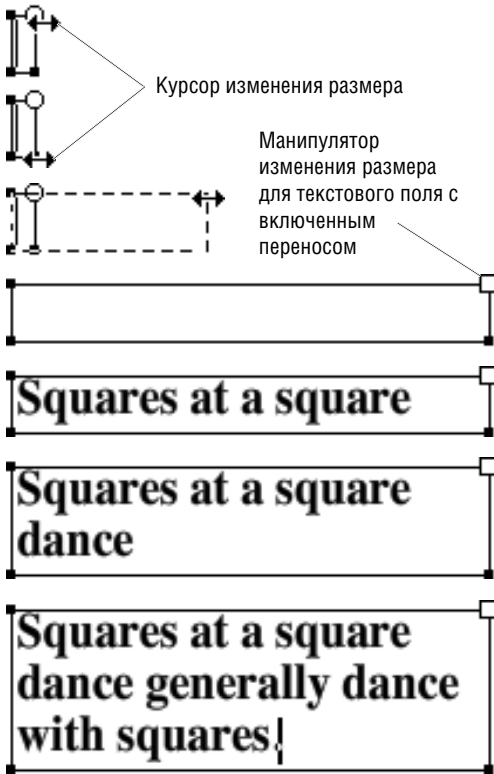


Рис. 3.4. В процессе ввода поле расширяется по горизонтали, подстраиваясь под длину текста. Переноса на новую строку не происходит.





**Рис. 3.5.** Щелкните мышью и потяните за любой манипулятор, чтобы создать текстовое поле фиксированной ширины. Правый верхний манипулятор примет форму квадратика, говорящего о том, что вводимый текст не будет по ширине выступать за пределы охватывающего прямоугольника. Поле будет расти только в высоту.

### Чтобы создать текст фиксированной ширины с переносом слов:

1. Выбрав инструмент «Текст», щелкните в том месте арены, где текст должен начинаться.
2. Подведите курсор к любому манипулятору изменения размера. Курсор примет форму двунаправленной стрелки.
3. Буксируйте манипулятор, пока ширина охватывающего прямоугольника не покажется достаточной (рис. 3.5). Манипулятор в правом верхнем углу превращается в квадратик; это говорит о том, для данного поля включен режим переноса слов.
4. Отпустите кнопку мыши. Внутри поля появится мигающая черта, обозначающая точку вставки.
5. Вводите текст.

Flash разбивает текст на строки, так чтобы не выходить за пределы области, ограниченной по ширине охватывающим прямоугольником.

**С**

Чтобы переместить текстовое поле, не меняя инструмента, наведите курсор на край охватывающего прямоугольника. Курсор примет форму стрелки выбора. Теперь можно двигать поле в любом направлении.

**С**

Чтобы восстановить исходное состояние текстового поля (с отключенным переносом), дважды щелкните по квадратному манипулятору в правом верхнем углу охватывающего прямоугольника. Он снова превратится в кружочек, говорящий о том, что поле может расширяться по горизонтали.

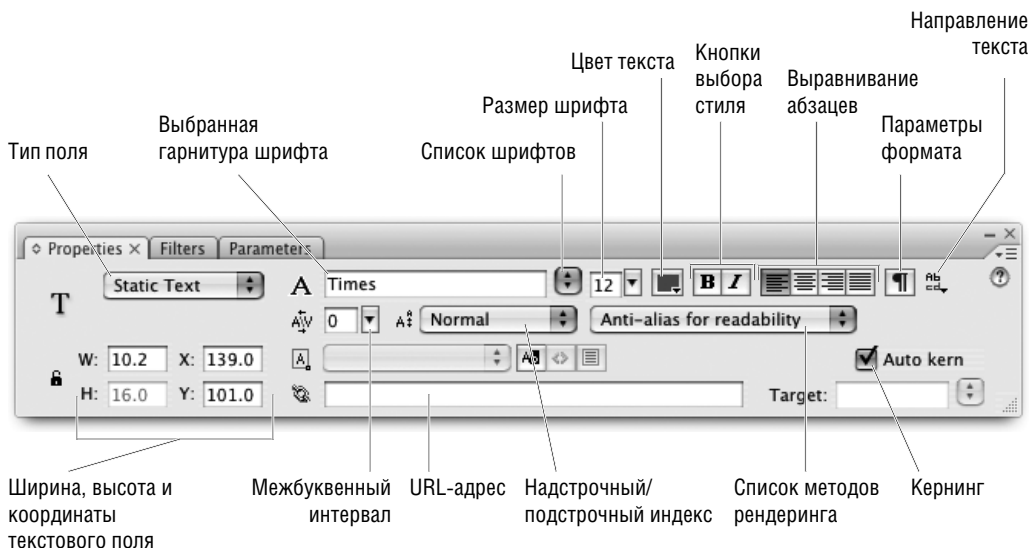
## Задание атрибутов текста

Атрибуты текста можно задать на контекстно-зависимой вкладке **Properties** в инспекторе свойств. Если выбран инструмент «Текст», на ней появляются элементы для задания шрифта, размера, стиля, межбуквенного интервала, межстрочного интервала и цвета, для управления *трекингом* (промежутки между буквами и словами в блоке выделенного текста), установки режима надстрочных или подстрочных индексов и создания гиперссылки с текста на URL. При выборе существующего текстового поля вместо метки с названием инструмента на вкладке отображаются все описанные выше атрибуты текста плюс атрибуты самого поля — размеры и координаты на арене. Когда нужно будет отличить эти две вкладки друг от друга, мы в этой книге будем говорить о *вкладке свойств инструмента «Текст»* и о *вкладке свойств текста*. Если же безразлично, о какой вкладке идет речь, то будем говорить о *вкладке текста*.

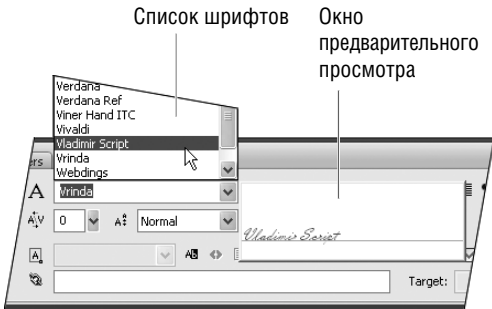
Атрибуты можно задать заранее, тогда во время ввода они будут применяться автоматически. А можно изменить атрибуты уже введенного текста. Инструмент «Текст» всегда пользуется теми атрибутами, которые заданы на вкладке текста (рис. 3.6).

При выполнении следующих упражнений не закрывайте панель инспектора свойств (если она еще не открыта, выберите из меню пункт **Window ⇒ Properties ⇒ Properties**).

**С** Атрибуты текста можно задавать не только на вкладке. Гарнитуру шрифта и его размер, стиль, выравнивание абзацев и трекинг можно задать с помощью меню **Text** (Текст). Оно позволяет как изменить свойства выбранного текста, так и загрузить свойства в инструмент «Текст».



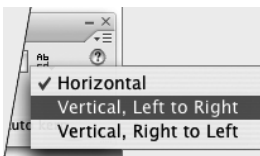
**Рис. 3.6.** При выборе инструмента «Текст» (или выборе текстового блока на арене) на вкладке **Properties** в инспекторе свойств отображаются атрибуты, которые будет применять инструмент (или которые будут применены к уже существующему тексту).



**Рис. 3.7.** Когда курсор перемещается по списку шрифтов на вкладке текста, в окне предварительного просмотра отображается текущий шрифт.

### Вертикальный текст

Статические текстовые поля позволяют создавать текст, расположенный в одну вертикальную колонку или перетекающий из одной колонки в другую. Текст может читаться слева направо (как английский) или наоборот (как японский). Когда активен инструмент «Текст», на вкладке **Properties** в инспекторе свойств, можно задать направление текста, выбрав его из списка **Change Orientation of Text** (Изменить ориентацию текста) (рис. 3.8). С вертикальными текстовыми полями можно работать так же, как с горизонтальными, в том числе задавать режим переноса; НО в этом случае слово переносится в соседнюю колонку, а не на следующую строку. При вводе текста символы размещаются друг под другом до тех пор, пока не будет достигнута нижняя граница охватывающего прямоугольника, а затем текст переходит в следующую колонку. Можно принудительно перейти к следующей колонке, введя символ конца абзаца.



**Рис. 3.8.** Если выбран инструмент «Текст» или текстовое поле, то на вкладке **Properties** отображается список направлений текста.

### Чтобы выбрать текст, к которому будут применяться атрибуты символов:

Выполните одно из следующих действий:

- выбрав инструмент «Текст», буксируйте мышью по существующему тексту, чтобы выделить его фрагмент;
- выбрав инструмент «Текст», щелкните по текстовому полю, чтобы выделить весь текст.

**С**

Инструмент «Стрелка» позволяет выбрать несколько текстовых полей и модифицировать все сразу. (Подробнее о выделении мы рассмотрим в главе 4.)

**С**

Параметры чувствительности к контакту, которые задаются в категории **General** диалогового окна **Preferences**, применимы также к выделению текстовых полей (о том, как выполняется такое выделение, см. главу 4).

### Чтобы выбрать один из установленных шрифтов:

1. Выберите текст, который хотите модифицировать.
2. На вкладке свойств текста в инспекторе свойств раскройте список шрифтов. Одновременно с раскрытием списка появляется окно предварительного просмотра шрифта (рис. 3.7).
3. Когда вы наводите курсор на элемент списка, имя соответствующего шрифта отображается в окне предварительного просмотра.
4. Щелкните по имени подсвеченного шрифта и закройте список. Имя шрифта появляется в поле **Font**, а Flash заменяет шрифт выделенного текста.

C

Можно также напрямую ввести имя шрифта в поле на вкладке текста. Это поле не чувствительно к регистру, но все равно печатать надо аккуратно. Если вы сделаете ошибку, то Flash подумает, что указанный шрифт не установлен, и подставит вместо него шрифт по умолчанию.

C

Шрифт можно также выбрать с помощью меню **Text ⇒ Font** (Текст ⇒ Шрифт).

C

Можно разрешить конечным пользователям копировать текст из статических полей. На этапе разработки выделите текст, который пользователям разрешено копировать, и на вкладке свойств текста щелкните по значку **Selectable Text** (Выбираемый текст) (это значок *Ab*, расположенный прямо под списком для создания надстрочных и подстрочных индексов).

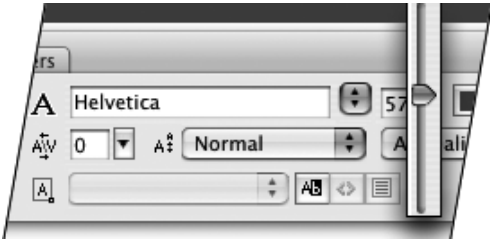
### Шрифты устройства

Назначая статическому полю один из установленных шрифтов, вы обычно хотите, чтобы при воспроизведении ролика текст выглядел точно так же, как на этапе разработки. Но откуда вам знать, установлен ли этот шрифт на компьютере конечного пользователя? Чтобы статический текст отображался в проигрывателе, как задумано, Flash сохраняет информацию о начертании букв в публикуемом SWF-файле (о публикации см. главу 17). Во время воспроизведения Flash Player использует сохраненную информацию для правильного отображения букв. Однако из-за этого растет размер SWF-файла. Шрифты устройства позволяют от нее избавиться. Задать шрифт устройства можно двумя способами: выбрать из списка шрифтов или установить режим **Use Device Fonts** (Использовать шрифты устройства) в списке методов рендеринга шрифта на вкладке текста в инспекторе свойств.

Выбирая шрифт устройства из списка шрифтов, вы сохраняете некоторый контроль над тем, как будет видеть текст пользователь, но отказываетесь от задания конкретного шрифта. В списке на вкладке текста и в меню **Text ⇒ Font** есть три шрифта устройства: **\_sans**, **\_serif** и **\_typewriter**. При воспроизведении ролика, содержащего статический текст, для которого был задан шрифт устройства, система конечного пользователя подставит наиболее близкий шрифт такого же типа. Если был выбран шрифт **\_serif**, подставляется шрифт с засечками (тонкие отрезки на концах литер, например, в шрифте Times Roman). Вместо шрифта **\_sans** (сокращение от *sans serif*, т.е. «без засечек») подставляется более простой шрифт без всяких украшений, примерами могут служить шрифты Helvetica и Arial. Шрифт **\_typewriter** заменяется моноширинным (т.е. все буквы имеют одинаковую ширину, как в шрифте пишущей машинки), пример — шрифт Courier.

Если выбрать конкретный шрифт для статического текста, а затем в списке методов рендеринга шрифта выставить режим **Use Device Fonts**, то система конечного пользователя получит максимальную свободу в выборе подставляемого шрифта. Она будет искать наилучшее соответствие среди установленных шрифтов, но нет гарантии, что подставит шрифт того же стиля, который был выбран вами при разработке.

Хотя применение шрифтов устройств позволяет уменьшить размер SWF-файла, результат часто получается непрезентабельным. Если вы собираетесь прибегнуть к этому методу, обязательно протестируйте ролик в разных системах, чтобы представлять себе, с чем могут столкнуться пользователи. Возможно, экономия на размере не стоит потери качества.



**Рис. 3.9.** Буксируя ползунок, можно интерактивно изменять размер шрифта выделенного на арене текста.

### Чтобы задать размер шрифта:

1. Выберите текст, который собираетесь модифицировать.
  2. На вкладке текста в инспекторе свойств дважды щелкните по полю размера шрифта (или щелкните один раз и буксируйте ползунок), чтобы выделить текущее значение.
  3. Введите требуемый размер шрифта в пунктах.
  4. Нажмите **Enter**.
- или*
5. Щелкните по треугольничку справа от поля размера шрифта. Появится ползунок (рис. 3.9).
  6. Буксируя ползунок, выберите значение между 8 и 96 пунктами. Flash показывает образец шрифта на арене.
  7. Щелкните мышью вне ползунка, чтобы зафиксировать новый размер шрифта.

### Антиальясинг

Антиальясинг (сглаживание) — это метод рендеринга (рисования прямых и кривых линий), при котором края делаются более плавными. В случае текста это означает, что буквы получаются слегка размытыми. Сглаживание текста, набранного крупным шрифтом, делает его более приятным для восприятия, но если шрифт мелкий, то текст становится нечетким и неразборчивым. В ранних версиях Flash применение антиальясинга к тексту, набранному мелким шрифтом, создавало затруднения при чтении. Технология Flash Type (механизм рендеринга шрифтов, используемый во Flash версии 8 и старше) позволяет применять антиальясинг к мелким шрифтам без потери качества при условии, что текст не анимирован.

**С**

Есть еще более быстрый способ — щелкните мышью по стрелочке ползунка и сразу начинайте буксировку. Когда вы отпустите кнопку мыши, Flash уберет ползунок и автоматически установит новый размер шрифта.

**С**

Размер, выходящий за пределы диапазона ползунка, придется ввести вручную.

**С**

Размер шрифта можно также выбрать из меню **Text ⇒ Size** (Текст ⇒ Размер).

### Чтобы задать метод рендеринга шрифта:

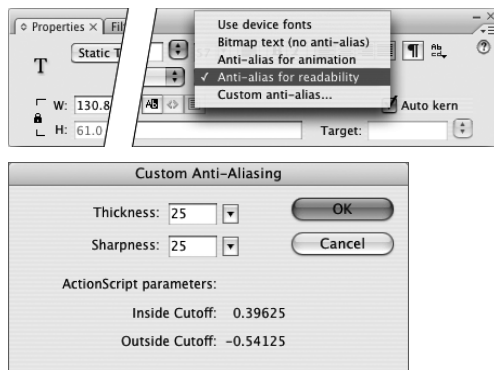
1. Выберите текстовое поле, которое собираетесь модифицировать.
2. На вкладке текста в инспекторе свойств выберите из меню **Anti-Aliasing** (Методы рендеринга шрифта) один из следующих пунктов (рис. 3.10):

**Use device fonts** (Применить шрифты устройства). Этот режим удобен, если уменьшение размера файла важнее возможности точно воспроизвести начертание шрифта на компьютере конечного пользователя (см. врезку «Шрифты устройства» выше).

**Bitmap text (no anti-alias)** (Растровый текст (без сглаживания)). Выберите этот режим, если хотите сохранить четкие, а не сглаженные края букв. В ролик будут встраиваться несглаженные шрифты. (Наличие растрового текста может заметно увеличить размер SWF-файла.)

**Anti-alias for animation** (Фильтрация-сглаживание для анимации). Этот режим устанавливайте для анимированных текстовых полей, а также в случае, когда планируете публиковать ролик для проигрывателя Flash Player версии 7 или еще более ранней. Flash встраивает шрифты, используемые в таких текстовых полях, но игнорирует часть информации о выравнивании и кернинге, чтобы ускорить воспроизведение. Для рендеринга не применяется технология Flash Type.

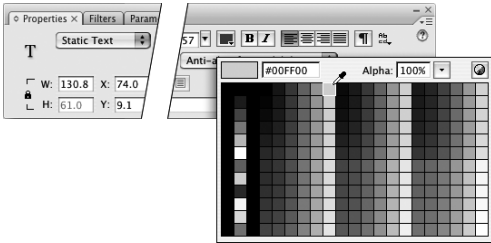
**Anti-alias for readability** (Фильтрация-сглаживание для читаемости). Это режим принимается по умолчанию при публикации для Flash Player версии 8 и старше (см. главу 17). Выбирайте его для текстовых полей, которые не планируете анимировать. Механизм рендеринга Flash Type рисует текст на арене в этом режиме, когда вы создаете FLA-файл. В момент публикации шрифты встраиваются в SWF-файл. При воспроизведении применяется Flash Type.



**Рис. 3.10.** Для управления степенью сглаживания выберите нужный режим из списка методов рендеринга на вкладке текста в инспекторе свойств. В режиме Custom Anti-Aliasing открывается диалоговое окно для более точного задания параметров сглаживания.

**Custom anti-alias** (Пользовательская фильтрация-сглаживание). При выборе этого режима открывается диалоговое окно Custom Anti-Aliasing, в котором можно задать, насколько размытыми должны быть сглаженные литеры. Параметры в разделе Sharpness and Thickness (Резкость и толщина) управляют смешением цветов литеры и фона, на котором она отображается.

**С** Даже в режимах, подразумевающих применение технологии Flash Type, при определенных обстоятельствах Flash отключает сглаживание. Когда выполняется перекоп, зеркальное отражение или искажение текстового поля, текст в нем не сглаживается. Технология Flash Type неприменима к шрифтам размером более 255 пунктов. (Отметим, что при масштабировании текстового поля в процессе разработки вы иногда можете запрашивать просмотр при размере шрифта более 255 пунктов, хотя в параметрах этого поля указан меньший размер.) В опубликованном ролике для Flash Player версии 8 и выше (SWF-файле) сглаживание применяется, но когда увеличенный текст отображается на арене, никакого сглаживания вы не увидите.



**Рис. 3.11.** Цвет текста, созданного инструментом «Текст», можно задать с помощью элемента управления цветом на вкладке текста в инспекторе свойств.



**Рис. 3.12.** Для задания полужирного и курсивного начертания служат кнопки Bold (левый рисунок) и Italic (правый рисунок) на вкладке текста в инспекторе свойств.

### Чтобы задать цвет текста:

1. Выберите текст, который собираетесь модифицировать.
2. На вкладке свойств текста щелкните по элементу **Text (Fill) Color** (Цвет текста (заливки)) (отметим, что во Flash текст является разновидностью заливки). Появляется стандартное окно образцов цвета (рис. 3.11).
3. Выберите цвет.

Поскольку Flash считает текст заливкой, то для изменения его цвета годятся все описанные в главе 2 способы задания атрибутов заливки.

**С** Flash отображает на вкладке текста атрибуты выбранного на арене текста. Иными словами, выбирая некий текст, вы тем самым загружаете в инструмент «Текст» все его атрибуты. Храните блоки текста с часто используемым форматированием на монтажном столе, чтобы можно было одним щелчком настроить инструмент нужным образом.

### Чтобы задать стиль:

1. Выберите текст, который собираетесь модифицировать.
2. На вкладке свойств текста в инспекторе свойств выполните одно из следующих действий:
  - для задания полужирного начертания нажмите кнопку **Bold** (рис. 3.12);
  - для задания курсивного начертания нажмите кнопку **Italic**;
  - для задания начертания, являющегося одновременно полужирным и курсивным, нажмите обе кнопки **Bold** и **Italic**.

**С** Для переключения между полужирным и обычным начертанием можно нажать комбинацию клавиш **Shift-⌘-B** (Mac) или **Ctrl-Shift-B** (Windows). (Обратите внимание на клавишу **Shift**; в других приложениях для работы с текстом она обычно не входит в состав комбинации клавиш, применяемой для той же цели.) Для переключения между курсивным и обычным начертанием можно нажать комбинацию клавиш **Shift-⌘-I** (Mac) или **Ctrl-Shift-I** (Windows).

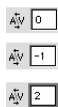
## Чтобы задать межбуквенный интервал (трекинг):

1. Выделите в текстовом поле текст, для которого хотите задать межбуквенный интервал.
  2. На вкладке свойств текста введите размер в пунктах в поле Letter Spacing (Межбуквенный интервал). Отрицательное значение уменьшает расстояние между буквами, положительное — увеличивает (рис. 3.13).
  3. Нажмите **Enter**.
- или
4. На вкладке свойств текста щелкните по треугольничку справа от поля Letter Spacing; появится ползунок.
  5. Буксируя ползунок, можно выбрать значение из диапазона от -60 до 60. В процессе буксировки Flash изменяет межбуквенный интервал в выделенном тексте на арене.
  6. Отпустите кнопку мыши, чтобы зафиксировать новое значение.

**C** Чтобы увеличить межбуквенный интервал в выделенном тексте, выполните команду **Text ⇒ Letter Spacing ⇒ Increase** (Текст ⇒ Межсимвольный интервал ⇒ Увеличить) или нажмите комбинацию клавиш **Option-⌘ — стрелка вправо** (Mac) либо **Ctrl-Alt — стрелка вправо** (Windows). Чтобы уменьшить межбуквенный интервал, выполните команду **Text ⇒ Letter Spacing ⇒ Decrease** (Текст ⇒ Межсимвольный интервал ⇒ Уменьшить) или воспользуйтесь аналогичной комбинацией клавиш, только вместо стрелки вправо нажимайте стрелку влево. Если включить в комбинацию еще и клавишу **Shift**, то величина шага удвоится.

**C** Межбуквенный интервал продолжает уменьшаться или увеличиваться все время, пока вы удерживаете вышеупомянутые комбинации клавиш.

**C** Чтобы восстановить исходный межбуквенный интервал, принятый для данного шрифта, выполните команду **Text ⇒ Letter Spacing ⇒ Reset** (Текст ⇒ Межсимвольный интервал ⇒ Сбросить) или нажмите комбинацию клавиш **Option-⌘ — стрелка вверх** (Mac) либо **Ctrl-Alt — стрелка вверх** (Windows).



To Shape the World  
To Shape the World  
To Shape the World

**Рис. 3.13.** Введите отрицательное значение, чтобы уменьшить расстояние между буквами, положительное — чтобы увеличить. Введите 0, если хотите восстановить межбуквенный интервал, встроенный в данный шрифт.

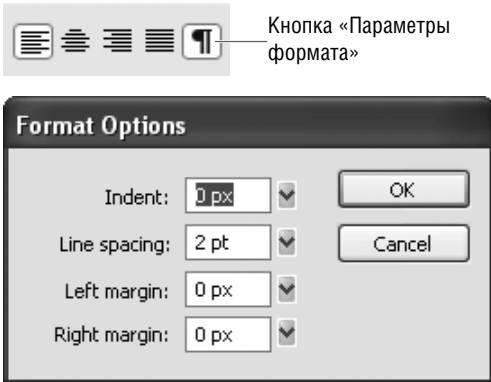
### Что такое кернинг?

Если *трекинг* определяет интервал между символами и словами в целой строке или абзаце, то кернинг относится только к расстоянию между парами литер. При разработке шрифта каждая литера считается отдельным элементом. Бывает так, что при написании рядом двух литер интервал между ними выглядит некрасиво. Например, если по соседству оказываются заглавная *T* и прописная *o*, то интервал будет казаться слишком большим из-за пустого места под верхней перекладиной *T*. Чтобы улучшить восприятие текста, можно сократить интервал между литерами, т.е. уменьшить кернинг. Наоборот, для некоторых литер, скажем, *t* и *i* соседство может казаться глазу слишком тесным. Тогда можно увеличить кернинг для такой пары.

Часто информация о подобных парах литер встраивается непосредственно в шрифт, и Flash принимает ее во внимание, если на вкладке текста отмечен флажок **Auto Kern** (Автоматически). Рекомендуется включать автокернинг, чтобы текст выглядел красивее.

Flash позволяет задавать кернинг и вручную вместо или в дополнение к автоматическому. Выберите пару литер, для которой нужно задать кернинг, и воспользуйтесь функцией **Letter Spacing**, чтобы уменьшить или увеличить расстояние между ними.





**Рис. 3.14.** На вкладке текста в инспекторе свойств нажмите кнопку **Edit Format Options** (Изменить параметры формата) (верхний рисунок), чтобы открыть диалоговое окно для задания таких атрибутов текста, как выравнивание, отступы, поля и межстрочный интервал (нижний рисунок).

## Задание атрибутов абзаца

Flash позволяет работать с абзацами, как в обычном текстовом редакторе. На вкладке текста в инспекторе свойств можно задать левое и правое поле, красную строку, межстрочный интервал и выравнивание (по левому или правому краю, по центру или по ширине) (рис. 3.14). Можно установить атрибуты абзаца заранее, тогда инструмент «Текст» будет применять их автоматически. А можно переформатировать уже введенный текст. Инструмент «Текст» применяет те параметры, которые в данный момент установлены на вкладке текста. В следующих упражнениях объясняется, как модифицировать существующий текст; для их выполнения оставьте открытой вкладку **Properties** в инспекторе свойств (если она закрыта, выполните команду **Window ⇒ Properties ⇒ Properties**).

### Чтобы выделить подлежащие модификации абзацы:

Выполните одно из следующих действий:

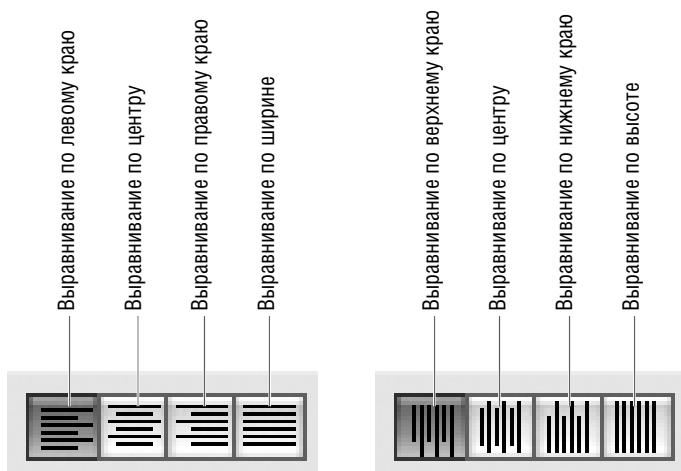
- выбрав инструмент «Текст», щелкните по абзацу, который хотите модифицировать;
- выбрав инструмент «Текст», буксируйте мышью для выделения нескольких смежных абзацев;
- выбрав инструмент «Стрелка», щелкните по тексту, чтобы выделить все абзацы в одном поле.

**Чтобы задать выравнивание абзаца:**

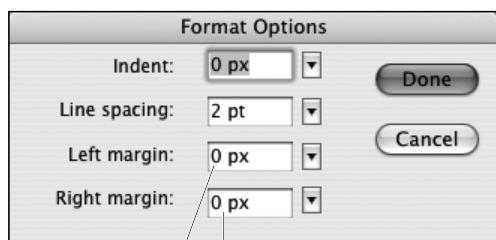
1. Выделите абзацы, которые хотите модифицировать.
2. На вкладке свойств текста в инспекторе свойств выполните одно из следующих действий (рис. 3.15):
  - чтобы выровнять горизонтальный текст по левому краю (вертикальный — по верхнему краю), нажмите первую слева кнопку;
  - чтобы выровнять горизонтальный или вертикальный текст по центру, нажмите вторую кнопку;
  - чтобы выровнять горизонтальный текст по правому краю (вертикальный — по нижнему краю), нажмите третью кнопку;
  - чтобы выровнять текст по ширине (все строки абзаца, кроме последней, растягиваются на всю ширину колонки), нажмите четвертую кнопку.

**C**

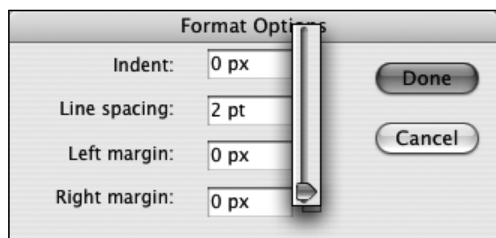
Чтобы выделить все абзацы в текстовом поле, щелкните инструментом «Текст» в любом месте поля, а затем выполните команду **Edit** ⇒ **Select All** (Редактирование ⇒ Выделить все).



**Рис. 3.15.** Кнопки выравнивания абзаца позволяют форматировать текст абзаца четырьмя способами. На каждой кнопке показано графическое представление соответствующего способа выравнивания. В зависимости от ориентации выделенного текста черточки расположены по горизонтали (левый рисунок) или по вертикали (правый рисунок).



Правое/нижнее поле  
Левое/верхнее поле



**Рис. 3.16.** Величину правого/левого (для горизонтального текста) или верхнего/нижнего поля можно ввести непосредственно в соответствующее поле диалогового окна **Format Options** (верхний рисунок) или воспользоваться ползунком (нижний рисунок).

### Чтобы задать поля:

1. На вкладке текста в инспекторе свойств нажмите кнопку **Edit Format Options** (на ней изображен символ абзаца). Открывается диалоговое окно **Format Options**.
2. В поле **Left Margin** (Левое поле) или **Right Margin** (Правое поле) введите величину нужного поля. Для полей действуют единицы измерения, установленные в окне свойств документа (см. главу 1).
3. Нажмите кнопку **Done** (рис. 3.16)

Указанные вами поля будут оставлены слева и справа от текста в охватывающем его прямоугольнике. Конечный пользователь не увидит полей, если вы не создадите текст, доступный для редактирования. В этом случае имеет смысл сделать видимой рамку текстового поля.

**С**

Для ввода новых значений можно также щелкнуть по треугольничку справа от поля **Margin**. Появится ползунок, с помощью которого можно выбрать значение от 0 до 72.

**С**

Заданные значения Flash учитывает при создании текстовых полей. Если величина поля равна 0 (по умолчанию), то при щелчке инструментом «Текст» по арене создается охватывающий прямоугольник такого размера, чтобы поместилась только мигающая вертикальная черта, обозначающая точку вставки. Если прямоугольник оказывается длиннее, чем вы ожидали, проверьте значения полей на вкладке текста в инспекторе свойств.

**С**

Flash может создавать поля шириной до 720 пикселей. Чтобы установить значение, большее тех 72 пикселей, которые допускает ползунок, введите его вручную в поле **Left Margin** или **Right Margin**.

**Чтобы задать отступ первой строки:**

- В диалоговом окне Format Options заполните поле Indent (Отступ) так же, как в предыдущих упражнениях. Flash отсчитывает отступ от левого поля. Если величина левого поля равна 0, то отступ отсчитывается от левого края охватывающего прямоугольника.

**Чтобы задать межстрочный интервал:**

- В диалоговом окне Format Options заполните поле Line Spacing (Межстрочный интервал) так же, как в предыдущих упражнениях. Если текст набран шрифтами разной высоты, то при вычислении межстрочного интервала берется больший шрифт (рис. 3.17).

**С**

Пункт — наиболее употребительная единица измерения при работе с текстом. Вне зависимости от того, какая единица измерения установлена в окне свойств документа, Flash всегда показывает межстрочный интервал в пунктах (*pt*).

**С**

Если вы работаете со шрифтом, для которого расстояние между строками оказывается больше, чем вам хочется, введите отрицательный межстрочный интервал. Допустимый диапазон — от -36 до 720 пунктов.

Squares at a square  
dance generally  
dance with squares.

Squares at a square  
dance Generally  
dance with squares.

**Рис. 3.17.** Межстрочный интервал для левого блока текста равен 0. То расстояние между строками, которое вы видите, определено в самом шрифте. Поскольку текст набран одним шрифтом, расстояния между верхней, средней и нижней строками одинаковы. В правом блоке текста межстрочный интервал тоже равен 0, но одна буква набрана шрифтом большего размера. Поэтому Flash увеличил расстояние между строками, чтобы эта буква уместилась.

# МОДИФИКАЦИЯ ПРОСТЫХ ГРАФИЧЕСКИХ ЭЛЕМЕНТОВ

---

## 4

Один из способов модификации графики в программе Adobe Flash CS3 Professional заключается в том, чтобы выбрать одну или несколько форм и изменить их атрибуты (например, цвет, размер или положение) на вкладке **Properties** в инспекторе свойств или на какой-нибудь другой панели.

Можно также модифицировать путь, определяющий форму элемента. Для некоторых операций, скажем для выпрямления линий, изменения кривых Безье и задания новых значений атрибутов, необходимо выбрать элементы. Для выполнения других, например изменения формы сегмента прямой или кривой линии, требуется, чтобы не было никаких выбранных элементов. Есть несколько операций, позволяющих редактировать элемент вне зависимости от того, выбран он или нет. В качестве примера приведем изменение цвета заливки с помощью инструмента «Ведро с краской».

В этой главе мы рассмотрим использование инструментов выбора, спецвыделения и лассо для выбора и модификации элементов, изученных в главе 2. Кроме того, мы покажем, как пользоваться вкладкой **Properties** инспектора свойств и другими панелями для изменения атрибутов.

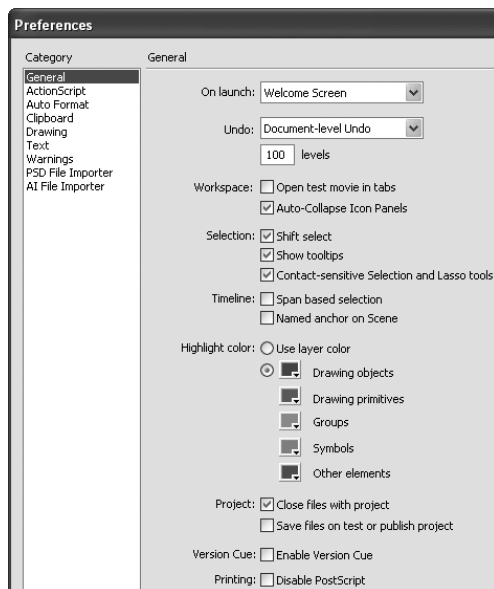
## Задание параметров выбора

Во Flash есть два основных способа выбора: щелкнуть по самому элементу или охватить элемент полностью либо частично эластичным контуром. Некоторые параметры позволяют точнее управлять этими способами. Для первого способа можно указать, необходимо ли для выбора нескольких элементов щелкать с нажатой клавишей **Shift** или очередной элемент добавляется в набор выбранных простым щелчком. Для выбора методом эластичного контура вы можете решить, будет ли графический объект считаться выбранным только, если контур охватывает его целиком или достаточно охватить какую-нибудь часть.

### Чтобы задать режимы работы инструментов выбора:

1. В меню **Flash** (Mac) или **Edit** (Windows) выберите пункт **Preferences** (Установки). Открывается диалоговое окно Preferences.
2. Выберите категорию **General** (Общие) (рис. 4.1).
3. В группе **Selection** (Выделение) отметьте любой из следующих флажков:

**Shift Select** (Выбор с нажатой клавишей Shift) (это режим по умолчанию). В этом режиме для добавления нового объекта в набор уже выбранных необходимо щелкать по нему мышью, удерживая клавишу **Shift**. Если флажок сброшен, то простой щелчок инструментом «Стрелка» добавляет объект в набор выбранных.



**Рис. 4.1.** В категории General в диалоговом окне Preferences имеются параметры для задания метода выбора.

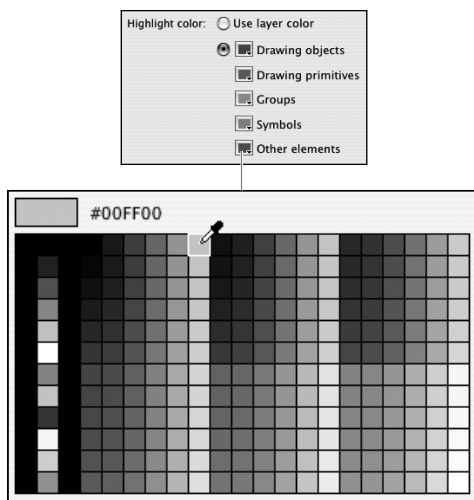
**Contact-Sensitive Selection and Lasso Tools**

(Инструменты выделение и лассо, чувствительные к контакту). Если этот флажок отмечен (по умолчанию так и есть), то всякий раз, как эластичный контур касается графического объекта, текстового поля (см. главу 3), сгруппированной формы (см. главу 5) либо экземпляра символа (см. главу 7), Flash выбирает этот объект. Если флажок сброшен, то объект считается выбранным только, если контур охватывает его целиком. Этот параметр не оказывает влияние на выбор комбинированных форм; всегда выбирается именно та часть комбинированной формы, которая охвачена эластичным контуром (см. следующий раздел «Выбор объектов»).

4. Нажмите **ОК**.

### Выберите способ индикации выделения по своему вкусу

Выбранные графические объекты на арене визуально выделяются. По умолчанию Flash CS3 применяет разные цвета для элементов разных типов (объектов-рисунков, примитивов, групп, символов и т.д.) В некоторых случаях такое выделение трудноразличимо. Если, например, цвет фона близок к цвету выделения, то понять, что именно выделено, нелегко. Цвета выделения можно настроить в категории **General** в диалоговом окне Preferences. Чтобы для выделения любого графического объекта в данном слое использовался цвет контуров в этом слое, установите переключатель **Highlight color** (Цвет выделения) в положение **Use Layer Color** (Использовать цвет слоя) (о слоях и представлении объектов в виде контуров см. главу 6). Чтобы для каждого типа графических объектов использовались разные цвета, отметьте переключатель, находящийся слева от списка типов (Drawing Objects, Drawing Primitives, Groups, Symbols и Other Elements — Объект-рисунок, Графические примитивы, Группы, Символы и Другие элементы). Чтобы задать цвет для некоторого типа объектов, щелкните по элементу управления цветом рядом с этим типом, а затем выберите цвет из списка образцов (рис. 4.2).



**Рис. 4.2.** В категории General в диалоговом окне Preferences можно задать цвета выделения выбранных элементов. Для этого щелкните по элементу управления цветом рядом с типом графического объекта.

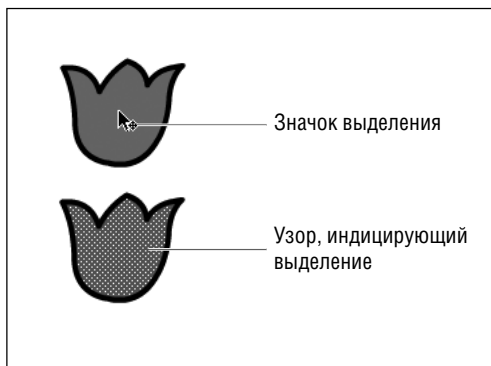
## Выбор объектов

Комбинированные формы, объекты-рисунки и примитивы, будучи выбранными, ведут себя по-разному. То, что представляется одиночной формой, на самом деле может состоять из нескольких сегментов. Инструмент «Прямоугольник» создает квадрат, обводка которого — это четыре разных отрезка. Если щелкнуть по одной стороне квадрата, созданного в режиме комбинированного рисования, то будет выбран только соответствующий ей отрезок. Чтобы выбрать всю обводку квадрата, придется выбрать каждый отрезок по отдельности (для добавления оставшихся трех сторон в набор выбранных можно щелкать мышью, удерживая клавишу **Shift**). Если квадрат создан как объект-рисунок или прямоугольный примитив, то Flash рассматривает все четыре сегмента обводки как единое целое. Щелчок по любой стороне такого квадрата приведет к выбору всей обводки.

Выбранные области комбинированных форм выделяются узором из точек. Убедитесь, что узором покрыты все те части обводки или заливки комбинированной формы, которые вы намеревались выбрать.

### Чтобы выбрать объекты щелчком мыши:

1. На панели инструментов выберите инструмент «Стрелка» или нажмите клавишу **V**.
2. Для выбора элементов, созданных как комбинированные формы, выполните одно из следующих действий:
  - чтобы выбрать заливку комбинированной формы, щелкните в любом месте заливки. Когда курсор-стрелка находится над заливкой, рядом с ним появляется значок выделения. Выбранную заливку Flash выделяет узором из точек (рис. 4.3);
  - чтобы выбрать обводку комбинированной формы, щелкните по любому сегменту обводки. Когда курсор-стрелка находится над обводкой, рядом с ним появляется небольшая дуга или уголок (рис. 4.4).



**Рис. 4.3.** Когда курсор находится над заливкой областью, рядом со стрелкой появляется значок выделения (перекрестье, говорящее о том, что инструмент готов переместить или выбрать объект). Щелкните в любом месте заливки. Выбранная заливка заполняется узором из точек.



**Рис. 4.4.** Дополнительный значок рядом с курсором показывает, какого рода точка обводки находится под ним.

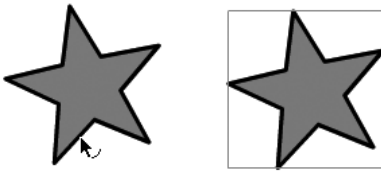
### Двойные щелчки мышью

Двойной щелчок по любому сегменту обводки связанной комбинированной формы выделяет все ее сегменты. Двойной щелчок по заливке комбинированной формы, которая обладает и заливкой, и обводкой, приводит к выбору того и другого.





**Рис. 4.5.** При щелчке по обводке комбинированной формы Flash выбирает и выделяет только один сегмент.



**Рис. 4.6.** Когда инструмент-стрелка находится над невыбранным объектом-рисунком (левый рисунок), рядом появляются те же значки, что для комбинированных форм (перекрестье обозначает готовность к перемещению/выбору заливки, дуга или уголок — то же самое для обводки). Щелчок в любой точке объекта-рисунка приводит к выбору его целиком, при этом появляется охватывающий прямоугольник (правый рисунок).



**Рис. 4.7.** Если стрелка находится над обводкой (верхний рисунок) или заливкой (средний рисунок) примитивной формы, рядом появляется значок выделения. Щелчок в любой точке формы приводит к выбору ее целиком (нижний рисунок).

Эти значки говорят о том, что инструмент находится над какой-то точкой сегмента, и указывают, над какой именно: гладкой или угловой. (Подробнее об этом см. врезку «О гладких и угловых точках».) Точечным узором Flash выделяет только тот сегмент, по которому вы щелкнули (рис. 4.5).

3. Чтобы выбрать элемент, созданный как объект-рисунок, щелкните по любой части формы. Рядом с курсором-стрелкой появится дуга или уголок, обозначающие то же самое, что для заливок и обводок комбинированных форм. Flash выбирает всю форму и выделяет ее, отображая охватывающий прямоугольник (рис. 4.6).
4. Чтобы выбрать элемент, созданный как примитив, щелкните по любой его части. Рядом с курсором-стрелкой появляется значок выделения, показывающий, находится ли курсор над заливкой или над обводкой. Flash выбирает всю форму и выделяет ее, отображая охватывающий прямоугольник (рис. 4.7).

5. Чтобы добавить элемент в набор выбранных, выполните одно из следующих действий:

- если активен режим выбора с нажатой клавишей **Shift** (режим по умолчанию), то удерживайте нажатой клавишу **Shift** при щелчке по элементу, который собираетесь добавить (рис. 4.8);
- если режим выбора с нажатой клавишей **Shift** выключен, то добавление происходит при простом щелчке по элементу.

**C**

Чтобы временно переключиться на инструмент-стрелку во время работы с другим инструментом, нажмите клавишу **⌘** (Mac) или **Ctrl** (Windows). Стрелка остается активным инструментом до тех пор, пока эта клавиша нажата.

**C**

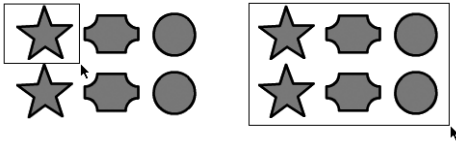
Чтобы выбрать все, что в данный момент находится на арене, выполните команду **Edit** ⇒ **Select All** (Редактирование ⇒ Выделить все) или нажмите комбинацию клавиш **⌘-A** (Mac) или **Ctrl-A** (Windows).

**C**

Охватывающий прямоугольник для объекта-рисунка круглой или неправильной формы легко различим, поскольку он находится вне формы. В случае прямоугольного примитива охватывающий прямоугольник можно распознать по наличию управляющих точек. А вот для объекта-рисунка прямоугольной формы охватывающий прямоугольник расположен точно по границе самой формы. Поэтому при некоторых сочетаниях цветов объекта-рисунка и выделения увидеть его не так-то просто. Если вы столкнетесь с этой проблемой, задайте контрастный цвет выделения в категории **General** в диалоговом окне Preferences (см. врезку «Выберите способ индикации выделения по своему вкусу» выше в этой главе).

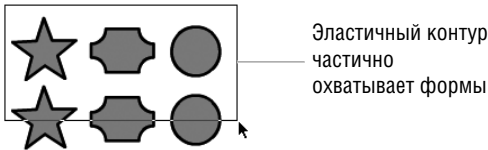


**Рис. 4.8.** Если оставлены настройки выбора по умолчанию, то для добавления новых сегментов в набор уже выбранных нужно щелкнуть по невыделенным сегментам или областям заливки, удерживая клавишу **Shift**.

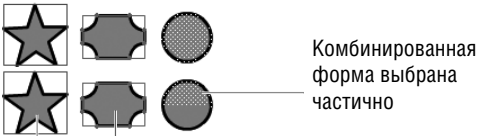


**Рис. 4.9.** Буксировка инструмента «Стрелка» создает эластичный прямоугольный контур (левый рисунок). Расположите начальную вершину так, чтобы можно было охватить им все элементы, которые вы намереваетесь выбрать (правый рисунок). Когда вы отпустите кнопку мыши, Flash выделит все охваченные элементы.

Буксируйте эластичный прямоугольный контур



Выбранные формы



Примитивная форма выбрана полностью  
Объект-рисунок выбран полностью

**Рис. 4.10.** Если активен режим выбора, чувствительный к контакту (по умолчанию), то в число выбранных объектов попадают все объекты-рисунки и примитивы, которых касается эластичный контур. Что касается комбинированных форм, то выбираются лишь те части, которые полностью охвачены контуром. (Здесь звезды — это объекты-рисунки, прямоугольники с вырезанными углами — примитивы, а круги — комбинированные формы.)

### Чтобы выбирать объекты эластичным контуром, чувствительным к контакту:

1. Убедитесь, что установлен режим выбора, чувствительный к контакту (см. раздел «Задание параметров выбора» выше).
2. Выберите инструмент «Стрелка».
3. Если вы начнете буксировать мышью, появится эластичный контур прямоугольника (рис. 4.9). Этот прямоугольник не является графическим элементом, а лишь обозначает границы области выбора.
4. Продолжайте буксировку, пока прямоугольник не охватит все комбинированные формы, которые вы собираетесь выбрать, и по крайней мере часть каждого объекта-рисунка.
5. Отпустите кнопку мыши.

**Комбинированные формы.** Flash выделяет те части заливки или обводки, которые попали внутри прямоугольника; все, что оказалось вовне, не выбирается (рис. 4.10).

**Объекты-рисунки и примитивные формы.** Если эластичный контур касается любой части объекта-рисунка или примитива, Flash выбирает его целиком, выделяя охватывающим прямоугольником.

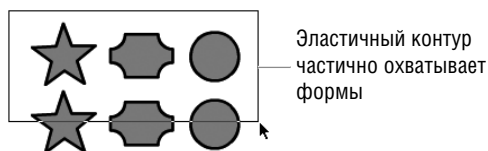
### Чтобы выбрать объекты эластичным контуром, не чувствительным к контакту:

1. Убедитесь, что режим выбора, чувствительный к контакту, выключен (см. раздел «Задание параметров выбора» выше).
2. Выполните то же, что в шагах 2-4 из предыдущего упражнения, но на этот раз полностью охватите контуром объекты-рисунки, которые хотите выбрать.

**Комбинированные формы.** Flash выделяет те части заливки или обводки, которые попали внутрь прямоугольника; все, что оказалось вовне, не выбирается (рис. 4.11).

**Объекты-рисунки и примитивные формы.** Flash выбирает только те объекты-рисунки и примитивы, которые полностью охвачены эластичным контуром. Если контур только касается такого элемента или охватывает его частично, элемент не выбирается.

Буксируйте эластичный прямоугольный контур



Эластичный контур частично охватывает формы

Выбранные формы



Комбинированная форма выбрана частично

Примитивная форма не выбрана

Объект-рисунок не выбран

**Рис. 4.11.** Если режим выбора, чувствительный к контакту, выключен, то выбираются только те части комбинированных форм, которые попадают внутрь эластичного контура. Объекты-рисунки и примитивы выбираются только тогда, когда целиком охвачены контуром. (Здесь опять звезды — это объекты-рисунки, прямоугольники с вырезанными углами — примитивы, а круги — комбинированные формы.)

**С**

Если линии или формы, которые вы хотите выбрать, расположены близко к другим линиям, то выбрать желаемое с помощью прямоугольного контура может оказаться затруднительно. Инструмент «Лассо» может создавать контур неправильной формы. Выберите лассо на панели инструментов или нажмите клавишу **L**. Буксируя инструмент, нарисуйте произвольную линию, охватывающую элементы, которые хотите выбрать (рис. 4.12).

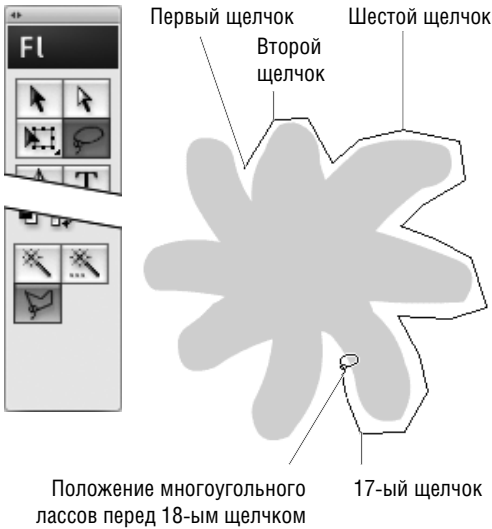
Замкните контур, вернувшись в начальную точку, и отпустите кнопку мыши. Flash выделит все объекты, попавшие внутрь лассо.

**С**

В режиме многоугольника лассо позволяет определить контур, состоящий из последовательности отрезков прямых. Выбрав инструмент «Лассо», щелкните по значку многоугольника в нижней части панели инструментов. Теперь можно начинать строить многоугольный контур, охватывающий нужные вам элементы (рис. 4.13). Чтобы зафиксировать выбор, щелкните мышью дважды.



**Рис. 4.12.** Лассо позволяет выбрать элементы, имеющие нерегулярную форму или расположенные настолько близко к другим элементам, что воспользоваться прямоугольным контуром невозможно. Все комбинированные формы, попавшие внутрь лассо, станут выделенными, как только вы отпустите кнопку мыши. Нужно ли полностью охватывать лассо объекты-рисунки и примитивы, чтобы выбрать их, зависит от того, включена ли чувствительность к контакту.



**Рис. 4.13.** В режиме многоугольника инструмент «Лассо» создает контур, состоящий из последовательности отрезков, которым можно охватить подлежащие выбору элементы. Двойной щелчок замыкает многоугольник, рисуя отрезок, соединяющий текущую точку с начальной.

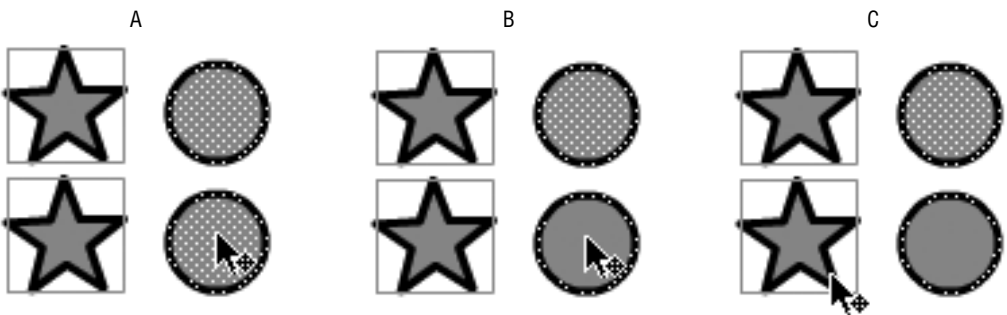
### Чтобы отменить выбор отдельных элементов:

1. Выберите инструмент «Стрелка» на панели инструментов.
2. Нажмите и удерживайте клавишу **Shift**.
3. Щелкайте мышью по тем выбранным объектам-рисункам, примитивам и заливкам или обводкам комбинированных форм, выбор которых следует отменить.

Flash снимает выделение с тех элементов, по которым вы щелкаете (рис. 4.14). Вне зависимости от действующего метода выбора, для отмены необходимо щелкать инструментом «Стрелка» с нажатой клавишей **Shift**.

**C** Чтобы отменить выбор всех элементов, выполните команду **Edit ⇒ Deselect All** (Редактирование ⇒ Отменить выбор всего) или нажмите комбинацию клавиш **Shift-⌘-A** (Mac) либо **Ctrl-Shift-A** (Windows).

**C** Для быстрой отмены выбора всех элементов, щелкните инструментом «Стрелка» по незазанятой области арены или монтажного стола.



**Рис. 4.14.** Укажите инструментом «Стрелка» на элемент, выбор которого нужно отменить (А). Отмените выбор, щелкнув по элементу с нажатой клавишей Shift (В). Повторите эту процедуру для отмены выбора следующего элемента (С). (Здесь звезды — это объекты-рисунки, а круги — комбинированные формы.)

## Работа с буфером обмена

Flash поддерживает стандартные операции вырезания, копирования и вставки, знакомые большинству пользователей. Имеются также операции специальной вставки графических элементов; вставленный элемент может располагаться в центре арены или в том месте, где находился изначально.

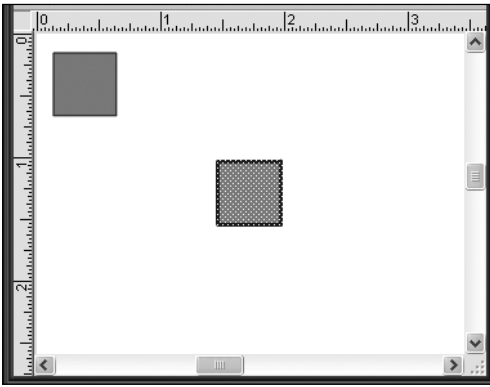
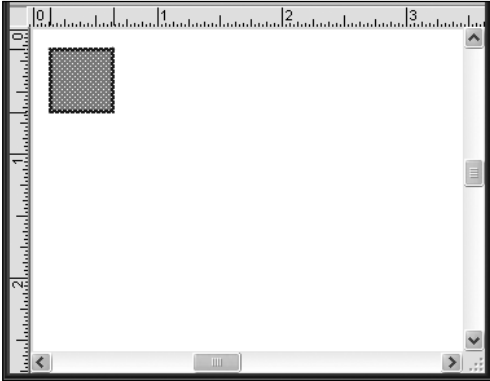
### Чтобы выполнить простейшие операции редактирования:

1. Выберите элементы, которые хотите удалить, вырезать или скопировать.
2. Откройте меню **Edit** (Редактирование) (рис. 4.15) и:
  - для удаления выбранных элементов выполните команду **Clear** (Очистить) или нажмите клавишу **Delete**;
  - для вырезания выбранных элементов выполните команду **Cut** (Вырезать) или нажмите комбинацию клавиш **⌘-X** (Mac) либо **Ctrl-X** (Windows);
  - для копирования выбранных элементов выполните команду **Copy** (Копировать) или нажмите комбинацию клавиш **⌘-C** (Mac) либо **Ctrl-C** (Windows);

Вырезанные или скопированные элементы остаются в буфере обмена до следующей операции вырезания или копирования. Чтобы извлечь содержимое буфера обмена, воспользуйтесь одной из команд вставки.

Edit	
Undo Change Selection	⌘Z
Repeat Cut	⌘Y
Cut	⌘X
Copy	⌘C
Paste in Center	⌘V
Paste in Place	⇧⌘V
Clear	⌫
Duplicate	⌘D
Select All	⌘A
Deselect All	⇧⌘A
Find and Replace	⌘F
Find Next	F3
Timeline	▶
Edit Symbols	⌘E
Edit Selected	
Edit in Place	
Edit All	

**Рис. 4.15.** В меню Edit есть команды вырезания, копирования и вставки для работы с графическими элементами.



**Рис. 4.16.** Если скопировать выбранный графический элемент (верхний рисунок), а затем выполнить команду **Edit ⇒ Paste in Center**, то Flash вставит копию элемента из буфера обмена в центр текущего вида арены (нижний рисунок).

### Чтобы вставить содержимое буфера обмена в центр окна:

- Выполните команду **Edit ⇒ Paste in Center** (Редактирование ⇒ Вставить в центр) или нажмите комбинацию клавиш **⌘-V** (Mac) либо **Ctrl-V** (Windows). Flash вставит содержимое буфера обмена в центр текущего вида арены (рис. 4.16).

### Чтобы вставить содержимое буфера обмена в то же место:

- Выполните команду **Edit ⇒ Paste in Place** (Редактирование ⇒ Вставить на месте) или нажмите комбинацию клавиш **Shift-⌘-V** (Mac) либо **Ctrl-Shift-V** (Windows).

Flash вставит содержимое буфера обмена в ту же позицию на арене, которую элемент занимал до копирования. Смысл этой команды становится понятен при работе со слоями или анимацией; иногда очень важно, чтобы элемент находился в том же месте, где и раньше, только в другом слое или кадре.

**C**

Еще один способ получить копию дает команда **Duplicate** (Дублировать). Выберите элементы, которые хотите скопировать. Выполните команду **Edit ⇒ Duplicate** (Редактирование ⇒ Копировать) нажмите комбинацию клавиш **⌘-D** (Mac) либо **Ctrl-D** (Windows). Flash создаст копии выбранных элементов. Дубликат будет располагаться на арене со смещением от оригинала. При этом он оказывается выбранным, чтобы предотвратить взаимодействие копии комбинированной формы с оригиналом. (Дополнительную информацию о взаимодействии между элементами см. в главе 5.) Команда **Duplicate** не изменяет содержимое буфера обмена.

**C**

Можно воспользоваться буксировкой и для дублирования нескольких элементов. Для этого сначала выберите их, а потом буксируйте инструментом «Стрелка» или «Лассо», удерживая нажатой клавишу **Option** (Mac) или **Alt** либо **Ctrl** (Windows).

**C**

Щелкните инструментом «Стрелка», удерживая нажатой клавишу **Option** (Mac) или **Alt** либо **Ctrl** (Windows), и буксируйте один какой-нибудь элемент, чтобы создать его копию. Не отпускайте клавишу-модификатор, пока не появится копия.

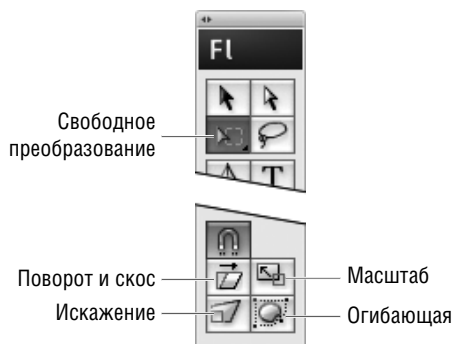
## Изменение размера графических элементов

Во Flash есть несколько способов изменить размер или *масштабировать* графические элементы. Можно интерактивно масштабировать выбранные элементы на арене, а можно задать конкретный коэффициент масштабирования или нужный размер на панели преобразований (Transform), на вкладке **Properties** в инспекторе свойств или на панели информации (Info).

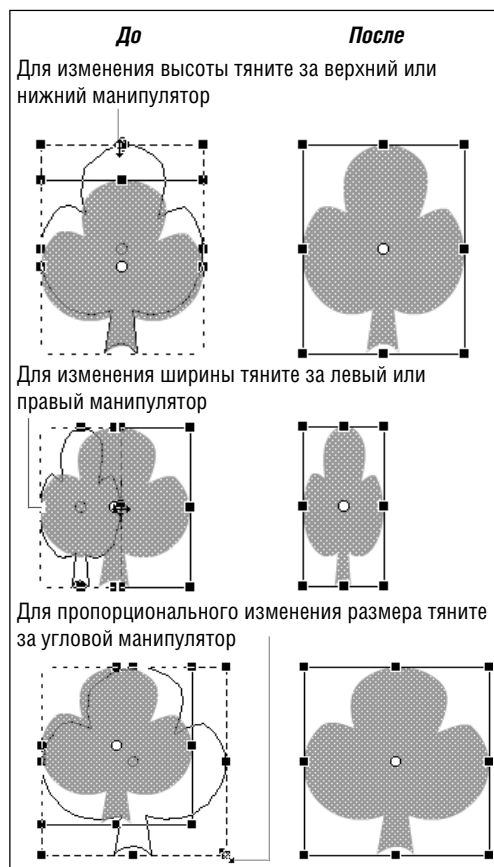
### Чтобы интерактивно изменить размер графического элемента:

1. На панели инструментов выберите инструмент «Свободное преобразование» (рис. 4.17).
2. Щелкните по элементу на арене, размер которого хотите изменить. Flash выделит элемент и разместит манипуляторы на всех сторонах и во всех углах охватывающего прямоугольника.
3. На панели инструментов выберите модификатор «Масштаб».
4. Наведите курсор на манипулятор. Курсор примет форму двунаправленной стрелки, указывающей направление, в котором элемент будет растягиваться или сжиматься, если потянуть за манипулятор.
5. Чтобы изменить размер графического элемента, выполните одно из следующих действий:
  - для изменения ширины щелкните мышью и потяните за один из манипуляторов на боковых сторонах;
  - для изменения высоты щелкните мышью и потяните за один из манипуляторов на верхней или нижней стороне;
  - для пропорционального изменения размеров элемента щелкните мышью и потяните за один из угловых манипуляторов.

Если тянуть в направлении центра элемента, размер уменьшается, в противном случае увеличивается (рис. 4.18).

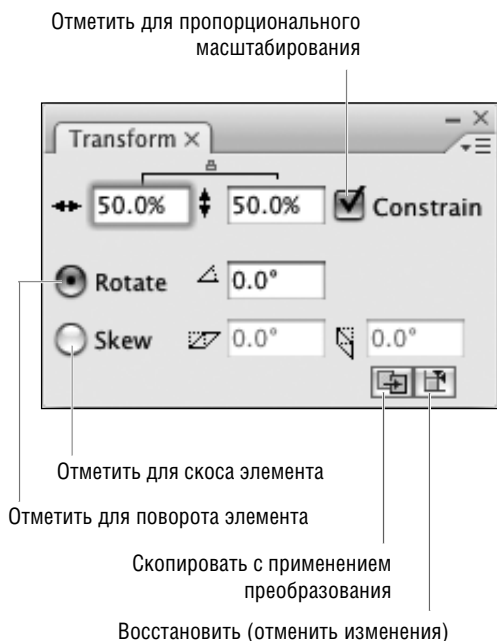


**Рис. 4.17.** Инструмент «Свободное преобразование» позволяет интерактивно масштабировать выбранные элементы.



**Рис. 4.18.** При активации модификатора «Масштаб» вокруг выбранного элемента появляются манипуляторы. Чтобы изменить размер, щелкните мышью и потяните за один из них.





**Рис. 4.19.** Чтобы изменить размер элемента, введите новые значения в поля Width и Height на панели преобразований. Если флажок Constrain отмечен, то при вводе значения в одно поле другое изменяется автоматически. Панель преобразований позволяет также задавать параметры для поворота и перекоса выбранных элементов.

**С**

В режиме по умолчанию новый размер графического элемента отсчитывается от управляющей точки, противоположной той, за которую вы тянете. Чтобы отсчитывать его от центра выбранного элемента, удерживайте нажатой клавишу **Option** (Mac) или **Alt** (Windows). Для символов (см. главу 7) дело обстоит прямо противоположным образом: по умолчанию символ масштабируется от точки трансформации (каковой является центр), а чтобы масштабировать от противоположной управляющей точки, нужно удерживать клавишу **Option** (Mac) или **Alt** (Windows).

### Чтобы изменить размер элемента с помощью панели преобразований:

1. Откройте панель преобразований и выберите элемент, размер которого хотите изменить. В полях Width (Ширина) и Height (Высота) на панели преобразований появится значение 100% (рис. 4.19).
2. Чтобы изменить размер элемента, выполните следующие действия:
  - для пропорционального изменения отметьте флажок **Constrain** (Ограничение) и введите новое значение в любое из полей Width или Height. Flash автоматически изменит значение в другом поле.
  - или*
  - если пропорции можно изменять, сбросьте флажок **Constrain** и введите новые процентные величины в поля Width или Height. Если значение меньше 100%, то элемент сжимается, иначе растягивается.
3. Нажмите **Enter**. Flash изменит размер элемента.

C

Панель преобразований масштабирует комбинированную форму абсолютно, пока она выбрана. Если отменить выбор формы, то ее текущий размер принимается за 100%. Для комбинированных форм возможно и относительное масштабирование. Выбрав форму, введите новые значения в поля Width и Height на панели преобразований, а затем отмените выбор. Снова выберите ту же форму (теперь в полях Width и Height будет находиться значение 100%). Введите процентный коэффициент масштабирования для второго преобразования.

C

Панель преобразований всегда абсолютно масштабирует объекты-рисунки, примитивы, текстовые поля, группы и символы (то есть применяет введенный коэффициент масштабирования к исходному размеру элемента).

C

Чтобы изменить размеры сразу нескольких элементов, выберите их и воспользуйтесь любым из описанных выше способов. Если вы пользуетесь панелью преобразований, то прямоугольник, охватывающий все выбранные элементы, масштабируется относительно своего центра, а все элементы внутри него сжимаются или растягиваются с одним и тем же коэффициентом. В случае применения инструмента «Свободное преобразование» охватывающий прямоугольник масштабируется относительно угла, противоположного тому, за который вы тянете.

C

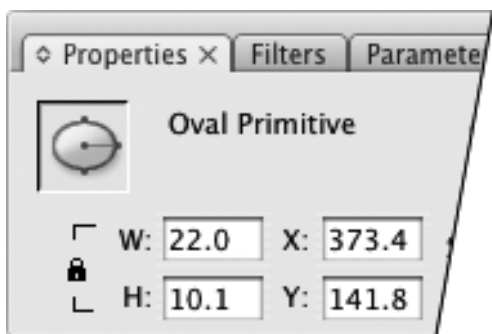
Если выбрать несколько элементов и менять их размеры с помощью панели преобразований, то Flash выполняет относительное масштабирование. Как только вы введете и подтвердите (нажатием **Enter**) новые значения ширины и высоты, Flash изменит размеры элементов и запишет в поля Width и Height значение 100%.

C

Чтобы быстро отменить изменение, выполненное с помощью панели преобразований, нажмите кнопку **Reset** (Восстановить) в правом нижнем углу панели или комбинацию клавиш **Shift-⌘-Z** (Mac) либо **Ctrl-Shift-Z** (Windows). Если элемент является комбинированной формой, то не следует предварительно отменять его выбор. Для объектов-рисунков, примитивов, текстовых полей (глава 3), групп (глава 5) и символов (глава 7) элемент можно выбрать и нажать кнопку **Reset** в любой момент, при этом будет восстановлен исходный размер.

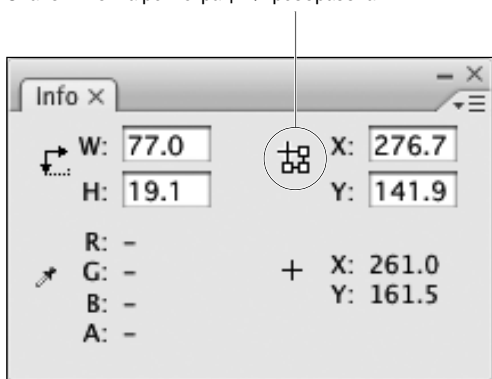
C

Чтобы преобразовать копию элемента, нажмите кнопку **Copy and Apply Transform** (Копировать и применить преобразование) на панели преобразований. Эта операция не очевидна в применении к комбинированным формам, так как Flash не сдвигает созданную копию. Поэтому не забудьте подвинуть ее сами перед тем, как отменять выбор.



**Рис. 4.20.** На вкладке свойств формы отображаются текущие ширина и высота охватывающего прямоугольника выбранного элемента. Чтобы изменить его размеры, введите новые значения. Если замок на значке заперт, то Flash сохраняет отношение ширины к высоте. Если замок открыт, то ширину и высоту можно изменять независимо.

Значок «Точка регистрации/преобразования»



**Рис. 4.21.** На панели информации можно ввести точные значения ширины и высоты элемента. Значок Registration/Transformation Point показывает, будут ли изменения отсчитываться от левого верхнего угла охватывающего прямоугольника или от точки трансформации. Чтобы применить новые значения к выбранному графическому элементу, нажмите Enter или щелкните мышью по арене.

### Чтобы изменить размер элемента с помощью инспектора свойств:

1. Откройте панель инспектора свойств и выберите элемент, размер которого хотите изменить.

В полях Width и Height на вкладке свойств формы отображаются текущие размеры охватывающего прямоугольника выбранной формы.

2. Чтобы изменить размер элемента, выполните следующие действия:

- для пропорционального масштабирования установите значок блокировки (слева от полей Width и Height) в положение, когда замок заперт. Введите новое значение в любое из этих полей (рис. 4.20). Значение в другом поле будет изменено автоматически.

или

- если пропорции разрешено изменять, установите значок блокировки в положение, когда замок открыт, и введите новые процентные коэффициенты в поля Width и Height.

3. Нажмите **Enter**. Flash изменит размер элемента.

**C**

Панель информации позволяет изменить размер выбранного элемента относительно точки его регистрации (левый верхний угол охватывающего прямоугольника) или точки трансформации. Для переключения режима щелкните по значку **Registration/Transformation Point** (Точка регистрации / трансформации) (рис. 4.21). Перекрестие в левом верхнем углу значка означает, что значения отсчитываются от точки регистрации, а кружочек — от точки трансформации. Значения в инспекторе свойств всегда вычислены относительно точки регистрации. (Дополнительную информацию см. на врезке «Как Flash отслеживает элементы» ниже в этой главе.)

## Позиционирование графических элементов

Если вас не устраивает положение некоторого элемента, его можно переместить. Чтобы позиционировать элемент, можно перетащить его по арене мышью или просто задать точные координаты. Вводить координаты можно на вкладке свойств формы в инспекторе свойств или на панели информации.

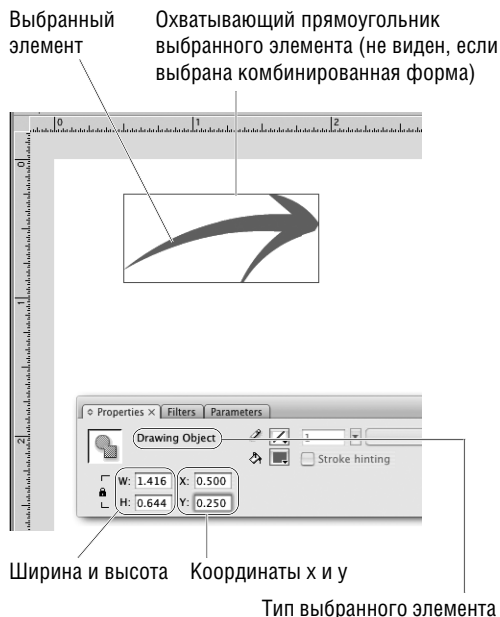
### Чтобы позиционировать элемент с помощью инспектора свойств:

1. Откройте вкладку **Properties** в инспекторе свойств и выберите элемент на арене. Текущие координаты точки регистрации выбранного элемента (к примеру, овального примитива) отображаются в полях *x* и *y* на вкладке **Properties** (рис. 4.22).
2. Для позиционирования элемента выполните одно или оба описанных ниже действия:
  - введите новую координату *x*, описывающую положение элемента по горизонтальной оси;
  - введите новую координату *y*, описывающую положение элемента по вертикальной оси;
3. Нажмите **Enter**, чтобы подтвердить введенные значения.

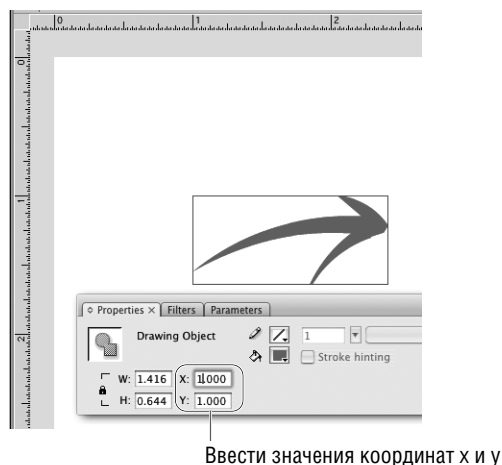
При выходе из поля в результате нажатия клавиши **Tab** или щелчка по другому полю, а также при нажатии клавиши **Enter** Flash фиксирует изменение, и элемент перемещается в новое положение (рис. 4.23).

**С**

По умолчанию точка трансформации соответствует центру охватывающего прямоугольника формы. Но это можно изменить. С помощью инструмента «Свободное преобразование» перетащите полый кружок, представляющий точку трансформации, в новое положение.



**Рис. 4.22.** На вкладке **Properties** в инспекторе свойств отображаются координаты *x* и *y* точки регистрации выбранного на арене элемента. Показанный объект-рисунок отстоит на 1/2 дюйма от правого края арены по горизонтали и на 1/4 дюйма от верхнего края по вертикали.



**Рис. 4.23.** При изменении координат на вкладке **Properties** в инспекторе свойств изменяется положение выбранного элемента. Теперь стрелка отстоит на 1 дюйм от правого края арены по горизонтали и на 3/4 дюйма от верхнего края по вертикали.

### Как Flash отслеживает элементы

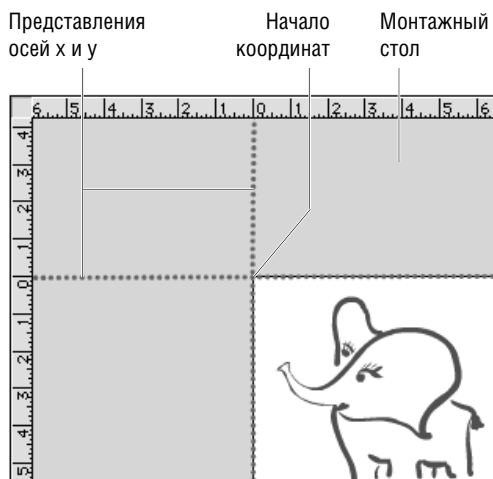
Для отслеживания положения и размеров элементов на арене Flash заключает каждый элемент в *охватывающий прямоугольник* — невидимый минимальный прямоугольник, внутри которого элемент помещается целиком. Сама арена рассматривается как гигантская координатная сетка, центр которой находится в левом верхнем углу арены (рис. 4.24). Единицы измерения по осям задаются в окне свойств документа (подробнее о документах см. главу 1). В инспекторе свойств и на панели информации отображаются размеры элемента (ширина и высота охватывающего прямоугольника) и его положение (координаты одной из важных точек элемента).

Размер элемента можно изменить путем ввода новых значений в поля Height и Width на вкладке **Properties** в инспекторе свойств или на панели информации. А ввод новых координат в поля x и y изменяет положение элемента на арене. (Дополнительную информацию об изменении размеров см. в разделе «Изменение размеров графических элементов» выше.)

В зависимости от того, на какой панели вводятся значения, для позиционирования элемента можно использовать две точки: *точку регистрации* и *точку трансформации*. Для комбинированных форм, объектов-рисунков и примитивов точка регистрации всегда расположена в левом верхнем углу охватывающего прямоугольника. Для символа положение точки регистрации задаете вы сами, и оно остается неизменным для всех его экземпляров. Для форм, текстовых полей и групп Flash помещает точку трансформации в центр охватывающего прямоугольника; ее положение для отдельных экземпляров одного символа можно задавать независимо (см. главу 7).

Инспектор свойств всегда отслеживает элементы по точке регистрации, а панель информации позволяет выбрать любую из двух точек. Выбор производится путем щелчка по значку Registration/Transformation. Если в левом верхнем углу этого значка отображается перекрестье, то положение и размеры элементов на панели информации приведены относительно точки регистрации, а если в правом нижнем углу отображается кружок, — то относительно точки трансформации.

В сценариях на языке ActionScript, которые динамически перемещают символы во время выполнения, для позиционирования всегда используется точка регистрации.



**Рис. 4.24.** Пунктирными линиями обозначены положения осей x и y относительно арены. Начало координат находится в левом верхнем углу арены.

## Отражение, поворот и перекос

Flash позволяет зеркально отражать, вращать и перекашивать выбранные элементы. Можно как манипулировать элементами произвольно с помощью модификатора **Rotate and Skew** (Вращение и наклон) инструмента «Свободное преобразование», так и применять различные команды для более точного выполнения операции.

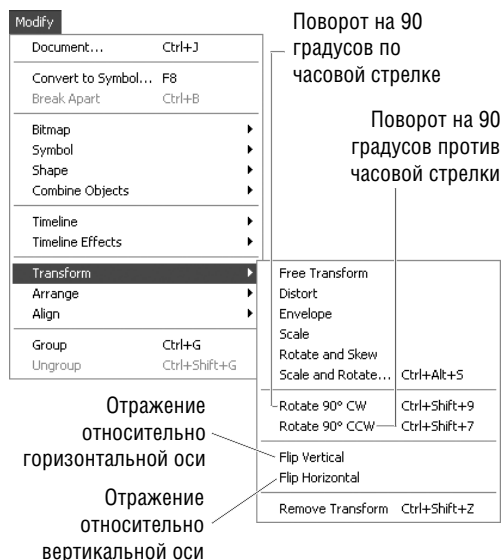
### Чтобы зеркально отразить графический элемент:

1. Выберите элемент, который хотите зеркально отразить.
2. Откройте меню **Modify** ⇒ **Transform** (Изменить ⇒ Преобразование).
3. Из этого меню (рис. 4.25) выберите одну из следующих команд:
  - чтобы отразить элемент относительно горизонтальной оси, выполните команду **Flip Vertical** (Отразить сверху вниз);
  - чтобы отразить элемент относительно вертикальной оси, выполните команду **Flip Horizontal** (Отразить слева направо).

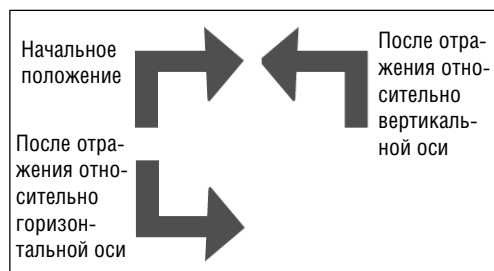
На рис. 4.26 показаны результаты обеих операций зеркального отражения.

**С**

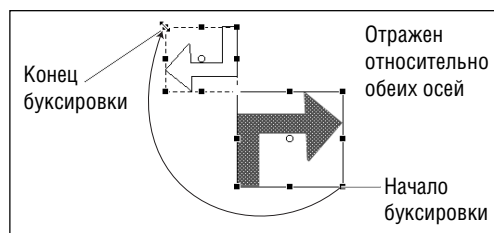
Можно одновременно выполнить зеркальное отражение и масштабирование, если воспользоваться модификатором **Scale** (Масштаб) инструмента «Свободное преобразование». Выбрав элемент и установив этот режим, тяните за какой-нибудь манипулятор в направлении прямо противоположного, пока он не пересечет его. Например, чтобы отразить выбранный элемент относительно обеих осей, тяните манипулятор из правого нижнего угла в левый верхний (рис. 4.27). В момент совмещения манипуляторов отраженный элемент будет совсем маленьким, но при продолжении буксировки прочь от левого верхнего угла растянется. Flash показывает текущий вид отраженного элемента; когда он достигнет желаемого размера, отпустите кнопку мыши.



**Рис. 4.25.** В меню **Modify** ⇒ **Transform** есть команды для зеркального отражения графических элементов относительно вертикальной и горизонтальной оси. Кроме них, имеются команды для поворота на угол 90 градусов как по, так и против часовой стрелки.



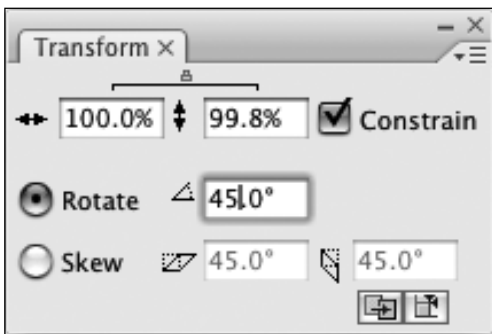
**Рис. 4.26.** Результат применения к элементу команд отражения из меню **Modify** ⇒ **Transform**.



**Рис. 4.27.** Модификатор «Масштаб» инструмента «Свободное преобразование» позволяет выполнять зеркальное отражение и масштабирование за одну операцию. На этом рисунке элемент отражен относительно обеих осей.



**Рис. 4.28.** Для поворота выбранного элемента на 90 градусов воспользуйтесь меню **Modify** ⇒ **Transform**. Повторите команду несколько раз, чтобы повернуть на угол 180, 270 или 360 градусов.



**Рис. 4.29.** Панель преобразования позволяет поворачивать элементы на точно заданный угол. Отметьте переключатель **Rotate** и введите угол поворота в градусах. Если значение положительно, то элемент поворачивается по часовой стрелке, иначе — против часовой стрелки.

### Чтобы повернуть элемент на 90 градусов:

1. Выберите элемент, который хотите повернуть.
2. Откройте меню **Modify** ⇒ **Transform** (рис. 4.28).
3. Из этого меню выберите одну из следующих команд:
  - чтобы повернуть элемент на 90 градусов по часовой стрелке, выполните команду **Rotate 90° CW** (Поворот на 90° по ЧС);
  - чтобы повернуть элемент на 90 градусов против часовой стрелки, выполните команду **Rotate 90° CCW** (Поворот на 90° против ЧС).

**C** Чтобы повернуть элемент относительно одного из его углов, а не точки трансформации, удерживайте во время буксировки клавишу **Option** (Mac) или **Alt** (Windows).

**C** Для поворота на угол, кратный 45 градусам, удерживайте клавишу **Shift**.

### Чтобы повернуть элемент на заданный угол:

1. Откройте панель преобразований (рис. 4.29). Если она еще не открыта, выберите пункт меню **Window** ⇒ **Transform** (Окно ⇒ Преобразование).
2. Отметьте переключатель **Rotate** (Повернуть).
3. Для задания направления и угла поворота, выполните одно из следующих действий:
  - для поворота против часовой стрелки введите отрицательное значение (от -1 до -360) в поле справа от переключателя **Rotate**;
  - для поворота по часовой стрелке введите положительное значение (от 1 до 360) в то же поле.
4. Нажмите **Enter**.

**С**

При работе с объектами-рисунками и примитивными формами можно в любой момент отменить изменение ширины или высоты, поворот и перекося, даже если выбор элемента уже отменен и были произведены другие модификации. Чтобы восстановить исходные ширину, высоту, угол поворота и перекося, выберите элемент и нажмите кнопку **Reset** (Сброс) на панели преобразований (рис. 4.30).



Сброс

**Рис. 4.30.** Чтобы отменить изменения, нажмите кнопку Reset на панели преобразований. Этот способ работает вне зависимости от того, были ли изменения выполнены с помощью панели преобразований или интерактивно инструментом «Свободное преобразование».

**С**

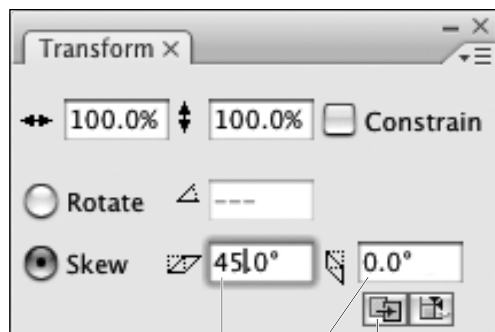
Для комбинированных форм кнопка **Reset** работает лишь до тех пор, пока форма остается выбранной. Если выбор был отменен, то для восстановления начального состояния формы придется воспользоваться командой **Undo** (Отменить) или панелью истории (см. главу 7).

### Чтобы перекосить элемент на заданный угол:

1. Откройте панель преобразований и выберите элемент, который хотите перекосить.
2. Отметьте на панели преобразований переключатель **Skew** (Перекося).
3. Задайте значения углов перекося по вертикали и по горизонтали в соответствующих полях (рис. 4.31).
4. Для выполнения преобразования нажмите **Enter**.

**С**

Чтобы перекосить копию выбранного элемента, нажмите кнопку **Copy and Apply Transform** на панели преобразований.



Перекося по горизонтали

Перекося по вертикали

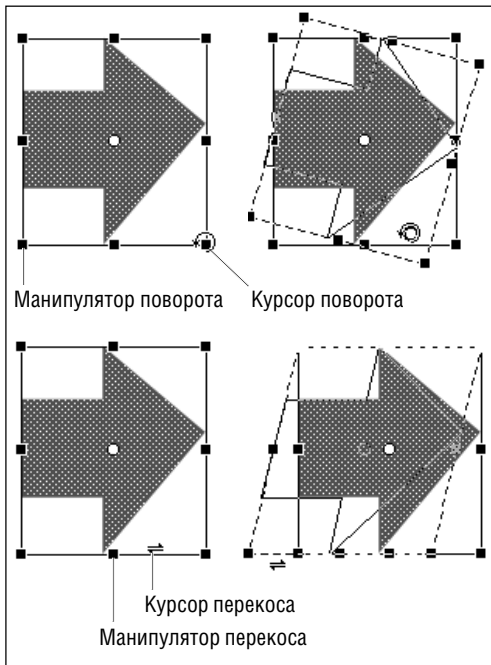
Копировать и применить преобразование

**Рис. 4.31.** Воспользуйтесь панелью преобразований для перекося выбранных элементов. Можно задавать разные углы перекося по горизонтали и по вертикали.





**Рис. 4.32.** Если отметить модификатор Rotate and Skew инструмента «Свободное преобразование», то появятся манипуляторы для интерактивного поворота и перекоса выбранного элемента.



**Рис. 4.33.** Отметив модификатор Rotate and Skew инструмента «Свободное преобразование», вы можете потянуть за один из угловых манипуляторов для поворота элемента (верхний рисунок). А если потянуть за боковой манипулятор, то элемент будет перекашиваться (нижний рисунок).

### Чтобы интерактивно повернуть или перекосить элемент:

1. Выберите элемент, который хотите повернуть или перекосить.
2. На панели инструментов выберите инструмент «Свободное преобразование», а затем щелкните по модификатору **Rotate and Skew** (рис. 4.32). На всех четырех сторонах и в углах охватывающего прямоугольника появляются квадратные манипуляторы.
3. Чтобы модифицировать выбранный элемент, выполните одно из следующих действий:
  - для поворота элемента подведите курсор к любому из угловых манипуляторов;
  - курсор принимает форму круговой стрелки. Когда вы начнете буксировать мышью, элемент станет поворачиваться вокруг точки трансформации (рис. 4.33);
  - чтобы перекосить элемент, подведите курсор к любому из боковых манипуляторов;
  - курсор принимает форму двусторонней полустрелки. Когда вы начнете буксировать мышью, элемент станет перекашиваться.
4. Отпустите кнопку мыши. Flash перерисует элемент в новом положении.

### С

Если выбрать объект инструментом «Свободное преобразование» и подвести курсор к угловому манипулятору, то он примет форму двунаправленной стрелки, обозначающей масштабирование. Но стоит чуть отвести курсор от манипулятора, как он превращается в круговую стрелку. В таком состоянии буксировка поворачивает объект. Подведите курсор к краю охватывающего прямоугольника между манипуляторами, и он примет форму двусторонней полустрелки. Теперь буксировка перекашивает объект.

## Искажение графических элементов

Модификатор **Distort** (Искажение) инструмента «Свободное преобразование» позволяет искажать некоторые графические элементы за счет изменения формы охватывающего прямоугольника. Можно сдвинуть один или несколько углов прямоугольника; манипулировать противоположными углами одновременно, чтобы превратить прямоугольник в трапецию; растягивать, сжимать и перекашивать прямоугольник с помощью боковых манипуляторов. Выбранные элементы растягиваются или сжимаются, оставаясь привязанными к охватывающему контуру. Модификатор **Distort** работает только для комбинированных форм и одиночных объектов-рисунков. К примитивным формам, текстовым полям (глава 3), группам (глава 5), символам (глава 7) и нескольким выбранным объектам-рисункам он неприменим.

### Чтобы произвольно исказить элемент:

1. Инструментом «Свободное преобразование» выберите элемент, который хотите исказить. Появится охватывающий прямоугольник с манипуляторами.
2. На панели инструментов отметьте модификатор **Distort** (рис. 4.34). Обратите внимание, что кружочек в центре прямоугольника отсутствует, это говорит о том, что вы работаете в режиме искажения.
3. Подведите курсор к любому манипулятору. Он примет форму полрой стрелки.

Свободное преобразование

Искажение

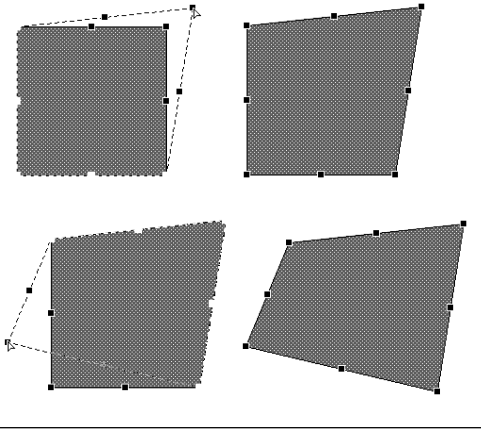


**Рис. 4.34.** Отметьте модификатор Distort инструмента «Свободное преобразование», чтобы можно было независимо изменять положение углов охватывающего прямоугольника.

### Искаженная перспектива

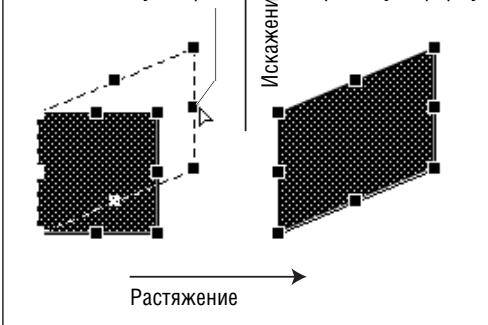
Начинающие художники быстро узнают, что имитировать глубину объектов, составленных из прямоугольников, не так уж сложно. Нужно лишь направить противоположные стороны вдоль воображаемых прямых, которые сходятся в удаленной точке горизонта — *точке схода*. При этом возникает иллюзия отдаления объекта. Изображение непрямоугольных форм в перспективе требует больше опыта и умения представить, как эти формы должны выглядеть. Модификатор **Distort** инструмента «Свободное преобразование» помогает в этом, так как заключает выбранную форму — круг, овал или произвольную кривую — в прямоугольник. Вам остается лишь подправить этот прямоугольник.

Тянуть за угловой манипулятор... ... чтобы исказить форму



**Рис. 4.35.** Используйте модификатор Distort инструмента «Свободное преобразование», чтобы изменить форму охватываемого элемента прямоугольника. Можно тянуть за каждый угловой манипулятор независимо от других.

Тянуть за боковой манипулятор... ... чтобы исказить и растянуть форму



**Рис. 4.36.** Если отмечен модификатор Distort, то буксировка боковых манипуляторов искажает выбранные элементы. Чтобы растянуть (сжать) элемент одновременно с искажением, тяните манипулятор в направлении от центра (к центру) исходной формы.

4. Чтобы изменить форму охватываемого прямоугольника, выполните одно из следующих действий:

- для перемещения одного угла охватываемого прямоугольника, подведите курсор к угловому манипулятору и отбуксируйте его в нужное положение. Все манипуляторы можно позиционировать независимо друг от друга (рис. 4.35);
- для перекашивания элемента подведите курсор к боковому манипулятору и отбуксируйте его в нужное положение. Элемент перекашивается в направлении буксировки;
- для растяжения элемента в процессе перекашивания, тяните манипулятор в направлении от центра (рис. 4.36);
- для сжатия элемента в процессе перекашивания, тяните манипулятор в направлении к центру.

5. Отпустите кнопку мыши.

**С**

Если выбрано несколько объектов-рисунков, то модификатор **Distort** инструмента «Свободное преобразование» неактивен. Чтобы исказить сразу несколько объектов-рисунков, необходимо объединить их в один объект (см. раздел «Конвертирование типов форм» ниже в этой главе).

**С**

Если выбрано несколько комбинированных форм и один объект-рисунок, то модификатор **Distort** доступен, и их можно исказить все сразу.

### Чтобы симметрично искажить графический элемент:

1. Выполните шаги 1 и 2 из предыдущего упражнения, чтобы подготовить элемент к искажению.
2. Чтобы выполнить коническое сведение элемента, выполните одно из следующих действий:

- чтобы верхняя сторона охватывающего прямоугольника стала уже нижней, нажмите клавишу **Shift** и, не отпуская ее, тяните за правый верхний угол в направлении левого верхнего угла или наоборот (рис. 4.37);
- чтобы верхняя сторона охватывающего прямоугольника стала шире нижней, нажмите клавишу **Shift** и, не отпуская ее, тяните за правый верхний угол в направлении от левого верхнего угла или наоборот.

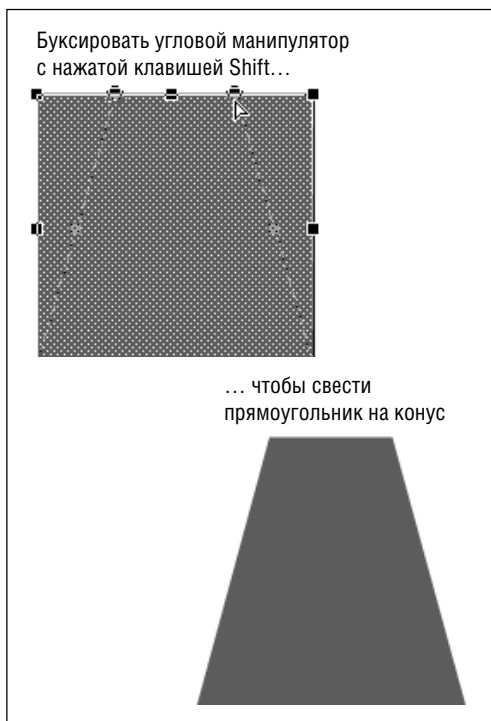
В процесс буксировки оба угловых манипулятора движутся синхронно, сближаясь или расходясь в разные стороны.

3. Отпустите кнопку мыши.

Flash перерисовывает охватывающий прямоугольник и все, что в нем находится. Если вы тянули манипуляторы в разные стороны, то новый контур сужается книзу, в противном случае — кверху. То же самое можно проделать с любой другой стороной прямоугольника и тем самым свести его на конус в любом направлении.

**C** Чтобы временно превратить курсор в полую стрелку, не выбирая модификатор **Distort**, нажмите клавишу **⌘** (Mac) или **Ctrl** (Windows). После этого можно будет буксировать с нажатой или не нажатой клавишей **Shift** с целью искажения выбранных элементов.

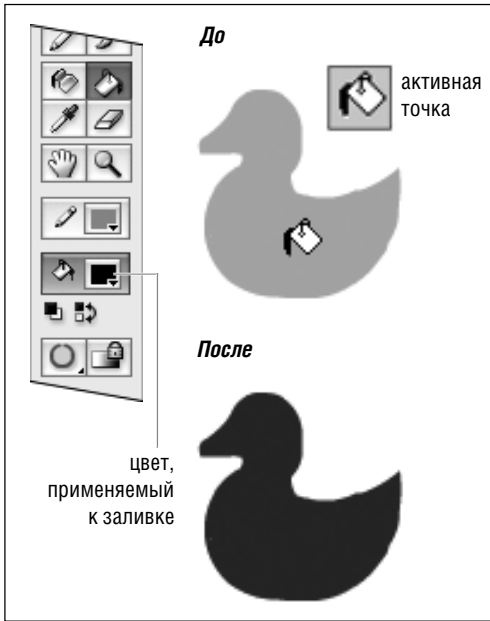
**C** Если при искажении графического элемента вы допустите ошибку и выполните команду **Edit** ⇒ **Undo**, чтобы отменить операцию, то элемент окажется выбранным инструментом «Свободное преобразование», но модификатор **Distort** будет неактивен. Чтобы возобновить искажение, придется вручную отметить его снова.



**Рис. 4.37.** Отметив модификатор **Distort** инструмента «Свободное преобразование», нажмите клавишу **Shift** и, не отпуская ее, буксируйте угловой манипулятор, чтобы свести охватывающий прямоугольник на конус.

**C** Хотя инструмент «Свободное преобразование» и применим к текстовым полям, группам и символам, но модификатор **Distort** для них не работает. Если среди выбранных объектов есть комбинированные формы и хотя бы одно текстовое поле, группа или символ, то модификатор **Distort** будет доступен, но искажаться будут только комбинированные формы.

**C** Инструмент «Свободное преобразование» можно использовать для искажения сразу нескольких комбинированных форм. Выберите те формы, которые хотите модифицировать и с помощью инструмента «Свободное преобразование» в режиме искажения измените вид охватывающего прямоугольника. Все находящиеся внутри формы изменяются как единое целое.



**Рис. 4.38.** Щелчок инструментом «Ведро с краской» по заливке применяет к ней цвет, заданный в элементе управления цветом заливки. С помощью этого приема можно изменять и существующие заливки.

## Модификация заливок и обводок

Flash предоставляет два способа модификации существующих заливок и обводок. Можно изменить атрибуты с помощью инструментов (ведро с краской для заливок и чернильница для обводок) или выбрать заливку либо обводку на арене, а затем ввести новые атрибуты в соответствующей панели. Для заливок и обводок с градиентом можно также изменить способ закрасивания формы градиентом с помощью инструмента «Преобразование градиента».

### Чтобы изменить цвет заливки с помощью ведра с краской:

1. На панели инструментов выберите ведро с краской или нажмите клавишу **K**.
2. Задайте новые атрибуты (глава 2).
3. Щелкните активной точкой ведра с краской (капелька на конце струи краски) в любой точке заливки, которую собираетесь изменить. Заливка может быть как выбранной, так и не выбранной. Цвет заливки изменится (рис. 4.38).

**С**

Если выбрано несколько комбинированных форм, то щелчок по любой из них ведром с краской изменяет цвет заливки и во всех остальных. Чернильница работает точно так же в применении к обводкам комбинированных форм (см. раздел «Чтобы изменить цвет обводки с помощью чернильницы» ниже). Однако, если выбрано несколько объектов-рисунков или примитивных форм, то с помощью этих инструментов невозможно за одну операцию изменить все заливки или обводки; придется либо щелкать по каждой в отдельности, либо воспользоваться панелью (см. врезку «Использование панелей для изменения выбранных заливок и обводок» ниже).

**С**

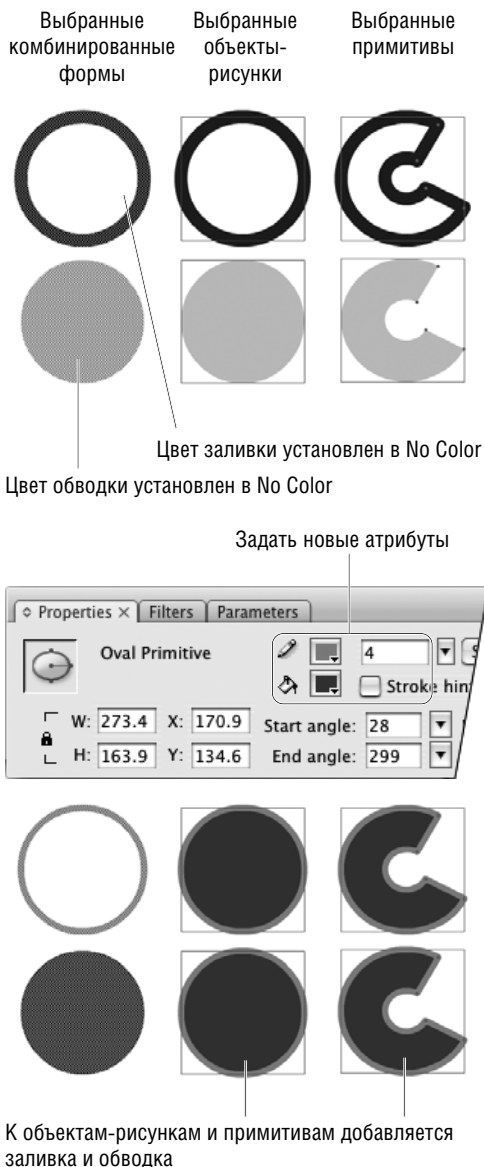
Чтобы выбрать цвет обводки или заливки и в дальнейшем использовать его как для обводок, так и для заливок, щелкните инструментом «Пипетка», удерживая клавишу **Shift**. Выбранный цвет загружается в элементы управления цветом заливок и обводок на панелях инструментов и цветов, а также на вкладках **Properties** в инспекторе свойств.

### Использование панелей для изменения выбранных заливок и обводок

Атрибуты заливки и обводки выбранного графического элемента можно изменить путем задания значений параметров на соответствующей панели. Нарисуйте, к примеру, овал с красной заливкой и сплошной синей обводкой шириной 1 пункт. Затем выберите всю форму (если овал создан как объект-рисунок или как примитив, то Flash выберет его автоматически). Теперь зайдите на вкладку **Properties** в инспекторе свойств; там отображаются атрибуты выбранной формы (или объекта-рисунка либо примитива). Сделайте элемент управления цветом заливки зеленым, овал перекрасится в зеленый цвет. Увеличьте ширину обводки или измените ее стиль, овал перерисует-ся с новыми параметрами.

Новые цвета выбранных заливок и обводок можно задавать на любой подходящей панели: цветов, образцов, инструментов или в инспекторе свойств. Если цвет обводки и заливки выбранной комбинированной формы установлен в No Color (Без цвета), то изменить его можно единственным способом — добавить заливку или обводку с помощью инструмента «Ведро с краской» или «Чернильница» соответственно (см. главу 2).

Если выбранная форма является объектом-рисунком или примитивом, то изменение атрибута заливки или обводки на панели приведет к добавлению недостающего элемента (рис. 4.39).

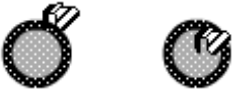


**Рис. 4.39.** Изменение атрибутов заливки или обводки объекта-рисунка или примитивной формы, для которых первоначально был установлен цвет No Color, заставит Flash добавлять недостающий элемент. Та же операция для выбранной комбинированной формы не дает никакого эффекта.

**Для комбинированных форм, объектов-рисунков и примитивов**



Если ничего не выбрано, щелкните по заливке или по обводке



Если выбраны и заливка, и обводка, щелкните по заливке или по обводке

**Только для комбинированных форм**



Если выбрана только заливка, щелкните по обводке



Если выбрана часть обводки, щелкните по выбранному сегменту



Предупреждение: щелчок по выбранной заливке никак не отражается на невыбранной обводке

**Рис. 4.40.** Необязательно выбирать обводку, чтобы изменить ее атрибуты. Достаточно щелкнуть по обводке или невыбранной заливке инструментом «Чернильница». Предупреждение: если выбрана заливка, то необходимо щелкать по самой обводке, щелчок по выбранной заливке не изменяет атрибуты невыбранной обводки.

**Чтобы изменить цвет обводки с помощью чернильницы:**

1. На панели инструментов выберите чернильницу или нажмите клавишу S.
2. Задайте новые атрибуты обводки (см. главу 2).
3. Щелкните активной точкой чернильницы (капля на конце вытекающей струи) одним из следующих способов:

- непосредственно по обводке;
- если у формы есть и обводка, и заливка, и обе выбраны, то по заливке;
- если у формы есть и обводка, и заливка, и выбрана только заливка, то по обводке;

К обводке применяются новые атрибуты (рис. 4.40).

**С**

Для экономии времени можно скопировать атрибуты заливки и обводки одного элемента и применить их к другому. На панели инструментов выберите пипетку. Чтобы скопировать цвет или градиент заливки, щелкните пипеткой в любой точке заливки. Чтобы скопировать все атрибуты обводки, щелкните пипеткой по обводке. Flash переключает инструменты: для заливок текущим становится инструмент «Ведро с краской», а для обводок — «Чернильница». Атрибуты элемента, по которому вы щелкали, появятся во всех соответствующих панелях. Так, тип и цвет заливки будут видны на панелях инструментов, цветов и на вкладке **Properties** в инспекторе свойств. Загруженное ведро с краской или чернильницу можно теперь использовать для применения этих атрибутов к различным графическим элементам.

# Чтобы изменить центр градиента:

1. На панели инструментов выберите из списка инструментов преобразования инструмент «Преобразование градиента» (рис. 4.41).
2. Укажите курсором на графический элемент, для которого хотите модифицировать градиент (заливку или обводку). Курсор принимает форму инструмента преобразования градиента.
3. Щелкните мышью.

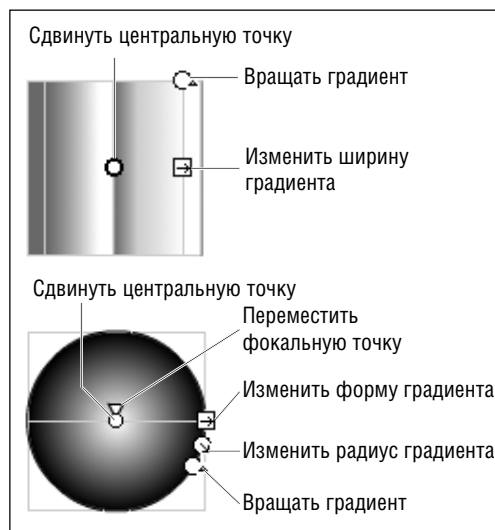
Появляются манипуляторы для изменения градиента (рис. 4.42) Можно вращать градиент, а также изменять его размер или центр.

4. Наведите курсор на центральный манипулятор в виде кружочка. Появляется значок перемещения.
5. Потяните за центральный манипулятор, чтобы изменить положение центра градиента (рис. 4.43).

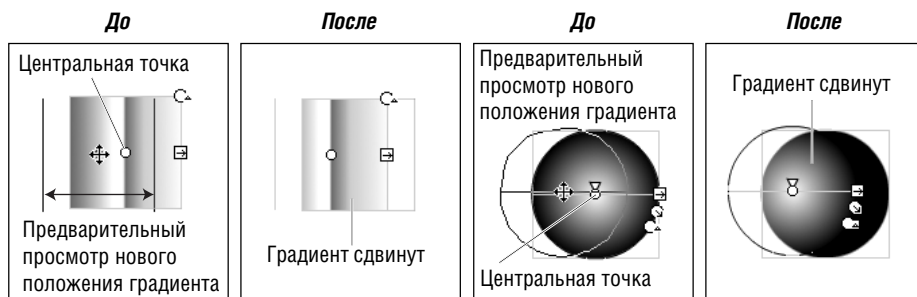
**С** При создании графического элемента с заблокированным градиентом Flash по умолчанию размещает градиент так, что его центр находится на левом краю арены. Если масштаб арены достаточно велик, то при выборе такого элемента инструментом преобразования градиента все манипуляторы могут оказаться вне видимой части арены. В таком случае создается впечатление, будто инструмент не работает. Если при щелчке по градиенту инструментом преобразования ничего не происходит, попробуйте уменьшить масштаб. Обычно при масштабе 50% все манипуляторы выбранного градиента видны.



**Рис. 4.41.** Инструменты «Свободное преобразование» и «Преобразование градиента» находятся в одном меню на панели инструментов. Выберите инструмент преобразования градиента для манипулирования внешним видом градиента заливок и обводок.

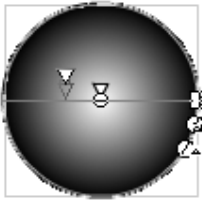
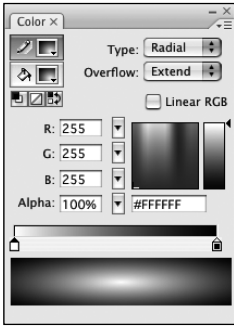


**Рис. 4.42.** При щелчке по градиенту инструментом преобразования градиента появляются манипуляторы.

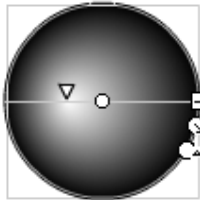


**Рис. 4.43.** Буксируйте центральный манипулятор, чтобы изменить положение центра градиента, которым закрашена форма.



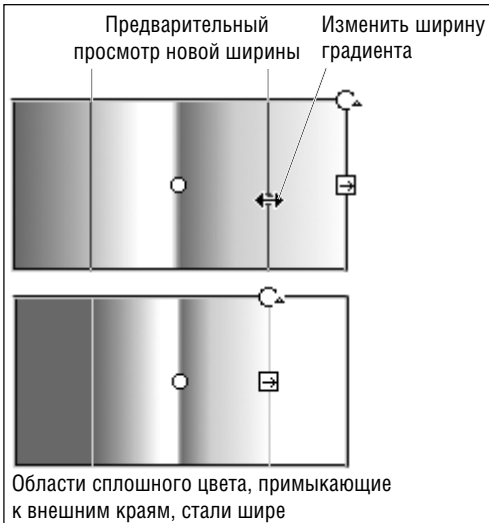


Предварительный просмотр нового положения фокальной точки



Фокальная точка сдвинута

**Рис. 4.44.** В этом градиенте происходит переход от белого цвета (слева) к черному (справа). Белый цвет в центре радиального градиента создает иллюзию освещенного трехмерного объекта. Меняя положение фокальной точки, можно имитировать перемещение источника света.



Области сплошного цвета, примыкающие к внешним краям, стали шире

**Рис. 4.45.** Для линейного градиента перетаскивание квадратного манипулятора ближе к центру сужает область перехода цветов.

### Чтобы изменить фокальную точку радиального градиента:

1. Выполните шаги 1-3 из предыдущего упражнения.
2. Наведите инструмент преобразования градиента на треугольный фокальный манипулятор. Курсор примет форму треугольника.
3. Перетащите фокальный манипулятор в новое положение (рис. 4.44).

Фокальная точка, вокруг которой концентрация центрального цвета градиента особенно велика, перемещается в новое место.

### Чтобы изменить градиент заливки или обводки:

4. Выбрав инструмент преобразования градиента, щелкните по элементу, градиент которого хотите модифицировать.
5. Чтобы изменить способ закрасивания заливки или обводки градиентом, выполните одно из следующих действий:

- для изменения ширины линейного градиента буксируйте квадратный манипулятор (рис. 4.45).

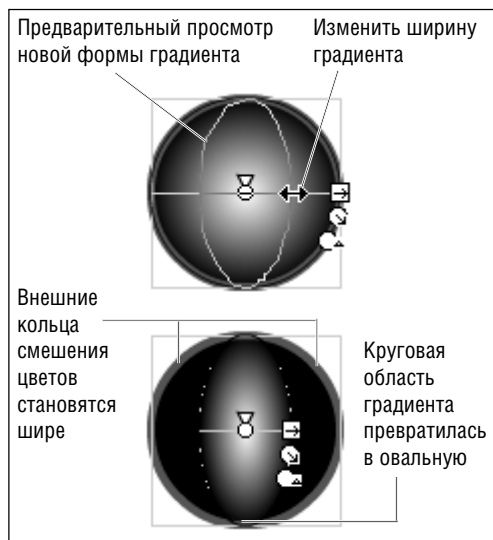
Курсор принимает форму двунаправленной стрелки. Буксировка в направлении к центру формы уменьшает ширину полосы, внутри которой происходит полный переход цветов, в направлении от центра — увеличивает.

- для изменения формы радиального градиента буксируйте квадратный манипулятор (рис. 4.46).

Курсор принимает форму двунаправленной стрелки. Буксировка в направлении к центру формы сплющивает овал, внутри которого происходит полный переход цветов, в направлении от центра — растягивает.

- для изменения радиуса радиального градиента буксируйте круглый манипулятор, который находится рядом с квадратным (рис. 4.47).

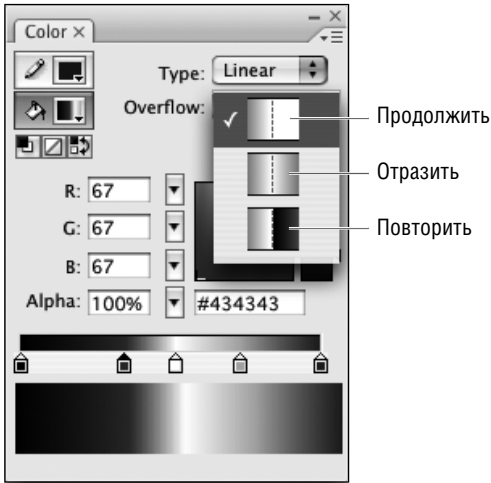
Курсор принимает форму стрелки внутри круга. Буксировка в направлении к центру формы уменьшает радиус овала, внутри которого происходит полный переход цветов, в направлении от центра — увеличивает.



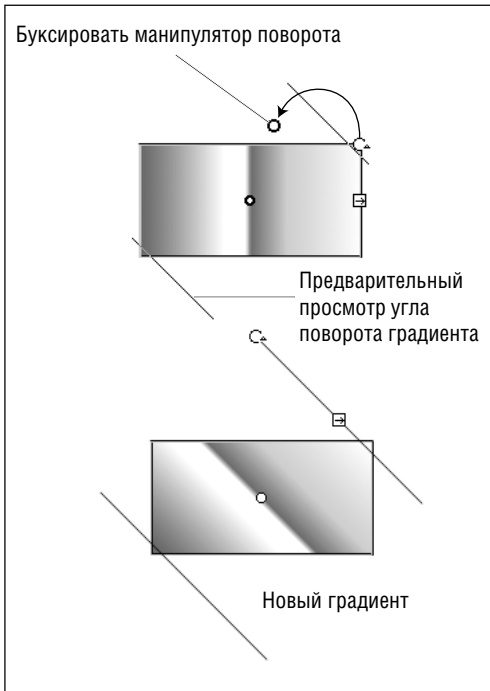
**Рис. 4.46.** Чтобы сплющить овал, в котором изменяются цвета, буксируйте квадратный манипулятор в сторону центра.



**Рис. 4.47.** Чтобы увеличить радиус градиента, буксируйте первый круглый манипулятор в сторону от центра.



**Рис. 4.48.** Если установлен режим публикации для Flash-плеера версии 8 или выше (см. главу 17), то на панели цветов присутствует меню **Overflow**. Вы можете указать, как должна закрашиваться форма, более широкая, чем сам градиент.



**Рис. 4.49.** Буксируя манипулятор поворота инструментом преобразования градиента, вы поворачиваете градиент вокруг его центра.

### Чтобы контролировать переполнение:

1. Выполните шаги, описанные в предыдущем упражнении, чтобы создать градиент, который уже содержит его формы.
2. Выполните одну из следующих команд в меню **Overflow** (Переполнение) на панели цветов (рис. 4.48):

- чтобы дополнять граничными цветами градиента, выполните команду **Extend** (Продолжить) (первый пункт меню);
- чтобы повторить градиент, инвертировав последовательность цветов, выполните команду **Reflect** (Отразить) (второй пункт меню);
- чтобы повторить градиент с той же последовательностью цветов, выполните команду **Repeat** (Повторить);

Отметим, что меню **Overflow** видно только тогда, когда установлен режим публикации для проигрывателя Flash Player версии 8 или выше (см. главу 17).

### Чтобы повернуть градиентную заливку:

1. Выбрав инструмент преобразования градиента, щелкните по заливке или обводке, градиент которой хотите модифицировать.
2. Для вращения градиента выполните одно из следующих действий (рис. 4.49):

- чтобы повернуть линейный градиент, тяните круглый манипулятор;
- чтобы повернуть радиальный градиент, тяните круглый манипулятор, находящийся дальше всего от квадратного.

Курсор принимает форму круговой стрелки. Вращать можно как по часовой стрелке, так и против.

### С

Буксируйте ведро с краской, чтобы вращать градиент одновременно с его применением. Чтобы ограничить угол поворота 0, 45 и 90 градусами, во время буксировки удерживайте нажатой клавишу **Shift**.

## Модификация форм: инструменты естественного рисования

Любые обводки и заливки, созданные во Flash, можно впоследствии редактировать. Можно изменить путь, определяющий комбинированную форму или объект-рисунок, естественным образом — с помощью инструмента «Стрелка» или же работать непосредственно с узлами пути и кривыми Безье, пользуясь для этого инструментами «Спецвыделение» и «Перо» (см. раздел «Модификация форм: кривые Безье» ниже). Примитивные формы можно модифицировать, манипулируя их управляющими точками с помощью инструментов «Стрелка» и «Спецвыделение», либо изменяя атрибуты в инспекторе свойств.

При использовании инструмента «Стрелка» комбинированная форма или объект-рисунок, который вы собираетесь модифицировать, не должен быть выбран. Если элемент выбран, то «Стрелка» перемещает его как единое целое. Всегда обращайте внимание на то, какую форму принимает курсор, когда указывает на изменяемый путь.

Выполняя следующие упражнения, отмените выбор элемента, с которым будете работать. В этом разделе мы будем иметь дело только с обводками, но все описанные приемы годятся и для изменения формы путей заливок (см. врезку «Пути заливок» ниже).

### О гладких и угловых точках

Инструменты «Стрелка» и «Спецвыделение» позволяют модифицировать криволинейные и прямолинейные сегменты элемента. Для этого инструмент «Спецвыделение» дает возможность двигать гладкие и угловые точки, определяющие путь элемента, а также работать с манипуляторами кривых Безье. Когда вы буксируете этим инструментом эластичный прямоугольник с целью охватить путь, Flash показывает манипуляторы Безье в гладких точках кривых. (В угловых точках манипуляторов нет.)

При работе с инструментом «Стрелка» Flash скрывает все эти технические детали. Для изменения формы линии вы просто тянете ее. Конечно, «Стрелка» знает обо всех гладких и угловых точках, но проявляется это только в изменении сопровождающего стрелку значка при взаимодействии с невыбранной прямой или кривой (рис 4.50).



Над пустым местом



Над выбранной линией  
на арене



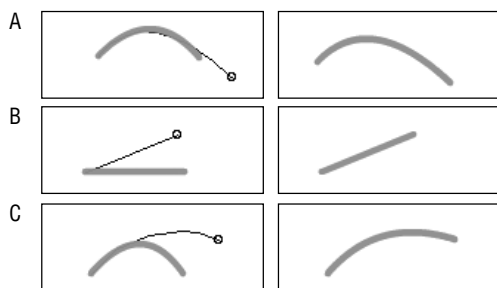
Над гладкой точкой  
невыбранной линии



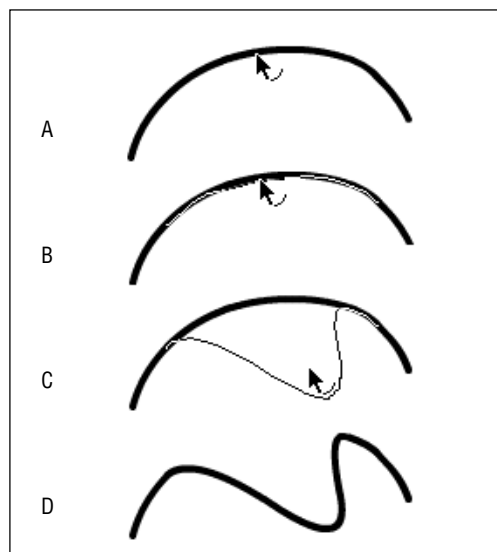
Над угловой точкой  
невыбранной линии

**Рис. 4.50.** Значки, появляющиеся рядом со стрелкой, показывают, что находится под курсором.

В случае инструмента «Стрелка» угловая точка появляется в конце невыбранного сегмента или там, где два сегмента, соединяясь, образуют резкий угол. Все, что между ними, считается гладкими точками. Если потянуть «Стрелкой» за гладкую точку, то вытянется небольшая дуга. Если же тянуть за угловую точку, то только она и перемещается.



**Рис. 4.51.** Тяните в сторону от исходного прямолинейного или криволинейного сегмента, чтобы удлинить его (А). Сдвиньте конечную точку, чтобы изменить направление сегмента (В, С).



**Рис. 4.52.** Щелкните по гладкой точке кривой (А). Flash активирует криволинейный сегмент (В). Перетащите сегмент в новое положение (С). Как только вы отпустите кнопку мыши, Flash перерисует кривую (D).

### Чтобы активировать конец сегмента инструментом «Стрелка»:

1. Наведите курсор на конечную точку невыбранного сегмента линии. Появится модификатор угловой точки.
2. Щелкните по конечной точке. Конец сегмента станет активным.
3. Переместите конечную точку одним из следующих способов:
  - отбуксируйте в сторону от существующей линии, чтобы удлинить сегмент;
  - отбуксируйте ближе к существующей линии, чтобы укоротить сегмент;
  - отбуксируйте под углом к исходной линии, чтобы поместить отрезок прямой в другое место или изменить форму криволинейного сегмента.

В процессе буксировки конечная точка превращается в кружочек, показывающий, что линия модифицируется; Flash при этом интерактивно рисует текущее положение сегмента (рис. 4.51).

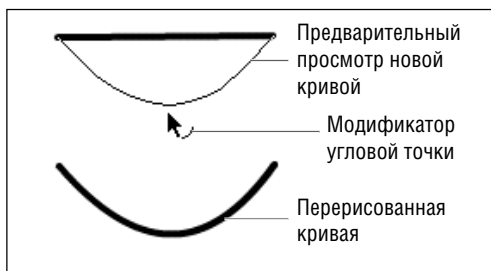
4. Отпустите кнопку мыши, и сегмент будет перерисован окончательно.

### Чтобы изменить форму кривой инструментом «Стрелка»:

1. Наведите курсор на гладкую точку невыбранного криволинейного сегмента. Появится модификатор гладкой точки.
2. Буксируя мышью, измените форму кривой (рис. 4.52). Flash рисует текущее положение сегмента.
3. После отпускания кнопки мыши сегмент будет перерисован окончательно.

**Чтобы превратить прямолинейный сегмент в криволинейный инструментом «Стрелка»:**

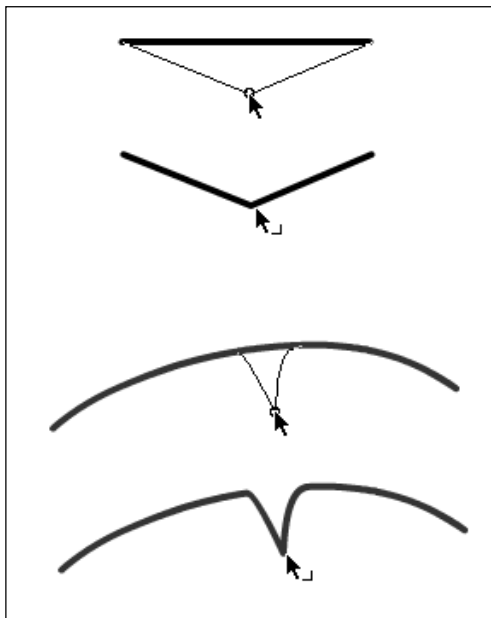
1. Наведите курсор на гладкую точку невыбранного прямолинейного сегмента. Появится модификатор гладкой точки.
2. Буксируя мышью, измените форму сегмента (рис. 4.53). Flash рисует текущее положение сегмента.
3. Отпустите кнопку мыши, и сегмент будет перерисован окончательно.



**Рис. 4.53.** Flash считает гладкими все точки прямолинейного сегмента, кроме концевых (верхний рисунок). Потянув за любую из них, вы создадите кривую линию (нижний рисунок).

**Чтобы создать новую угловую точку «Стрелка»:**

1. Наведите курсор на гладкую точку невыбранного прямолинейного или криволинейного сегмента. Появится модификатор гладкой точки.
2. Щелкните мышью, одновременно нажав клавишу **Option** (Mac) или **Ctrl** (Windows). После короткой паузы рядом со стрелкой появится модификатор угловой точки, а там, куда указывает курсор, возникнет кружочек. Вы приступили к созданию новой угловой точки.
3. Буксируя мышью, модифицируйте сегмент и добавьте новую угловую точку (рис. 4.54).



**Рис. 4.54.** Щелчок с нажатой клавишей Option (Mac) или Ctrl (Windows) создает новую угловую точку. Если потянуть в сторону от прямолинейного сегмента, образуется V-образная линия (верхний рисунок). Если исходный сегмент был криволинейным, то эта операция приводит к появлению V-образного выступа, сходящегося в одну точку (нижний рисунок).



**Рис. 4.55.** Пользуйтесь инструментом «Спецвыделение» для модификации пути.

## Модификация форм: кривые Безье

Инструмент «Спецвыделение» позволяет видеть узлы пути и манипулировать ими. Можно переместить узел, чтобы изменить путь, или, манипулируя управляющими точками кривой Безье, модифицировать наклон и крутизну кривой. С помощью трех инструментов работы с узлами или инструмента «Перо» можно добавлять и удалять их, преобразовывать гладкую точку в угловую либо наоборот. Во Flash есть два стиля отображения узлов: полый (по умолчанию) и сплошной кружок. Чтобы задать стиль, выберите пункт **Preferences** (Установки) из меню **Flash** (Mac) или **Edit** (Редактирование) (Windows). В диалоговом окне выберите категорию **Drawing** (Рисунок). В разделе **Pen Tool** (Инструмент «Перо») отметьте или сбросьте флажок **Show Solid Points** (Показывать сплошные точки) и нажмите **OK**. Во всех упражнениях ниже предполагается, что этот флажок отмечен.

### Пути заливок

Хотя заливка без обводки не имеет контура, путь, описывающий ее форму, все же существует. Инструменты «Стрелка», «Спецвыделение» и «Перо» могут менять форму пути заливки так же, как и пути обводки (рис. 4.56).



**Рис. 4.56.** Если курсор указывает на край заливки, то рядом с инструментом «Стрелка» появляется модификатор гладкой или угловой точки. Щелчок по краю заливки активирует часть пути, оконтуривающего форму (левый рисунок). Если выбрать край заливки инструментом «Спецвыделение», то будет выделен полный путь и все его узлы. Чтобы изменить форму заливки, можно сместить узлы и манипуляторы Безье (правый рисунок).

### Чтобы увидеть путь и узлы:

1. Выберите инструмент «Спецвыделение» или нажмите клавишу **A** (рис. 4.55). Курсор примет форму полую стрелки.
2. Щелкните по прямой или кривой линии на арене, которую собираетесь модифицировать. Flash выделяет путь и его узлы. Для манипулирования конкретной точкой ее нужно выбрать напрямую.

**C**

Если выбран инструмент «Спецвыделение», то команда **Select All** (Выделить все) — **⌘-A** (Mac) или **Ctrl-A** (Windows) — выбирает пути и узлы всех графических элементов на арене и монтажном столе.

### Чтобы выбрать узел:

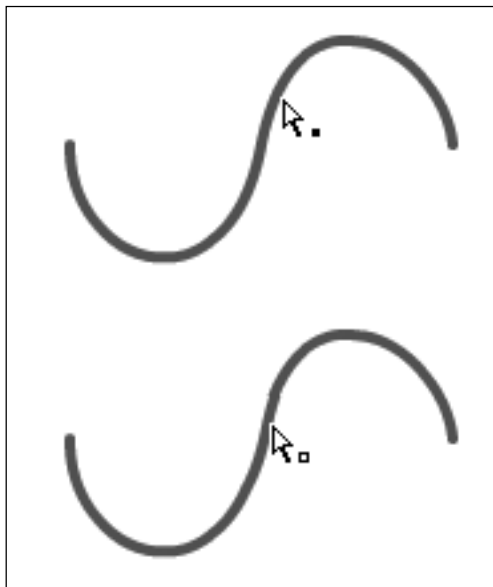
1. Выберите инструмент «Спецвыделение».
2. Наведите курсор на путь, который хотите модифицировать.

Когда курсор находится над сегментом линии, рядом с полый стрелкой появляется сплошной квадратик, а когда непосредственно над узлом — полый квадратик (рис. 4.57).

3. Щелкните по узлу. Узел выделяется, и появляются ассоциированные с ним манипуляторы Безье.

**С**

Инструмент «Спецвыделение» позволяет выбрать несколько точек на пути. Проведите эластичный прямоугольник, охватывающий нужные точки. Flash выделит весь путь и те узлы, которые попали внутрь прямоугольника.



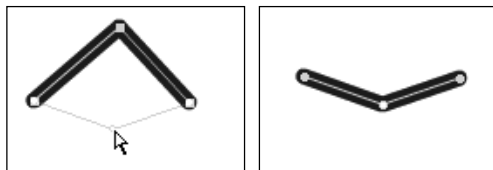
**Рис. 4.57.** Когда инструмент «Спецвыделение» готов выбрать путь целиком, рядом с ним появляется сплошной квадратик (верхний рисунок). Если квадратик полый (нижний рисунок), то инструмент готов к выбору и манипулированию единственным узлом.

### Чтобы передвинуть угловую точку:

1. С помощью инструмента «Спецвыделение» выделите путь и узлы элемента, который собираетесь модифицировать.
2. Наведите курсор на угловую точку.
3. Перетащите угловую точку в новое место. Flash перерисовывает путь (рис. 4.58).

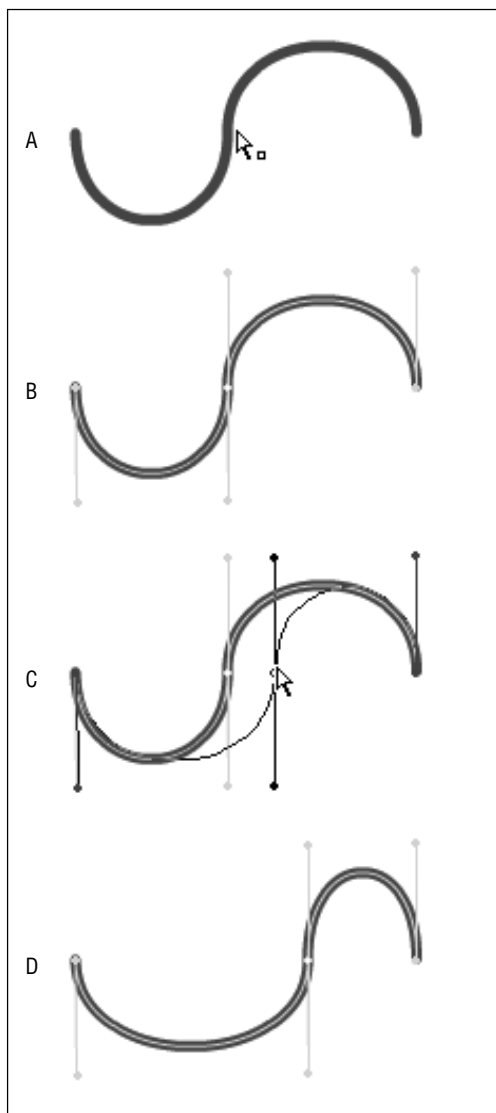
**С**

Угловые точки часто легко опознать и не выделяя путь. Перетаскивать их можно, даже если путь предварительно не выделен. Но если вы щелкнете не точно по точке, то перемещаться будет весь путь, а не только нужная вам точка.



**Рис. 4.58.** С помощью инструмента «Спецвыделение» перетащите угловую точку в новое место.





**Рис. 4.59.** При выборе узла (A) Flash выделяет весь путь (B). Для модификации пути можно перетаскивать узел (C). После отпущания кнопки мыши и путь, и узел остаются выделенными (D).

### Чтобы переместить гладкую точку:

1. С помощью инструмента «Спецвыделение» выделите путь и узлы элемента, который собираетесь модифицировать.
2. Наведите курсор на гладкую точку. Появится модификатор узла.
3. Перетащите точку в новое место (рис. 4.59). Во время перетаскивания Flash рисует текущую кривую. После того как гладкая точка перемещена, путь остается выделенным, а манипуляторы Безье активными, так что манипуляции с кривой можно продолжать.

### Чтобы изменить форму кривой с помощью манипуляторов Безье:

1. Щелкните инструментом «Спецвыделение» по кривой, которую собираетесь модифицировать.
2. Щелкните по любому узлу на кривой. Появляются манипуляторы Безье.
3. Потяните за какой-нибудь из манипуляторов Безье. Курсор принимает форму сплошной стрелки.
4. Для модификации кривой выполните одно из следующих действий:
  - чтобы сделать кривую более крутой, отведите манипулятор Безье подальше от кривой в направлении горба;
  - чтобы сделать кривую более пологой, подведите манипулятор ближе к ней;
  - чтобы изогнуть кривую в противоположном направлении, тяните манипулятор в сторону, противоположную текущему направлению горба.

- чтобы увеличить глубину горба, ответьте манипулятор дальше от узла;
- чтобы уменьшить глубину горба, подведите манипулятор ближе к узлу;

В процессе манипулирования Flash рисует текущее положение кривой (рис. 4.60).

С

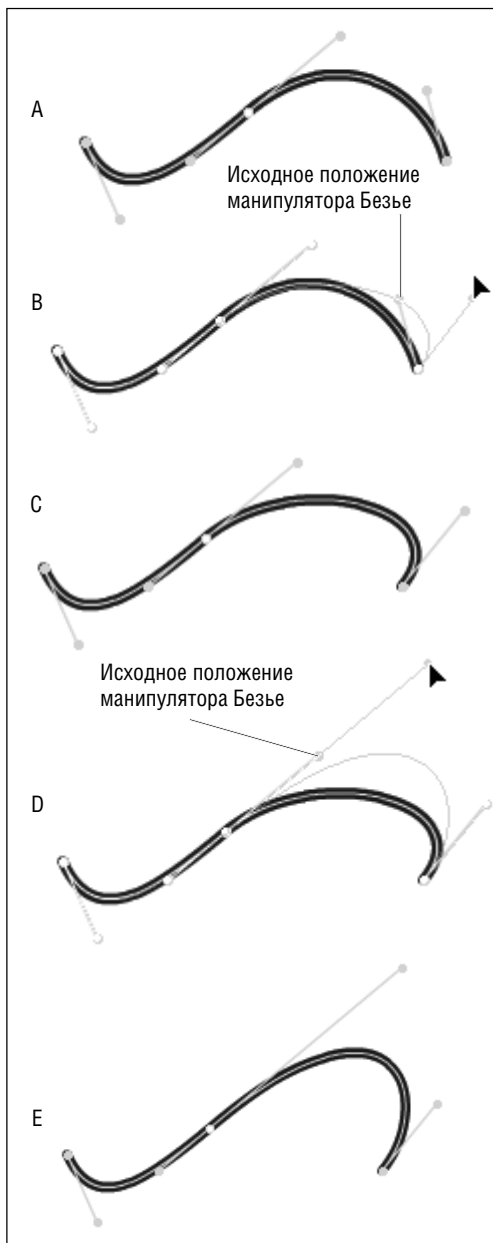
У гладкой точки на стыке двух криволинейных сегментов есть два манипулятора Безье. По умолчанию они действуют согласованно, так что изменяются кривые по обе стороны от узла. Но можно работать и с одним манипулятором (с одной кривой). Для этого удерживайте нажатой клавишу **Option** (Mac) или **Alt** (Windows) при буксировке манипулятора инструментом «Спецвыделение». Или в момент, когда узел выбран и манипуляторы Безье активны, выберите инструмент «Преобразовать узловую точку». Курсор в форме крышки будет буксировать один манипулятор независимо от другого.

С

Чтобы быстро выбрать узел и активировать его манипуляторы Безье, нарисуйте инструментом «Спецвыделение» эластичный контур, охватывающий кривую, которую хотите модифицировать. Даже если путь не был выделен, Flash выберет все кривые и узлы, попавшие внутрь контура, и активирует их манипуляторы.

С

Выбранные узлы можно перемещать клавишами со стрелками. Чтобы увеличить шаг перемещения, удерживайте нажатой клавишу **Shift**.



**Рис. 4.60.** Когда выбраны какие-то узлы, появляются их манипуляторы Безье (А). Если отклонить манипулятор от кривой (В), та становится более крутой (С). Если приблизить манипулятор к кривой, она станет более пологой. Если оттащить манипулятор дальше от узла (D), то глубина горба увеличится (Е), а если подтащить поближе, — уменьшится.

## Режимы инструмента «Перо»

Во Flash CS3 появилось три новых инструмента для работы с кривыми Безье: «Добавить узловую точку», «Удалить узловую точку» и «Преобразовать узловую точку». В предыдущих версиях все это делал инструмент «Перо». Он по-прежнему способен выполнять многие из этих операций. Когда перо находится над ареной или над существующим путем, рядом с курсором появляются различные модификаторы, обозначающие текущую функцию пера (рис. 4.61).

**Создать начальный узел.** Маленькая буква х означает, что перо готово поместить первую точку пути. Чтобы начать создание пути, щелкните по пустому месту на арене.

**Создать последующие узлы.** В процессе добавления в путь дополнительных узлов у пера нет никаких модификаторов. Щелчок по арене в этом режиме создаст новый узел и сегмент.

**Добавить узел.** Когда курсор указывает на сегмент выбранного пути между узлами, рядом с пером появляется плюси́к. Щелчок в этом режиме добавит новый узел в существующий путь. Если щелкнуть между двумя угловыми точками, добавится новая угловая точка, а если между двумя гладкими (или между угловой и гладкой) — новая гладкая точка.

**Преобразовать гладкую точку в угловую.** Когда курсор указывает на гладкую точку выбранного пути, рядом с ним появляется крышка (уголок). Если в этот момент щелкнуть мышью, гладкая точка преобразуется в угловую.

**Удалить угловую точку.** Когда курсор указывает на угловую точку выбранного пути, рядом с ним появляется минус. Если в этот момент щелкнуть мышью, угловая точка будет удалена.

**Продолжить путь.** Когда курсор указывает на конечной узел (первую или последнюю точку незамкнутого пути), рядом с ним появляется косая черта. Щелчок привязывает перо к существующему пути; последующие щелчки по арене добавля́ют в путь новые узлы.

**Замкнуть путь.** Когда курсор указывает на начальный узел создаваемого пути, рядом с ним появляется кружочек. (Этот модификатор появляется и тогда, когда перо используется для создания пути комбинированной формы, и курсор указывает на конечную точку в пути другой комбинированной формы.)

**Соединить с путем объекта-рисунка.** Когда вы создаете новый путь, и курсор указывает на один из конечных узлов пути существующего объекта-рисунка, рядом с ним появляется значок звена цепи.



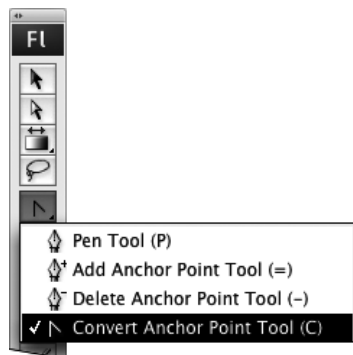
**Рис. 4.61.** Форма курсора, соответствующего инструменту «Перо», изменяется, информируя о том, какова функция инструмента в данный момент.

### Чтобы преобразовать угловую точку во гладкую:

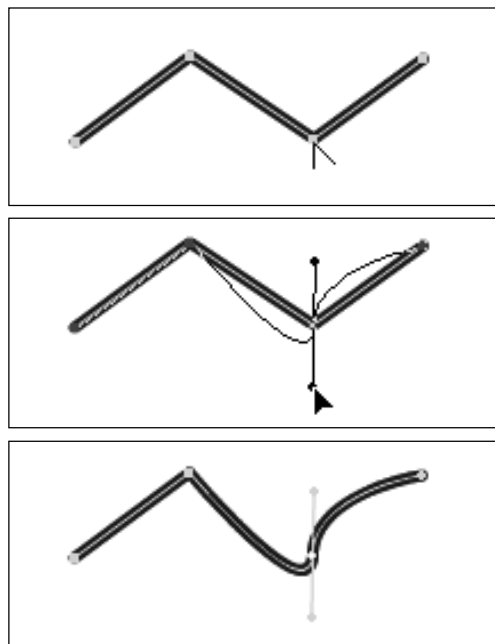
1. С помощью инструмента «Спецвыделение» выделите путь и узлы, которыми хотите модифицировать.
2. На панели инструментов щелкните по текущему инструменту Безье и в открывшемся меню выберите инструмент «Преобразовать узловую точку» (рис. 4.62). Курсор принимает форму крышки.
3. Укажите курсором на угловую точку.
4. Чтобы активировать манипуляторы Безье, щелкните мышью и буксируйте. Flash преобразует угловую точку в гладкую с манипуляторами Безье (рис. 4.63). В процессе буксировки манипуляторы расходятся и двигаются, изменяя форму кривой.

**С**

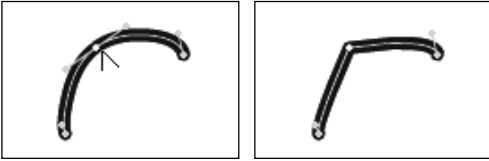
Преобразовать угловую точку в гладкую можно также с помощью инструмента «Спецвыделение». Наведите курсор, имеющий форму полый стрелки, на выбранную угловую точку, а затем тяните его в сторону от этой точки, удерживая клавишу **Option** (Mac) или **Alt** (Windows); возникнут манипуляторы Безье.



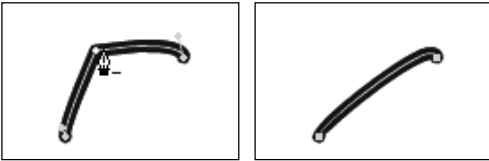
**Рис. 4.62.** Во Flash CS3 есть четыре инструмента для модификации узлов и кривых Безье, принадлежащих пути. Щелкните по текущему инструменту Безье на панели инструментов, и в открывшемся меню увидите весь набор.



**Рис. 4.63.** Чтобы преобразовать угловую точку в гладкую (с манипуляторами Безье) с помощью инструмента «Преобразовать узловую точку», щелкните (верхний рисунок) и потяните угловую точку. В этом случае точка не сдвинется с места, зато для нее появятся манипуляторы (средний рисунок). Когда вы отпустите кнопку мыши, Flash перерисует кривую (нижний рисунок).



**Рис. 4.64.** Щелчок по гладкой точке инструментом «Преобразование узловой точки» (левый рисунок) превращает ее в угловую (правый рисунок). Путь перерисовывается.



**Рис. 4.65.** Щелчок по узлу инструментом «Удаление узловой точки» удаляет этот узел. Путь перерисовывается.

### Чтобы преобразовать гладкую точку в угловую:

1. С помощью инструмента «Спецвыделение» выделите путь, который хотите модифицировать.
2. На панели инструментов из меню инструментов Безье выберите инструмент «Преобразовать узловую точку».
3. Наведите курсор в форме крышки на гладкую точку.
4. Щелкните по гладкой точке.

Flash преобразует гладкую точку в угловую, удаляя при этом манипуляторы Безье и сглаживая криволинейный путь (рис. 4.64).

### С

Работая с инструментом «Перо», вы можете временно обратиться к инструменту преобразования узла, удерживая клавишу **Option** (Mac) или **Alt** (Windows).

### Чтобы удалить узел:

1. С помощью инструмента «Спецвыделение» выделите путь, который хотите модифицировать.
2. На панели инструментов щелкните по текущему инструменту Безье и в открывшемся меню выберите инструмент «Удалить узловую точку». Курсор принимает форму пера с модификатором удаления (знак минус).
3. Укажите курсором на узел и щелкните мышью. Flash удаляет узел и изменяет форму пути между соседними с ним узлами (рис. 4.65).

### С

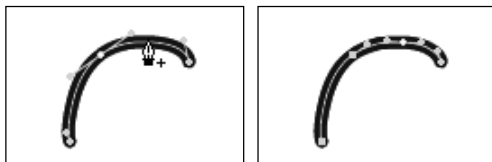
Чтобы временно обратиться к инструменту удаления узла при работе с инструментом «Добавление узловой точки», нажмите и удерживайте клавишу **Option** (Mac) или **Alt** (Windows).

### С

Удалить узлы в режиме работы с инструментом «Спецвыделение» можно также клавишами **Backspace** или **Delete**, но результат может показаться вам обескураживающим. Если выбранный узел соединяет два сегмента, то нажатие **Delete** удаляет узел и изменяет форму пути. Однако, если узел лежит на пересечении трех или более сегментов, то удаляется не только он сам, но и все примыкающие к нему прямо- и криволинейные сегменты. Поэтому так можно удалить больше, чем хотелось.

### Чтобы добавить точку в путь:

1. С помощью инструмента «Спецвыделение» выделите путь, который хотите модифицировать.
  2. На панели инструментов выберите инструмент Безье »Добавить узловую точку». Курсор принимает форму пера с модификатором плюс.
  3. Наведите курсор на путь и выполните одно из следующих действий:
    - щелкните между двумя угловыми точками, чтобы создать новую угловую точку;
    - щелкните между двумя гладкими точками, чтобы создать новую гладкую точку;
    - щелкните между угловой и гладкой точкой, чтобы создать новую гладкую точку.
- Flash добавляет новую точку (рис. 4.66).



**Рис. 4.66.** Если перо находится между двумя узлами, рядом с ним появляется плюси́к (левый рисунок). В этом режиме щелчок по пути добавляет новый узел (правый рисунок).

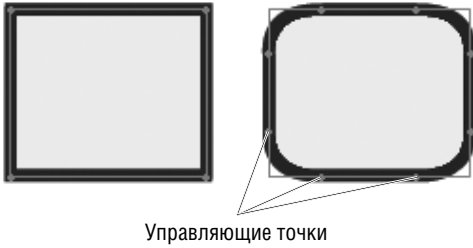
**С** Чтобы временно обратиться к инструменту добавления узла при работе с инструментом «Удаление узловой точки», нажмите и удерживайте клавишу **Option** (Mac) или **Alt** (Windows).

### Чтобы продолжить путь:

1. На панели инструментов выберите инструмент «Перо».
2. Укажите курсором на любой *концевой узел*. Появляется модификатор продолжения пути — значок косой черты.
3. Щелкните по концевому узлу. Перо присоединяется к нему, как если бы вы только что добавили точку.
4. Щелкайте мышью для добавления последующих узлов, как было описано в главе 2.

**С** Для быстрого переключения между любым из инструментов Безье и инструментом «Спецвыделение» можно воспользоваться клавиатурой: клавиша **A** выбирает инструмент «Спецвыделение», **P** — инструмент «Перо», **=** — инструмент добавления узла, **-** (минус) — инструмент удаления узла, **C** — инструмент преобразования узла.

**С** Присоединить новый путь к существующему можно, если путь и перо находятся в одном и том же режиме рисования. С помощью пера расставьте узлы пути. Когда будете готовы соединить пути, укажите курсором на узел того пути, который хотите продолжить. Рядом с пером появляется один из двух модификаторов. Если перо работает в режиме рисования комбинированных форм, и вы пытаетесь присоединиться к пути комбинированной формы, то появится модификатор замыкания пути (полный кружок). Если перо работает в режиме рисования объектов, и вы пытаетесь присоединиться к пути объекта-рисунка, то появится модификатор соединения путей (звено цепи). Щелкните по узлу старого пути, и Flash соединит два однотипных пути.



**Рис. 4.67.** Для прямоугольного примитива имеется по две управляющих точки с каждой стороны скругленного угла. Если радиус закругления равен 0, то эти точки совмещены.

## Модификация путей примитивных форм

Инструменты «Прямоугольный примитив» и «Овальный примитив» создают формы с предопределенными путями, на которые наложены ограничения. Свойства формы вы задаете сами. Произвольно изменить контур примитивной формы, как в случае комбинированных форм и объектов-рисунков, невозможно (например, нельзя ни добавить, ни переместить точки на пути, описывающем примитивный овал, и тем самым превратить его в бесформенную кляксу). Изменять свойства, характеризующие примитив, можно путем буксировки управляющих точек на самой форме или задавая новые значения в инспекторе свойств.

### Чтобы интерактивно изменить свойства прямоугольного примитива:

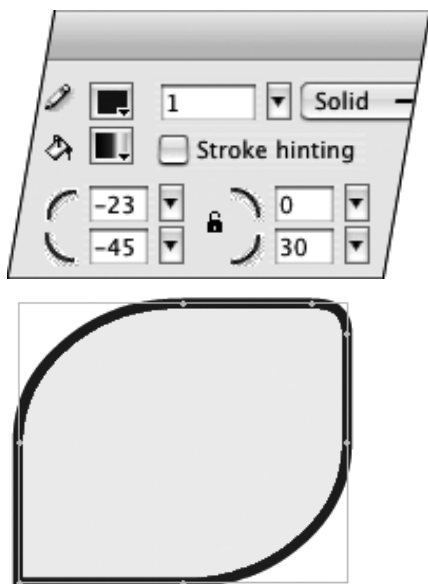
1. С помощью инструмента «Стрелка» или «Спецвыделение» выберите на арене интересующий вас прямоугольный примитив. Появляется охватывающий прямоугольник с управляющими точками (рис. 4.67). Вблизи каждого угла есть две управляющие точки. Если угол резкий, то есть радиус закругления равен 0, то эти точки совмещены. В случае же скругленного угла управляющие точки находятся по обе стороны угловой дуги. Эти точки работают синхронно — буксировка одной приводит к перемещению другой.
2. Укажите курсором на одну из управляющих точек. Курсор принимает форму сплошной стрелки без хвостика.

3. Чтобы модифицировать форму, выполните одно из следующих действий:

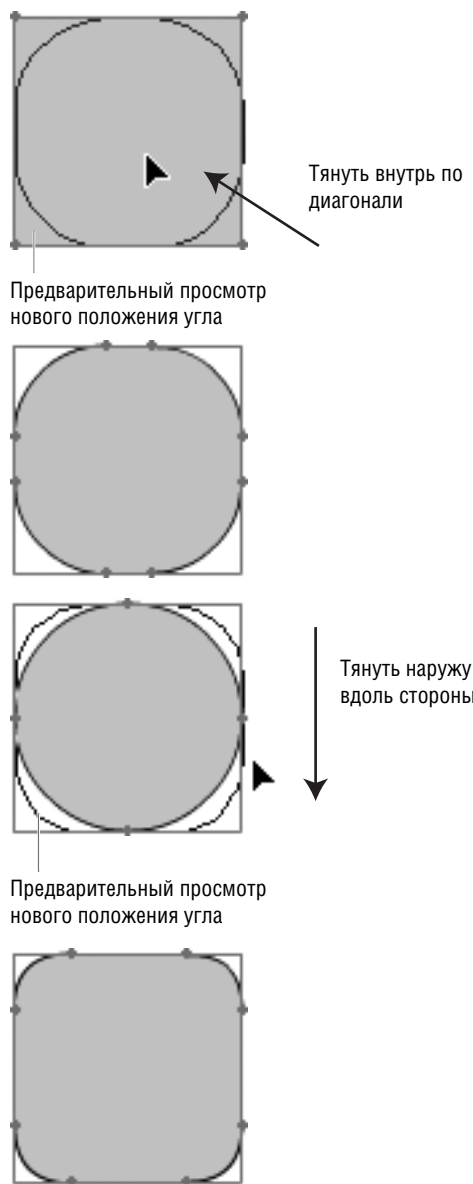
- для увеличения радиуса закругления буксируйте точку внутрь прямоугольника;
- для уменьшения радиуса закругления буксируйте точку наружу;

Можно буксировать как по диагонали в направлении центра или от него, так и по вертикали или горизонтали в любую сторону (рис. 4.68).

**С** По умолчанию Flash ограничивает радиус закругления угла так, чтобы во всех четырех углах радиус был одинаков. Чтобы создать углы с разными радиусами закругления, выберите инструмент «Прямоугольный примитив», перейдите на вкладку **Properties** в инспекторе свойств и щелкните по значку ограничения радиуса угла, так чтобы запертый замок стал открытым. Теперь можно буксировать управляющие точки каждого угла независимо друг от друга (рис. 4.69).

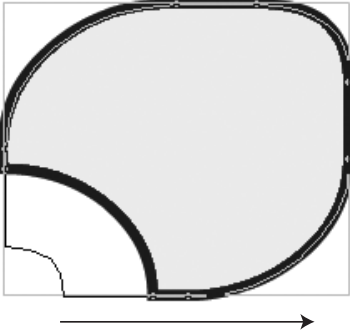


**Рис. 4.69.** Если разблокировать кнопку ограничения радиуса угла, то для каждого угла можно будет ввести свой радиус.

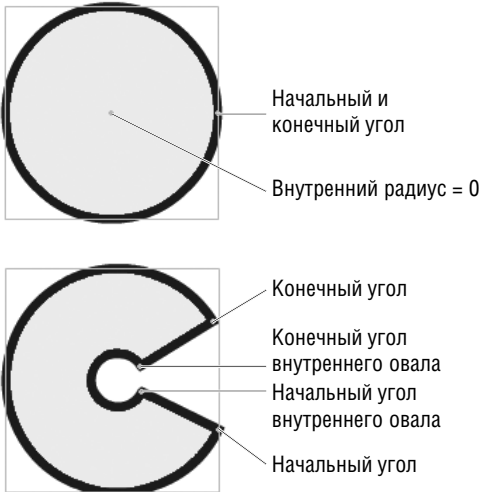


**Рис. 4.68.** По мере увеличения радиуса угла становятся различимы две управляющие точки на концах дуги. Чтобы увеличить радиус, буксируйте точку внутрь, чтобы уменьшить — наружу.





**Рис. 4.70.** Буксирование управляющей точки вырезанного угла (с отрицательным радиусом) изменяет размер выреза. Но если довести ее до угловой точки, то угол станет прямым и при последующей буксировке будет создан уже не вырезанный, а скругленный угол. Чтобы восстановить вырезанный угол, придется снова ввести отрицательный радиус в инспекторе свойств.



**Рис. 4.71.** Управляющие точки овального примитива позволяют изменять начальный и конечный угол внешнего и внутреннего овала.

### Чтобы точно задать

#### новые свойства прямоугольного примитива:

1. Выбрав на арене прямоугольный примитив, перейдите на его вкладку в инспекторе свойств.
2. Чтобы изменить форму прямоугольника, выполните следующие действия:
  - для создания закругленных углов введите положительные значения радиусов закругления;
  - для создания вырезанных углов введите отрицательные значения радиусов закругления.

Подробнее о задании различных свойств прямоугольных примитивов см. главу 2.

### С

Нельзя преобразовать резкий или скругленный угол в вырезанный путем буксировки управляющих точек; необходимо вручную ввести отрицательное значение в инспекторе свойств. Но скорректировать величину уже имеющегося выреза путем буксировки управляющей точки вполне возможно (рис. 4.70).

### Чтобы интерактивно изменить свойства овального примитива:

1. С помощью инструмента «Стрелка» или «Спецвыделение» выберите на арене интересующий вас овальный примитив.

Появляется охватывающий прямоугольник с управляющими точками. У овального примитива есть четыре управляющие точки: одна пара для начального и конечного угла внешнего овала и другая — для углов внутреннего овала (рис. 4.71). Если внешний и внутренний угол одинаковы, то эти точки совмещены.

2. Подведите курсор к управляющей точке. Курсор принимает форму сплошной стрелки без хвостика.
3. Чтобы изменить форму овала, выполните одно из следующих действий:
  - для изменения начального угла буксируйте соответствующую ему управляющую точку вокруг овала в любом направлении (рис. 4.72);
  - для изменения конечного угла буксируйте соответствующую ему управляющую точку вокруг овала в любом направлении;
  - для увеличения внутреннего радиуса (расширения дырки бублика) буксируйте управляющую точку наружу (рис. 4.73);
  - для уменьшения внутреннего радиуса буксируйте управляющую точку внутрь.

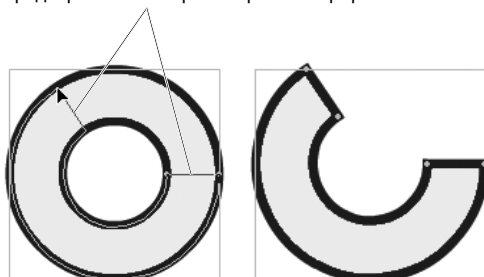
#### Чтобы точно задать новые свойства овального примитива:

1. Выбрав на арене овальный примитив, перейдите на его вкладку в инспекторе свойств.
2. Введите новые значения в поля Start Angle (Начальный угол), End Angle (Конечный угол), Inner Radius (Внутренний радиус).  
 Подробнее о задании различных свойств овальных примитивов см. главу 2.

**С**

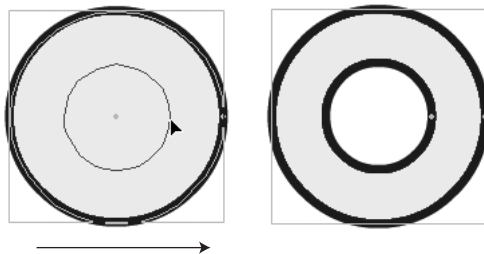
Хотя произвольно изменять пути примитивных форм невозможно, но можно использовать примитив как отправную точку для создания формы, допускающей произвольное изменение. После того как свойства примитива будут заданы, преобразуйте его в комбинированную форму или объект-рисунок (см. раздел «Конвертирование типов форм» ниже). Получившуюся форму уже можно модифицировать как угодно описанными выше способами.

Предварительный просмотр новой формы

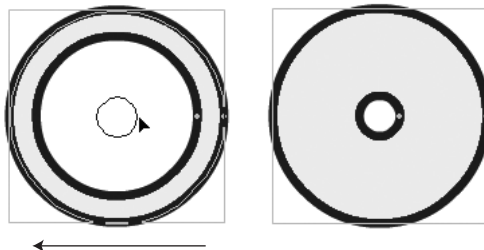


**Рис. 4.72.** Чтобы изменить начальный или конечный угол овального примитива, буксируйте управляющие точки, расположенные на его внешнем крае, по часовой стрелке (или в другую сторону).

Результат увеличения внутреннего радиуса



Результат уменьшения внутреннего радиуса



**Рис. 4.73.** Чтобы изменить радиус внутреннего овала, буксируйте управляющие точки, расположенные на внутреннем крае. Буксировка в направлении к центру овала сжимает внутреннюю дырку, в направлении от центра — расширяет.

### Модификация форм инструментом «Ластик»

Ластик позволяет модифицировать комбинированные формы и объекты-рисунки путем удаления частей заливок и обводок. (К примитивным формам ластик применим только в режиме крана.) Чтобы воспользоваться ластиком, щелкните им по арене и начинайте стирать так же, как обычным ластиком с бумаги.

У инструмента «Ластик» есть пять режимов, выбираемых из меню ластика на панели инструментов. Они управляют тем, как ластик взаимодействует с заливками и обводками. Для простых форм в одном слое обычно имеет смысл установить режим **Erase Normal** (Нормальная очистка); в этом случае удаляются все линии и заливки, по которым проходит ластик. В остальных режимах есть ограничения на элементы, к которым инструмент применим. В режиме **Erase Lines** (Очистить линии) стираются только обводки, в режиме **Erase Selected Fills** (Очистка выделения) — только выбранные заливки, а в режиме **Erase Inside** (Внутренняя очистка) — только заливка, в которой вы начали процесс стирания. Эти режимы могут пригодиться при работе со сложной графикой, состоящей из нескольких элементов (см. главу 5).

Еще один параметр ластика — это модификатор **faucet** (Кран). Кран ускоряет стирание обводок и заливок для форм любого типа. Щелчок ластиком по не выбранной обводке в режиме крана стирает все входящие в состав этой обводки сегменты. Если выбрано несколько обводок комбинированных форм или заливок, то щелчок по любому из выбранных элементов, удаляет все выбранные. Если выбрано несколько объектов-рисунков или примитивных форм, то кран удаляет лишь ту обводку или заливку, по которой вы щелкнули. (Отмена изменений примитивов не всегда работает надежно.)

Кроме того, в режиме крана модифицируются сложные настройки ластика. Например, если установлен режим стирания линий, то стереть заливку не получится. Однако, если, не выходя из этого режима, установить дополнительно режим крана, то щелчок по заливке сотрет ее. Применение ластика — самый быстрый способ полностью очистить арену, для этого достаточно дважды щелкнуть по значку ластика на панели инструментов.

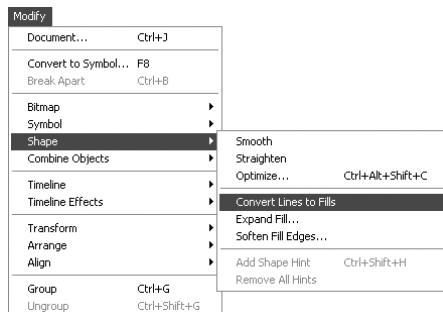
## Конвертирование форм из одного вида в другой

Во Flash есть много разновидностей форм: заливки, обводки, комбинированные формы, объекты-рисунки, примитивы и текст. Некоторые формы можно конвертировать в другие. Например, заливки легко преобразуются в обводки, комбинированные формы — в объекты-рисунки и наоборот, примитивные формы — в комбинированные или в объекты-рисунки. Текст — особый тип заливки — можно конвертировать в обычную комбинированную форму. Однако преобразовать комбинированную форму или объект-рисунок в примитивную форму невозможно.

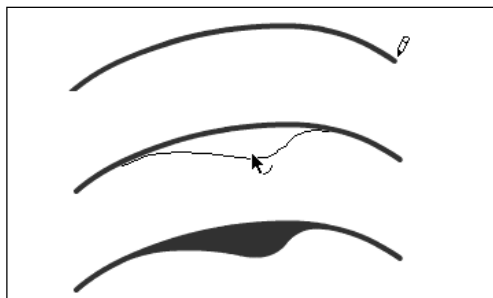
### Чтобы конвертировать обводку в заливку:

1. Щелкните по обводке на арене.
2. Выполните команду **Modify** ⇒ **Shape** ⇒ **Convert Lines to Fills** (Изменить ⇒ Форма ⇒ Преобразовать линии в заливку) (рис. 4.74).

Flash преобразует обводку в заливку точно такой же формы. Теперь можно редактировать путь, составляющий контур обводки, как если бы вы работали с заливкой, созданной инструментом «Кисть» (рис. 4.75).



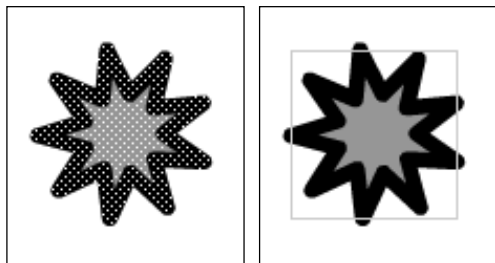
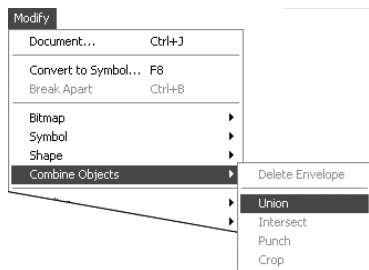
**Рис. 4.74.** Чтобы преобразовать заливки в обводки, выполните команду **Modify** ⇒ **Shape** ⇒ **Convert Lines to Fills**.



**Рис. 4.75.** Можно преобразовать в заливку показанную на верхнем рисунке обводку, которая была нарисована карандашом. У контура заливки будет свой редактируемый путь (средний и нижний рисунки).

### Текстовые заливки

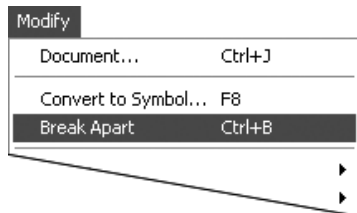
Во Flash текст считается особым видом заливки, но литеры ведут себя не так, как формы созданные инструментами рисования. Все литеры рассматриваются как единый текст внутри контейнера, который интерпретируется как один графический объект. Flash позволяет *расщепить* текст, то есть разбить один текстовый блок, содержащий редактируемые символы, на несколько блоков по одному символу в каждом. В результате вы получаете возможность масштабировать, позиционировать и искажать отдельные символы. Размещение букв в отдельных текстовых полях удобно и для анимации текста. (С техникой анимации вы познакомитесь в главах 8-11). Но и такие однобуквенные поля остаются графическими объектами. Чтобы преобразовать текстовую заливку в заливки настоящих комбинированных форм, придется выполнить расщепление дважды: один раз, чтобы разбить на отдельные текстовые поля, а второй — чтобы преобразовать каждую букву в комбинированную форму. Этот прием полезен, когда имеется текст небольшого объема, который вы не можете (или не хотите) передавать каждому конечному пользователю, но при этом требуется, чтобы текст выглядел в готовом продукте точно так же, как в среде разработки.



Выбранная  
комбинированная форма  
(заливка и обводка)

После конвертации  
в объект-рисунок

**Рис. 4.76.** Выполните команду **Modify** ⇒ **Combine Objects** ⇒ **Union** для конвертирования выбранной на арене формы в объект-рисунок.



**Рис. 4.77.** Выполните команду **Modify** ⇒ **Break Apart** для конвертирования выбранного на арене объекта-рисунка в комбинированную форму.

### Чтобы конвертировать комбинированную форму или примитив в объект-рисунок:

1. Выберите комбинированную форму или примитив на арене.
2. Выполните команду **Modify** ⇒ **Combine Objects** ⇒ **Union** (Изменить ⇒ Объединить объекты ⇒ Объединение) (рис. 4.76).

**C**

Команды из меню **Modify** ⇒ **Combine Shapes** (**Union** (Объединение), **Intersect** (Пересечение), **Punch** (Подшивка) и **Chop** (Кадрировать)) применяются к нескольким выбранным объектам-рисункам и примитивам; три последних работают для перекрывающихся объектов-рисунков или примитивов. Смысл их в том, что объекты-рисунки или примитивы конвертируются в комбинированные формы (чтобы они могли взаимодействовать между собой), а затем результат преобразуется обратно в объект-рисунок. Отметим, что после объединения примитив перестает быть примитивом. Дополнительную информацию об объединении форм см. в главе 5.

### Чтобы конвертировать объект-рисунок или примитив в комбинированную форму:

1. Выберите объект-рисунок или примитив на арене.
2. Выполните команду **Modify** ⇒ **Break Apart** (Изменить ⇒ Разделить) или нажмите комбинацию клавиш **⌘-B** (Mac) либо **Ctrl-B** (Windows) (рис. 4.77).

Flash конвертирует выбранный объект-рисунок или примитив в комбинированную форму, получившаяся форма остается выбранной.

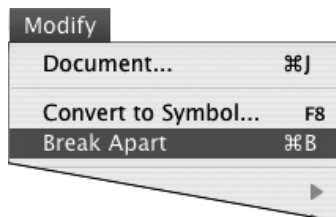
### Чтобы расщепить блок текста на однобуквенные текстовые поля:

1. Выберите на арене блок текста.
2. Выполните команду **Modify** ⇒ **Break Apart** (рис. 4.78).

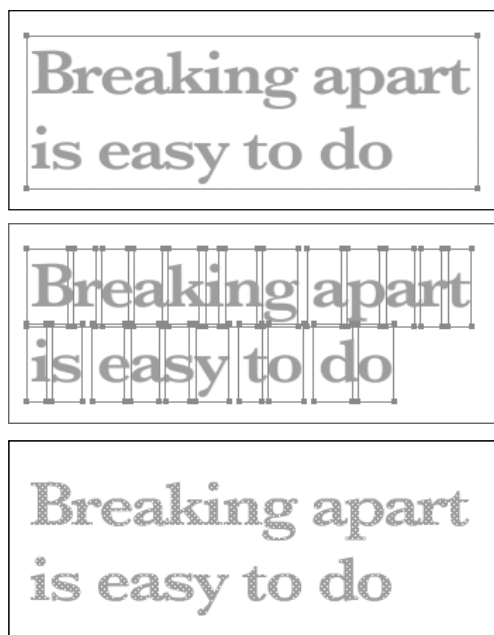
Каждая буква помещается в отдельное текстовое поле, и все эти поля оказываются выбранными. Ширина полей установлена в точном соответствии с шириной буквы. Каждую букву можно редактировать по отдельности, но группа букв больше не является единым целым.

### Чтобы преобразовать буквы в комбинированные формы:

1. Выполните описанные в предыдущем упражнении шаги, чтобы поместить каждую букву в отдельное текстовое поле.
2. Выполните команду **Modify** ⇒ **Break Apart**. Эта вторая по счету команда расщепления преобразует редактируемые буквы в формы на арене (рис. 4.79). Их можно редактировать, как любую заливку, но изменить атрибуты текста с помощью инструмента «Текст» уже не получится.



**Рис. 4.78.** Выполните команду **Modify** ⇒ **Break Apart**, чтобы поместить каждую букву текста в отдельное текстовое поле.



**Рис. 4.79.** После первой команды **Break Apart** текст (верхний рисунок) преобразуется в набор однобуквенных текстовых полей (средний рисунок), а повторная команда преобразует каждую букву в комбинированную форму (нижний рисунок).

# ОДНОСЛОЙНАЯ СОСТАВНАЯ ГРАФИКА

---

## 5

В главах 2 и 4 вы научились создавать и модифицировать простые формы, составленные из обводок (линий) и заливок. Но в ролике наверняка будет присутствовать много форм, и придется различными способами сочетать их обводки и заливки. Можно, например, составить из овалов и прямоугольников фигурку робота. Для эффективной работы с составной графикой необходимо понимать, как взаимодействуют различные графические элементы — комбинированные формы, объекты-рисунки и примитивы, — расположенные в одном или нескольких слоях. В этой главе вы узнаете, как работать с несколькими графическими элементами, находящимися в одном слое документа Flash. (Дополнительную информацию о слоях см. в главе 6.)

У двух инструментов рисования Flash — кисти и ластика — есть специальные режимы, предназначенные для использования с несколькими обводками и заливками в одном слое. Выполняя упражнения из этой главы оставьте режим, установленный по умолчанию (Paint Normal для кисти и Erase Normal для ластика), если явно не оговорено противное.

### Когда комбинированные формы взаимодействуют

Можно считать, что каждый кадр Flash-ролика — это стопка прозрачных пленок. В терминологии Flash каждая такая пленка называется слоем. Графический элемент, присутствующий в верхнем слое, скрывает от взгляда то, что находится под ним в нижних слоях, — точно так же, как при рисовании на пленках.

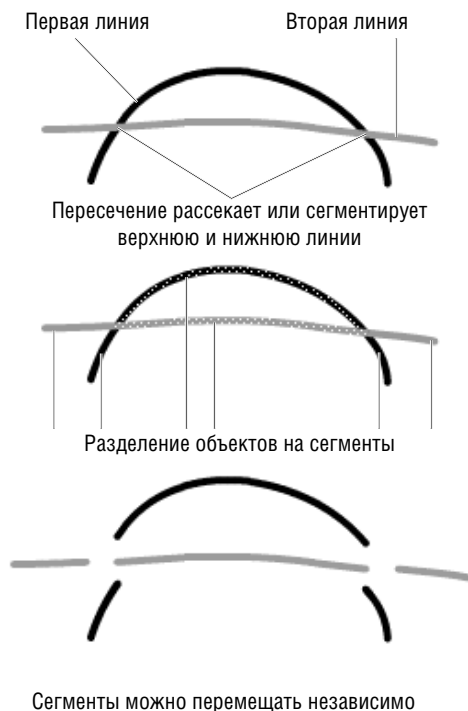
Представьте себе, что ролик состоит из двух слоев. Если нарисовать желтый квадратик в нижнем слое, а затем перейти на верхний и нарисовать большой красный квадрат поверх желтого, то, хотя последний никуда не делся, вы его не увидите — мешает верхний квадрат.

Но комбинированные формы, находящиеся в одном слое, взаимодействуют почти так же, как если бы вы рисовали их акварелью, не давая высохнуть. Ниже коротко описано, как взаимодействуют линии (обводки) и формы (заливки), созданные в одном слое в комбинированном режиме рисования.

#### Когда взаимодействуют линии, принадлежащие комбинированным формам

Пересекающиеся линии, принадлежащие комбинированным формам, нарисованным в одном слое, влияют друг на друга. Нарисуйте в комбинированном режиме сначала одну линию, а потом другую, так чтобы она пересекала первую. Вторая линия пересекает первую. Вторая линия *разделяет* — или, в терминологии Flash, — *сегментирует* первую. Сегментация происходит вне зависимости от того, нарисованы линии одним цветом или разными, но увидеть ее проще, когда выбраны контрастные цвета (рис. 5.1).

Можно предположить, что вторая линия пройдет поверх первой, но так бывает не всегда. Нарисуйте сначала красную линию, а потом синюю. Когда вы отпустите кнопку мыши, синяя линия поднырнет под красную. Flash упорядочивает линии в стопке по шестнадцатеричному значению цвета обводки. Чем больше значение, тем выше окажется линия в пределах данного слоя. Нарисованная в комбинированном режиме линия с цветом 663399 всегда будет выше линии с цветом 333399.



**Рис. 5.1.** Когда при рисовании в комбинированном режиме одна линия пересекает другую, каждое пересечение приводит к созданию отдельного сегмента.



### Когда пересекаются линии и заливки, принадлежащие комбинированным формам

Даже если заливка не имеет обводки, у нее все равно есть путь (невидимый контур, описывающий форму заливки). При работе в комбинированном режиме рисования пути заливок могут рассекать обводки других форм. Рисуя линии над заливкой, можно получить много мелких сегментов. Попробуйте нарисовать несколько линий (обводок) карандашом и дуг (заливок) — кистью, установив при этом комбинированный режим рисования. Если заливка пересекает обводку, то заливка останется целостным объектом, а обводка будет сегментирована (рис. 5.2). Если же обводка пересекает заливку, то обводка рассекает заливку, а путь заливки рассекает обводку (рис. 5.3).



**Рис. 5.2.** Если заливка, нарисованная в комбинированном режиме, перекрывает обводку, то заливки сегментирует обводку. Взглянув на узор, обозначающий выделение, вы поймете, что заливка осталась целостным объектом.



**Рис. 5.3.** Если линия, нарисованная в комбинированном режиме, перекрывает заливку, то путь линии сегментирует заливку, а путь заливки сегментирует линию.

### Когда пересекаются заливки, принадлежащие комбинированным формам

Если пересекающиеся заливки, созданные в комбинированном режиме, имеют один и тот же цвет, то последняя нарисованная добавляется к предыдущей комбинированной форме (рис. 5.4). Если же цвета различны, то более новая заливка замещает старую в областях перекрытия (рис. 5.5).

Заливки, нарисованные кистью



**Рис. 5.4.** Если пересекающиеся заливки, созданные в комбинированном режиме, имеют один и тот же цвет, то Flash создает из них единую комбинированную форму.



**Рис. 5.5.** Если пересекающиеся заливки, созданные в комбинированном режиме, имеют разные цвета, то они не сливаются, а остаются отдельными объектами. Второй овал на рисунке замещает первый там, где они пересекаются.

## Работа с группами

*Группа* — это виртуальный контейнер, содержащий графические элементы. Группы выполняют несколько функций. Они предотвращают взаимодействие выбранных комбинированных форм, а также фиксируют атрибуты форм и сохраняют пространственные отношения между графическими элементами. Хотя группировать можно также объекты-рисунки и примитивные формы, в этом разделе мы будем изучать только взаимодействия комбинированных форм, поэтому отождим кнопку **Object Drawing** (Рисование объектов) на панели инструментов.

### Чтобы создать группу:

1. Выберите на арене один или несколько объектов, как описывалось в главе 4 (рис. 5.6).
2. Выполните команду **Modify** ⇒ **Group** (Изменить ⇒ Группировать) или нажмите комбинацию клавиш **⌘-G** (Mac) либо **Ctrl-G** (Windows).

Flash сгруппирует элементы, поместив их внутрь охватывающего прямоугольника (рис. 5.7). Наличие на экране охватывающего прямоугольника показывает, что выбрана группа. Если никакой группы не выбрано, то прямоугольник не виден.

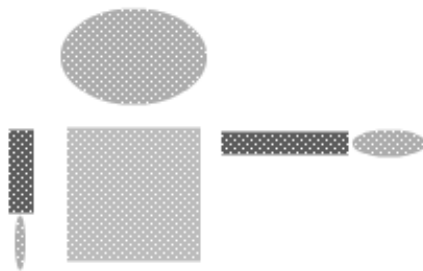
**С**

Если выполнить команду **Modify** ⇒ **Group**, когда ни один объект не выбран, вы попадаете в режим редактирования группы: любые объекты, рисуемые на арене, становятся членами новой группы.

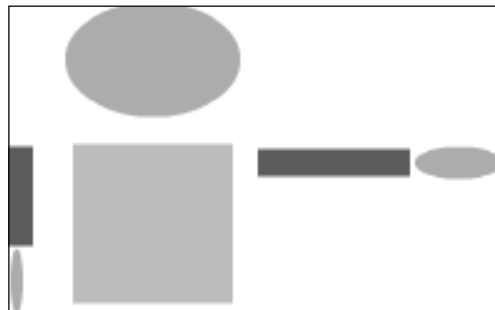
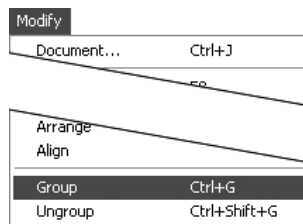
### Чтобы разгруппировать объекты:

3. Выберите группу, которую собираетесь реформировать.
4. Выполните команду **Modify** ⇒ **Group** (Изменить ⇒ Группировать) или нажмите комбинацию клавиш **Shift-⌘-G** (Mac) либо **Ctrl-Shift-G** (Windows).

Flash сотрет охватывающий прямоугольник, а все входившие в группу элементы оставит выбранными.



**Рис. 5.6.** Первый шаг при создании группы — выбор форм, которые должны войти в нее.



**Рис. 5.7.** Команда **Modify** ⇒ **Group** помещает выбранные объекты внутрь охватывающего прямоугольника.

## С

Команда расщепления символов (**Modify** ⇒ **Break Apart** или ⌘-B (Mac) либо Ctrl-B (Windows)) позволяет также расформировывать группы. Но для групп, содержащих формы разных типов, результат может оказаться неожиданным. Если в группу входит хотя бы одна комбинированная форма, то команда **Break Apart** преобразует все формы в комбинированные, даже если первоначально это были объекты-рисунки или примитивы. Расщепление группы, не содержащей комбинированных форм, не изменяет типы входивших в нее элементов. Чтобы сохранить исходный тип каждого члена группы, расформировывайте ее командой **Modify** ⇒ **Ungroup**.

## С

Взаимодействия между обводками и заливками имеют место не только при рисовании, но и при копировании или перемещении формы. Будьте осторожны, когда помещаете копии заливок и обводок комбинированных форм в один и тот же слой; вы можете случайно добавить или удалить часть находящейся под копией формы. При расформировании группы, перекрывающей комбинированные формы в том же слое, формы сегментируют друг друга.

### Предотвращение взаимодействий

Формы, нарисованные в комбинированном режиме, ведут себя так, как будто краска еще не высохла. Ниже описывается несколько способов «высушить» краску и тем самым предотвратить взаимодействия.

**Объекты-рисунки.** В режиме рисования объектов создаются комбинированные формы, но Flash изолирует их внутри контейнера. Заливки и обводки внутри контейнера объекта-рисунка не взаимодействуют с заливками и обводками, которые не принадлежат этому контейнеру, но путь и атрибуты заливок и обводок объекта-рисунка можно модифицировать прямо на арене (см. главу 4). Можно также открыть контейнер объекта-рисунка и работать с находящимися в нем комбинированными формами непосредственно.

**Примитивные формы.** Инструменты «Прямоугольный примитив» и «Овальный примитив» создают объекты-рисунки, имеющие свойства, которые характеризуют их форму. Flash помещает такие объекты в контейнер, налагающий еще больше ограничений. Атрибуты заливок и обводки примитива можно модифицировать непосредственно с помощью инструментов «Ведро с краской» и «Чернильница». Характеристические свойства формы (например, радиус закругления для прямоугольного примитива) можно изменять, воздействуя инструментом «Стрелка» на управляющие точки, но произвольно модифицировать контур формы запрещается. Можно открыть контейнер примитива, чтобы модифицировать находящийся внутри объект-рисунок, но тогда форма будет конвертирована в объект-рисунок навсегда.

**Группы.** Сгруппированные заливки и обводки перестают взаимодействовать с другими заливками и обводками. Кроме того, сгруппированные элементы нельзя редактировать напрямую. Чтобы модифицировать атрибуты форм, включенных в группу, необходимо войти в специальный режим редактирования.

**Символы.** Заливки и обводки внутри символов не взаимодействуют с заливками и обводками других графических элементов. Редактируются символы тоже в специальном режиме (см. главу 7).

**Слои.** Комбинированные формы, находящиеся в разных слоях, не взаимодействуют между собой (см. главу 6).

**Чтобы предотвратить взаимодействие между комбинированными формами в одном слое:**

1. Выберите на панели инструментов «Овал» в режиме комбинированного рисования.
2. Для обводки задайте цвет No Color, а для заливки — красный.
3. Нарисуйте на арене большой овал (рис. 5.8).
4. Переключитесь на инструмент «Стрелка» и выберите только что нарисованный овал.
5. Чтобы сделать овал частью группы, выполните команду **Modify** ⇒ **Group** (рис. 5.9).
6. Снова перейдя на панель инструментов, выберите «Овал» и какой-нибудь другой цвет.
7. Нарисуйте на арене овал поменьше внутри первого овала (рис. 5.10).

Как только вы закончите рисовать новый овал, он сразу же скроется под овалом, включенным в группу (рис. 5.11). Объясняется это тем, что сгруппированные объекты всегда оказываются в стопке выше не сгруппированных (см. врезку «Порядок в стопке» ниже).

8. Переключившись на инструмент «Стрелка», расположите большой овал так, чтобы было видно овал поменьше (рис. 5.12).
9. Отмените выбор большого овала, но выберите маленький (рис. 5.13).
10. Чтобы сделать маленький овал частью группы, выполните команду **Modify** ⇒ **Group**.

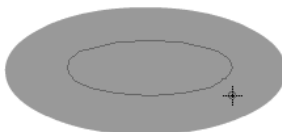
Flash нарисует охватывающий прямоугольник вокруг маленького овала и поместит его на вершину стопки (рис. 5.14). Группа, созданная последней, всегда помещается поверх всех остальных. Теперь можно располагать овалы в любом месте арены, и они не будут взаимодействовать между собой.



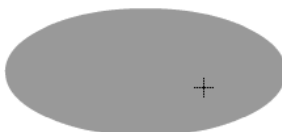
**Рис. 5.8.** Овал до включения в группу.



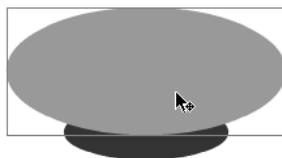
**Рис. 5.9.** Овал после включения в группу.



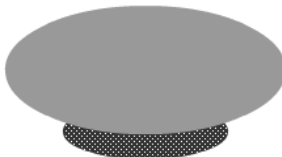
**Рис. 5.10.** Нарисуйте второй овал поверх того, что включен в группу.



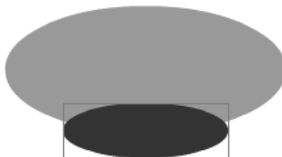
**Рис. 5.11.** Овал, не включенный в группу, оказывается в стопке под тем, что включен в группу.



**Рис. 5.12.** Перетащите овал, включенный в группу, так чтобы другой овал был виден.

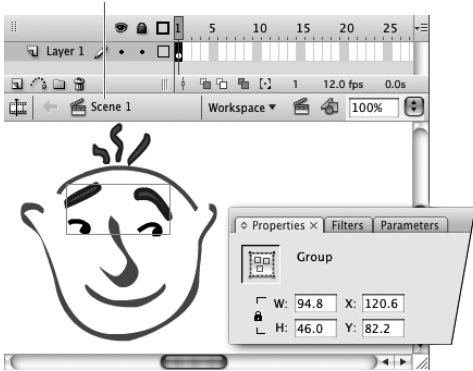


**Рис. 5.13.** Выберите меньший овал.



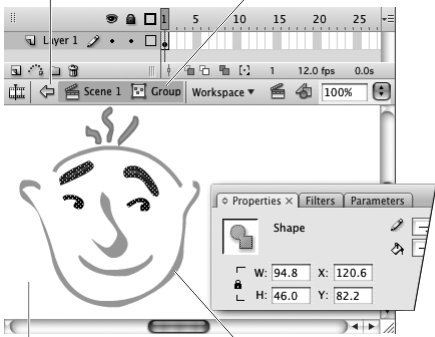
**Рис. 5.14.** После включения в группу маленький овал, входящий в группу, созданную последней, оказывается на вершине стопки.

## Режим редактирования документа



Нажмите кнопку Back для возврата в режим редактирования документа

Режим редактирования группы



Дважды щелкните мышью в любой точке вне группы для возврата в режим редактирования документа

Закрашенные серым цветом формы не принадлежат редактируемой группе

**Рис. 5.15.** Глаза и брови входят в выбранную группу (верхний рисунок). В режиме редактирования группы (нижний рисунок) выбранные формы можно изменять; остальные элементы на арене выкрашены серым цветом, следовательно, редактировать их невозможно.

## Редактирование групп

Группу можно преобразовывать как единое целое (масштабировать, вращать, перекашивать), но редактировать отдельные формы в ней так, как это делается для не сгруппированных форм, нельзя. Чтобы начать редактирование форм, являющихся частью группы, воспользуйтесь командой **Edit Selected** (Редактировать выделение).

### Чтобы отредактировать содержимое группы:

1. Выберите инструмент «Стрелка».
2. Выберите на арене группу, которую собираетесь отредактировать.
3. Выполните команду **Edit ⇨ Edit Selected**.

Flash переходит в режим редактирования группы (рис. 5.15), о чем свидетельствует панель редактирования над ареной. Прямоугольник, охватывающий выбранную группу, исчезает, и Flash перекрашивает в серый цвет все элементы на арене, не являющиеся частью группы. Серые элементы недоступны для редактирования, они лишь служат контекстом для редактирования выбранной группы.

4. Внесите изменения в элементы, являющиеся частью группы.
5. Для возврата в режим редактирования документа выполните одно из следующих действий:

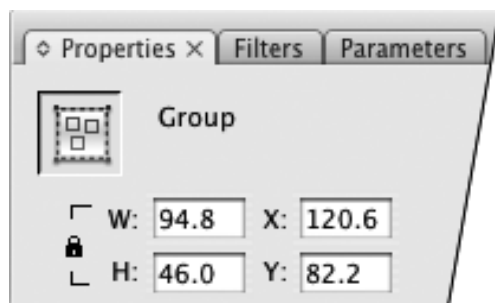
- выполните команду **Edit ⇨ Edit All** (Редактирование ⇨ Редактировать все);
- дважды щелкните по пустому месту на арене или на монтажном столе;
- щелкните по имени текущей сцены на панели редактирования;
- нажмите кнопку **Back** (Назад) на панели редактирования.

**С** Чтобы быстро перейти в режим редактирования группы, дважды щелкните по входящему в группу элементу инструментом «Стрелка».

**С** На вкладке **Properties** в инспекторе свойств можно посмотреть и изменить координаты прямоугольника, охватывающего выбранную группу (рис. 5.16).

**С** При редактировании группы, вложенной в другую группу, нажатие кнопки **Back** возвращает на один уровень вверх в иерархии вложенности.

**С** Перейти в режим группового редактирования одного выбранного элемента позволяет команда **Edit** ⇨ **Edit in Place** (Редактирование ⇨ Редактировать на месте). При редактировании групп эта команда ничем не отличается от команды **Edit** ⇨ **Edit Selected**. Но при редактировании символов они отличаются (см. главу 7).



**Рис. 5.16.** В инспекторе свойств отображаются высота, ширина и координаты охватывающего прямоугольника для выбранной на арене группы. Чтобы изменить любой из этих параметров, просто введите новые значения.

### Редактирование внутри контейнера объекта-рисунка

В главе 4 было показано, как модифицировать объекты-рисунки, находящиеся на арене (по существу, вы обращаетесь к ним через соответствующий контейнер). Но можно манипулировать и непосредственно комбинированными формами внутри этого контейнера. Для этого следует открыть контейнер и работать в режиме редактирования объекта-рисунка. Для входа в этот режим не существует команды в меню, но двойной щелчок по любому объекту-рисунку на арене открывает его контейнер и позволяет редактировать содержимое. (Отметим, что двойной щелчок по примитивной форме приводит к появлению диалогового окна, в котором предлагается преобразовать примитив в объект-рисунок и открыть контейнер для редактирования содержимого.)

В режиме редактирования объекта-рисунка, как и в режиме редактирования группы, выбранные формы отображаются своим цветом, а все остальные — серым. В этом режиме можно модифицировать или удалить исходные комбинированные формы либо добавить новые. Работая в режиме редактирования объекта-рисунка, вы можете создать новые объекты-рисунки или примитивы, но при возврате в режим редактирования документа Flash конвертирует исходный объект-рисунок в составляющие его комбинированные формы, добавляет новые объекты-рисунки или примитивы, выбирает все эти элементы и объединяет их в группу.

Для возврата в режим редактирования документа годятся те же способы, что и при редактировании группы (например, нажатие кнопки **Back** на панели редактирования).

## Управление порядком в стопке

Все текстовые поля, сгруппированные объекты, объекты-рисунки и примитивы, находящиеся в одном слое, организуются в стопку, как если бы они размещались в подслоях, наложенных поверх всех комбинированных форм, не входящих ни в какую группу. Порядок относительно стопки определен даже в том случае, когда объекты не перекрывают друг друга. Если на одном краю арены находится группа, а на другом — объект-рисунок, то невозможно понять, который из них выше в стопке, а который — ниже. Однако, если перетащить объекты так, чтобы они перекрывались, то порядок станет очевидным. (Символы, о которых вы узнаете в главе 7, — это еще один тип графических объектов, которые находятся в стопке выше не сгруппированных комбинированных форм.)

### Порядок в стопке

Комбинированные формы, находящиеся в одном слое, не покидают его и сегментируют друг друга, если занимают одно и то же место на арене. Все графические объекты (объекты-рисунки, примитивы, текстовые поля, группы и символы) занимают определенное место в стопке. По умолчанию Flash располагает каждый новый графический объект поверх предшествующих; самый последний созданный объект оказывается поверх всех остальных (рис. 5.17). Объект, расположенный в стопке выше, закрывает те объекты, что оказались под ним.



**Рис. 5.17.** На этом рисунке схематически показан порядок графических объектов в стопке, принимаемый по умолчанию. Объект, созданный последним, оказывается на вершине стопки. Комбинированные формы всегда находятся в самом низу.

Порядок графических объектов в стопке можно изменить командами из меню **Modify** ⇒ **Arrange** (Изменить ⇒ Упорядочить). Объекты можно перемещать вверх и вниз по стопке на один уровень или сразу сделать объект самым верхним либо самым нижним.

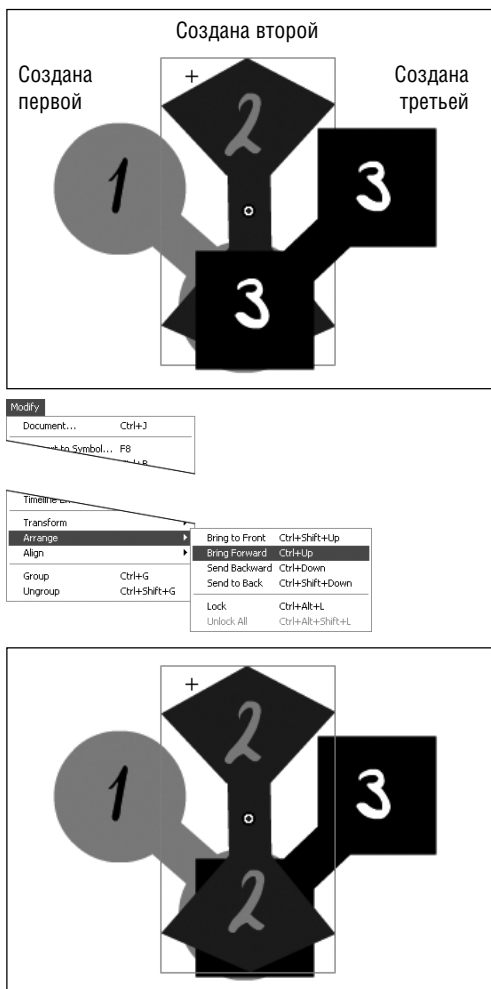
### Чтобы изменить положение в стопке на один уровень:

1. Создайте на арене по меньшей мере три графических объекта. Это могут быть сгруппированные формы, объекты-рисунки, примитивы и текстовые поля в любом сочетании.
2. Выберите один из графических объектов.
3. Из меню **Modify** ⇒ **Arrange** выберите один из следующих пунктов:
  - чтобы поднять выбранный элемент на один уровень, выполните команду **Bring Forward** (Перенести вперед) или нажмите комбинацию клавиш **⌘**—**стрелка вверх** (Mac) либо **Ctrl**—**стрелка вверх** (Windows);
  - чтобы опустить выбранный элемент на один уровень, выполните команду **Send Backward** (Перенести на задний план) или нажмите комбинацию клавиш **⌘**—**стрелка вниз** (Mac) либо **Ctrl**—**стрелка вниз** (Windows).

Flash перемещает выбранный элемент на один уровень вверх или вниз (рис. 5.18).

### Чтобы переместить элемент в начало или в конец стопки:

1. Выберите на арене один из созданных в предыдущем упражнении объектов.
2. Из меню **Modify** ⇒ **Arrange** выберите один из следующих пунктов:
  - чтобы переместить элемент в начало (на вершину) стопки, выполните команду **Bring To Front** (На передний план) или нажмите комбинацию клавиш **⌘**—**Shift**—**стрелка вверх** (Mac) либо **Ctrl**—**Shift**—**стрелка вверх** (Windows);

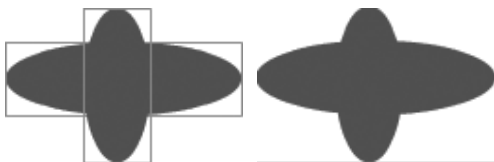


**Рис. 5.18.** Каждая гантелеобразная фигура на этом рисунке — отдельная группа (верхний рисунок). Выполните команду **Modify** ⇒ **Arrange** ⇒ **Bring Forward** (средний рисунок), чтобы переместить выбранную группу на один уровень вверх в стопке (верхний рисунок).

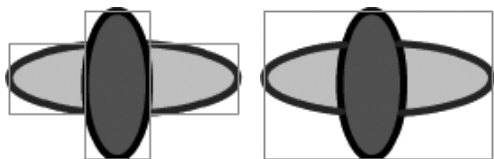
- чтобы переместить элемент в конец (самый низ) стопки, выполните команду **Send To Back** (Отправить назад) или нажмите комбинацию клавиш **⌘**—**Shift**—**стрелка вниз** (Mac) либо **Ctrl**—**Shift**—**стрелка вниз** (Windows).

Flash перемещает выбранный элемент на вершину или в самый низ стопки.





**Рис. 5.19.** Применение команды **Modify** ⇒ **Combine Objects** ⇒ **Union** к одноцветным заливкам объекта-рисунка или примитива (левый рисунок) объединяет заливки и создает единый объект-рисунок (правый рисунок).



**Рис. 5.20.** Применение команды **Modify** ⇒ **Combine Objects** ⇒ **Union** к разноцветным объектам-рисункам (или примитивам) (левый рисунок) приводит к тому, что заливки и обводки замещают и сегментируют друг друга, как обычные комбинированные формы. Получившиеся формы объединяются в единый объект-рисунок (правый рисунок).



**Рис. 5.21.** Попробуйте дважды щелкнуть по новому объекту-рисунку после применения команды **Union** к разноцветным объектам-рисункам (или примитивам). В режиме редактирования объекта-рисунка вы увидите, как формы сегментируют друг друга. Каждый фрагмент обводки или заливки — это отдельная форма.

## Объединение объектов-рисунков и примитивов

Объекты-рисунки и примитивы не взаимодействуют ни между собой, ни с комбинированными формами даже тогда, когда перекрываются. Но можно заставить их взаимодействовать с помощью команд из меню **Modify** ⇒ **Combine Objects** (Изменить ⇒ Объединить объекты). Объединенные примитивы Flash преобразует в объекты-рисунки.

### Чтобы объединить несколько объектов-рисунков или примитивов:

1. С помощью инструментов рисования, работающих в режиме рисования объектов, или инструментов «Прямоугольный примитив» либо «Овальный примитив» создайте перекрывающиеся формы:
  - нарисуйте две или более перекрывающиеся заливки одного цвета;
  - нарисуйте две или более перекрывающиеся формы с заливками и обводками; в каждой форме применяйте разные цвета заливок и обводок.
2. Выберите перекрывающиеся заливки одного цвета.
3. Выполните команду **Modify** ⇒ **Combine Objects** ⇒ **Union**. Обе заливки становятся единым объектом-рисунком (рис. 5.19).
4. Выберите перекрывающиеся формы разного цвета.
5. Повторите шаг 3.

Заливки и обводки форм сегментируют друг друга, но в результате получается единый объект-рисунок, содержащий все сегменты (рис. 5.20).

**С** Команду **Modify** ⇒ **Combine Objects** ⇒ **Union** можно использовать и для объединения набора комбинированных форм, объектов-рисунков и примитивов в единый объект-рисунок.

**С** Для редактирования отдельной комбинированной формы внутри объекта-рисунка дважды щелкните по ней мышью (рис. 5.21).

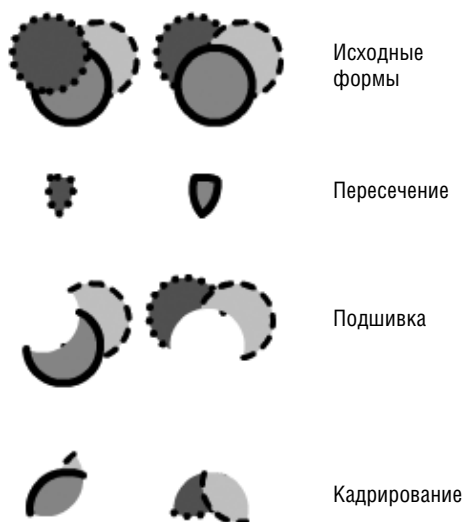
### Чтобы использовать один объект-рисунок для удаления части другого:

1. Воспользуйтесь инструментами рисования в режиме рисования объектов или инструментами «Прямоугольный примитив» либо «Овальный примитив» для создания двух или более перекрывающихся форм с различными заливками и обводками.
2. Из меню **Modify** ⇒ **Combine Objects** выберите один из следующих пунктов:

**Intersect** (Пересечение) сохраняет заливки и обводки только там, где перекрываются все выбранные формы, остальные удаляются. Результирующая форма (или формы) наследуют атрибуты заливок и обводок от формы, расположенной на вершине стопки.

**Punch** (Подшивка) использует верхнюю форму как фигурный резак, удаляющий все, что находится непосредственно под ним. (Представьте себе, что остается от куска теста после того, как вы надавите на него формовщиком печенья; вот именно это и делает команда Punch). Оставшиеся формы сохраняют свои атрибуты.

**Crop** (Кадрировать) использует верхнюю форму как фигурный резак, выбирающий все, что находится под ним (Снова прибегнем к аналогии с формовщиком печенья, но на этот раз нас будет интересовать то, что осталось внутри него). Оставшиеся формы сохраняют свои атрибуты (рис. 5.22).



**Рис. 5.22.** Последние три команды в меню **Modify** ⇒ **Combine Objects** дают разные результаты в зависимости от того, какой объект находится на вершине стопки. Команда **Intersect** создает новую форму, являющуюся пересечением всех выбранных, и присваивает ей атрибуты верхней формы. В случае команды **Punch** форма на вершине стопки вырезает части находящихся под ней форм, а оставшиеся формы сохраняют свои атрибуты. Команда **Crop** также применяет верхнюю форму к остальным, но на этот раз удаляет все, что не находится под ней; оставшиеся формы сохраняют свои атрибуты.

**C**

Если выбраны лишь комбинированные формы, то в меню **Modify** ⇒ **Combine Objects** доступна только команда **Union**. Можно воспользоваться ей вместо группирования комбинированных форм. Эта команда сохраняет пространственные отношения между формами, но позволяет изменять заливки и обводки непосредственно на арене, как описано в главе 4.

**C**

Если после выполнения команды **Modify** ⇒ **Combine Objects** ⇒ **Intersect** все формы исчезли, значит, не было ни одной точки, которая принадлежала бы им всем. На первый взгляд, очевидно, но если вы выбрали много форм или они достаточно сложные, то понять, пересекаются они или нет, не так-то просто.

# МНОГОСЛОЙНАЯ ГРАФИКА

---

## 6

В программе Adobe Flash CS3 Professional иллюзия трехмерности создается путем наложения графических элементов. В главе 6 вы научились делать это в пределах одного слоя, накладывая друг на друга объекты-рисунки, примитивные формы, группы и символы. Однако, чем больше элементов находится в слое, тем сложнее манипулировать ими и отслеживать порядок в стопке. Слои упрощают решение этой задачи.

Можно считать, что документ Flash представляет собой стопку диафильмов (длинная acetatная пленка, разбитая на кадры). Каждый диафильм — это аналог слоя Flash. Формы, нарисованные на верхнем диафильме, скрывают то, что находится в нижних. А там, где на верхней пленке ничего нет, проступают элементы, нарисованные на нижних.

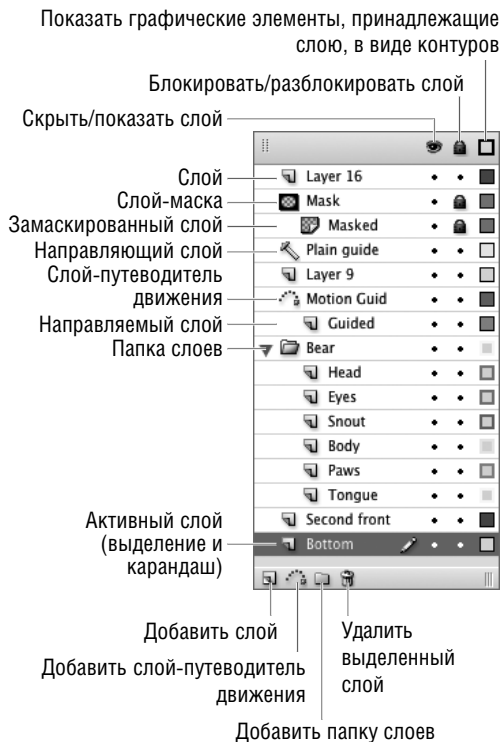
Распределяя элементы по разным слоям, становится проще следить за порядком в стопке и при необходимости изменять его. Можно придвинуть формы ближе к зрителю, перенести на верхний слой. Кроме того, формы из разных слоев не взаимодействуют, поэтому не нужно думать о группировке комбинированных форм и о том, как бы случайно не удалить нужную форму. Слои можно скрывать и показывать, им можно присваивать текстовые метки. Все это упрощает работу со слоями и элементами в документах Flash.

## Знакомство с функциями временной шкалы для работы со слоями

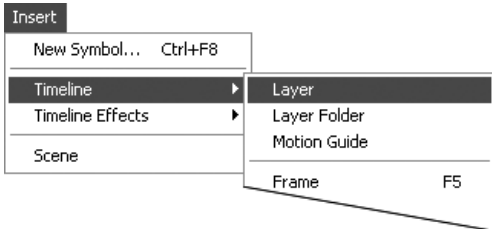
Графически Flash представляет каждый слой как горизонтальный раздел на временной шкале. Для просмотра и манипулирования этими разделами имеются специальные элементы управления. Существует ряд удобных функций для работы с графикой, размещенной в слоях, например, просмотр множества элементов в виде контуров и назначение каждому контуру своего цвета, чтобы было видно, какие элементы каким слоям принадлежат. Слой можно блокировать, чтобы случайно не изменить его содержимое, или скрыть, чтобы не загромождать арену. Можно создать специальные направляющие слои, которые помогают позиционировать элементы, маски для избирательного сокрытия и показа содержимого слоев и путеводители для анимации движения вдоль пути (см. главу 9).

Сложные ролики содержат десятки слоев. Просмотр такой насыщенной временной шкалы и навигация по ней может оказаться утомительным и непростым делом. Flash позволяет создавать папки для организации слоев в ролике. Так, можно поместить в одну папку все слои, относящиеся к одному персонажу или элементу. Flash считает папку частным случаем слоя, поэтому методы для добавления и удаления папок слоев похожи на те, что применяются к отдельным слоям. Однако в папках слоев не хранится графическое содержимое, и для них нет ни кадров, ни опорных кадров на временной шкале. (Опорными называются специальные кадры, в которые помещают графические элементы, мы познакомимся с ними в главе 8).

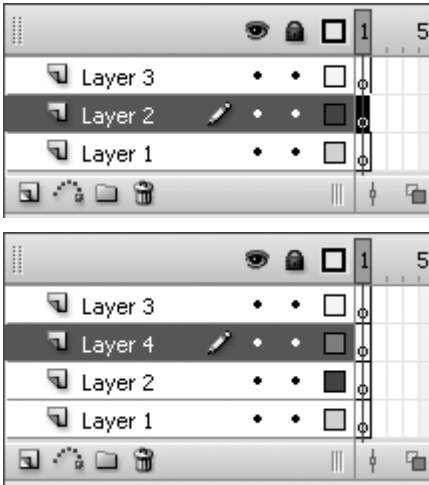
На рис. 6.1 представлен обзор важнейших средств для работы со слоями на временной шкале.



**Рис. 6.1.** На временной шкале графически представлены все слои, входящие в ролик. Папки слоев позволяют организовать слои в сложном ролике. Многим операциям создания и манипулирования слоями и папками соответствуют кнопки на панели временной шкалы.



**Рис. 6.2.** Для добавления нового слоя на временную шкалу выполните команду **Insert ⇒ Timeline ⇒ Layer**.



**Рис. 6.3.** Выберите слой, который должен оказаться под вновь вставленным (верхний рисунок); Flash вставляет новый слой прямо над выбранным и присваивает ему имя по умолчанию (нижний рисунок).

## Создание и удаление слоев и папок

При создании компонентов сцены в ролике вы можете добавлять столько слоев и папок, сколько необходимо.

### Чтобы добавить слой или папку:

1. На временной шкале выберите слой или папку.

Flash всегда добавляет новый слой или папку прямо над выбранным, поэтому выберите слой, который должен будет оказаться под новым. Чтобы добавить слой в самый низ шкалы, сначала создайте его где-нибудь, а затем перетащите мышью в конец стопки.

2. Чтобы добавить слой, выполните одно из следующих действий:

- Выполните команду **Insert ⇒ Timeline ⇒ Layer** (Вставить ⇒ Временная шкала ⇒ Слой) (рис. 6.2).
- Нажмите кнопку **Insert Layer** (Вставить слой) в полосе состояния временной шкалы (значок в виде листа бумаги с загнутым уголком).

Flash добавит новый слой и присвоит ему имя по умолчанию, например, Layer 4 (рис. 6.3).

3. Для добавления папки выполните одно из следующих действий:

- Выполните команду **Insert** ⇒ **Timeline** ⇒ **Layer Folder** (Вставить ⇒ Временная шкала ⇒ Папка слоев).
- Нажмите кнопку **Insert Layer Folder** (Вставить папку слоев) в полосе состояния временной шкалы (значок в виде папки).

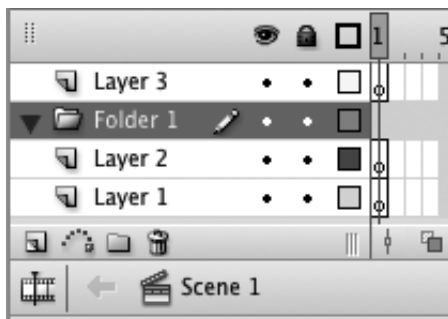
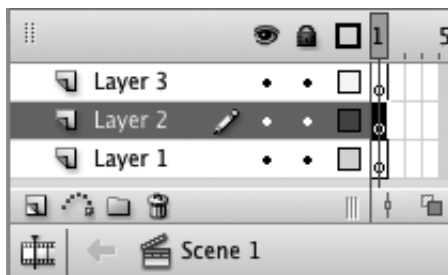
Flash добавит новую папку слоев и присвоит ей имя по умолчанию, например, *Folder 1* или *Folder 2* (рис. 6.4). Числа в именах, присваиваемых по умолчанию, основаны на количестве уже созданных слоев или папок в активной сцене, а не на количестве существующих в данный момент слоев или на текущей позиции. За слоями и папками слоев Flash следит независимо, поэтому первая созданная папка будет называться *Folder 1*, даже если слоев уже создано много.

### Чтобы удалить слой или папку:

1. Выберите на временной шкале слой или папку, которую хотите удалить.
2. Нажмите кнопку **Delete Layer** (значок с изображением мусорной корзины) (рис. 6.5).

Flash удаляет выбранный слой (и все принадлежащие ему кадры) или папку (вместе со всеми находящимися в ней слоями).

**C** В контекстном меню для слоев есть несколько команд, которые дублируются только кнопками на временной шкале, в частности, **Delete Layer** и **Delete Folder** (рис. 6.6). Чтобы открыть это меню, щелкните кнопкой мыши, удерживая клавишу **Control** (Mac) или правой кнопкой мыши (Windows).



**Рис. 6.4.** Выберите слой, который должен оказаться под новой папкой (верхний рисунок). Flash создаст новую папку над выбранным слоем (нижний рисунок). Имя новой папки присваивается исходя из числа уже созданных в текущей сцене папок.



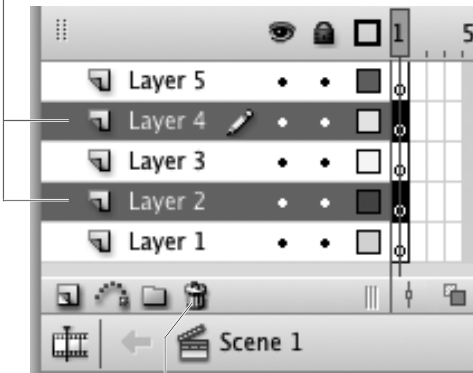
Удалить слой или папку

**Рис. 6.5.** Нажмите кнопку **Delete Layer**, чтобы удалить выбранный слой или папку.



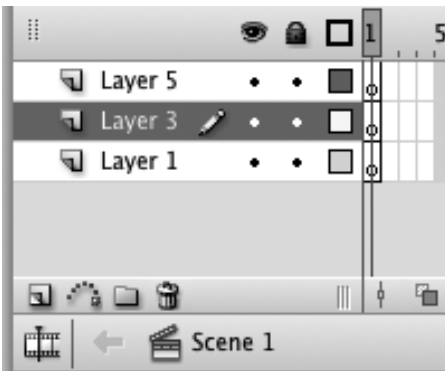
**Рис. 6.6.** Контекстное меню слоя дает быстрый доступ к командам, в том числе и к тем, которые дублируются только кнопками, например, **Delete Layer**. Чтобы открыть контекстное меню, щелкните кнопкой мыши, удерживая клавишу **Control** (Mac) или правой кнопкой мыши (Windows) по значку интересующего вас слоя.

Слой, выбранный для удаления



Нажать для удаления выбранных слоев

После удаления



**Рис. 6.7.** Чтобы удалить несмежные слои, щелкните по слоям, которые хотите выбрать с нажатой клавишей **(⌘)** (Mac) или правой кнопкой мыши (Windows). Затем нажмите кнопку **Delete Layer**.

### Чтобы удалить несколько слоев или папок:

1. На временной шкале выберите первый из тех слоев или папок, которые собираетесь удалить.
2. Щелкните мышью, удерживая клавишу **(⌘)** (Mac) или **Ctrl** (Windows) по каждому из остальных слоев/папок, которые нужно удалить. Этот способ позволяет выбирать не только соседние слои (рис. 6.7).

3. Нажмите кнопку **Delete Layer** (значок с изображением мусорной корзины).

Flash удаляет выбранные слои или папки с временной шкалы. Если среди выбранных были непустые папки, появится диалоговое окно с предупреждением о том, что вместе с папкой будут удалены все находящиеся в ней слои.

4. Чтобы удалить папку вместе со всеми находящимися в ней слоями, нажмите **Yes**.

или

Чтобы отменить операцию удаления, нажмите **No**.

**С**

Чтобы выбрать диапазон соседних слоев, щелкните по последнему из них, а затем, удерживая клавишу **Shift**, щелкните по первому. Flash выберет оба указанных вами слоя и все промежуточные.

**С**

Выбранные слои можно перетащить на кнопку **Delete Layer**, чтобы удалить их одной операцией.

**С**

Нельзя удалить все вообще слои, присутствующие на временной шкале. Если вы выберете все слои и папки и нажмете кнопку **Delete Layer**, то Flash удалит все, кроме самого нижнего. Если нижний слой находится в папке, Flash все равно оставит его, но извлечет из папки и сделает обычным слоем.

**С**

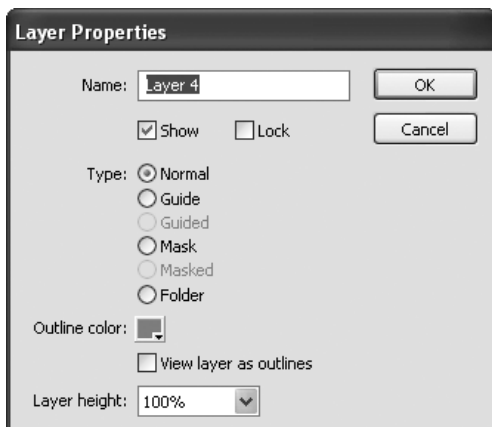
Можно также щелкнуть по выбранным слоям или папкам с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows), чтобы открыть контекстное меню, содержащее команды удаления.

## Управление слоями и папками

**Свойства слоя** — это параметры, которые определяют внешний вид и функционирование слоя. Папки слоев во Flash представляют собой особый вид слоев. Слойам и папкам можно присваивать имена. Можно скрывать или показывать слои и папки, блокировать во избежание непреднамеренного изменения и просматривать их содержимое в виде контуров. В общем случае Flash предлагает два способа управлять свойствами выбранного слоя или папки: задать значение свойства в диалоговом окне Layer Properties (Свойства слоя) или с помощью кнопок на панели временной шкалы.

### Чтобы задать значения в окне свойств слоя:

1. На временной шкале выберите слой, чьи свойства хотите задать или изменить.
2. Выберите пункт меню **Modify** ⇒ **Timeline** ⇒ **Layer Properties** (Изменить ⇒ Временная шкала ⇒ Свойства слоя). Открывается окно Layer Properties (рис. 6.8).
3. Чтобы присвоить слою имя, введите текст в поле Name (Имя).
4. Чтобы задать тип слоя, отметьте один из следующих переключателей:
  - Normal (Обычный) (принимается по умолчанию);
  - Guide (Направляющий) (помогает позиционировать элементы, не является частью опубликованного ролика);
  - Guided (Направляемый) (используется совместно со слоем-путеводителем для анимации вдоль пути);
  - Mask (Маска) (скрывает и показывает элементы, входящие в связанные слои);
  - Masked (Маскируемый) (содержит элементы, подлежащие маскированию);
  - Folder (Папка) (применяется для организации слоев).
5. Чтобы скрыть содержимое слоя (или папки слоев), сбросьте флажок **Show** (Показать). Чтобы показать содержимое, отметьте этот флажок.



**Рис. 6.8.** Тип и другие характеристики слоя можно задать в окне свойств слоя. Большинство из них можно задать и непосредственно на панели временной шкалы. Единственные свойства, которые нельзя задать таким образом, — это Layer Height (Высота слоя), Outline Color (Цвет контура) и тип «Направляющий» (который превращает слой в набор направляющих линий для точного позиционирования элементов).

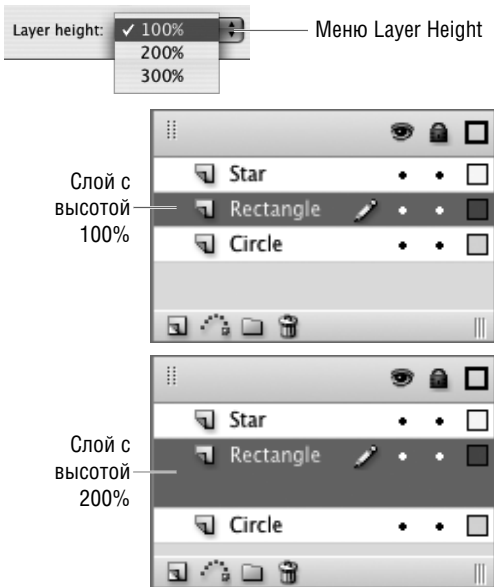
### Диалоговое окно свойств слоя и элементы управления на панели временной шкалы

Если вам нужно лишь задать видимость слоя, заблокировать его содержимое или просмотреть элементы в виде контуров, то это можно сделать как в окне свойств слоя, так и с помощью элементов управления на панели временной шкалы. Задание свойства в диалоговом окне не делает его более постоянным, чем в результате нажатия кнопки.

Окно свойств слоя обладает функциями, не имеющими аналогов на панели временной шкалы: создание слоев-направляющих, изменение высоты слоя на временной шкале, выбор цвета контуров и смена типа.

Но на панели временной шкалы есть возможность создавать слои-путеводители, которая в диалоговом окне отсутствует.





**Рис. 6.9.** Flash предлагает два увеличенных размера слоя на временной шкале. Увеличенные слои особенно полезны при работе со звуком. Чтобы увеличить высоту выбранного слоя, задайте процент в списке **Layer Height** в окне свойств слоя.

6. Чтобы предотвратить изменение содержимого слоя или папки, отметьте флажок **Lock** (Блокировка). Чтобы разрешить изменения, сбросьте этот флажок.
7. Чтобы просмотреть содержимое слоя в виде контуров, отметьте флажок **View layer as outlines** (Просмотр слоя в виде контуров). Если этот флажок сброшен, то показываются сплошные графические элементы.
8. Чтобы задать цвет контуров для данного слоя, воспользуйтесь элементом управления цветом контура (см. раздел «Создание сплошных цветов и градиентов» в главе 2). Это свойство можно изменить только в окне свойств слоя.
9. Чтобы изменить высоту слоя или папки на временной шкале, выберите процентное значение из раскрывающегося списка **Layer Height** (рис. 6.9).

Это свойство можно изменить только в окне свойств слоя.

#### 10. Нажмите **OK**.

Flash применяет заданные параметры к текущему слою.

**C** Чтобы изменить размер графического представления всех слоев на временной шкале, выберите размер из меню **Frame View**, которое находится в правом верхнем углу временной шкалы. Пункты **Preview** (Просмотр) и **Preview in Context** (Просмотр в контексте) выводят миниатюрные изображения содержимого каждого кадра, принадлежащего слою.

**C** Чтобы заставить Flash выделять все выбранные графические элементы слоя одним и тем же цветом, откройте диалоговое окно параметров (из меню **Flash** (Mac) или **Edit** (Windows) выберите пункт **Preferences**), и в категории **General** (Общие) отметьте переключатель **Use Layer Color** (Использовать цвет слоя) в разделе **Color Highlight** (Цвет выделения).

**C** Двойной щелчок по значку страницы с уголком или с папкой в области конкретного слоя либо по значку папки на панели временной шкалы открывает окно свойств этого слоя или папки.

**C** Если для папки слоев отмечен флажок **View layer as outlines**, то Flash отображает в виде контуров все слои, находящиеся в этой папке. Хотя для папки и можно задать цвет контура, он не влияет на то, что вы видите на арене. Если для папки включен режим просмотра в виде контуров, то каждый слой из папки отображается цветом, заданным именно для этого слоя.

## Задание свойств слоев на панели временной шкалы

На панели временной шкалы слои и папки слоев представлены горизонтальной полосой, содержащей имя и три кнопки для управления представлением слоя или папки на арене. С помощью этих кнопок слой или папку можно скрыть (то есть временно сделать все элементы невидимыми), заблокировать (сделать элементы видимыми, но не изменяемыми) или показать содержимое в виде контуров. Описанные возможности полезны, когда приходится редактировать много элементов, находящихся в разных слоях.

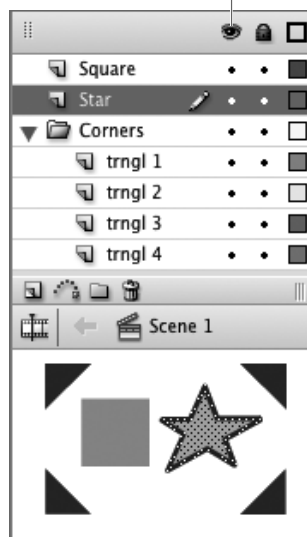
### Чтобы переименовать слой или папку:

1. Дважды щелкните по имени слоя или папки на временной шкале. Flash активирует текстовое поле для ввода имени.
2. Введите новое имя.
3. Нажмите **Enter** или щелкните вне поля имени.

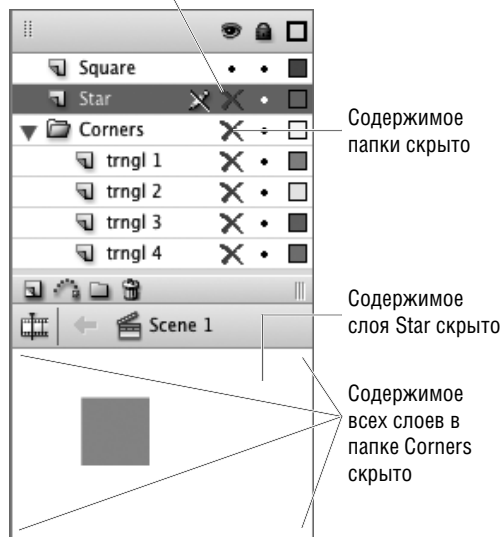
### Чтобы скрыть/показать содержимое слоя или папки:

- На панели временной шкалы щелкните по значку в колонке с изображением глаза, относящемуся к тому слою или папке, которую вы собираетесь скрыть или показать (рис. 6.10). Состояние переключается между «видимый» и «скрытый». Красная буква X в колонке говорит, что слой (папка) скрыт; содержимое скрытого слоя не показывается на арене. Кружочек означает, что слой видимый. Состояние показа не влияет на окончательный ролик. При публикации содержимое всех скрытых слоев и папок.

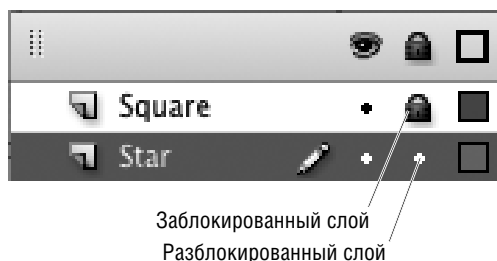
Переключает видимость содержимого слоя



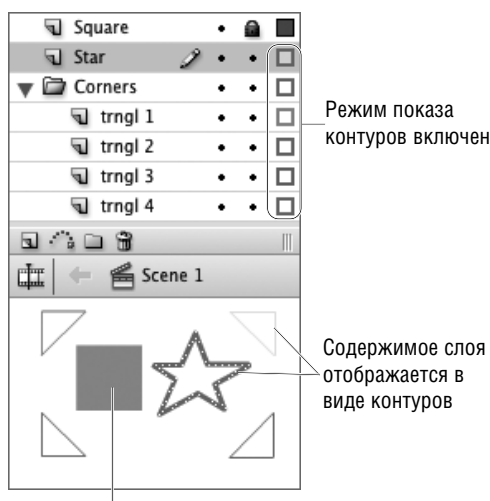
Содержимое слоя скрыто



**Рис. 6.10.** Колонка с изображением глаза в заголовке управляет видимостью слоев. Каждый из шести показанных элементов (верхний рисунок) отображается в отдельном слое. Четыре угловых треугольника находятся в папке Corners. Если скрыть слой Star, то звезда исчезнет с арены; если скрыть еще и папку Corners, то пропадут и треугольники (нижний рисунок).



**Рис. 6.11.** Замок означает, что слой заблокирован. Его содержимое видно на арене, но недоступно для редактирования.



Содержимое остальных слоев отображается в виде сплошных форм

**Рис. 6.12.** Полный квадратик в последней колонке означает, что графические элементы слоя отображаются в виде контуров. Если задать такой режим для папки, то он автоматически будет установлен для всех находящихся в ней слоев.

### Чтобы заблокировать/разблокировать слой или папку:

- На панели временной шкалы щелкните по значку в колонке с изображением замка, относящемуся к тому слою или папке, которую вы собираетесь заблокировать или разблокировать (рис. 6.11). Состояние переключается между «заблокирован» и «разблокирован». Запертый замок в колонке говорит, что слой (папка) заблокирован; содержимое такого слоя показывается на арене, но редактировать его нельзя. Кругочек означает, что слой разблокирован; его содержимое доступно для редактирования. Состояние блокировки не влияет на окончательный ролик.

### Чтобы просмотреть содержимое слоя или папки в виде контуров/сплошных форм:

- На панели временной шкалы щелкните по значку в колонке с изображением квадратика, относящемуся к тому слою (папке), содержимое которого вы собираетесь просматривать в виде контуров или сплошных форм (рис. 6.12). Состояние переключается между «контур» и «сплошные формы». Если в колонке отображается только цветной контур квадратика, то содержимое слоя показывается на арене в виде контуров. Если же квадратик сплошной, то содержимое показывается полностью. Для обычных слоев цвет контура совпадает с цветом контуров на арене. (Для папок цвет контура не имеет значения; квадратик лишь индицирует режим показа.) Состояние показа не влияет на окончательный ролик.

## С

Задание режима показа контуров для папки означает, что содержимое всех слоев, находящихся в этой папке, будет показываться в виде контуров. Чтобы представить содержимое отдельного слоя в виде сплошных форм, щелкните по квадратнику, соответствующему этому слою.

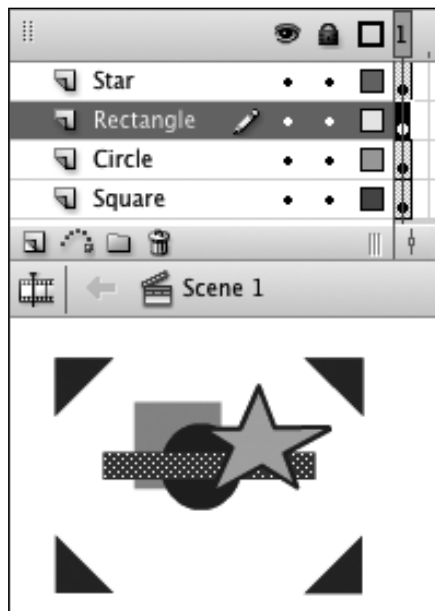
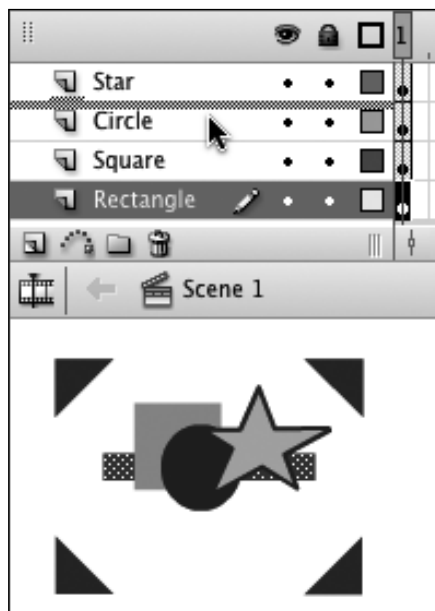
## Организация слоев с помощью папок

Организовать слои для удобства работы с ними можно, позиционируя их на временной шкале или распределив по папкам. Слои позволяют изменять порядок сразу нескольких элементов в стопке за одну операцию. Можно, например, поместить все элементы из одного слоя на вершину стопки; для этого достаточно перетащить слой в начало списка на временной шкале. В результате принадлежащие слою элементы окажутся на переднем плане (перекрывая элементы из остальных слоев) во всех кадрах ролика. Перестановка папки на временной шкале меняет порядок в стопке для всех слоев, находящихся в этой папке. Поместив слои в папку, вы получаете возможность скрыть или показать их все, просто свернув или раскрыв эту папку на временной шкале.

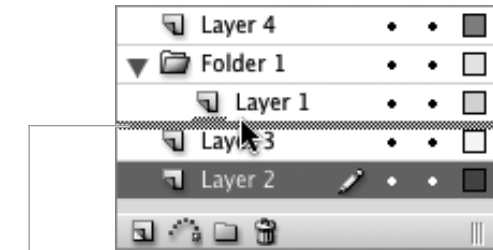
### Чтобы переупорядочить слои или папки:

1. На панели временной шкалы укажите курсором на значок того слоя или папки, который хотите переставить в другое место. Папка может быть закрыта или открыта (см. раздел «Чтобы скрыть/показать слои внутри папок на панели временной шкалы» ниже).
2. Перетащите слой мышью. Flash индицирует то место, где окажется слой после перетаскивания.
3. Подтащите индикатор к тому месту в списке, куда хотите поместить слой (рис. 6.13)
4. Отпустите кнопку мыши.

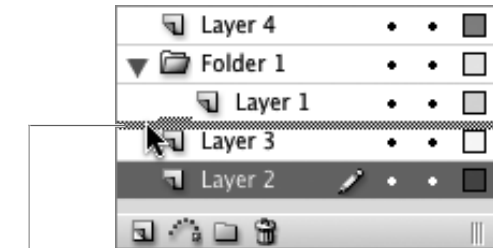
Flash перемещает слой или папку в новое место и оставляет его выбранным. Если был выбран одиночный слой, то его содержимое будет выбрано на арене; содержимое слоев, принадлежащих выбранной папке, не выбирается.



**Рис. 6.13.** Серая линия (верхний рисунок) обозначает новое положение перетаскиваемого слоя (в данном случае слоя Rectangle). Отпустите кнопку мыши, чтобы расположить слой в новом месте. Flash выбирает этот слой и все его содержимое (нижний рисунок). Прямоугольная форма переместилась в стопке с нижней позиции в позицию под звездой.

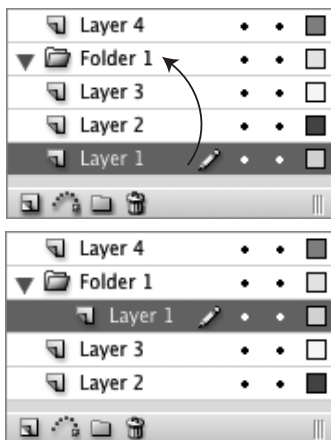


Слой окажется внутри папки



Слой окажется вне папки

**Рис. 6.14.** Если индикатор перемещения слоя оказывается под открытой папкой, то необходимо сообщить Flash, хотите ли вы, чтобы слой оказался внутри или вне папки. Выступ на индикаторе показывает, где будет находиться слой. Чтобы оставить слой вне папки, буксируйте влево, пока выступ не окажется под значком папки, а чтобы поместить в папку, буксируйте вправо, пока выступ не окажется под значком слоя.



**Рис. 6.15.** Когда перетаскиваемый слой оказывается над папкой, индикатор исчезает (верхний рисунок). Отпустите кнопку мыши, чтобы поместить слой в эту папку. Flash выберет и сам слой, и его содержимое (нижний рисунок).

**С**

Расположить слой под открытой папкой чуть сложнее. Необходимо решить, что вы хотите: чтобы слой находился внутри или вне папки. Когда индикатор окажется ниже последнего слоя в открытой папке, следите за выступом на нем; он обозначает, куда будет помещен слой. Чтобы поместить слой вне открытой папки, буксируйте влево, пока выступ не окажется под значком папки (рис. 6.14). Чтобы поместить слой внутрь папки, буксируйте вправо, пока выступ не окажется под значком слоя.

### Чтобы переместить существующий слой в папку:

1. Загрузив документ с несколькими слоями и папками, укажите курсором на значок того слоя на панели временной шкалы, который хотите переместить в папку.
2. Перетащите слой на папку, где он должен находиться. В процессе буксировки Flash индицирует новое место слоя толстой серой линией; как только курсор окажется точно над папкой, линия исчезнет (рис. 6.15).
3. Отпустите кнопку мыши.

Flash переместит слой в папку и сделает его первым в ней. При этом перед именем слоя на временной шкале появится отступ, слой окажется выбранным на временной шкале, а его содержимое — на арене.

**С**

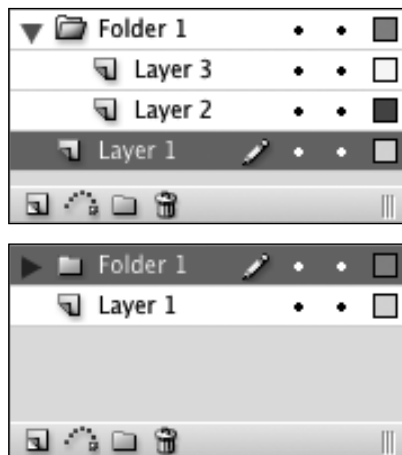
Если вы не хотите, чтобы перемещенный слой был первым в папке, не отпускайте кнопку мыши над папкой. Вместо этого расположите индикатор между какими-нибудь двумя слоями в этой папке. Тогда после отпускания кнопки слой окажется именно в этом месте.

### Чтобы скрыть/показать слои внутри папок на панели временной шкалы

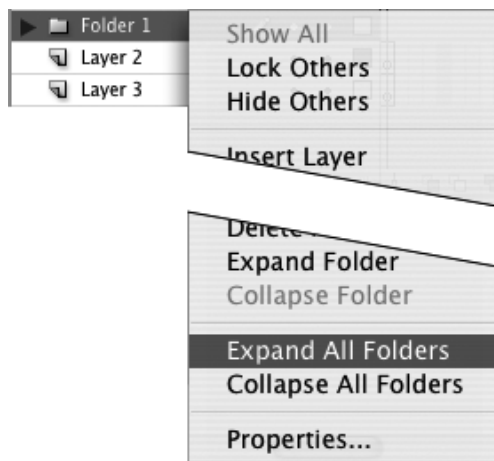
- На панели временной шкалы щелкните по треугольничку слева от значка папки (рис. 6.16). Состояние папки изменяется с открытой на закрытую или наоборот. Если треугольничек указывает вправо, то появляется значок закрытой папки, и на временной шкале исчезают все слои, находящиеся в этой папке. Если же треугольничек указывает вниз, то появляется значок открытой папки, и находящиеся в ней слои видимы.

С

С помощью контекстного меню можно сразу открыть или закрыть все папки в текущем ролике. Откройте меню, щелкнув по любому слою на временной шкале кнопкой мыши с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows). Выберите из меню команду **Expand All Folders** (Развернуть все папки) или **Collapse All Folders** (Свернуть все папки) (рис. 6.17).

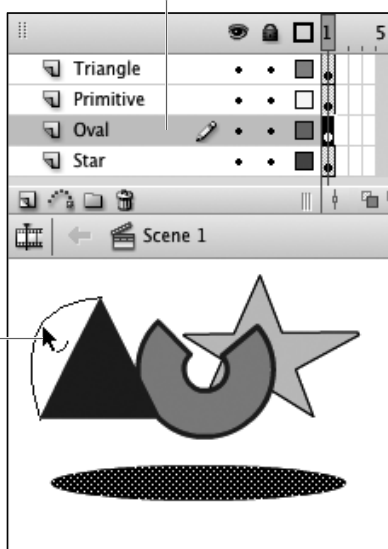


**Рис. 6.16.** Щелчок по треугольничку слева от значка папки переключает ее состояние с «открыта» (верхний рисунок) на «закрыта» (нижний рисунок).



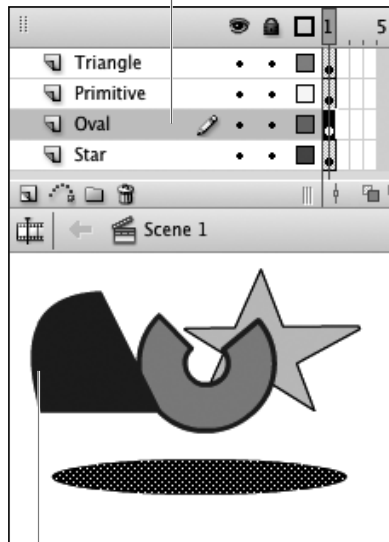
**Рис. 6.17.** Чтобы открыть или закрыть сразу все папки, откройте контекстное меню, щелкнув по любому слою на временной шкале кнопкой мыши с нажатой клавишей Control (Mac) или правой кнопкой мыши (Windows). Затем выберите команду **Expand All Folders** или **Collapse All Folders**.

Активный слой



С помощью инструмента «Стрелка» модифицируете форму в неактивном слое

Активный слой не изменился



Flash перерисовывает форму

**Рис. 6.18.** Активным слоем является Oval, но вы тем не менее можете редактировать комбинированные формы и объекты-рисунки, расположенные в неактивных слоях. Модификация контура формы инструментом «Стрелка» не приводит к активации слоя, содержащего эту форму.

## Работа с графикой в нескольких слоях

Если вы не заблокировали формы, не заблокировали и не скрыли слои, то для редактирования доступны комбинированные формы и объекты-рисунки, принадлежащие любому слою. К ним можно применять инструменты «Стрелка», «Ведро с краской» и «Чернильница». К примитивам применимы инструменты «Ведро с краской» и «Чернильница». В активный (выбранный в данный момент) слой можно добавлять новые формы.

### Чтобы активировать слой:

- Чтобы выбрать слой, в который будут добавляться графические объекты, выполните одно из следующих действий:

- щелкните по имени слоя на временной шкале.

Область, содержащая имя слоя, выделяется синим цветом, а справа от имени появляется значок карандаша. Flash выбирает все графические объекты, принадлежащие этому слою.

- щелкните по графическому объекту на арене.

Flash выбирает этот объект, выделяет имя слоя серым цветом и помещает справа от имени значок карандаша.

### Чтобы редактировать контуры комбинированной формы или объекта-рисунка в неактивном слое:

1. Загрузите документ, содержащий комбинированные формы или объекты-рисунки, расположенные в двух или более слоях. Инструментом «Стрелка» укажите на форму, принадлежащую неактивному слою (для которого не отображается значок карандаша). Появляется значок гладкой или угловой точки.
2. Буксируйте контур формы в неактивном слое.
3. Отпустите кнопку мыши.

Flash перерисовывает форму (рис. 6.18). Слой, который был активным в начале операции, таким и остается.

С

В отличие от комбинированных форм и объектов-рисунков, использование инструмента «Стрелка» для модификации контура примитива, расположенного в неактивном слое, делает этот слой активным. Начать модификацию — например, изменить расположение точек, управляющих радиусом закругления прямоугольного примитива, — можно даже тогда, когда примитив не выбран. Но по завершении операции Flash выбирает этот примитив на арене, а содержащий его слой становится активным на временной шкале.

С

Инструментами Безье можно модифицировать только формы, расположенные в активном слое.

### Чтобы изменить заливку в неактивном слое:

1. Загрузив документ, содержащий формы, расположенные в двух или более слоях, выберите инструмент «Ведро с краской».
2. На панели цветов задайте новый цвет заливки.
3. Укажите ведром с краской на форму, принадлежащую неактивному слою, и щелкните кнопкой мыши. Flash закрасит форму выбранным цветом, но слой останется неактивным (рис. 6.19).

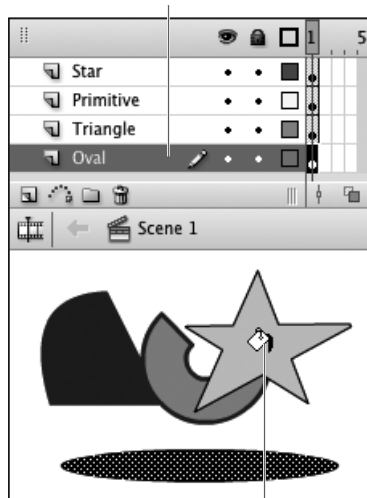
С

При работе с комбинированными формами возьмите за правило создавать каждую форму в отдельном слое. Тогда при необходимости будет несложно изменить порядок в стопке. Не повредит, если вы будете помещать в отдельные слои также объекты-рисунки и примитивы. От количества слоев размер файла готового ролика не зависит.

С

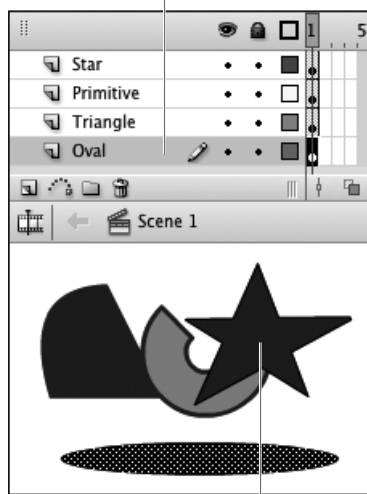
Не забывайте, что после выбора нового цвета перекрасятся все выбранные заливки, расположенные в активном слое. Чтобы сохранить цвет заливки, отмените ее выбор перед выполнением шага 2.

Активный слой



С помощью инструмента «Ведро с краской» измените цвет заливки в неактивном слое

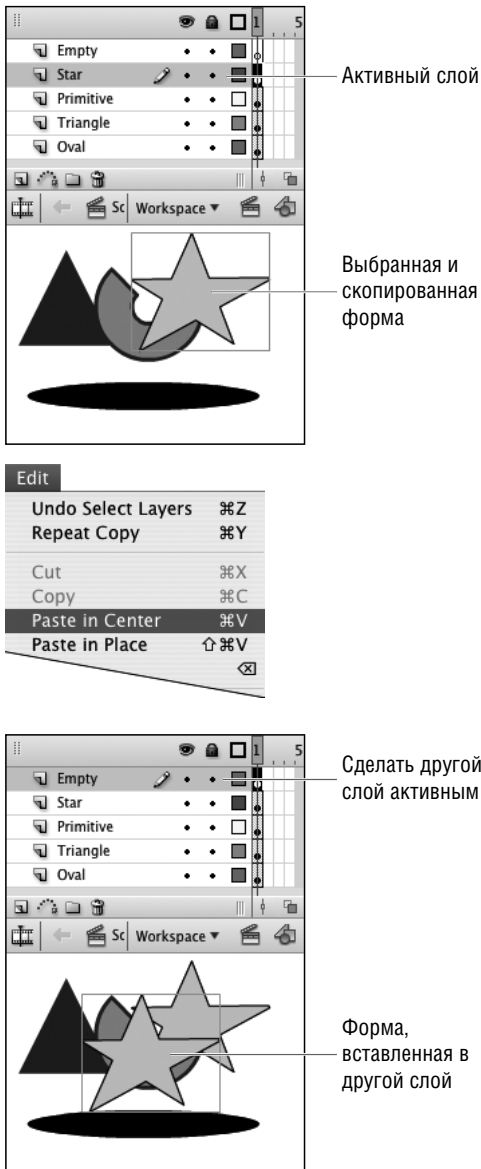
Активный слой не изменился



Flash перекрашивает заливку в новый цвет

**Рис. 6.19.** Использование ведра с краской для изменения цвета заливок в неактивном слое не активирует этот слой.





**Рис. 6.20.** Для копирования формы из одного слоя в другой необходимо выбрать форму (верхний рисунок), скопировать ее, выбрать конечный слой и вставить в него копию. Команда **Paste in Center** (средний рисунок) размещает вставленную форму в центре окна (нижний рисунок).

## Межслойные операции вырезания и вставки

Flash позволяет создавать графику только в активном слое документа. Но можно копировать, вырезать и удалять элементы из любого видимого, разблокированного слоя. Можно выбрать элементы в нескольких слоях, вырезать их, а затем вставить в какой-нибудь другой слой. Или вырезать элементы по одному из некоторого слоя, а затем перераспределить их между другими слоями.

### Чтобы вставить элемент из одного слоя в другой:

1. Создайте или откройте документ с несколькими слоями.
2. Поместите в каждый слой, кроме одного, хотя бы по одному элементу.

Чтобы с элементами было проще работать, создайте их как объекты-рисунки или примитивы либо включите каждую комбинарованную форму в отдельную группу. Один слой оставьте пустым.

3. Выберите на арене какую-нибудь форму. Flash подсветит соответствующий ей слой на временной шкале серым цветом.
4. Выполните команду **Edit ⇒ Copy** (Редактирование ⇒ Копировать).
5. На временной шкале выберите пустой слой.
6. Выполните команду **Edit ⇒ Paste in Center** (Редактирование ⇒ Вставить в центр).

Flash вставит копию формы в пустой слой и расположит ее в центре окна (рис. 6.20). При желании можно сдвинуть форму в другое место.

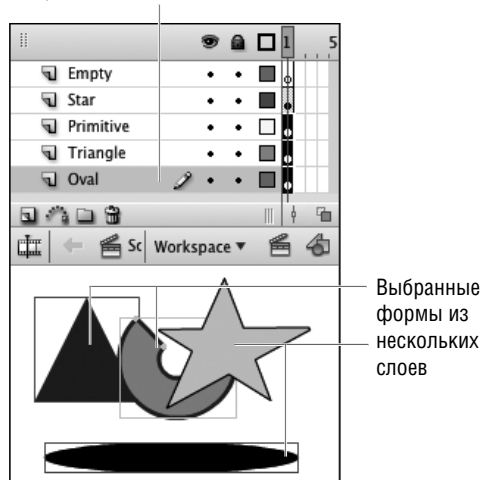
**Чтобы воспользоваться командой Paste in Place для копирования между слоями:**

1. Загрузите документ, в котором несколько слоев содержат формы, а один слой пуст, и выберите одну форму. На временной шкале слой, содержащий выбранную форму, становится активным.
2. Добавьте в набор выбранных еще какие-нибудь формы, как описано в главе 4. На временной шкале слой, содержащий последнюю выбранную форму, становится активным.
3. Выполните команду **Edit ⇒ Cut** (Редактирование ⇒ Вырезать). Flash удаляет выбранные формы (рис. 6.21).
4. На временной шкале выберите пустой слой.
5. Выполните команду **Edit ⇒ Paste in Place** (Редактирование ⇒ Вставить на месте) (рис. 6.22). Flash вставит все формы в те же места на арене, где они были раньше, но уже в другой слой (рис. 6.23). Если временно скрыть активный слой, то этих форм не будет видно на арене.

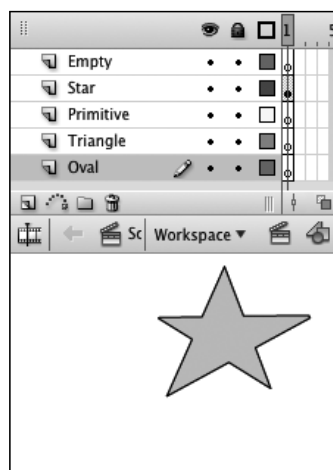
### Два способа вставки

Flash предлагает две команды вставки: **Paste in Center** и **Paste in Place**. Команда **Paste in Center** помещает элементы в центр открытого окна. (Отметим, что центр окна обязательно совпадает с центром арены; если нужно вставить в центр арены, то следует центрировать арену в открытом окне.) Команда **Paste in Place** помещает элемент в точку с теми же координатами, которые были у него до вырезания или копирования. Эта команда полезна, когда нужно сохранить точное взаимное расположение элементов на сцене при перемещении их из одного слоя в другой.

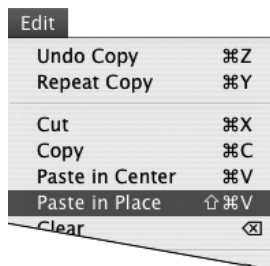
Активным является слой, содержащий последний выбранный элемент



После вырезания форм

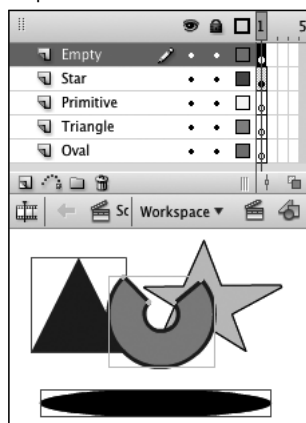


**Рис. 6.21.** Первый шаг перемещения элементов из разных слоев в один заключается в том, чтобы выбрать и вырезать необходимые элементы. Затем нужно будет вставить их в новый активный слой.

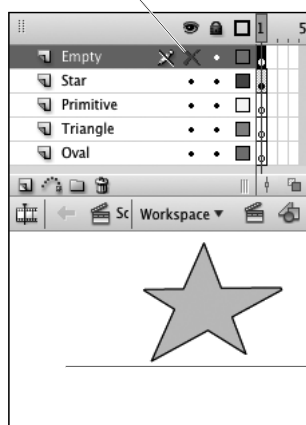


**Рис. 6.22.** Для вставки элементов в исходные позиции выполните команду Edit ⇒ Paste in Place.

Формы после вставки на месте



Щелкнуть, чтобы скрыть слой



Слой, содержащий формы, теперь скрыт

**Рис. 6.23.** Команда Paste in Place вставляет элементы в новый слой. При этом у каждой формы сохраняются те координаты, которые она имела в исходном слое. Скройте новый слой, если хотите убедиться, что элементы действительно перенесены из тех слоев, где находились раньше.

**С** После выбора элемента на арене активным на временной шкале становится слой, содержащий этот элемент. Когда вы перемещаете элементы между слоями, полезно помнить, что справедливо и обратное: при выборе слоя на временной шкале Flash выбирает на арене все элементы, принадлежащие этому слою.

**С** Цвет выделения активного слоя на временной шкале изменяется в зависимости от того, кто владеет фокусом: временная шкала или арена. Синий цвет выделения означает, что фокус принадлежит временной шкале (например, потому что вы только что щелкнули по какому-нибудь слою на ней). Серое выделение говорит о том, что фокус сейчас у арены (например, вы только что выбрали элемент на арене или вставили на арену элемент с помощью команды Paste). Это тонкое различие помогает понять, какую операцию вы только что завершили.

**С** Процедура вырезания элементов и последующего выполнения команды **Paste in Place** отнимает много времени, так как приходится переключаться с одного слоя на другой. Чтобы автоматизировать этот процесс, воспользуйтесь командой **Distribute to Layers** (Распределить по слоям) (см. раздел «Распределение графических элементов по слоям» ниже в этой главе).

### Куда попадают элементы после вставки?

В документе Flash в каждый момент времени может быть только один активный слой. Все вновь создаваемые формы попадают в активный слой. То же относится к копиям элементов и экземпляров символов; если скопировать и вставить некий элемент, то копия окажется в активном слое. Точно так же, в активный слой попадает экземпляр символа, который вы перетаскиваете из окна библиотеки.

## Распределение элементов по слоям

В процессе рисования элементов для ролика вы можете забыть о необходимости создавать новый слой для каждого элемента. Использование команд **Cut** и **Paste in Place** отнимает время. Команда **Distribute to Layers** (Распределить по слоям) позволяет автоматизировать эту процедуру; она помещает каждый из выбранных элементов в отдельный слой. Это особенно удобно, если вы создаете анимацию путем интерполяции движения, когда каждый анимируемый элемент обязан находиться в отдельном слое. (Об интерполяции см. главу 9.)

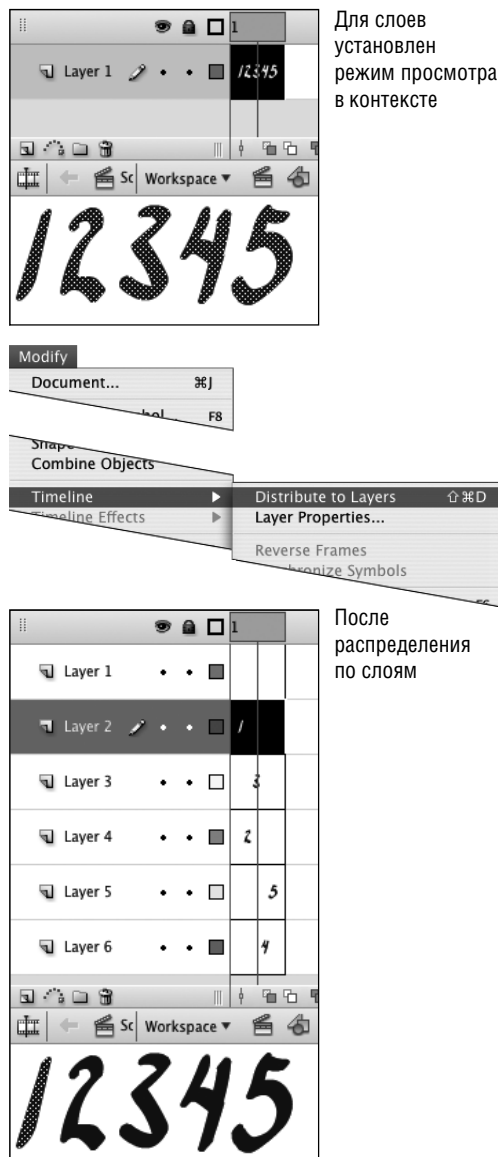
### Чтобы поместить выбранные элементы в отдельные слои:

1. Откройте новый документ и создайте на арене несколько форм, принадлежащих одному слою.
2. Выполните команду **Edit ⇨ Select All** (Редактирование ⇨ Выделить все). Flash выделяет все формы.
3. Выполните команду **Modify ⇨ Timeline ⇨ Distribute to Layers** (Изменить ⇨ Временная шкала ⇨ Распределить по слоям) или нажмите комбинацию клавиш **Shift-⌘-D** (Mac) либо **Ctrl-Shift-D** (Windows) (рис. 6.24).

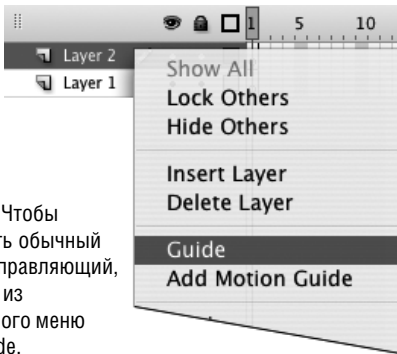
Flash создает для каждой формы свой слой и добавляет эти слои в конец временной шкалы. Каждая форма занимает на арене то же место, что и раньше, однако находится в другом слое.

**C** Команда **Distribute to Layers** работает как для выбранных графических объектов (текстовых полей, объектов-рисунков, примитивов, групп и символов), так и для комбинированных форм. Flash помещает каждый выбранный графический объект в отдельный слой, при этом различные элементы объекта остаются связанными.

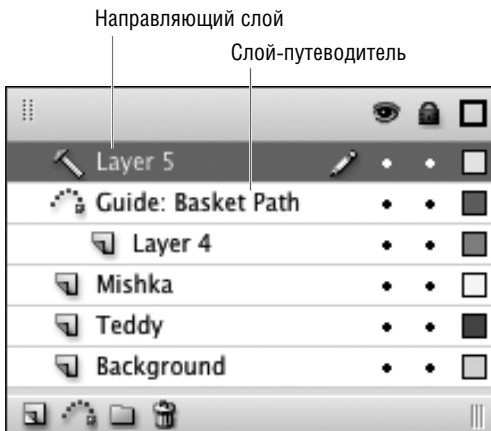
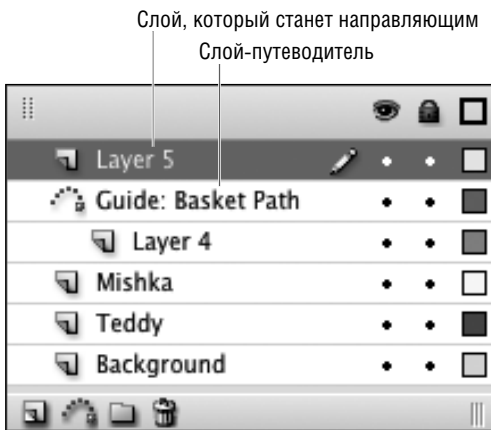
**C** После выполнения команды **Distribute to Layers** невыбранные элементы остаются там же, где и были. В новые слои перемещаются только те элементы, которые были выбраны.



**Рис. 6.24.** Если выбрать элементы на арене (верхний рисунок) и выполнить команду **Modify ⇨ Timeline ⇨ Distribute to Layers** (средний рисунок), то для каждого элемента автоматически выполняется команда вырезания и вставки в новый слой. Новые слои располагаются в том же порядке, в каком вы помещали элементы на арену. Элементы на этом рисунке добавлялись в порядке следования их номеров, поэтому изображение числа 1 оказывается в первом из созданных слоев (нижний рисунок).



**Рис. 6.25.** Чтобы превратить обычный слой в направляющий, выберите из контекстного меню пункт **Guide**.



**Рис. 6.26.** Выберите слой (верхний рисунок) и определите его как направляющий. На временной шкале направляющие слои идентифицируются Т-образным значком; сравните его со значком слоя-путеводителя (нижний рисунок).

## Работа с направляющими слоями

Flash предлагает два вида направляющих слоев: направляющие и путеводители. Обычные направляющие слои могут содержать любые элементы: линии, формы или символы. Содержимое такого слоя служит просто эталоном для размещения элементов на арене. Flash не включает графическое содержимое направляющих слоев в готовый ролик.

Слой-путеводитель содержит единственную линию, которая направляет движение анимированного элемента вдоль пути (см. главу 9). Следует запомнить и еще одно различие — при создании путеводителя Flash самостоятельно добавляет новый слой на временную шкалу. Для создания направляющих слоев вы должны переопределить роль какого-то уже существующего слоя.

### Чтобы создать обычный направляющий слой:

1. Выполните одно из следующих действий:
  - создайте новый слой на временной шкале (например, нажав кнопку **Insert Layer**). Flash выберет новый слой;
  - выберите уже существующий слой.
2. Щелкните кнопкой с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows) и выберите из контекстного меню пункт **Guide** (Направляющий) (рис. 6.25).

Выбранный слой станет направляющим, а перед его именем появится Т-образный значок (рис. 6.26). Если хотите, можете переименовать слой, чтобы было понятно, что он является направляющим.

3. Чтобы с направляющими элементами было проще работать, выберите из меню пункт **View ⇒ Snapping** (Просмотр ⇒ Привязка) и выполните одну из команд в подменю:

- команду **Snap to Objects** (Привязка к объектам) (рис. 6.27).
- команду **Snap Align** (Выравнивание с привязкой).

Flash отображает направляющие выравнивания, когда перетаскиваемая форма или графический объект оказывается близко к другой форме или графическому объекту. Это упрощает выравнивание по элементам, которые находятся в направляющем слое.

C

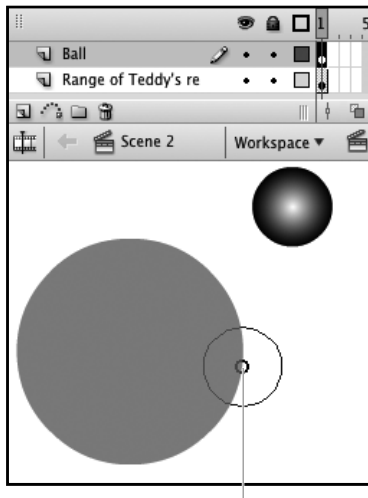
Разместив направляющие элементы там, где они должны быть в конкретной сцене, заблокируйте направляющий слой, чтобы случайно не сдвинуть эти элементы, когда будете рисовать в других слоях.

C

Название команды **Snap to Guide** (Привязка к направляющим) наводит на мысль, что она может оказаться полезной для привязки к элементам в направляющем слое, однако это не так. Здесь имеются в виду направляющие, которые вытягиваются из линеек (см. главу 1).

C

Поскольку графические элементы, находящиеся в направляющих слоях, не включаются в готовый ролик, то можете использовать их как записную книжку, оставляя заметки для других людей, которые будут работать над тем же файлом.

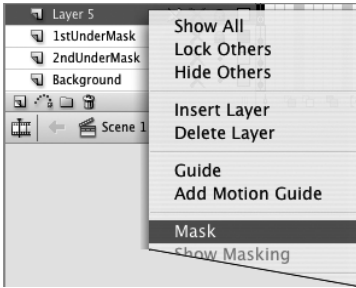


Центральные точки совмещаются, когда вы буксируете элемент

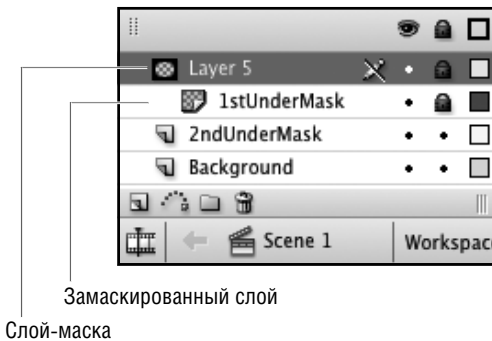
**Рис. 6.27.** Выполните команду **View ⇒ Snapping ⇒ Snap to Objects** (верхний рисунок), чтобы перетаскиваемые элементы принудительно совмещались с другими линиями или формами, например, теми, которые присутствуют в направляющем слое.

C

Размещая новые слои под направляющим, легко непреднамеренно преобразовать направляющий слой в слой-путеводитель. Перетаскивая какой-нибудь слой под направляющий, обращайтесь внимание на индикатор места вставки и значок в направляющем слое. Чтобы не изменить тип направляющего слоя, буксируйте влево, пока выступ на индикаторе не окажется под Т-образным значком в направляющем слое. Если вы хотите создать слой-путеводитель, буксируйте вправо; значок в направляющем слое изменится на составленную из точек дугу, обозначающую путеводитель (подробнее о путеводителях см. главу 9).



**Рис. 6.28.** Чтобы сделать слой маской, выберите пункт **Mask** из контекстного меню слоя.



**Рис. 6.29.** Значок слоя-маски имитирует наложение темной маски на шахматную доску. Имена замаскированных слоев отображаются с отступом от левого края и помечаются значком с шахматным узором.

## Работа со слоями-масками

Слой-маска — это особый вид слоя, который позволяет скрывать и показывать элементы, находящиеся в слоях, расположенных под ним. В готовом ролике формы из слоя-маски становятся «дырками», через которые проглядывают элементы из связанных слоев.

### Чтобы создать слой-маску:

1. Выполните одно из следующих действий:
  - создайте на временной шкале новый слой (например, нажав кнопку **Insert Layer**). Этот слой становится выбранным;
  - выберите уже существующий слой.

Вообще говоря, разумно создавать (или выбирать) слой, расположенный непосредственно над слоем, в котором имеется контент, подлежащий маскированию. Но можно создать слой-маску отдельно, а позже связать с ним маскируемые слои.

2. Щелкните по слою кнопкой мыши с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows), чтобы открыть его контекстное меню и выполните команду **Mask** (Маска) (рис. 6.28)

Чтобы реализовать эффект маскирования, Flash автоматически определяет этот слой как маску, привязывает к маске слой, расположенный непосредственно под ним, и блокирует оба слоя (рис. 6.29). При желании можно переименовать слой, чтобы сразу было ясно, что это маска.

**С**

Поместить слой под список замаскированных слоев не так-то просто. Когда индикатор места вставки оказывается ниже последнего замаскированного слоя, вы можете сделать одно из двух: добавить слой и замаскировать его или оставить на основном уровне временной шкалы. Для различения этих случаев на индикаторе вставки есть визуальные ключи. Чтобы новый слой стал замаскированным, отбуксируйте его немного вправо, пока выступ на линии индикатора не окажется под значком сложенной страницы. А чтобы новый слой остался на основном уровне временной шкалы, отбуксируйте его влево, пока выступ не окажется под значком маски.

**С**

Чтобы быстро связать с маской существующие слои, перетащите их на временной шкале, так чтобы они оказались непосредственно под маской или под одним из привязанных к ней слоев.

**С**

Привязав к маске один слой, можно затем привязать и дополнительные. Выберите привязанный (замаскированный) слой, а затем выполните те же действия, что при создании нового слоя. Flash добавит новые слои прямо над выбранным и сделает их замаскированными.

**Чтобы создать маску:**

1. Создайте один или несколько слоев, содержащих графические элементы, которые должны быть видны только сквозь маску.
2. Создайте слой-маску над слоями с замаскированным содержимым и убедитесь, что этот слой выбран, видимый и разблокирован.

На панели временной шкалы слой должен быть выделен, а в колонках со значками глаза и замка должны быть кружочки (не буква X и не значок запертого замка).

3. С помощью инструментов рисования создайте в слое-маске заливку (рис. 6.30). В качестве масок Flash воспринимает только заливки, все обводки в слое-маске игнорируются.

**С**

Лучше в качестве маски ограничиться одной комбинированной формой или одним графическим объектом (объектом-рисунком, примитивом, группой, символом или текстовым полем со статическим текстом). Если слой-маска содержит несколько комбинированных форм, то поначалу маска, может быть, и сработает. Но стоит подвинуть любой кусочек формы, как Flash перестанет рассматривать его как часть маски. Если слой-маска содержит комбинированную форму и графический объект, то для создания маски Flash использует только комбинированную форму. Если в слое-маске есть несколько графических объектов, то используется лишь самый нижний из них (подробнее о порядке в стопке см. главу 5).

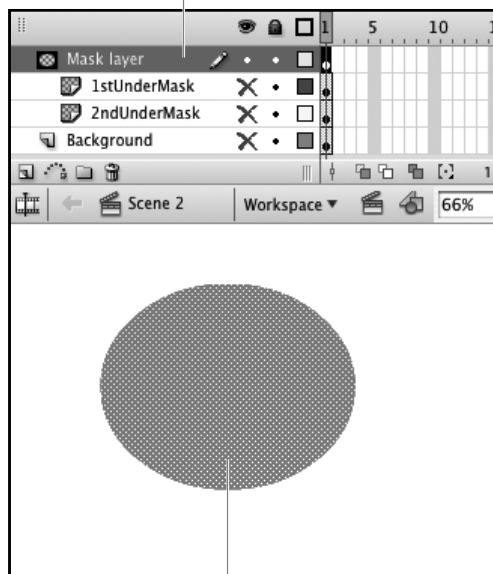
**С**

Если вы непременно хотите включить в маску несколько форм, выберите их и преобразуйте в единый объект-рисунок с помощью команды **Modify** ⇒ **Combine Objects** ⇒ **Union**.

**С**

Слой-маску и замаскированные слои можно настроить также в диалоговом окне **Layer Properties** (см. раздел «Управление слоями и папками» выше).

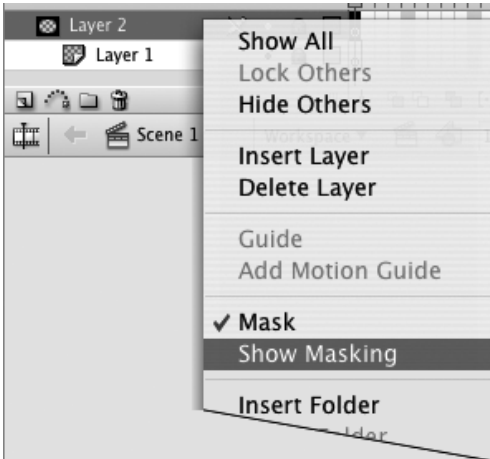
Когда слой определяется как маска...



... эта комбинированная форма становится маской

**Рис. 6.30.** Содержимое слоя, выступающего в роли маски, мало чем отличается от любого другого содержимого. Для создания «дырок» в маске необходимо использовать формы-заливки; все такие формы должны находиться на одном подуровне слоя-маски. Другими словами, следует либо использовать только комбинированные формы, либо объединить все формы в один объект-рисунок.





**Рис. 6.31.** Команда Show Masking в контекстном меню слоя блокирует все слои, связанные с выбранным слоем-маской.

### Чтобы увидеть, как действует маска:

- Заблокируйте слой-маску и все связанные с ним слои.

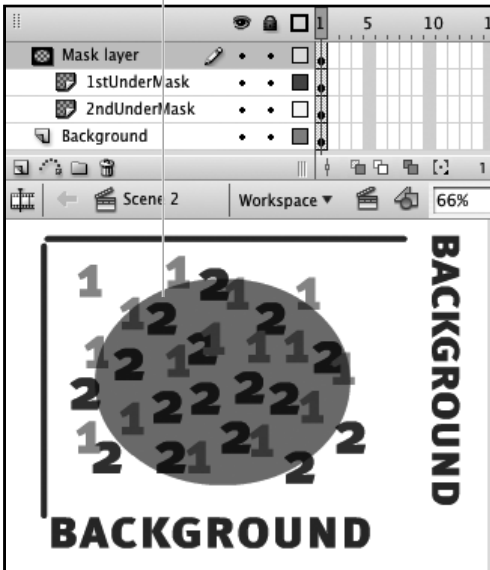
или

1. Щелкните кнопкой мыши с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows) по слою-маске или замаскированному слою.
2. Из контекстного меню выберите пункт **Show Masking** (Показать маскирование) (рис.6.31).

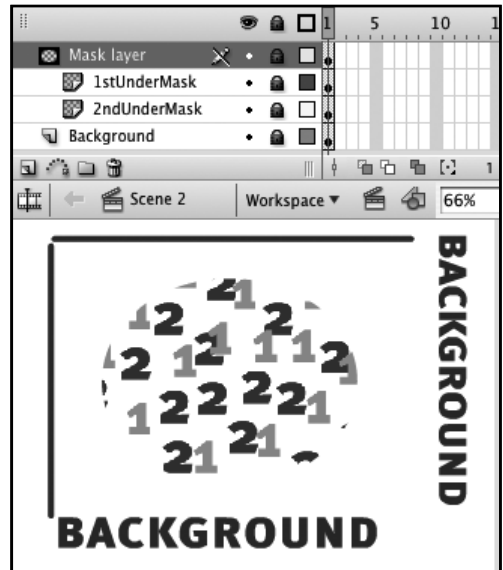
Flash автоматически заблокирует слой-маску и все связанные с ним.

В режиме редактирования документа вы должны заблокировать слой-маску и находящиеся под ним замаскированные слои, чтобы увидеть маску в действии (рис. 6.32). Без блокирования слоев эффект маски можно наблюдать в одном из тестовых режимов Flash (см. главу 8).

Прозрачная заливка позволит увидеть, что будет проглядывать сквозь маску



Маскирование не включено



Маскирование включено

**Рис. 6.32.** Определив слой-маску и замаскированные слои, вы должны заблокировать их, если хотите наблюдать эффект маски в режиме редактирования документа.

**Чтобы изменить маску:**

1. На временной шкале выберите слой-маску.
2. Убедитесь, что этот слой является видимым и разблокированным.
3. Создайте или измените заливки любым из способов, описанных в предыдущих главах.

**C**

Чтобы разорвать связь между маской и связанными с ней слоями, переопределите тип замаскированного слоя в окне Layer Properties.

**C**

Если удалить слой-маску, то Flash сделает все слои, которые были с ней связаны, обычными.

**C**

Не забывайте, что для маскирования требуется процессорное время. Наличие слишком большого числа масок может понизить частоту кадров в готовом ролике. Кроме того, замаскированные области публикуются и занимают место в файле.

**О масках**

Слой-маску можно уподобить конверту с окошком (в таких вы получаете счета). В конверте может быть целая пачка испещренных цифрами документов, но снаружи он выглядит как белый лист бумаги с прорезью, в которой видны ваше имя и адрес. Слой-маска — это и есть окошко, а замаскированные слои — находящиеся в конверте документы.

Во Flash окошко в конверте создается путем рисования в слое-маске. (В главе 11 вы узнаете, что это окошко можно анимировать для создания специальных эффектов.) Форма-заливка в слое-маске становится окошком в готовом ролике. Через это окошко видно то, что находится в связанных (замаскированных) слоях, составляющих содержимое конверта. Внутри конверта может быть несколько слоев, которые ведут себя как обычные слои Flash.

Но тут возникают некоторые тонкости. Любая область конверта (слоя-маски), оставленная чистой, скрывает расположенные под ней области замаскированных слоев. Но части *не связанных* слоев, которые не находятся в конверте, остаются видимыми.

# РАБОТА С СИМВОЛАМИ

---

В предыдущих главах вы научились создавать и редактировать статические графические изображения. Но в конце-то концов хотелось бы эти изображения анимировать и использовать их снова и снова. Зачастую один и тот же элемент появляется несколько раз в одном, а иногда даже в разных роликах. Сохранить графический элемент для повторного использования можно в библиотеке. Но для этого нужно сначала преобразовать его в *символ*.

У любого документа Flash есть своя библиотека, в которой хранятся созданные символы и другие *активы* — повторно используемые элементы, импортируемые в ролик (см. врезку «Терминология, связанная с библиотеками»).

В этой главе вы научитесь работать с библиотеками и создавать символы, являющиеся статическими изображениями. В последующих главах мы обратимся к созданию анимированных символов и кнопок (см. главы 11 и 12), к использованию растровой графики (глава 14), добавлению звука (глава 15) и видео (глава 16).

## Терминология, связанная с библиотеками

Любой элемент, хранящийся в библиотеке Flash, называется *активом* (asset). Конкретнее, графические изображения, созданные с помощью инструментов рисования и сохраненные в библиотеке, называются *символами*; хранящиеся в библиотеке шрифты — *шрифтовыми символами*; а импортированные звуки, видеоклипы и растровые изображения (которые только в библиотеке и могут храниться) так и называются: *звуки*, *видеоклипы* и *растровые изображения*. Копия библиотечного актива, используемого в ролике, называется *экземпляром* этого актива.

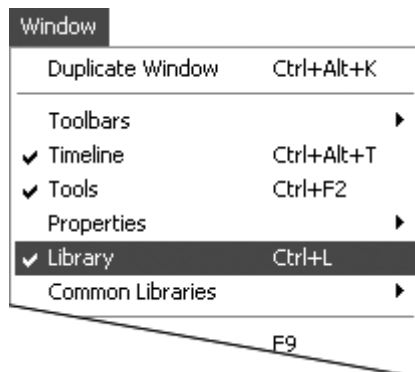
## Знакомство с библиотечной панелью

Панель Library предлагает несколько способов просмотра содержимого библиотеки и позволяет создать набор папок для организации символов, звуков, видеоклипов и растровых изображений. На этой панели отображается информация о времени последней модификации актива и о том, сколько раз он используется в ролике. Также на библиотечной панели есть кнопки и меню для быстрого выполнения операций с символами: создания новых папок, переименования и удаления элементов.

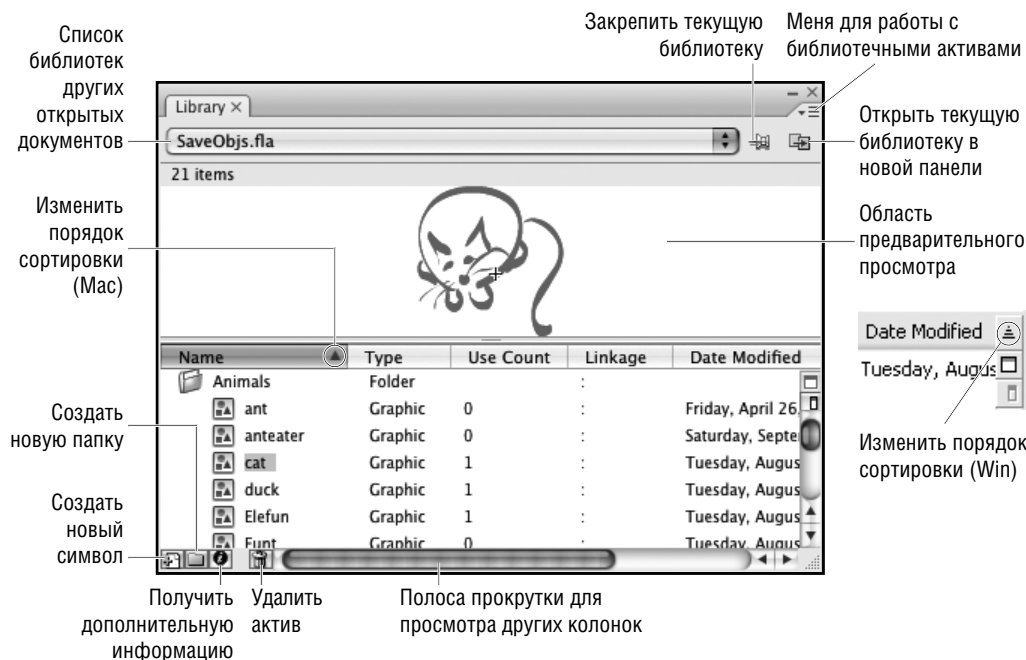
### Чтобы открыть библиотеку текущего ролика:

- Выберите пункт меню **Window** ⇒ **Library** (Окно ⇒ Библиотека), нажмите клавишу **F11** или комбинацию клавиш **(⌘)-L** (Mac) либо **Ctrl-L** (Windows) (рис. 7.1).

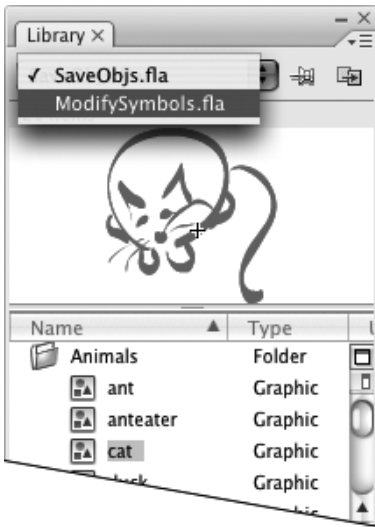
Библиотечная панель появляется на рабочем столе (рис. 7.2). На ней представлены библиотеки всех документов Flash, открытых в текущий момент.



**Рис. 7.1.** Чтобы открыть библиотечную панель активного документа Flash, выберите из меню пункт **Window** ⇒ **Library**.



**Рис. 7.2.** На библиотечной панели перечислены активы, связанные с текущим документом. Список отсортирован по выбранной колонке; чтобы изменить порядок сортировки на противоположный, нажмите кнопку **Sort Order**.



**Рис. 7.3.** На библиотечной панели присутствует список библиотек всех открытых документов. Чтобы просмотреть другую библиотеку, не меняя активный документ, выберите ее из списка.

### Чтобы просмотреть библиотеку другого открытого документа:

1. Откройте два или более документов Flash.
2. Открыв библиотечную панель, выполните одно из следующих действий:
  - из списка открытых документов на библиотечной панели выберите нужный вам неактивный документ (рис. 7.3);

или

- выберите открытый документ, чья библиотека вас интересует.

Если документ отображается на вкладке, щелкните по этой вкладке; если в отдельном окне — по этому окну. В любом случае документ станет активным.

На библиотечной панели появится список активов для выбранного документа.

## С

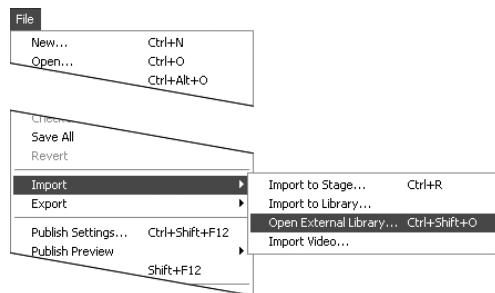
В каждый момент времени на библиотечной панели отображаются активы только одного документа. Если вы хотите видеть библиотеки сразу для нескольких документов, откройте несколько библиотечных панелей. Для этого на открытой библиотечной панели нажмите кнопку **New Library Panel** (Новая панель библиотеки) (значок в виде двух документов со стрелкой, расположенный под меню операций). Flash откроет в новом окне панели текущую библиотеку, то есть в данный момент на двух панелях показывааются активы одного и того же документа. Теперь на любой из панелей откройте другую библиотеку.

## С

Количество меню, из которых можно открыть ту или иную библиотеку, столь велико, что поначалу приводит в трепет. Поэтому приведем краткую сводку. Чтобы открыть библиотечную панель для уже открытого активного документа, выполните команду **Window ⇒ Library**. Чтобы открыть библиотеку уже открытого, но не активного документа, выберите его в меню открытых документов на библиотечной панели. Чтобы открыть библиотеку закрытого документа, выполните команду **File ⇒ Import ⇒ Open External Library**. Чтобы открыть одну из общих библиотек, выполните команду **Windows ⇒ Common Libraries** (см. врезку "Что такое внешние библиотеки?")

### Чтобы открыть библиотеку закрытого документа:

1. Выполните команду **File ⇒ Import ⇒ Open External Library** (Файл ⇒ Импорт ⇒ Открыть внешнюю библиотеку) или нажмите комбинацию клавиш **Shift-⌘-O** (Mac) либо **Ctrl-Shift-O** (Windows) (рис. 7.4).
2. Найдите файл, чью библиотеку хотите открыть, выберите его и нажмите кнопку **Open** (Открыть). На рабочем столе появится библиотечная панель, и представленные на ней символы можно использовать в других роликах.

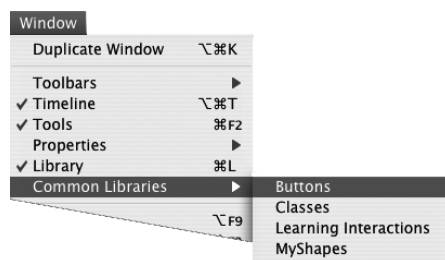


**Рис. 7.4.** Для получения доступа к символам из библиотеки закрытого документа выполните команду **File ⇒ Import ⇒ Open External Library**.

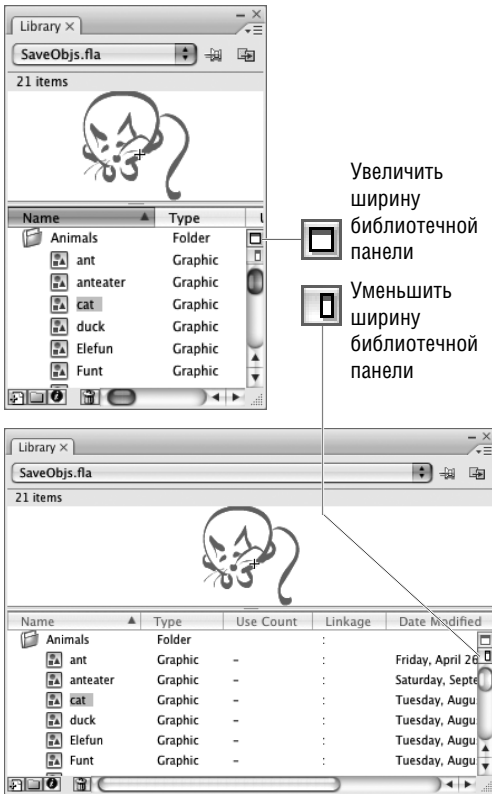
### Что такое общие библиотеки?

Из меню можно получить доступ к набору *общих библиотек*, так сказать библиотеке библиотек. В дистрибутив Flash входят три общих библиотеки, но вы можете добавить и свои собственные. Библиотеки, представленные в меню **Common Libraries**, — это файлы, которые находятся в подпапке Libraries папки Configuration уровня приложения (см. врезку «Папка Configuration» в главе 1). Владелец административных привилегий может добавить в эту папку FLA-файлы. Если таких привилегий у вас нет, то придется создать подпапку с именем Libraries в папке Configuration уровня пользователя. Любые файлы, помещенные в папку Libraries, появятся в меню **Common Libraries** после перезагрузки приложения (рис. 7.5). При выборе пункта этого меню открывается не сам файл, а библиотека.

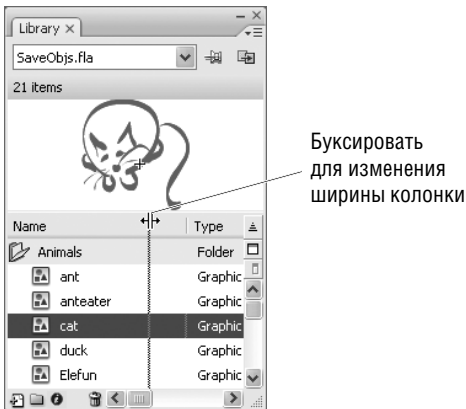
Можно создать общую библиотеку, в которой будут храниться все символы, звуки, видеоклипы и растровые изображения для рабочего проекта, доступного из главного меню. По мере создания или импорта активов помещайте их копии в специальный файл с осмысленным названием, например MyShapes. Сделайте этот файл одной из своих общих библиотек. При выборе пункта MyShapes из меню **Window ⇒ Common Libraries** Flash откроет библиотеку, содержащую все активы, необходимые вашему проекту.



**Рис. 7.5.** С помощью меню **Common Libraries** можно быстро добраться до библиотек Flash-документов, хранящихся в папке Libraries, которая доступна всем пользователям.



**Рис. 7.6.** Чтобы увеличить ширину библиотечной панели, нажмите кнопку Wide State, чтобы уменьшить — кнопку Narrow State.



**Рис. 7.7.** Чтобы изменить ширину колонки списка, буксируйте разделитель колонок.

### Чтобы изменить ширину библиотечной панели:

- Открыв библиотечную панель, выполните одно из следующих действий:
- чтобы перейти от узкого представления к широкому, нажмите кнопку **Wide State** (Широкая область просмотра библиотеки) (рис. 7.6). Flash увеличит ширину панели вдвое по сравнению со стандартной;
- чтобы перейти от широкого представления к узкому, нажмите кнопку **Narrow State** (Узкая область просмотра библиотеки). Flash восстановит стандартную ширину панели.

### Чтобы изменить ширину колонок:

1. Находясь на библиотечной панели, укажите курсором на разделитель колонок. Курсор принимает форму двусторонней стрелки для перемещения разделителя.
2. Буксируйте разделитель мышью (рис. 7.7).

**C**

Flash запоминает, сколько раз использовался экземпляр символа, но в колонке Use Count (Счетчик использования) значение не обновляется автоматически. Чтобы установить режим автоматического обновления, выберите из меню библиотечной панели пункт **Keep Use Counts Updated** (Обновлять счетчики использования). (В таком режиме работа Flash может замедлиться.) Чтобы самостоятельно обновить счетчики, когда возникнет такая необходимость, выберите из меню команду **Update Use Counts Now** (Немедленно обновить счетчики использования).

**C**

Если вы сомневаетесь, стоит ли делать графический элемент символом, лучше сделайте. Позже всегда можно будет расщепить его на исходные формы. При публикации ролика неиспользуемые символы не экспортируются (если только вы явно об этом не попросите). Следовательно, неиспользуемые символы не увеличивают размер SWF-файла. Можете убедиться в этом, запросив отчет о размере при публикации ролика (см. главу 17).

## Иерархия библиотек

Flash позволяет создавать подпапки для иерархической организации библиотечных активов.

### Чтоб создать папку в библиотеке:

1. Откройте библиотечную панель.
  2. Для выбора места создания выполните одно из следующих действий:
    - чтобы создать корневую папку, выберите какой-нибудь элемент на верхнем уровне;
    - чтобы добавить подпапку, выберите элемент внутри той папки, где она должна находиться (не выбирайте саму родительскую папку).
  3. Чтобы создать новую папку, выполните одно из следующих действий:
    - из меню панели в правом верхнем углу окна выберите пункт **New Folder** (Создать папку) (рис. 7.8);
    - нажмите кнопку **New Folder** (Создать папку) в нижней части окна (рис. 7.9).
- Flash создаст новую папку, выберет ее и поместит курсор в текстовое поле.
4. Введите имя папки.
  5. Нажмите **Enter**.

C

Чтобы отсортировать элементы на библиотечной панели, щелкните по заголовку соответствующей колонки. Например, для сортировки по имени щелкните по заголовку колонки **Name** (Имя).

C

Чтобы изменить направление сортировки, нажмите кнопку **Sort**. В Windows эта кнопка (полосатый треугольник) находится справа от заголовков всех колонок; в Mac она выглядит как треугольник справа от заголовка выбранной колонки (рис. 7.2).

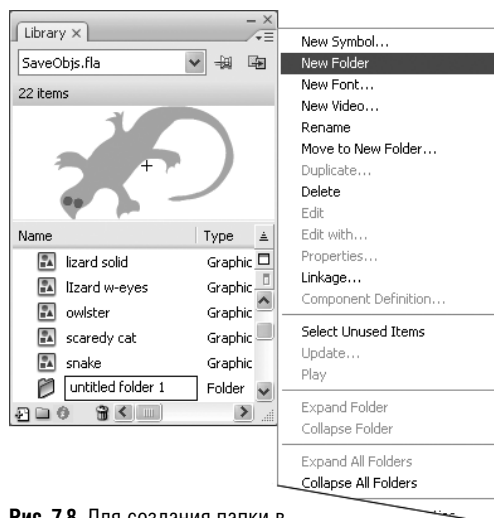


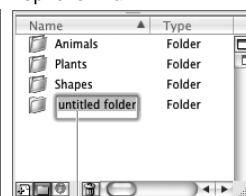
Рис. 7.8. Для создания папки в библиотеке выполните команду **New Folder** в меню библиотечной панели.

Выбрать элемент на верхнем уровне...



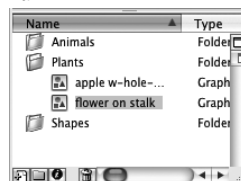
Кнопка **New Folder**

для добавления новой корневой папки



Ввести имя папки

Выбрать элемент внутри папки...



Для добавления подпапки

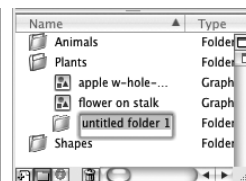
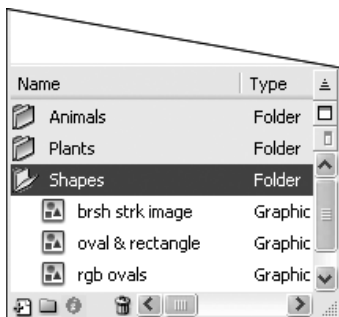
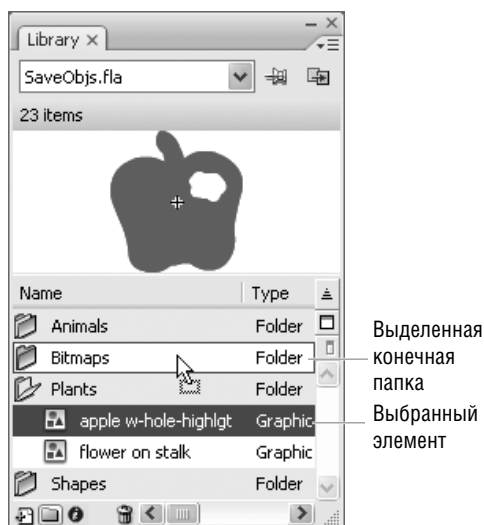


Рис. 7.9. Для создания новых папок и подпапок нажмите кнопку **New Folder**.





**Рис. 7.10.** Откройте папки, чтобы увидеть их содержимое на библиотечной панели.



**Рис. 7.11.** Можно перетаскивать элементы из одной папки в другую. Когда конечная папка готова к бросанию элемента, она подсвечивается.

### Чтобы выполнить операции над папками:

1. На библиотечной панели выберите закрытую папку.
2. Чтобы открыть закрытую папку, выполните одно из следующих действий:
  - дважды щелкните по значку папки;
  - из меню библиотечной панели выберите пункт **Expand Folder** (Развернуть папку).
 На панели появляется содержимое папки (рис. 7.10).
3. Чтобы закрыть открытую папку, выполните одно из следующих действий:
  - дважды щелкните по значку папки;
  - из меню библиотечной панели выберите пункт **Collapse Folder** (Свернуть папку).

**С** Чтобы раскрыть сразу все папки, выберите из меню библиотечной панели пункт **Expand All Folders** (Развернуть все папки).

### Чтобы переместить элементы из одной папки в другую:

1. Открыв библиотечную панель, выберите элемент, который собираетесь переместить.
  2. Перетащите выбранный элемент на значок конечной папки.
- Flash выделит конечную папку (рис. 7.11).
3. Отпустите кнопку мыши.

**С** Чтобы быстро переместить элемент в новую папку, выберите из меню библиотечной панели пункт **Move to New Folder** (Переместить в новую папку). Открывается диалоговое окно, в котором следует ввести имя новой папки. Введите имя и нажмите **Enter**. Flash создаст папку на том же уровне, на котором находится выбранный элемент, и переместит в нее этот элемент.

## Преобразование графических элементов в символы

Не вся графика во Flash-роликах представляет собой символы; чтобы сделать из созданного элемента символ, нужно предпринять определенные шаги. Можно преобразовать в символы уже имеющиеся элементы или создать символ с нуля в редакторе символов. Символы хранятся в библиотеке того документа, в котором были созданы. Можно скопировать символ из одного документа в другой или из библиотеки в документ; в этом случае в библиотеке каждого документа будет храниться своя копия символа.

Стандартная библиотека документа Flash содержит все символы, используемые в этом документе.

В следующем упражнении создаются статические графические символы. Но можно также преобразовать графику в символы, представляющие собой анимацию (глава 11) или кнопки (глава 12).

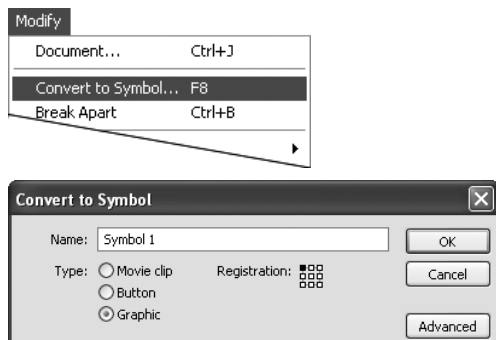
### Зачем нужны символы?

Для уменьшения размера файла во Flash используется векторная графика. Каждая векторная форма представляет собой набор команд для визуализации. Уже по этой причине векторные формы хранятся эффективно. Символы позволяют повторно использовать элементы, чтобы не дублировать векторные формы.

Символ можно уподобить базовому кулинарному рецепту. Представьте себе ресторан, в котором подаются три вида супа: куриная лапша, куриный суп-пюре с рисом и куриный суп с овощами. При этом за каждый котел отвечает отдельный повар. Шеф-повар мог бы повторить каждому повару все шаги, необходимые для приготовления куриного бульона, но это заняло бы слишком много времени. Если в ресторане есть базовый рецепт бульона, то каждый повар мог бы воспользоваться им, а затем выполнить дополнительные шаги, отличающие одно блюдо от другого: добавить лапшу, рис и сливки либо картошку, морковку и горох.

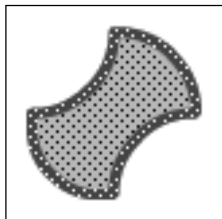
Вот так же работают и символы во Flash. В библиотеке хранится полный рецепт. А каждый экземпляр символа на арене содержит лишь команды, которые говорят, с какого рецепта начать и как его изменить, например: взять рецепт красного треугольника, но вдвое увеличить его, заменить красный цвет синим и повернуть на 45 градусов по часовой стрелке. Поскольку символ может содержать другие символы, то разбиение графических элементов на мельчайшие части окупается; каждую часть следует сделать символом, а затем объединять их в более крупные символы или графику.

Каждый экземпляр символа (каждое место, где один и тот же символ встречается в документе) хранит ссылку на главный символ. Вместо того чтобы включать полный набор команд для каждого экземпляра, Flash оставляет лишь указание о том, что следует повторить одни и те же команды, возможно, с некоторыми модификациями. При таком способе повторного использования элементов размер опубликованного файла оказывается небольшим. Кроме того, если вы решите изменить внешний вид какого-то символа (например, использовать в логотипе круг, а не овал), то достаточно будет обновить лишь главный графический элемент, а не повторять одно и то же изменение снова и снова.

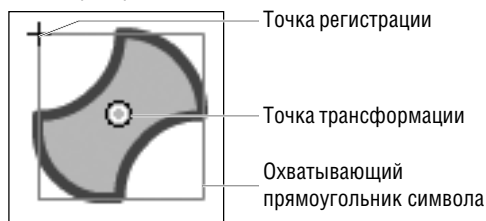


**Рис. 7.12.** Чтобы преобразовать выбранный элемент в символ, выполните команду **Modify** ⇒ **Convert to Symbol**. В диалоговом окне **Convert to Symbol** введите имя символа, задайте его тип и точку регистрации. (Кнопка **Advanced** дает доступ к дополнительным параметрам, описывающим совместное использование и импорт/экспорт.)

Выбранная комбинированная форма



После преобразования в символ



**Рис. 7.13.** Выбранная на арене комбинированная форма выделяется узором из точек. После преобразования ее в символ единственным признаком выделения остается охватывающий прямоугольник. Перекрестье показывает, где находится точка регистрации, а кружочек обозначает точку трансформации символа.

## Чтобы преобразовать графический элемент в символ:

1. Выберите на арене один или несколько графических элементов, которые хотите преобразовать в символ.
2. Выполните команду **Modify** ⇒ **Convert to Symbol** (Изменить ⇒ Преобразовать в символ) или нажмите клавишу **F8**.  
Открывается диалоговое окно **Convert to Symbol** (рис. 7.12). Flash присваивает символу имя по умолчанию, например **Symbol 16**, зависящее от того, сколько символов уже было создано ранее.
3. Если имя по умолчанию вам не нравится, введите другое.
4. В качестве типа символа выберите **Graphic** (Графика).
5. Чтобы задать точку регистрации символа (см. врезку «Точка регистрации и точка трансформации символа»), щелкните по одному из квадратиков в поле **Registration** (Регистрация). По умолчанию Flash регистрирует символ по левому верхнему углу охватывающего прямоугольника. Можете вместо этого выбрать любой другой угол, центр или середину стороны.
6. Нажмите **OK**.

Flash добавляет новый символ в библиотеку. Выбранные на арене графические элементы становятся экземпляром символа. Появляется перекрестье, которое обозначает точку регистрации, и кружочек в центре символа, называемый точкой трансформации (рис. 7.13). Редактировать элемент прямо на арене больше нельзя, необходимо открывать его в одном из режимов редактирования символов.

## С

Графический символ может состоять из одной или нескольких комбинированных форм, объектов-рисунков, примитивов, групп. В состав символа могут даже входить другие символы. Все, что выбрано на арене в момент выполнения команды **Convert to Symbol**, становится частью символа.

## С

Для быстрого преобразования графических элементов в символ, выберите их на арене и перетащите в нижнюю половину библиотечной панели документа. Открывается диалоговое окно **Convert to Symbol**. Назовите и охарактеризуйте символ, как описано в предыдущем упражнении.

## С

Поле «Регистрация» присутствует только в диалоговом окне **Convert to Symbol**. Это означает, что у вас есть лишь один шанс воспользоваться им для установки точки регистрации в одной из предопределенных точек охватывающего прямоугольника. Однако позже можно войти в режим редактирования символа и изменить положение графических элементов относительно его точки регистрации (см. раздел «Редактирование главных символов» ниже).

## О типах символов

Во Flash для каждого символа необходимо задать тип. (В предыдущих версиях это называлось поведением символа.) Существует три типа: графический, кнопка и фрагмент ролика (клип).

Как и следовало ожидать, графические символы — это графические элементы, которые могут быть анимированными. Типы отличаются друг от друга способом взаимодействия с временной шкалой при воспроизведении ролика. Графические символы синхронизированы со шкалой. Статический графический символ занимает один кадр ролика (как и всякий графический элемент). Трехкадровый анимированный графический символ занимает три кадра ролика (см. главу 11).

Кнопки обладают собственной временной шкалой протяженностью в четыре кадра. Экземпляр кнопки занимает один кадр главной временной шкалы ролика, но выводит четыре кадра при взаимодействии с пользователем (см. главу 12).

У клипа есть своя многокадровая временная шкала, которая воспроизводится независимо от главной шкалы ролика (см. главу 11).

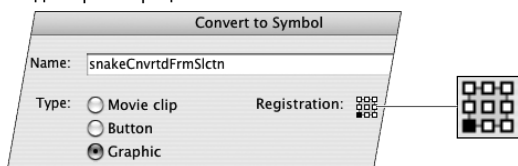
### Точка регистрации и точка трансформации символа

Точка регистрации (отображается в виде перекрестья) — это точка, которую Flash использует для регистрации графического объекта, то есть для того чтобы назначить объекту те же координаты, которые он имел на арене во время разработки. Для объектов-рисунков, примитивов, групп и текстовых полей точкой регистрации всегда является левый верхний угол охватывающего прямоугольника. Для символов положение точки регистрации можно задавать более гибко (рис. 7.14). Когда выбранные на арене элементы преобразуются в символ, в качестве точки регистрации можно указать одну из девяти предопределенных точек (см. раздел «Преобразование графических элементов в символы» выше). Если символ создается с нуля в редакторе, то положение точки регистрации определяется путем размещения элементов вокруг значка перекрестья на арене (см. раздел «Создание символов с нуля» ниже). Положение точки регистрации остается неизменным для всех экземпляров одного и того же символа.

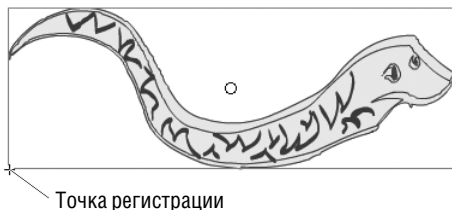
Точка трансформации (представлена кружочком) — это опорная точка, которую Flash использует для трансформации символа. Например, при повороте символа инструментом «Свободное преобразование» точкой трансформации является центр вращения. Точку трансформации можно использовать в операциях привязки. По умолчанию Flash помещает точку трансформации в центр главного символа, но для отдельных его экземпляров положение этой точки можно изменить с помощью инструмента свободного преобразования.

На информационной панели и на вкладке **Properties** в инспекторе свойств отображаются координаты, определяющие положение выбранного экземпляра символа на этапе разработки. На вкладке **Properties** показаны также координаты точки регистрации символа, а на информационной панели вы можете сами указать, какие координаты показывать: точки трансформации или точки регистрации. Для этого служит кнопка **Registration/Transformation Point** (подробнее об использовании информационной панели и вкладки **Properties** в инспекторе свойств см. главу 4).

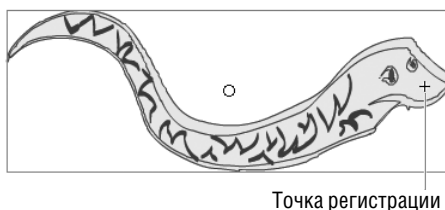
#### Модель регистрации



#### Преобразованный символ



#### Символ, созданный с нуля



**Рис. 7.14.** Когда выбранные графические элементы преобразуются в символ, точку регистрации можно поместить в центр, любой угол или середину любой стороны охватывающего прямоугольника. Тот квадратик в модели регистрации, по которому вы щелкнете, и станет точкой регистрации символа. При создании символа с нуля точку регистрации можно поместить в любое место.

## Создание символов с нуля

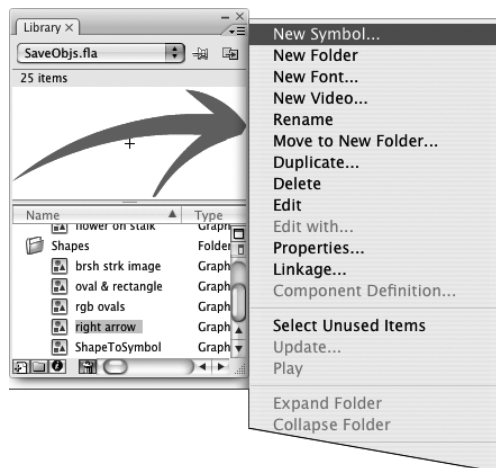
Можно избежать процедуры преобразования, описанной в предыдущем разделе, если создавать символы с нуля в редакторе символов. При этом вам доступны все инструменты, кадры и слои, но Flash с самого начала определяет создаваемый элемент как символ.

### Чтобы создать новый символ:

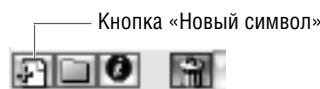
1. Для входа в режим редактирования символов выполните одно из следующих действий:
  - выполните команду **Insert ⇒ New Symbol** (Вставить ⇒ Новый символ) или нажмите комбинацию клавиш **⌘-F8** (Mac) либо **Ctrl-F8** (Windows);
  - в меню библиотечной панели выберите пункт **New Symbol** (рис. 7.15);
  - нажмите кнопку **New Symbol** в левом нижнем углу библиотечной панели (рис. 7.16).

Открывается диалоговое окно Create New Symbol.

2. Введите имя символа.
3. В качестве типа символа укажите Graphic (Графика).



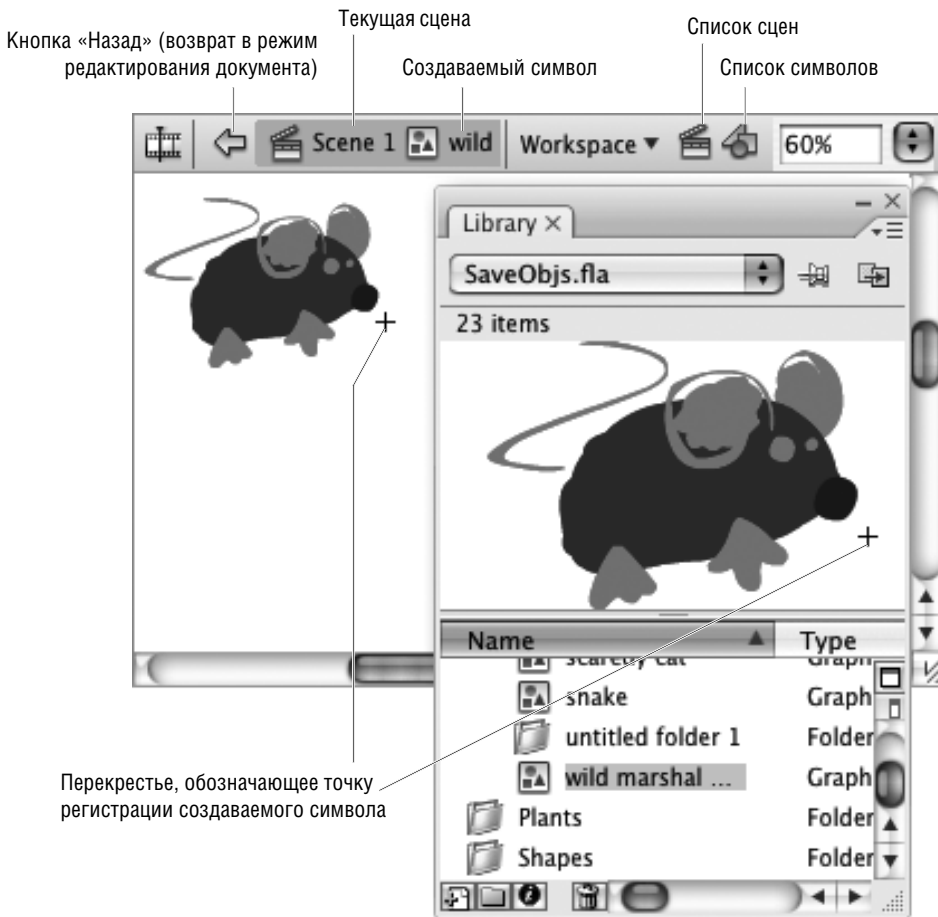
**Рис. 7.15.** Чтобы создать новый символ с нуля, выберите пункт **New Symbol** из меню библиотечной панели.



**Рис. 7.16.** Чтобы создать новый символ с нуля, нажмите кнопку **New Symbol** на библиотечной панели.

**4. Нажмите ОК.**

Flash переходит в режим редактирования символов. На панели редактирования отображается введенное вами имя символа, в центре арены появляется перекрестье, а монтажный стол исчезает (рис. 7.17). Перекрестье обозначает точку регистрации символа.

**5. Создавайте графическое изображение на арене редактора символов точно так же, как в обычной среде разработки.**

**Рис. 7.17.** В режиме редактирования символов на панели редактирования, которая находится над ареной, отображается имя символа, с которым вы работаете.

6. Чтобы вернуться в режим редактирования документов, выполните одно из следующих действий:

- выполните команду **Edit ⇨ Edit Document** (Редактирование ⇨ Редактировать документ). Произойдет возврат к текущей сцене;
- нажмите кнопку **Назад** или щелкните по ссылке **Current Scene** (Текущая сцена) на панели редактирования (рис. 7.18);
- из выпадающего меню **Edit Scene** (Изменить сцену) на панели редактирования выберите сцену. Flash вернет вас на указанную сцену (рис. 7.19).

C

При создании новых символов подумайте о том, как должна функционировать точка регистрации в готовом символе, и размещайте графические элементы относительно перекрестья соответствующим образом. Вы хотите, чтобы данный символ выравнивался по центру? Тогда элементы должны равномерно окружать перекрестье. Или выравнивание должно происходить по какой-то особой точке (скажем, при регистрации птички это может быть кончик клюва)? Тогда так и располагайте элементы.

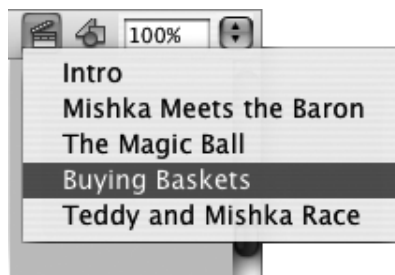
C

При входе в режим редактирования существующего символа перекрестье может оказаться вне окна просмотра. Чтобы поместить его в центр окна, выполните команду **View ⇨ Magnification ⇨ Show Frame** (Просмотр ⇨ Масштаб ⇨ Показать кадр).

Кнопка «Назад»      Ссылка «Текущая сцена»



**Рис. 7.18.** Чтобы вернуться в режим редактирования документов, нажмите кнопку Назад или щелкните по ссылке Current Scene.



**Рис. 7.19.** Выберите из меню Edit Scene на панели редактирования сцену, к которой хотите вернуться.

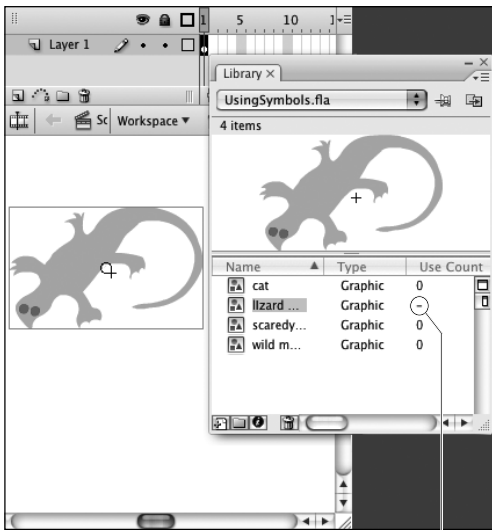
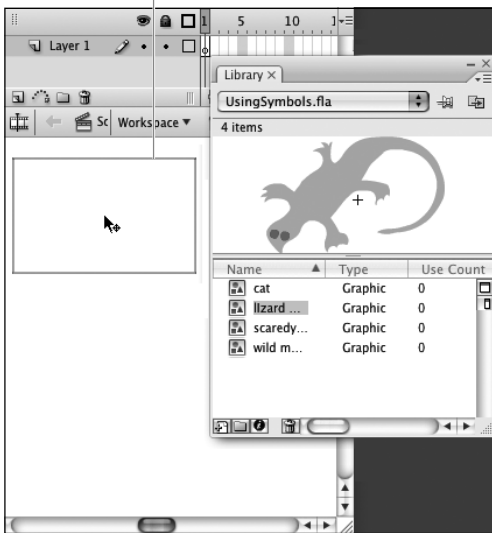
### Где я нахожусь?

При редактировании символов в документе Flash текущее окно переключается в режим редактирования символов. Легко перепутать, что именно вы редактируете — документ или символ. Поэтому запомните следующие визуальные ключи, которые говорят о том, что сейчас вы находитесь в режиме редактирования символов.

В режиме редактирования символов на панели редактирования отображается имя сцены и редактируемого символа, а также активируется кнопка **Назад**. Монтажного стола не видно. Кроме того, на арене появляется значок перекрестья, обозначающий точку регистрации символа. Если вы попали в режим редактирования символов в результате выполнения команды **Edit in Place**, то все элементы на арене, кроме редактируемых, затенены. Если не считать этих особенностей, временная шкала, арена и все инструменты выглядят и работают точно так же, как в режиме редактирования документов.



Предварительный показ места символа на арене



Счетчик использования следует обновить

**Рис. 7.20.** В результате перетаскивания символа с библиотечной панели на арену (верхний рисунок) Flash размещает символ на арене, выбирает его и обновляет внутренний счетчик использования символа (нижний рисунок). Если не задан режим автоматического обновления счетчиков, то об изменении говорит прочерк в колонке Use Count. Чтобы увидеть актуальное значение, выполните команду Update Use Counts Now из меню библиотечной панели.

## Использование экземпляров символа

Экземпляр символа — это указатель на полное описание символа. Символы способствуют уменьшению размера файла. Если преобразовать присутствующие на арене графические элементы в символ, то на арене останется всего один экземпляр символа. Чтобы повторно воспользоваться этим символом или символом, созданным в режиме редактирования, необходимо взять его копию из библиотеки и поместить на арену.

### Чтобы включить экземпляр символа в ролик:

1. На временной шкале выберите слой и опорный кадр, в котором должен появиться графический символ.
- Flash может помещать символы только в опорные кадры. Если вы укажете промежуточный кадр, то символ будет помещен в предшествующий ему опорный кадр (подробнее об опорных кадрах см. главу 8).
2. Откройте библиотеку, содержащую символ.
3. Найдите на библиотечной панели нужный символ и щелкните по нему мышью. Flash подсветит выбранный символ и покажет его в области предварительного просмотра.
4. Наведите курсор на область предварительного просмотра.
5. Перетащите копию символа на арену. В процессе буксировки Flash с помощью прямоугольного контура показывает, где окажется символ (рис. 7.20).
6. Отпустите кнопку мыши. Flash помещает символ на арену и выбирает его.

**С**

Чтобы быстро поместить экземпляр символа, перетащите его имя прямо с библиотечной панели на арену, не используя область предварительного просмотра.

## Модификация экземпляров символа

Внешний вид отдельных экземпляров символа можно изменить, не меняя сам главный символ. Размер и положение экземпляра можно изменять (в частности, масштабировать и поворачивать), как и для любого другого элемента, с помощью подходящих инструментов, панелей информации и преобразований, а также вкладки **Properties** в инспекторе свойств (см. главу 4).

Можно также изменить цвет и прозрачность экземпляра символа, но это делается не так, как для комбинированных форм, объектов-рисунков и примитивов. Необходимо воспользоваться списком **Color** (Цвет) на вкладке **Properties** в инспекторе свойств.

### Чтобы изменить цвет экземпляра:

1. Выберите экземпляр символа на арене.
2. Перейдите на вкладку **Properties** в инспекторе свойств. На ней отображаются параметры выбранного символа.
3. Выберите из списка **Color** то цветовое свойство, которое хотите изменить, а именно: **Brightness** (Яркость) (долю черного или белого в цвете), **Tint** (Тон), **Alpha** (прозрачность) и **Advanced** (Дополнительно) (одновременное изменение коэффициента прозрачности и RGB-компонентов).
4. Введите новые значения параметров, как описано в последующих упражнениях.
5. Нажмите **Enter**, чтобы подтвердить изменения. Flash применит новые параметры к символу на арене.

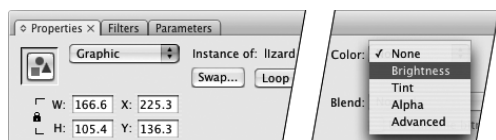
### Рекомендации по трансформации символов

Размеры и ориентацию экземпляров символа можно изменять так же, как для графических объектов (см. главу 4). Можно ввести точные значения на вкладке **Properties** в инспекторе свойств или на панелях информации либо преобразований. Или модифицировать экземпляр символа интерактивно с помощью инструмента «Свободное преобразование». Ниже приводятся некоторые советы по работе с этим инструментом.

Если для масштабирования экземпляра символа применяется инструмент «Свободное преобразование», то он масштабируется относительно точки трансформации экземпляра (по умолчанию это центр символа). Чтобы масштабирование производилось относительно угла охватывающего прямоугольника, противоположного тому, за который вы тянете, удерживайте клавишу **Option** (Mac) или **Alt** (Windows).

Для конкретных экземпляров символа положение точки трансформации можно изменить. Выбрав инструмент «Свободное преобразование», щелкните по белому кружочку в центре экземпляра символа. Перетащите кружок в другое место.

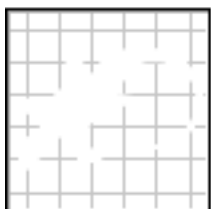
Чтобы совместить точку трансформации с точкой регистрации, выберите экземпляр инструментом «Свободное преобразование» и дважды щелкните по точке трансформации.



**Рис. 7.21.** Параметр Brightness позволяет изменить яркость экземпляра символа. Положительное значение делает символ светлее, отрицательное — темнее.



Задана яркость -100 процентов



Задана яркость 100 процентов

**Рис. 7.22.** Если задать предельные значения яркости, то символ станет черным или белым.



**Рис. 7.23.** Параметры, связанные с оттенком, позволяют изменить цвет экземпляра символа.

### Чтобы изменить яркость экземпляра:

1. На шаге 3 предыдущего упражнения выберите из списка **Color** пункт Brightness. Появится поле для ввода процентной величины яркости (рис. 7.21).
2. Введите значение. Если яркость равна -100, то символ становится черным, 0 оставляет исходную яркость, 100 делает символ белым (рис. 7.22).

### Чтобы изменить цвет экземпляра:

1. На шаге 3 первого упражнения из этого раздела выберите из списка **Color** пункт Tint. Появятся поля для задания оттенка (рис. 7.23).
2. Чтобы задать новый цвет, выполните одно из следующих действий:
  - в поля Red, Green и Blue введите новые значения компонентов цвета;
  - щелкните по элементу управления цветом и выберите цвет из появляющегося окна образцов.
3. Введите процент в поле Tint Amount (Интенсивность оттенка).

Это поле говорит о том, какую долю нового цвета подмешать в существующие цвета. Если задать значение 100, то все линии и заливки в символе перекрасятся в новый цвет. При меньших значениях происходит частичное смешение цветов; можно сказать, что на символ наложен цветной фильтр.

**Чтобы изменить прозрачность экземпляра:**

1. На шаге 3 первого упражнения из этого раздела выберите из списка **Color** пункт **Alpha**. Появится поле для ввода процентного коэффициента прозрачности (рис. 7.24).
2. Введите коэффициент. Значение 0 соответствует полностью прозрачному символу, а значение 100 — полностью непрозрачному.

**C**

Можете не нажимать **Enter** для подтверждения правильности ввода, а просто щелкнуть мышью где-нибудь на вкладке **Properties** или на арене.

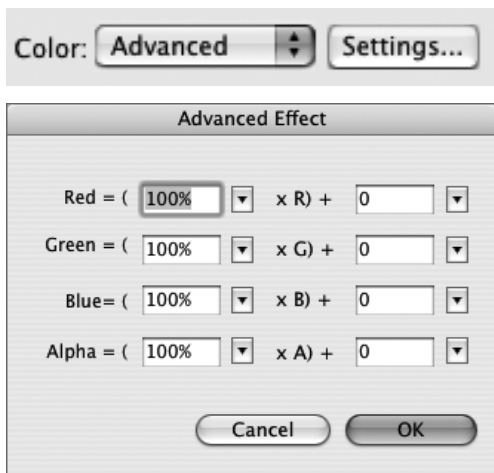


**Рис. 7.24.** Чтобы изменить прозрачность экземпляра символа, задайте цветовой параметр Alpha на вкладке Properties в инспекторе свойств.

### Дополнительные цветовые параметры

Если выбрать из списка **Color** пункт **Advanced**, то можно будет одновременно задать RGB-компоненты и прозрачность цвета символа. Ползунки, которые появляются при щелчке по стрелкам в левой колонке, управляют тем, какая часть RGB-компонентов и коэффициента прозрачности, определенных в главном символе, должна сохраниться в данном экземпляре. Ползунки, соответствующие стрелкам в правой колонке, позволяют увеличить или уменьшить значения исходных компонентов цвета и прозрачности на указанную величину.

Представьте себе символ, состоящий из трех овалов чисто красного, чисто зеленого и чисто синего цвета. Пусть еще задана прозрачность 50%. Изменение красного ползунка в левой колонке (текущая доля красного цвета) коснется только красного овала. Доля красного в зеленом и синем овале равна 0, поэтому ее удвоение ничего не даст. Если сдвинуть вверх ползунок в правой колонке, то красный цвет добавится ко всем овалам. По мере изменения значения они начнут менять цвет. (Если двигать правый ползунок вниз, уменьшая тем самым долю красного, то синий и зеленый овалы не изменятся, так как красного в них не было изначально.)



**Рис. 7.25.** Выберите в списке цветов пункт **Advanced** и нажмите кнопку **Settings** на вкладке свойств (верхний рисунок), чтобы открыть диалоговое окно **Advanced Effect** (нижний рисунок). Введите новое значение цвета и коэффициента прозрачности символа.

### Чтобы одновременно изменить оттенок и прозрачность:

1. На шаге 3 первого упражнения из этого раздела выберите из списка **Color** пункт **Advanced**. Справа от списка находится кнопка **Settings** (Параметры).
2. Нажмите кнопку **Settings**. Откроется диалоговое окно **Advanced Effect** (рис. 7.25). В нем имеются ползунки и текстовые поля для изменения RGB-компонентов и коэффициента прозрачности цвета.
3. Изменяя значения, можете настроить цвет и прозрачность данного экземпляра символа (см. врезку «Дополнительные цветовые параметры»).
4. Чтобы изменения вступили в силу, нажмите **OK**.



Для интерактивного просмотра новых цветов откройте ползунков справа от нужного текстового поля (например, поля **Tint Amount**). При буксировке ползунка цвет символа на арене будет изменяться.

### Эффекты временной шкалы: сомнительный источник символов

Команды из меню **Insert** ⇒ **Timeline Effects** (Вставить ⇒ Эффекты временной шкалы) — это попытка помочь неопытному пользователю Flash в создании нескольких распространенных специальных эффектов и простых анимаций. В результате выполнения этих команд Flash создает символы за вас (иногда статические, описанные в этой главе, а иногда — динамические, о которых речь пойдет в главе 11). Но применять изложенные в этой главе методы для повторного использования или модификации созданных таким образом символов не следует, иначе вы рискуете испортить эффект. Если дважды щелкнуть по готовому символу эффекта на арене, то ничего не произойдет. Если выбрать готовый символ и попробовать применить к нему команду редактирования (например, **Edit** ⇒ **Edit Selected**), то будет выдано предупреждение о том, что, продолжив выполнение, вы потеряете возможность редактировать эффект. Однако при попытке повторно использовать готовый символ или один из вложенных в него символов, необходимых для создания эффекта, никакого предупреждения не будет. Вообще говоря, эффекты временной шкалы — это не лучший способ создания анимации и эффектов во Flash. Поэтому рассматривать их в этой книге мы не будем.

## Замена одного экземпляра символа другим

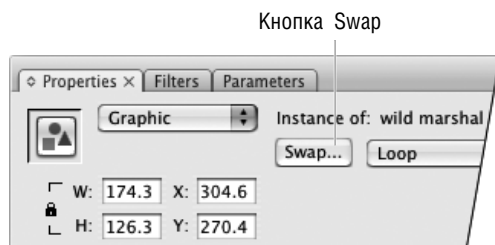
Flash позволяет заменять один экземпляр символа другим, сохраняя все модификации, произведенные над последним. Например, если вы хотите изменить логотип в некоторых, но не во всех местах сайта, то можете создать новый логотип как отдельный символ и подставить его в нужные места. (Чтобы изменить внешний вид каждого экземпляра, лучше отредактировать сам главный символ, как описано в разделе «Редактирование главных символов» ниже). Для замены символов понадобится вкладка **Properties** в инспекторе свойств (рис. 7.26).

### Чтобы заменить символы:

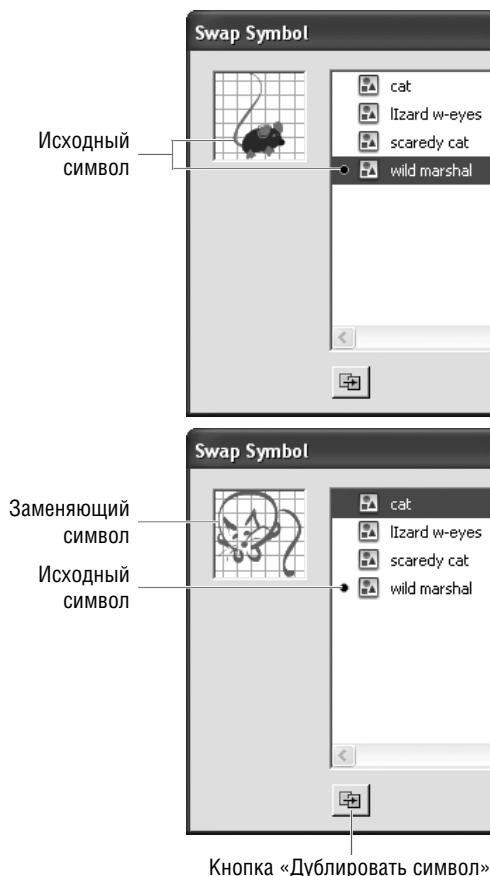
1. Выберите на арене экземпляр символа, который хотите заменить.
2. На вкладке **Properties** в инспекторе свойств нажмите кнопку **Swap** (Замена).

Открывается окно **Swap Symbol**, в котором перечислены все символы из библиотеки текущего документа (рис. 7.27). Имя модифицируемого символа подсвечено. На платформе Windows слева от имени отображается еще и маркер.

3. Выберите из списка заменяющий символ. В Windows исходный символ остается промаркированным. Flash выделяет новый символ и показывает его в области предварительного просмотра.



**Рис. 7.26.** Кнопка Swap на вкладке Properties позволяет заменить выбранный экземпляр символа другим символом из того же документа.



**Рис. 7.27.** Выберите заменяющий символ из списка и нажмите OK, чтобы вставить его вместо текущего.



Немодифицированная  
версия мыши

Немодифицированная  
версия кошки



Мышь после  
масштабирования и  
поворота, но до замены



После замены:  
масштабирование  
и поворот применяются  
к кошке

**Рис. 7.28.** При замене символов все модификации, произведенные над выбранным экземпляром, применяются и к заменяющему символу.

#### 4. Нажмите **ОК**.

Flash помещает новый символ в то место на арене, где находился старый, и применяет к нему те же модификации, которые были произведены над прежним экземпляром (рис. 7.28).

**C**

Чтобы быстро заменить символ, дважды щелкните по новому символу в окне Swap Symbol. Flash произведет замену и закроет окно.

**C**

Кнопка **Duplicate Symbol** (Дублировать символ) в диалоговом окне позволяет скопировать символ, выбранный в списке (рис. 7.27). Если вы собираетесь слегка изменить главную версию заменяющего символа для данного экземпляра, но хотите также сохранить и текущую, нажмите эту кнопку, введите имя символа в появляющемся диалоговом окне и нажмите **ОК**. Убедитесь, что в списке выделена только что созданная копия и нажмите **ОК**. Позже вы сможете изменить главный символ, для которого был создан дубликат.

## Редактирование главных символов

Созданный ранее символ можно изменить в режиме редактирования символов. В отличие от модификации экземпляра символа, которая затрагивает только экземпляр, находящийся на арене, а главный символ, хранящийся в библиотеке, оставляет без изменения, модификации, сделанные в режиме редактирования символов, применяются и к главному символу, и ко всем его экземплярам в ролике. Войти в режим редактирования символов можно несколькими способами.

### Чтобы попасть в режим редактирования символов с арены:

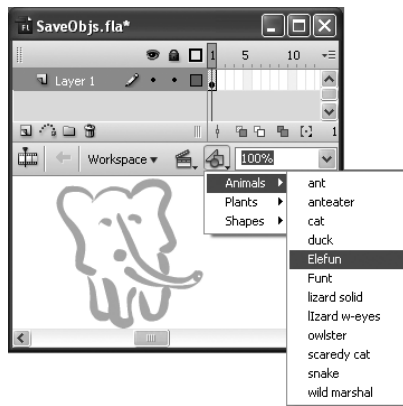
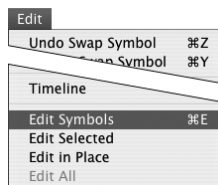
1. Выберите на арене символ, который собираетесь изменить.
2. Чтобы открыть редактор символов, выполните одно из следующих действий:
  - выполните команду **Edit ⇒ Edit Symbols** (Редактирование ⇒ Редактировать символы) или нажмите комбинацию клавиш **⌘-E** (Mac) либо **Ctrl-E** (Windows) (рис. 7.29);
  - выполните команду **Edit ⇒ Edit Selected** (Редактирование ⇒ Редактировать выделение);
  - выберите нужный символ из раскрывающегося списка символов на панели редактирования (рис. 7.30).

Flash открывает редактор символов в текущем окне. Для редактирования символа можно применять все способы, используемые для модификации графики и создания или удаления контента.

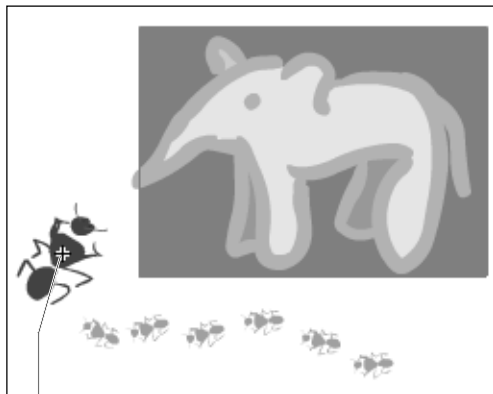
**C**

Команда **Edit in Place** (Редактировать на месте) позволяет изменять главный символ в контексте арены, когда все другие элементы затенены (рис. 7.31). Для этого выберите экземпляр символа и выполните команду **Edit ⇒ Edit in Place**. Любые внесенные изменения отразятся на всех экземплярах данного символа.

**Рис. 7.29.** Команда **Edit ⇒ Edit symbols** переводит вас из режима редактирования документа в режим редактирования символов. Если на арене выбран какой-то символ, то перейти в режим его редактирования позволяет также команда **Edit ⇒ Edit Selected**



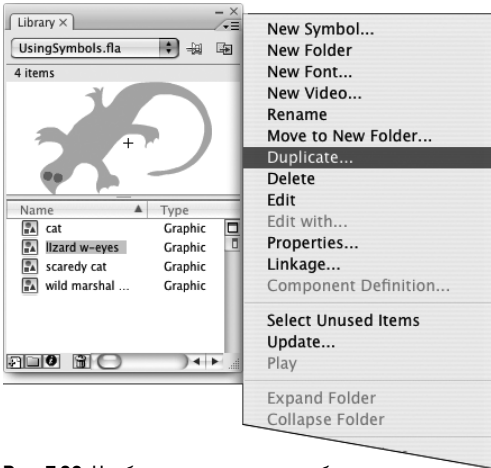
**Рис. 7.30.** Выбор символа из раскрывающегося списка **Edit Symbol** на панели редактирования переводит вас в режим редактирования символов.



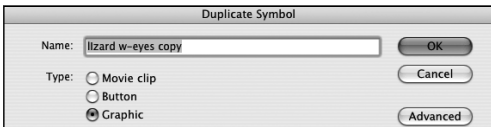
Редактируемый символ

**Рис. 7.31.** Команда **Edit in Place** позволяет видеть символ в контексте других элементов на арене. Экземпляр этого символа отображается в своем цвете, а остальные элементы затенены. Изменения, произведенные над экземпляром в этом режиме, отражаются на главном символе и на всех остальных его экземплярах в ролике.

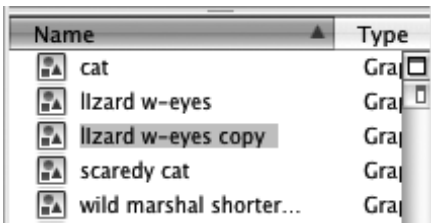




**Рис. 7.32.** Чтобы сделать копию выбранного символа, выберите из меню библиотечной панели команду **Duplicate**.



**Рис. 7.33.** В диалоговом окне **Duplicate Symbol** можно ввести имя символа (по умолчанию ему присваивается имя, состоящее из имени исходного символа и слова *copy*) и назначить ему тип.



**Рис. 7.34.** Flash добавляет дубликат в библиотеку на том же уровне иерархии, что и оригинал.

## Дублирование главного символа

Хотя можно в любой момент модифицировать экземпляры символа на арене, но если вы собираетесь многократно использовать один и тот же вариант символа, то имеет смысл продублировать главный символ, а затем модифицировать дубликат и тем самым создать новый видеоизмененный главный символ.

### Чтобы создать дубликат символа:

1. Выберите на библиотечной панели символ, который хотите продублировать.
2. Из меню панели выберите команду **Duplicate** (Дублировать) (рис. 7.32). Открывается диалоговое окно **Duplicate Symbol**, в котором дубликату присвоено имя по умолчанию (рис. 7.33).
3. Если хотите, можете изменить имя символа.
4. Выберите тип символа, например, **Graphic**.
5. Нажмите **OK**.

Flash добавит новый символ в библиотеку на том же уровне иерархии, что и исходный (рис. 7.34). Дубликат никоим образом не связан с оригиналом. Можно изменять дубликат, не затрагивая оригинал, и наоборот.

## Удаление главных символов

Удалять символы несколько сложнее, чем формы и группы на арене. Удалить один экземпляр символа с арены просто, для этого годятся те способы вырезания и удаления графики, которые мы рассматривали в главе 4. Удалить символ из библиотеки тоже нетрудно, но надо иметь в виду, что экземпляры удаляемых символов еще могут использоваться в ролике.

### Чтобы удалить один символ из библиотеки:

1. Выберите на библиотечной панели символ, который собираетесь удалить.
2. Для удаления символа выполните одно из следующих действий:
  - нажмите кнопку **Delete** (Удалить) в нижней части окна (значок с изображением мусорной корзины) (рис. 7.35);
  - из меню панели выберите команду **Delete** (рис. 7.36).

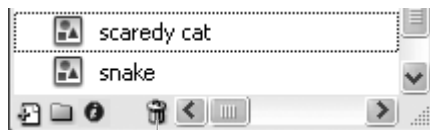
### Чтобы удалить папку символов из библиотеки:

1. Выберите папку, которую хотите удалить.
2. Выполните те же действия, что на шаге 2 предыдущего упражнения.

Flash удаляет папку и все содержащиеся в ней символы.

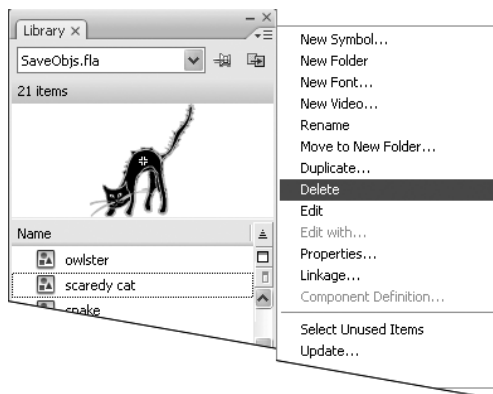
**С**

Прежде чем удалять что-нибудь из библиотеки, обращайте внимание на счетчики использования (рис. 7.37). Вряд ли вы захотите удалять символ, который встречается в ролике, а если это вложенный символ, то не сразу и заметишь. Предыдущие версии Flash в этом случае выдавали предупреждение, но во Flash CS3 этого уже нет.

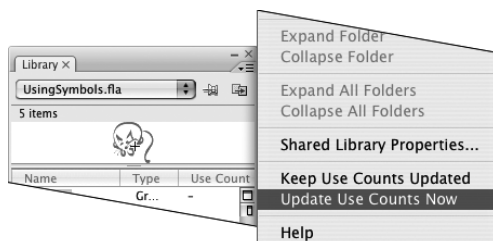


Удалить выбранный элемент

**Рис. 7.35.** Щелкните по значку мусорной корзины, чтобы удалить выбранный элемент из библиотеки.

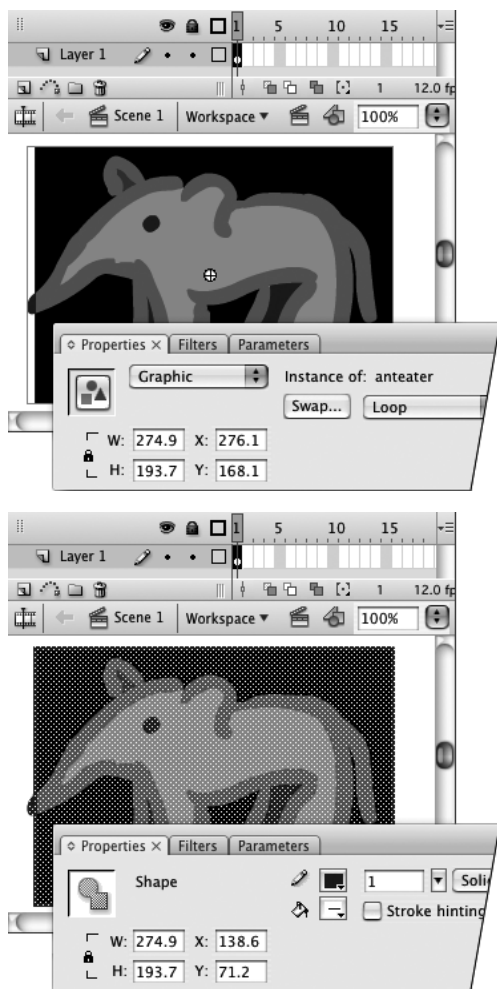


**Рис. 7.36.** Для удаления выбранных символов из активного документа можно также выбрать команду **Delete** из меню библиотечной панели.



Name	Type	Use Count
scaredy ...	Gr...	0
lizard w...	Gr...	2
cat	Gr...	2

**Рис. 7.37.** Не забывайте посмотреть на счетчики использования, прежде чем удалять символ. Для этого предварительно выполните команду **Update Use Counts Now** из меню библиотечной панели, а потом убедитесь, что счетчик использования равен 0.



**Рис. 7.38.** На вкладке Properties в инспекторе свойств видно, что выбранный элемент является экземпляром символа (верхний рисунок). Чтобы разорвать его связь с главным символом, выполните команду **Modify** ⇒ **Break Apart**. Теперь вкладка Properties говорит о том, что выбрано несколько форм, а не экземпляр символа (нижний рисунок).

## Преобразование экземпляров символа в графические элементы

Иногда желательно разорвать связь между помещенным в ролик экземпляром символа и самим главным символом. Например, если вам хочется перерисовать форму только в одном экземпляре. Чтобы преобразовать символ обратно в независимую форму или в набор форм, его необходимо расщепить.

### Чтобы разорвать связь с символом:

1. Выберите на арене экземпляр символа, для которого хотите разорвать связь.
2. Выполните команду **Modify** ⇒ **Break Apart** (Изменить ⇒ Разделить) или нажмите комбинацию клавиш **⌘-B** (Mac) либо **Ctrl-B** (Windows) (рис. 7.38).

Flash разывает связь с символом из библиотеки и оставляет составные элементы символа выбранными. На вкладке **Properties** в инспекторе свойств теперь отображается информация не об экземпляре символа, а о выбранных формах.

Если исходный символ содержал сгруппированные элементы, то они останутся таковыми и после разрыва связи. Наоборот, не сгруппированные не будут включены в группу. Символы, входившие в состав расщепленного символа, остаются экземплярами соответствующих главных символов. Теперь составные элементы можно редактировать, как описывалось в предыдущих главах.

### Отмена операции на уровне объекта

Любой пользователь компьютера знаком с командой отмены (Undo). Комбинации клавиш **⌘-Z** (Mac) или **Ctrl-Z** (Windows) — это универсальная реакция на допущенные ошибки. В программе Flash CS3 есть два вида отмены: на уровне документа и на уровне объекта. (В версии Flash 8 они тоже есть, а в более ранних — нет.)

На уровне документа Flash запоминает все выполненные отменяемые действия. Открыв панель истории (**Window ⇒ Other Panels ⇒ History** (Окно ⇒ Другие панели ⇒ История)), вы увидите список выполненных вами операций. На уровне объекта также запоминаются отменяемые действия, но проводится различие между теми, которые были выполнены при работе над основным документом и при работе над главными символами.

Для отмены на уровне объекта операции над каждым главным символом отслеживаются по отдельности. Всякий раз, как вы создаете главный символ, Flash заводит новый список на панели истории специально для этого символа. На протяжении всего рабочего сеанса (при условии, что вы ничего не изменяли на панели истории самостоятельно) каждая операция редактирования главного символа заносится в историю. Если вы перейдете к редактированию другого символа, на панели истории будут отображаться выполненные над ним операции. После выхода из режима редактирования символов и возврата в режим редактирования документа, на панели истории остаются только операции, выполненные над основным документом.

Отметим, что модификации, произведенные над экземпляром символа на арене, считаются частью истории документа, а не главного символа.

Чтобы сменить тип операции отмены, выберите из меню **Flash** (Mac) или **Edit** (Windows) пункт **Preferences** (Установки). В категории **General** выберите в списке **Undo** (Отменить) тот вариант, который вам нужен. Чтобы отслеживать историю изменения каждого главного символа, установите режим Object-level Undo (Отменить на уровне объекта), а если нужно вести общий список — режим Document-level Undo (Отменить на уровне документа). Лучше устанавливать режим отмены в начале сеанса, иначе при переключении режима Flash очистит текущий список на панели истории.

# ПОКАДРОВАЯ АНИМАЦИЯ

---

8

Покадровая анимация традиционно применялась до наступления эры компьютеров. Даже кино- и видеофильмы с натуральным движением — это тоже разновидность покадровой анимации. Камера передает движение, снимая неподвижные сцены с большой частотой. Принцип анимации состоит в том, чтобы показать элементы на различных фазах движения.

Чтобы имитировать естественное перемещение персонажей в мультфильме, художники-мультипликаторы, работавшие на студиях Уолта Диснея или «Уорнер Бразерс» с 1930-х по 1960-е годы, должны были рисовать сотни картинок, каждая из которых чуть-чуть отличалась от предыдущей. Затем каждое изображение становилось отдельным кадром на пленке.

Персонажи (или их части) и прочая графика рисовались на прозрачных листах целлулоида — *келях*. Затем кели накладывались друг на друга и получалось изображение, составляющее один кадр. Такая техника позволяла экономить время за счет повторного использования тех частей картинки, которые не менялись на протяжении нескольких кадров.

В Adobe Flash CS3 Professional тоже возможна покадровая анимация, когда различное содержимое помещается в разные кадры. Кадры, в которых размещается новое содержимое, во Flash называются *опорными*.

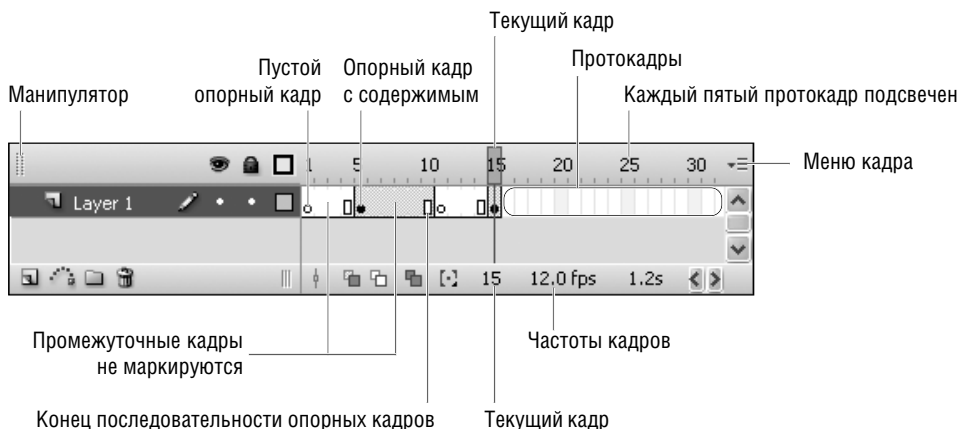
## Использование временной шкалы

На панели временной шкалы есть пять вариантов просмотра кадров и два способа предварительного просмотра содержимого кадра в миниатюре. Ролик Flash может состоять из сотен кадров. Полосы прокрутки временной шкалы позволяют получить доступ к кадрам, которые оказались за пределами окна. Можно на время скрыть временную шкалу, чтобы высвободить больше места для арены. Можно также отстыковать окно временной шкалы и сделать его плавающим или изменить размер окна, чтобы в нем помещалось больше или меньше кадров.

На рис. 8.1 показана временная шкала для ролика, содержащего один слой и 15 кадров.

### Чтобы изменить размер временной шкалы независимо от документа:

1. Откройте новый документ Flash. По умолчанию появляется временная шкала.
2. Укажите курсором на манипулятор (текстурированная область в левом углу полосы заголовка временной шкалы).
3. Потяните за него, чтобы отстыковать панель временной шкалы от края окна документа. Серый контур показывает новое положение панели.
4. Выбрав подходящее место, отпустите кнопку мыши. Временная шкала теперь отображается в отдельном окне изменяемого размера.



**Рис. 8.1.** На временной шкале представлены все кадры ролика. Если щелкнуть по любому кадру, его содержимое появится на арене.

**С**

Чтобы вновь пристыковать панель временной шкалы, щелкните по манипулятору и подтащите ее к верхнему краю окна документа. Курсор должен находиться над панелью редактирования или над полосой заголовка документа. Когда вы отпустите кнопку мыши, панель пристыкуется.

**С**

Панель временной шкалы можно пристыковать к любому краю окна документа. Подтяните ее за манипулятор к нужному краю. При отпускании кнопки мыши панель пристыковывается.

**С**

Чтобы переместить плавающее окно временной шкалы, следует тянуть за полосу заголовка, а не за манипулятор.

**С**

У пункта меню панели **Placement** (Размещение) есть подменю, содержащее команды для отстыковки временной шкалы или пристыковки ее к любому краю окна документа.

### Представление временной шкалы

При создании нового документа Flash на временной шкале виден единственный слой с сотнями мелких ячеек. Первая ячейка обведена сплошным черным контуром, и в ней находится полный маркер; контуры остальных ячеек серые. Каждая пятая ячейка закрашена серым цветом. Ячейка с черным контуром и маркером представляет *опорный кадр*, ячейки с серым контуром еще не определены и называются *промежуточными*.

По мере добавления опорных кадров контуры соответствующих ячеек в *последовательности опорного кадра* становятся черными.

Для пустого опорного кадра (для которого на арене нет никакого содержимого) маркер на временной шкале будет полным. Если же содержимое имеется, то маркер сплошной. Кадры, находящиеся между опорными, называются *промежуточными*.

Для любого промежуточного кадра, следующего за опорным кадром, который имеет содержимое, это содержимое и отображается на арене. Для последнего из подряд идущих промежуточных кадров на временной шкале показан полный маркер. Если выбрать представление кадров со штриховкой (**Tinted Frames** (Окрашенные кадры)) (это режим по умолчанию), то промежуточные кадры с содержимым выделяются еще и штриховкой.

### Чтобы скрыть/показать временную шкалу:

- Нажмите кнопку **Hide Timeline** (Скрыть временную шкалу) на панели редактирования.

Панель временной шкалы убирается или появляется (рис. 8.2). Двойной щелчок по имени сцены или по серой области на панели редактирования также переключает видимость временной шкалы. Отметим, что кнопка **Hide Timeline** видна только в случае, когда временная шкала активна и пристыкована к верхнему краю окна документа.

### Чтобы закрыть/открыть временную шкалу:

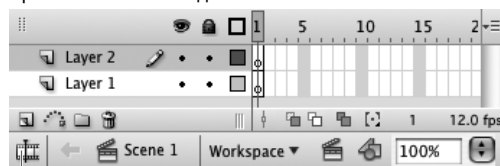
- Выберите из меню пункт **Window ⇒ Timeline** (Окно ⇒ Временная шкала) или нажмите комбинацию клавиш **Option-⌘-T** (Mac) либо **Ctrl-Alt-T** (Windows). Если временная шкала была открыта, то Flash закроет ее. Чтобы вернуть временную шкалу на место, повторно выполните то же самое действие.

### Чтобы изменить размеры кадров на временной шкале:

- В меню временной шкалы выберите способ показа кадров (рис. 8.3).

Flash изменит размеры кадров на временной шкале в соответствии с вашим выбором (см. врезку «Размеры кадров»).

#### Временная шкала видна

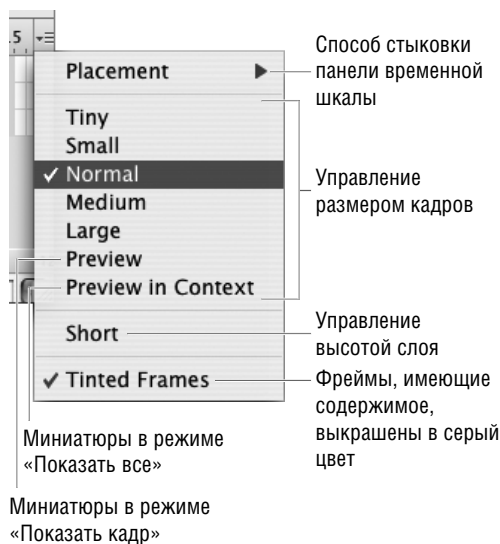


Кнопка «Скрыть временную шкалу»

#### Временная шкала скрыта



**Рис. 8.2.** Чтобы на время скрыть панель временной шкалы, нажмите кнопку Hide Timeline; чтобы показать скрытую шкалу, нажмите ту же кнопку еще раз.



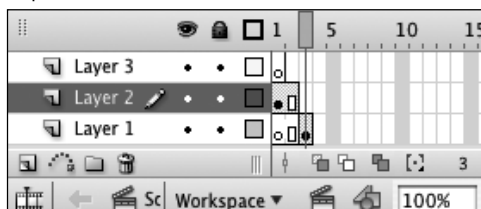
**Рис. 8.3.** С помощью меню панели можно управлять представлением кадров на временной шкале.



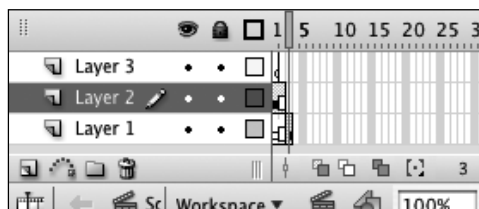
### Размеры кадров

В меню панели временной шкалы есть пункты для задания размера кадров. Если в документе много слоев и кадров, то имеет смысл выбрать режим **Tiny** (Мелкие), тогда на шкале уместится наибольшее число кадров, но понять, что на них изображено, будет сложно. Режимы **Normal** (Нормальные) и **Medium** (Средние) удобны, когда нужно различать опорные кадры с содержимым, пустые опорные кадры и промежуточные кадры. В режимах **Preview** (Предварительный просмотр) и **Preview in Context** (Предварительный просмотр в контексте) на временной шкале показываются миниатюрные изображения кадров. На рис. 8.4 представлены некоторые режимы просмотра кадров.

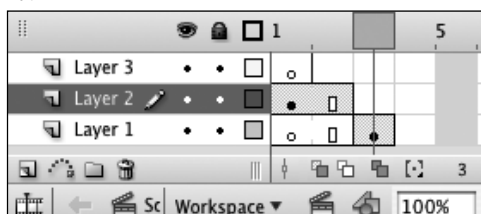
Нормальные



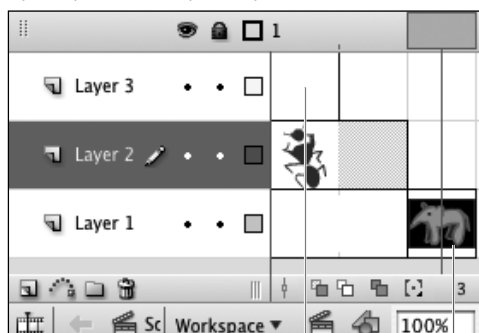
Мелкие



Крупные



Предварительный просмотр



Пустой опорный фрейм

Опорный фрейм с содержимым

**Рис. 8.4.** Flash может показывать кадры на временной шкале, в разных масштабах от мелкого до крупного. Можно также установить режим предварительного просмотра содержимого кадров.

## Создание опорных кадров

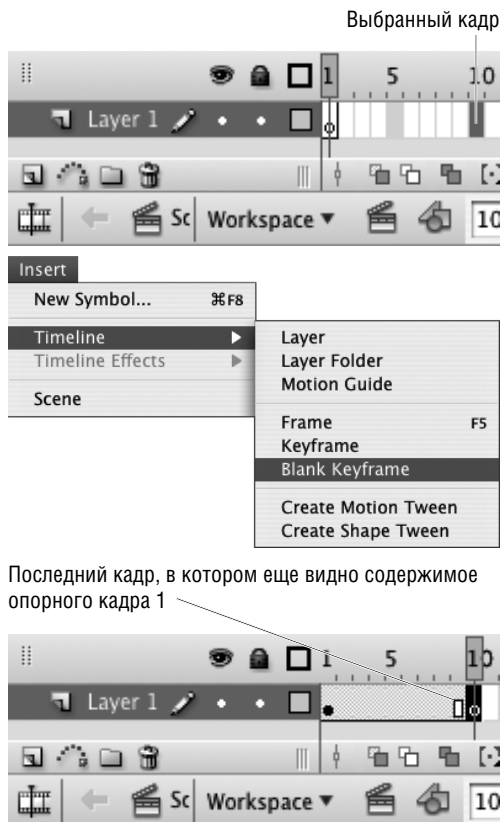
Flash предлагает две команды для создания опорных кадров. Команда **Insert** ⇒ **Timeline** ⇒ **Blank Keyframe** (Вставить ⇒ Временная шкала ⇒ Пустой опорный кадр) вставляет опорный кадр без содержимого; применяйте ее, когда нужно полностью изменить картинку на арене. Команда **Insert** ⇒ **Timeline** ⇒ **Keyframe** (Вставить ⇒ Временная шкала ⇒ Опорный кадр) вставляет опорный кадр, дублирующий содержимое предыдущего опорного кадра в том же слое. Она полезна, если необходимо модифицировать содержимое предыдущего опорного кадра.

**С** В упражнениях этого раздела вы будете обращаться к командам работы с кадрами из главного меню. Однако эквивалентные команды есть также в контекстном меню кадра. Чтобы открыть его, щелкните мышью с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows).

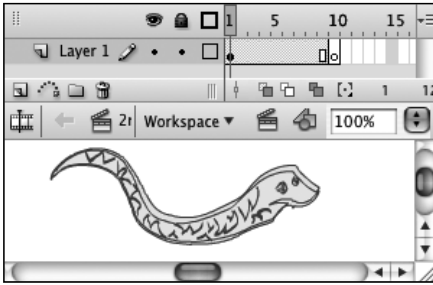
### Чтобы добавить пустой опорный кадр в конец ролика:

1. Создайте новый документ Flash. По умолчанию новый документ содержит один слой и один пустой опорный кадр в первой позиции.
2. На временной шкале щелкните по протокадру в позиции 10.
3. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Blank Keyframe** (рис. 8.5).

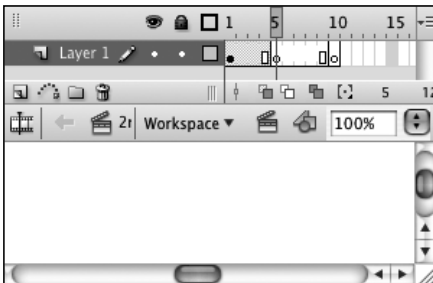
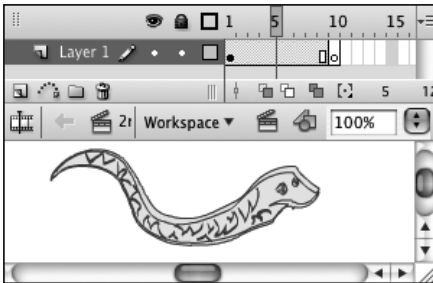
Flash отражает новое состояние на временной шкале. В кадре 9 появляется полый маркер, а между кадрами 9 и 10 появляется черная линия. Она обозначает конец последовательности одного опорного кадра и начало следующей. Вместо серых линий, разделяющих протокадры, Flash рисует риски, а выделение каждого пятого протокадра серым цветом убирается.



**Рис. 8.5.** Выберите кадр на временной шкале (верхний рисунок) и выполните команду **Insert** ⇒ **Timeline** ⇒ **Blank Keyframe** (средний рисунок). Будет добавлен новый пустой опорный кадр (нижний рисунок).



**Рис. 8.6.** Когда вы наполняете опорный кадр содержимым, Flash помечает его на временной шкале сплошным маркером. Серая штриховка в промежуточных кадрах означает, что в них отображается то же содержимое, что и в предшествующем опорном кадре. Полный прямоугольник — это признак конца последовательности промежуточных кадров, отображающих одно и то же содержимое. Полным маркером-кружком обозначается пустой опорный кадр.



**Рис. 8.7.** В момент преобразования промежуточного кадра с содержимым в пустой опорный кадр Flash удаляет с арены содержимое этого кадра. Кадры 6–9 были заштрихованы, так как в них отображалось содержимое кадра 1 (верхний рисунок). Но после вставки пустого опорного кадра в позицию 5 (нижний рисунок) штриховка исчезла, так как теперь эти кадры повторяют содержимое последнего предшествующего им опорного кадра, каковым является пустой кадр 5.

### Чтобы создать пустой опорный кадр в середине ролика:

1. Выполните те же шаги, что в предыдущем упражнении, для создания однослойного ролика с 10 кадрами.
2. Щелкните по кадру 1 на временной шкале.
3. Нарисуйте на арене какую-нибудь форму.

Flash обновит временную шкалу, добавив сплошной маркер к кадру 1 (рис. 8.6). Если установлен режим Tinted Frames (по умолчанию), то кадры 1–9 будут заштрихованы серым цветом. Это означает, что кадр 1 имеет содержимое, и оно будет показываться вплоть до кадра 9 в этом слое. В кадре 9 присутствует полный прямоугольник, обозначающий конец последовательности промежуточных кадров, в которых отображается содержимое опорного кадра 1.

В кадре 10 маркер по-прежнему полный, так как содержимого в нем еще нет.

4. В области временной шкалы над слоями щелкните по числу 5 или перетащите точку воспроизведения на пятый кадр. Flash покажет содержимое кадра 5 на арене. Обратите внимание, что в этом промежуточном кадре отображается то же содержимое, что в предшествующем ему опорном кадре 1.
5. Выполните команду **Insert** ⇨ **Timeline** ⇨ **Blank Keyframe**. Flash преобразует выбранный промежуточный кадр в опорный и очистит арену (рис. 8.7).

### Чтобы продублировать содержимое предшествующего опорного кадра:

1. Откройте новый документ и поместите какое-нибудь содержимое в кадр 1.
2. На временной шкале выберите кадр 3. Точка воспроизведения не перемещается в область протокадра, но кадр 3 выделяется.
3. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**.

Flash создает новый опорный кадр, дублирует содержимое кадра 1 в кадр 3 и помечает кадр 3 сплошным маркером, а кадр 2 — полым прямоугольником (рис. 8.8). Содержимое кадров 1 и 3 никак не связано между собой.

С

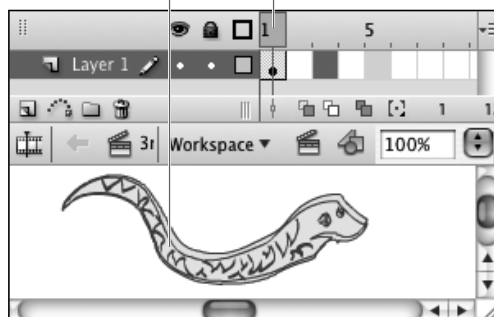
Слово *insert* (вставить) в применении к опорным кадрам не совсем точно. Команда **Insert** ⇒ **Timeline** ⇒ **Keyframe** добавляет кадры только, если вы предварительно выберете протокадр. Если же выбран существующий промежуточный кадр, то он преобразуется в опорный, а общая длина ролика не изменяется. Команда же **Insert** ⇒ **Timeline** ⇒ **Frame** всегда добавляет в ролик новый кадр.

С

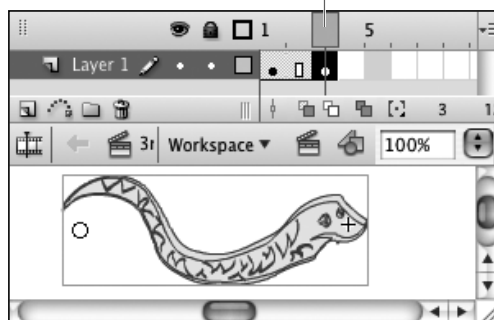
Невозможно вставить пустой опорный кадр между двумя соседними опорными кадрами. Сделав кадры 5 и 6 опорными, выберите кадр 5 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Blank Keyframe**. Точка воспроизведения переместится на кадр 6, но никакого нового пустого опорного кадра Flash не вставит. Если выбран кадр 5, то следует выполнять команду **Insert** ⇒ **Timeline** ⇒ **Frame**. Тогда в позиции 6 будет создан промежуточный кадр. Теперь выберите кадр 5 или 6 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Blank Keyframe**; кадр 6 будет преобразован в опорный.

Содержимое первого опорного кадра

Текущий кадр



Дубликат первого опорного кадра



**Рис. 8.8.** Команда **Insert** ⇒ **Timeline** ⇒ **Keyframe** создает опорный кадр с таким же содержимым, как у предшествующего опорного кадра в том же слое.

### Опорные кадры: вставка и преобразование

Помимо команд **Insert ⇒ Timeline ⇒ Keyframe** и **Insert ⇒ Timeline ⇒ Blank Keyframe**, Flash располагает командами для преобразования промежуточных кадров в опорные, а именно **Modify ⇒ Timeline ⇒ Convert to Keyframes** (Изменить ⇒ Временная шкала ⇒ Преобразовать в опорные кадры) (клавиша F6) и **Modify ⇒ Timeline ⇒ Convert to Blank Keyframes** (Изменить ⇒ Временная шкала ⇒ Преобразовать в пустые опорные кадры) (клавиша F7). Такие же команды есть в контекстном меню кадра.

Какую команду выбрать — вставки или преобразования, — зависит от количества выбранных в данный момент кадров и от того, сколько кадров вы хотите создать. Команды из меню **Insert** создают только один ключевой кадр вне зависимости от того, сколько кадров выбрано, а команды из меню **Modify** создают по одному опорному кадру для каждого выбранного.

Если выбран всего один кадр, то команды **Insert ⇒ Timeline ⇒ Keyframe** и **Modify ⇒ Timeline ⇒ Convert to Keyframe** эквивалентны. Если выбран один протокадр или промежуточный кадр, то обе команды преобразуют его в опорный и дублируют содержимое предшествующего опорного кадра (если таковой существует). Если же выбран опорный кадр, за которым следует промежуточный или протокадр, то следующий за ним кадр преобразуется в опорный с таким же содержанием, как в выбранном кадре. Ни та, ни другая команда не оказывает влияния на выбранный опорный кадр, если следующий за ним тоже является опорным.

Если выбрано несколько протокадров и промежуточных кадров, то команда **Insert ⇒ Timeline ⇒ Keyframe** создает единственный опорный кадр, обычно в том месте, на которое указывает точка воспроизведения (если выбраны кадры в конце ролика, то новый опорный кадр создается на месте последнего выбранного). Остальные выбранные кадры становятся промежуточными.

Если выбрано несколько протокадров и промежуточных кадров, то команда **Modify ⇒ Timeline ⇒ Convert to Keyframes** создает опорный кадр на месте каждого выбранного.

Для пустых опорных кадров эти команды работают аналогично. **Insert ⇒ Timeline ⇒ Blank Keyframe** создает один опорный кадр в том месте, на которое указывает точка воспроизведения, а остальные кадры становятся промежуточными. **Modify ⇒ Timeline ⇒ Convert to Blank Keyframe** создает пустой опорный кадр на месте каждого выбранного.

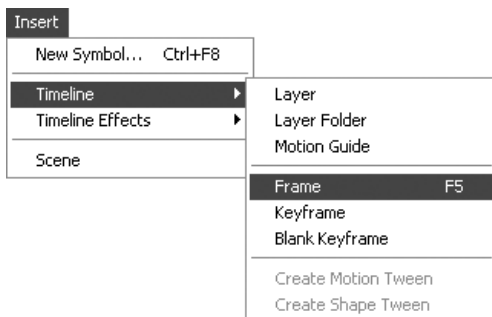
## Создание промежуточных кадров

Кадры между опорными в каком-то смысле привязаны к предшествующему опорному кадру. В них отображается содержимое последнего, а предназначены они для создания промежуточной анимации (см. главы 9 и 10). Flash ясно визуализирует связь между этими кадрами, подсвечивая их и помечая последний кадр в последовательности полым прямоугольником.

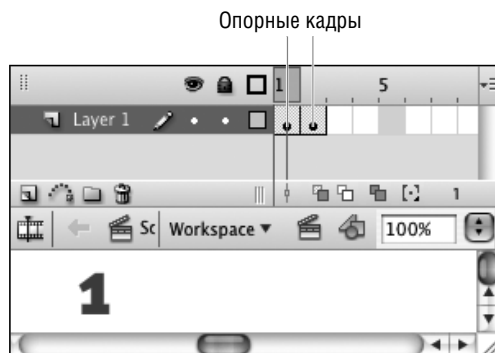
### Чтобы добавить промежуточные кадры:

1. Откройте или создайте документ с непустыми опорными кадрами в позициях 1 и 2.
2. На временной шкале поместите точку воспроизведения на кадр 1.
3. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame** или нажмите **F5** (рис. 8.9).

Flash добавит промежуточный кадр (рис. 8.10). Теперь ролик содержит опорный кадр в позиции 1, промежуточный — в позиции 2 и еще один опорный — в позиции 3.

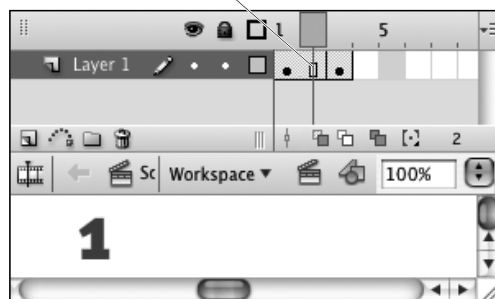


**Рис. 8.9.** Для добавления промежуточных кадров выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame**.



Временная шкала до выполнения команды **Insert** ⇒ **Timeline** ⇒ **Frame**

Промежуточный кадр



Временная шкала после выполнения команды **Insert** ⇒ **Timeline** ⇒ **Frame**

**Рис. 8.10.** Команда **Insert** ⇒ **Timeline** ⇒ **Frame** добавляет промежуточный кадр после того, на который указывает точка воспроизведения.

### Что такое опорные и промежуточные кадры?

На заре мультипликации для создания огромного числа рисунков, необходимых для покадровой анимации, требовалась целая армия художников. Чтобы сократить расходы, киностудии разбивали весь объем работы на категории в зависимости от квалификации художника и оплаты. В самом начале создавались спецификации на каждого персонажа. Затем наступала очередь этапа раскадровки, на котором вчерне описывался сценарий фильма. И, наконец, вступали в дело художники, которым предстояло нарисовать сотни келей, чтобы оживить персонажей.

Чтобы процесс не вышел из-под контроля, аниматоры разбивали каждое движение на последовательность *опорных* кадров, определяющих основные моменты движения, и прочих, которые содержали постепенные изменения с целью имитации перемещения. Последние назывались *промежуточными кадрами*. На опорных кадрах фиксировались значимые изменения в положении персонажа или сцены. Представьте себе последовательность из 25 кадров, в начале которой Багз Бани стоит лицом к аудитории, а в конце поворачивает голову направо, чтобы взглянуть на Даффи Дака. Для этой сцены необходимо два опорных кадра — Багз анфас и Багз в профиль — и 23 промежуточных.

Раньше некоторые художники специализировались исключительно на опорных кадрах, тогда как в задачу других (обычно ниже оплачиваемых) входило создание промежуточных. Последние копировали рисунки с опорных кадров и вносили в них небольшие изменения, чтобы намеченное движение уложилось в отведенное число кадров, не нарушая иллюзию непрерывности. В главах 9 и 10 вы узнаете, как заставить Flash выполнять такую низко квалифицированную работу. Программа может взять на себя скучную генерацию промежуточных кадров для определенных видов анимации.

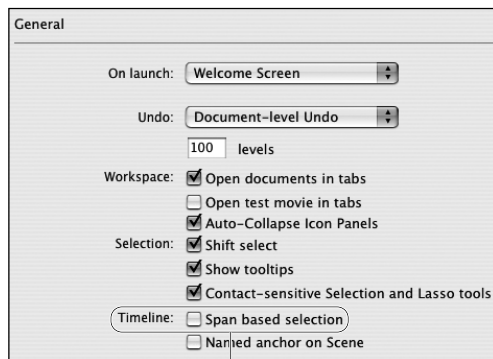
Во Flash опорные кадры необходимы для описания любого изменения содержимого или фона, как бы мало оно ни было. Термин *промежуточные кадры* во Flash не применяется, любой кадр, отличный от опорного, называется просто *кадром*. Но для ясности мы в этой книге все же называем такие кадры промежуточными.

## Выбор кадров

Во Flash есть два способа выбора кадров на временной шкале. По умолчанию установлен одиночный режим выбора, когда каждый кадр рассматривается по отдельности. А в групповом режиме кадры считаются членами *последовательности опорного кадра*, которая состоит из самого опорного кадра и всех следующих за ним промежуточных. В групповом режиме щелчок по любому кадру в середине последовательности выбирает всю последовательность. Если явно не оговорено противное, то мы будем предполагать, что установлен одиночный режим выбора.

### Чтобы установить режим выбора:

1. Выберите из меню **Flash** (Mac) или **Edit** (Windows) пункт **Preferences** (Установки). Откроется диалоговое окно Preferences.
2. Перейдите в категорию **General** (Общие). В правой части окна отображаются общие параметры, как показано на рис. 8.11.
3. В разделе Timeline (Временная шкала) установите режим выбора кадров:
  - чтобы действовал групповой режим, отметьте флажок **Span Based Selection** (Выделение на основе диапазона) (рис. 8.12);
  - чтобы действовал одиночный режим, сбросьте флажок **Span Based Selection** (рис. 8.13);



Timeline: ☐ Span based selection

**Рис. 8.11.** Установите режим выбора кадров на временной шкале в категории General в диалоговом окне Preferences.

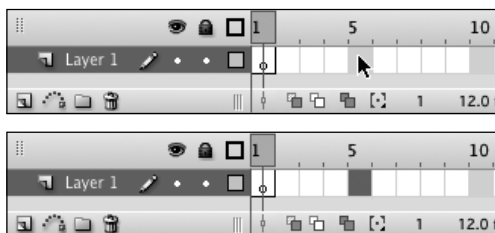
Timeline: ☒ Span based selection

**Рис. 8.12.** Если отметить флажок Span Based Selection, то будет выбираться вся последовательность опорного кадра.

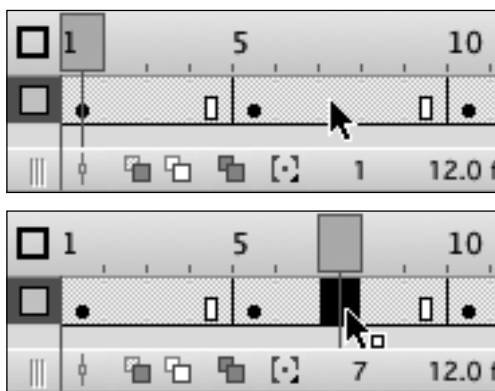
Timeline: ☐ Span based selection

**Рис. 8.13.** Если сбросить флажок Span Based Selection, то будут выбираться отдельные кадры.





**Рис. 8.14.** Протокадры в обоих режимах выбора обрабатываются одинаково. Чтобы выбрать протокадр, укажите на него курсором (верхний рисунок) и щелкните мышью (нижний рисунок).



**Рис. 8.15.** В одиночном режиме выбора кадров щелчок по кадру в середине последовательности (верхний рисунок) выбирает только этот кадр (нижний рисунок).

### Чтобы выбрать не определенные кадры:

Для работы с протокадрами выполните одно из следующих действий:

- чтобы выбрать один протокадр, щелкните по нему мышью (рис. 8.14);
- чтобы выбрать два протокадра и все кадры между ними, щелкните мышью по первому и последнему, удерживая клавишу **Shift**;
- чтобы выбрать несмежные протокадры, щелкайте по ним с нажатой клавишей **⌘** (Mac) или **Ctrl** (Windows).

Процедура выбора еще не определенных кадров не зависит от установленного режима выбора.

### Чтобы выбрать определенные кадры по отдельности:

Установив режим одиночного выбора кадров, выполните следующие действия:

- чтобы выбрать опорный кадр, щелкните по нему;
- чтобы выбрать последний кадр в последовательности опорного кадра, щелкните по нему;
- чтобы выбрать кадр в середине последовательности опорного кадра, щелкните по нему (рис. 8.15);
- чтобы выбрать всю последовательность опорного кадра, дважды щелкните по любому входящему в нее кадру;
- чтобы выбрать кадры вдобавок к уже выбранным, щелкните по какому-нибудь из них с нажатой клавишей **Shift**. Flash выберет все кадры между последним выбранным и тем, по которому вы щелкнули;
- чтобы выбрать диапазон кадров, щелкните с нажатой клавишей **Shift** по первому и последнему кадру в этом диапазоне или в момент, когда не выбрано ни одного кадра, щелкните по первому кадру диапазона и буксируйте мышью до последнего кадра.

### Чтобы выбрать определенные кадры в групповом режиме:

Установив режим группового выбора кадров, выполните следующие действия:

- чтобы выбрать опорный кадр, щелкните по нему;
- чтобы выбрать последний кадр в последовательности опорного кадра, щелкните по нему;
- чтобы выбрать один промежуточный кадр, щелкните по нему с нажатой клавишей **⌘** (Mac) или **Ctrl** (Windows);
- чтобы выбрать всю последовательность опорного кадра, щелкните по любому входящему в нее кадру (рис. 8.16) или щелкните с нажатой клавишей **Shift** по первому либо последнему кадру последовательности;
- чтобы добавить последовательности к уже выбранным, щелкните по какой-нибудь из них. Необязательно выбирать только смежные последовательности (рис. 8.17);
- чтобы выбрать диапазон кадров на платформе Windows, буксируйте по нему мышью с нажатой клавишей **Ctrl**.

**С**

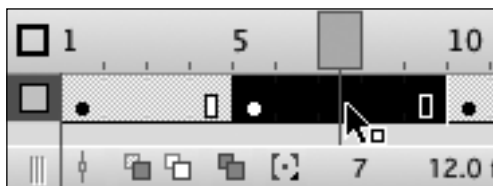
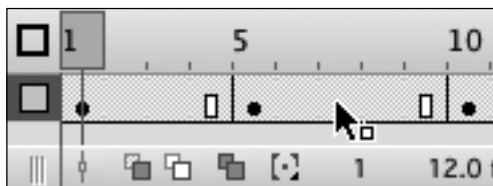
В режиме группового выбора первый и последний кадр последовательности становятся «добавляющими». Если что-то уже выбрано, то щелчок по любому из них добавляет к выбранному всю последовательность.

**С**

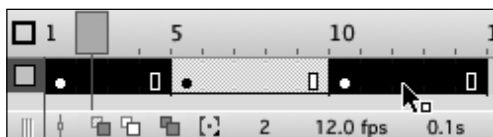
В обоих режимах можно выбрать все кадры в одном слое, если щелкнуть по имени слоя. В режиме группового выбора все кадры можно выбрать также, дважды щелкнув по любому кадру.

**С**

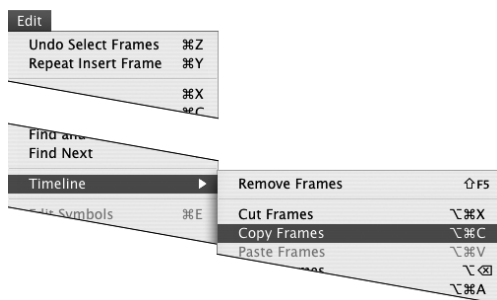
В обоих режимах можно выбирать несмежные кадры, щелкая по ним мышью с нажатой клавишей **⌘** (Mac) или **Ctrl** (Windows).



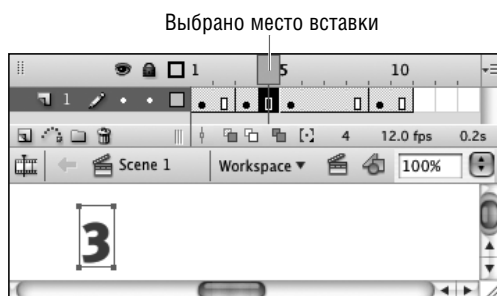
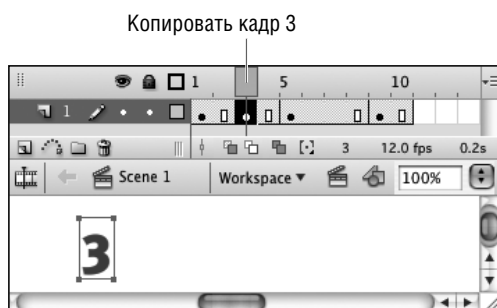
**Рис. 8.16.** В режиме группового выбора щелчок по кадру в середине последовательности (верхний рисунок) выбирает всю последовательность (нижний рисунок).



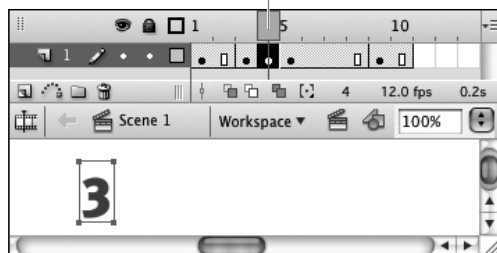
**Рис. 8.17.** В режиме группового выбора щелчком с нажатой клавишей Shift можно выбрать несмежные последовательности опорного кадра.



**Рис. 8.18.** В меню **Edit** имеются специальные команды для копирования и вставки кадров на временной шкале.



Flash вставляет содержимое скопированного кадра 3 в кадр 4



**Рис. 8.19.** При вставке кадра с новым содержимым на место промежуточного кадра Flash преобразует этот кадр в опорный.

## Манипулирование кадрами в одном слое

Нельзя копировать и вставлять кадры с помощью стандартных команд **Copy** и **Paste**, применимых к графическим элементам. В меню **Edit** (Редактирование) есть специальные команды для этой цели. Кроме того, Flash позволяет перетаскивать выбранные кадры мышью в новое место на временной шкале.

Для выполнения следующих упражнений откройте новый документ Flash. Создайте ролик из 10 кадров и сделайте кадры 1, 3, 5, 9 опорными. С помощью инструмента «Текст» поместите в каждый опорный кадр текстовое поле и введите в него номер кадра; так будет проще следить за тем, где оказался кадр в результате выполнения операции.

### Чтобы скопировать и вставить один кадр:

1. Выберите на временной шкале кадр 3.
2. Выполните команду **Edit** ⇒ **Timeline** ⇒ **Copy Frames** (Редактирование ⇒ Временная шкала ⇒ Копировать кадры) или нажмите комбинацию клавиш **Option-⌘-C** (Mac) либо **Ctrl-Alt-C** (Windows) (рис. 8.18).

Flash скопирует выбранный кадр в буфер обмена.

3. Щелкните по кадру 4 на временной шкале, чтобы отметить место вставки скопированного кадра.
4. Выполните команду **Edit** ⇒ **Timeline** ⇒ **Paste Frames** (Редактирование ⇒ Временная шкала ⇒ Вставить кадры) или нажмите комбинацию клавиш **Option-⌘-V** (Mac) либо **Ctrl-Alt-V** (Windows)

Flash вставит скопированный кадр в позицию 4 (рис. 8.19).

5. Вставьте еще одну копию в позицию 5 (рис. 8.20).

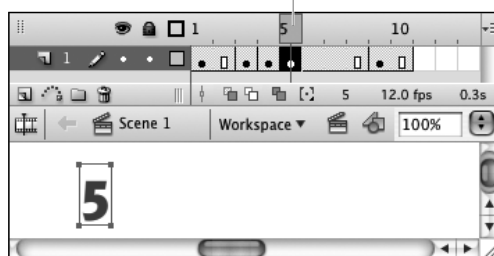
Flash заменяет содержимое опорного кадра 5 содержимым опорного кадра 3.

6. Вставьте еще одну копию в позицию 12.

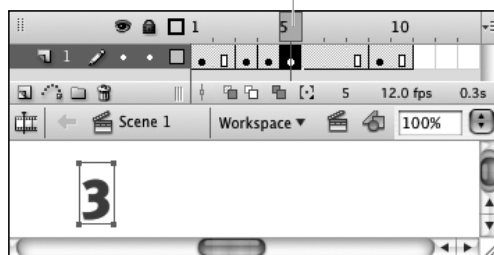
Flash продлевает ролик, освобождая место для вставленного кадра. Отметим, что точка воспроизведения не перемещается на протокадр 12, пока вы не вставите копию.

- Можно скопировать и вставить сразу несколько кадров; для этого на шаге 1 предыдущего упражнения выберите диапазон кадров.
- Чтобы скопировать и вставить содержимое опорного кадра, можно скопировать промежуточный кадр с тем же содержимым. В момент вставки Flash создаст новый опорный кадр.
- Предупреждение: Flash всегда заменяет содержимое выбранного кадра содержимым вставленного (в случае вставки нескольких кадров — содержимым первого из них). Если забыть об этом, можно случайно стереть содержимое нужных опорных кадров. Дабы этого не случилось, вставляйте скопированные кадры на место промежуточных или пустых опорных кадров. Потом лишние опорные кадры можно будет удалить.

Выбран опорный кадр 5



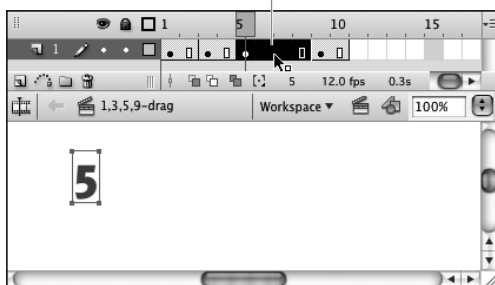
Вставлено содержимое опорного кадра 3



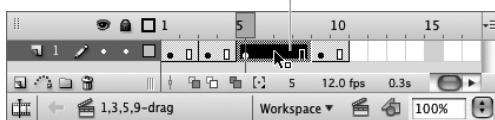
**Рис. 8.20.** При вставке кадра с новым содержимым на место опорного кадра Flash заменяет содержимое последнего.

**Выбор и перетаскивание: любой режим**

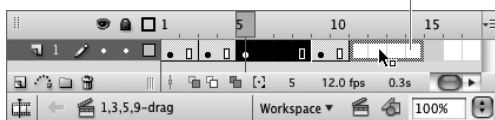
Выбранные кадры



Кадры готовы к перетаскиванию

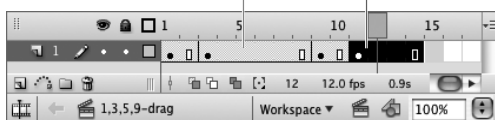


Предварительный просмотр нового места размещения

**После бросания: режим одиночного выбора**

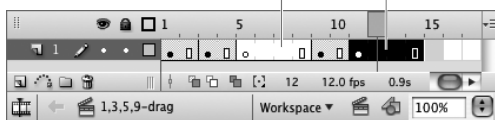
Исходный опорный кадр удален

Кадры в новом месте

**После бросания: режим группового выбора**

Исходный опорный кадр сохранен, но его содержимое стерто

Кадры в новом месте



**Рис. 8.21.** Процедура перетаскивания кадров по временной шкале одинакова в обоих режимах выбора (верхний рисунок). Однако ее результаты совершенно различны (нижний рисунок). В групповом режиме выбранные опорные кадры удаляются со своих позиций и заменяются промежуточными. В одиночном режиме опорные кадры остаются на прежних местах, но их содержимое стирается.

**Чтобы переместить кадры буксировкой:**

1. В том же документе, что и раньше, выберите последовательность, которая начинается с опорного кадра 5 и кончается кадром 8.
2. Укажите курсором на любой из выбранных кадров.
3. Начинаяте буксировать мышью. Flash дополнительно выделяет выбранные кадры штриховкой. Заштрихованный прямоугольник показывает текущее положение выбранных кадров на временной шкале в процессе буксировки.
4. Чтобы переместить выбранные кадры в конец ролика, отбуксируйте прямоугольник за последний определенный кадр в область протокадров и отпустите кнопку мыши.

Flash добавит кадры в конец ролика, в них будет то же содержимое, что в кадрах 5-8. В режиме группового выбора содержимое из кадров 5-8 стирается, а сами эти кадры включаются в последовательность предшествующего опорного кадра. В режиме одиночного выбора содержимое тоже стирается, но в позиции 5 остается пустой опорный кадр (рис. 8.21).

5. Чтобы переместить выбранные кадры в начало ролика, отбуксируйте их к кадру 1 и отпустите кнопку мыши.

Отбуксированные кадры замещают содержимое кадров 1-4, в позиции 5 больше нет опорного кадра.

C

Чтобы путем перетаскивания скопировать выбранные кадры, удерживайте во время буксировки клавишу **Option** (Mac) либо **Alt** (Windows).

C

Какой бы режим выбора ни был установлен в данный момент, нажатие клавиши **⌘** (Mac) либо **Ctrl** (Windows) позволяет временно переключиться в другой режим. В одиночном режиме этот модификатор превращает курсор в двунаправленную стрелку для расширения последовательности опорного кадра. Если модификатор нажат в групповом режиме, то курсор принимает форму стрелки для выбора отдельных кадров.

C

Если во время модификации кадров на временной панели вы допустите ошибку, то операцию можно отменить командой **Edit** ⇒ **Undo** (Редактирование ⇒ Отмена). Flash сохраняет на случай отмены историю выбора и отказа от выбора кадров. Чтобы отменить такие операции, как перетаскивание кадров в целях перемещения или расширения последовательностей, команду **Undo**, возможно, придется выполнить несколько раз, так как кое-какие действия Flash в этом случае выполняет незримо для вас.

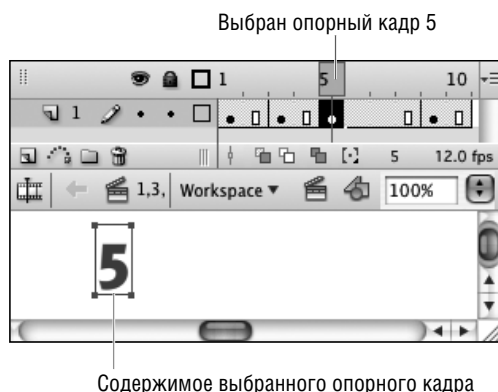
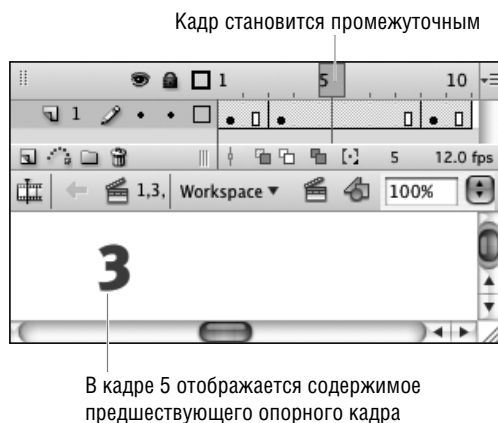
### Расширение последовательности опорного кадра

В режиме группового выбора курсор принимает форму двунаправленной стрелки, когда оказывается над опорным кадром или над конечным кадром последовательности. В этот момент можно перетащить невыбранный опорный или конечный кадр влево или вправо, чтобы соответственно уменьшить или увеличить длину последовательности.

Изменение длины последовательности, окруженной другими последовательностями, имеет ряд особенностей. Flash не позволяет расширяющейся последовательности «пожрать» содержимое других опорных кадров. Однако длина соседней последовательности может уменьшиться. В документе, с которым вы все это время работали, установите режим группового выбора и укажите курсором на кадр 4 (конечный в последовательности опорного кадра 3); этот кадр не должен быть выбран. Когда курсор примет форму двунаправленной стрелки, тащите кадр 4 вправо. Добравшись до кадра 7, вы обнаружите, что дальше «не тянется». Отпустите кнопку мыши. Последовательность, начинающаяся с кадра 3, теперь дошла до кадра 7. Содержимое бывшего опорного кадра 5 все еще существует, но перешло в кадр 8.

Чтобы увеличить длину последовательности, не изменяя длины соседних, выберите саму эту последовательность или какой-нибудь кадр внутри нее и выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame** (Вставить ⇒ Временная шкала ⇒ Кадр) либо нажмите **F5**. Flash добавит в выбранную последовательность промежуточный кадр, а все следующие за ней последовательности сдвинет вправо по временной шкале.

Если при перетаскивании длина последовательности уменьшается, то Flash создает последовательности пустого опорного кадра, чтобы заполнить лакуны между концом изменяемой последовательности и началом соседней. Чтобы просто сократить число промежуточных кадров, можно воспользоваться командой **Remove Frames** (см. раздел «Удаление кадров» ниже).

*Перед очисткой опорного кадра**После очистки опорного кадра*

**Рис. 8.22.** Команда **Modify** ⇒ **Timeline** ⇒ **Clear Keyframe** стирает с арены содержимое выбранного опорного кадра и преобразует его в промежуточный. Общая длина ролика не изменяется.

**Удаление кадров**

Во Flash есть две команды для удаления кадров: **Clear Keyframe** (Очистить ключевой кадр) и **Remove Frames** (Удалить кадры). Команда **Clear Keyframe** преобразует выбранный опорный кадр (или диапазон кадров) в промежуточный и удаляет из ролика содержимое опорных кадров. Эта команда не изменяет число кадров в ролике. Команда **Remove Frames** удаляет кадры из ролика (вместе с содержимым, если кадры опорные) и, стало быть, уменьшает общее число кадров. В следующих упражнениях вы будете работать с тем же документом, что и раньше.

**Чтобы преобразовать опорный кадр в промежуточный:**

1. Выберите на временной шкале опорный кадр 5.
2. Выполните команду **Modify** ⇒ **Timeline** ⇒ **Clear Keyframe** или нажмите **Shift-F6**.

Flash убирает из кадра 5 маркер (обозначавший, что это был опорный кадр) и стирает содержимое кадра. Кадр 5 становится промежуточным и в нем отображается содержимое кадра 3 (рис. 8.22). Общее число кадров в ролике не изменяется.

### Чтобы удалить один кадр из ролика:

1. Восстановите документ-пример в исходное состояние (опорные кадры в позициях 1, 3, 5, 9) и выберите на временной шкале кадр 4. Это промежуточный кадр, ассоциированный с опорным кадром 3.
2. Выполните команду **Edit ⇒ Timeline ⇒ Remove Frames** (Редактирование ⇒ Временная шкала ⇒ Удалить кадры) или нажмите **Shift-F5**.

Flash удаляет кадр 4, уменьшая на единицу общую длину ролика (рис. 8.23). Опорный кадр 5 теперь занимает четвертую позицию на шкале.

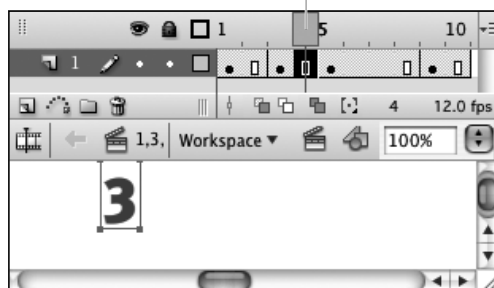
3. Выберите опорный кадр 3 и снова выполните команду **Edit ⇒ Timeline ⇒ Remove Frames**.

Flash удаляет выбранный кадр вместе с содержимым и уменьшает длину ролика на единицу. Содержимое опорного кадра, который находился в позиции 4, теперь перемещается в кадр 3.

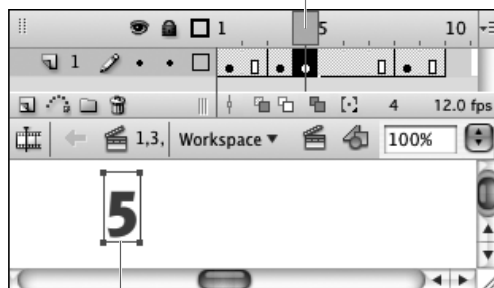
**С**

Flash не позволяет с помощью команды **Clear Keyframe** превратить первый опорный кадр в промежуточный, но удалить его можно. Если выбрать все кадры в слое и выполнить команду **Edit ⇒ Timeline ⇒ Remove Frames**, то будет удалено все, кроме протокадров. Чтобы наполнить слой содержимым, необходимо снова добавить опорный кадр в позицию 1.

Выбранный промежуточный кадр



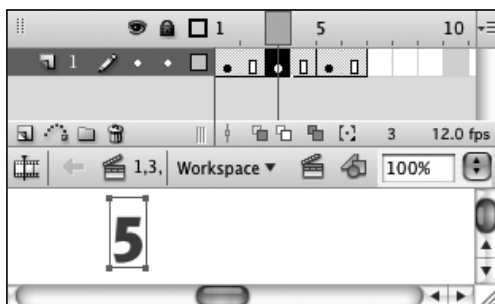
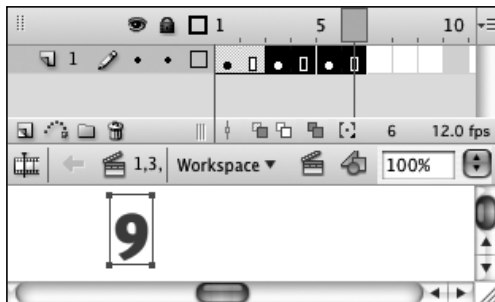
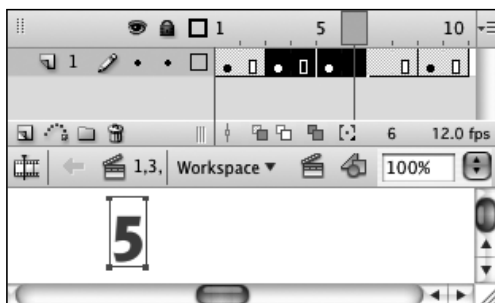
После удаления



Содержимое бывшего опорного кадра 5 теперь находится в кадре 4

**Рис. 8.23.** Команда **Edit ⇒ Timeline ⇒ Remove Frames** удаляет кадры из ролика, уменьшая тем самым его длину.





**Рис. 8.24.** Команда **Edit ⇒ Timeline ⇒ Remove Frames** позволяет удалять выбранный диапазон кадров. Поскольку последовательность опорного кадра (кадры 3–4) включена в диапазон целиком (верхний рисунок), то Flash не только уменьшает общее число кадров, но и удаляет содержимое этой последовательности (средний рисунок). Если выбрать только часть последовательности (кадры 5 и 6 в исходном документе), то она станет короче, но ее содержимое сохранится (нижний рисунок).

### Чтобы удалить диапазон кадров:

1. Восстановите документ-пример в исходное состояние (опорные кадры в позициях 1, 3, 5, 9) и выберите на временной шкале кадры 3–6.
2. Выполните команду **Edit ⇒ Timeline ⇒ Remove Frames**. Flash удаляет все выбранные кадры (рис. 8.24).

**С**

В режиме одиночного выбора можно быстро заменить содержимое одного опорного кадра содержимым другого. Выберите промежуточный кадр, в котором отображается нужное вам содержимое, и перетащите его на тот опорный кадр, содержимое которого требуется заменить.

## Создание простой покадровой анимации

В традиционной мультипликации иллюзия движения создается за счет демонстрации ряда мало отличающихся изображений. Процесс создания каждого из таких рисунков и помещения их в опорные кадры называется *покадровой анимацией*. Если же создаются только ключевые изображения, а Flash самостоятельно генерирует необходимые переходные рисунки, то мы говорим об *анимации с построением промежуточных изображений* или, короче, *интерполяции* (tweening). Подробнее об интерполяции вы узнаете в главах 9 и 10. Классический пример покадровой анимации — прыгающий мяч. Для нее требуется всего три кадра.

### Чтобы подготовить начальный опорный кадр:

1. Создайте новый документ и назовите его Frame-by-Frame Bounce. По умолчанию Flash создает документ с одним слоем, первый кадр которого является опорным. Выполните команду **View ⇒ Grid ⇒ Show Grid** (Просмотр ⇒ Сетка ⇒ Показать сетку), чтобы было удобнее размещать графику.
2. Выберите на временной шкале опорный кадр 1. С помощью меню временной шкалы установите режим предварительного просмотра в контексте (**Preview in Context**). Так будет проще следить за тем, что происходит.
3. На панели инструментов выберите инструмент «Овал».
4. Задайте цвет обводки No Color (Без цвета).
5. Ближе к верхнему краю арены нарисуйте круг (рис. 8.25), он и будет мячом. Сделайте его побольше. Для более эффективного использования графики преобразуйте этот круг в символ (см. главу 7).

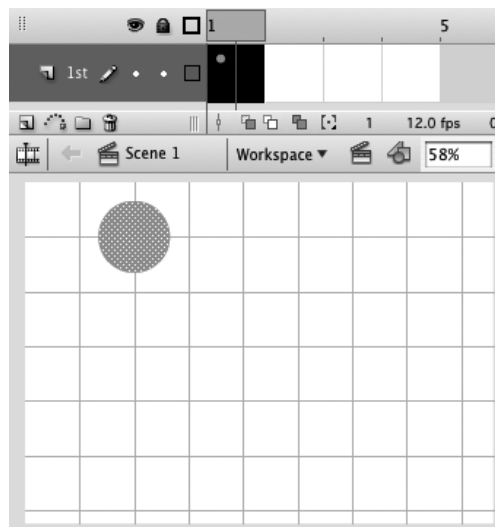
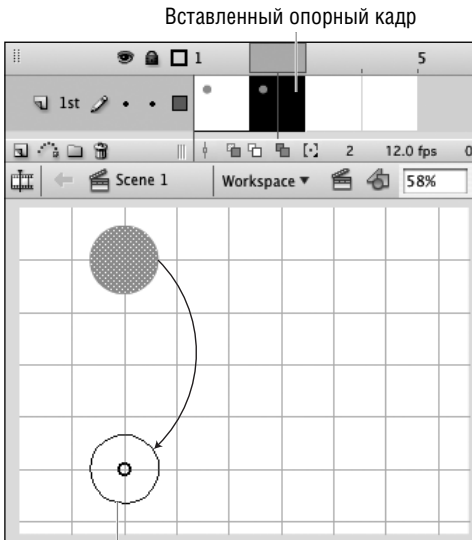


Рис. 8.25. В опорном кадре 1 нарисуйте круг у верхнего края арены, он станет прыгающим мячом.

### Когда применять покадровую анимацию

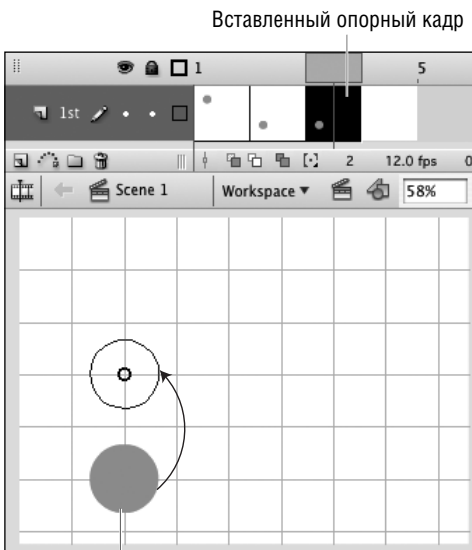
Чем больше кадров в покадровой анимации, тем меньше различия между ними, и, следовательно, тем более плавным получается движение. Однако лишние опорные кадры увеличивают размер ролика и тем самым время загрузки для тех, кто просматривает ролик через Web. Ваша цель — найти золотую середину.

В примере с прыгающим мячом можно уменьшить размер ролика, сделав мяч символом, тогда добавление еще одного опорного кадра мало скажется на размере файла. Но в реальных роликах, если вообще удастся воспользоваться символами, то лучше прибегнуть к более эффективной технике анимации — *интерполяции*. Поэтому покадровую анимацию лучше оставить для тех случаев, когда формы постоянно изменяются, и эти изменения нужно контролировать очень точно. В остальных случаях применяйте инструменты автоматического построения промежуточных движений и форм (см. главы 9 и 10).



Перетащите мяч в другое место

**Рис. 8.26.** Выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**, чтобы продублировать в кадре 2 мяч, изображенный в кадре 1. Затем перетащите мяч в новое положение.



Перетащите мяч в другое место

**Рис. 8.27.** Выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**, чтобы продублировать в кадре 3 мяч, изображенный в кадре 2. И снова измените положение мяча.

### Чтобы создать второй опорный кадр:

1. Выберите на временной шкале кадр 2.
2. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe** (Вставить ⇒ Временная шкала ⇒ Опорный кадр). Flash создает в позиции 2 опорный кадр, являющийся точной копией кадра 1.
3. В опорном кадре 2 выберите мяч и переместите его к нижнему краю арены (рис. 8.26).

### Чтобы создать третий опорный кадр:

1. Выберите на временной шкале кадр 3.
2. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe** (Вставить ⇒ Временная шкала ⇒ Опорный кадр). Flash создает в позиции 3 опорный кадр, являющийся точной копией кадра 2.
3. В опорном кадре 3 выберите мяч и переместите его в середину арены (рис. 8.27).

Все! Хотите верить, хотите нет, но вы создали все необходимое для анимации прыгающего мяча. Чтобы посмотреть, как это работает, щелкните по очереди по кадрам 1, 2 и 3. Изменение содержимого на арене и есть пример очень грубой анимации.

## Предварительный просмотр действия

Хотя для просмотра ролика можно просто по очереди щелкнуть каждый кадр, Flash предлагает и более развитые средства для тестирования анимации. В окне пульта управления (Controller) имеются кнопки воспроизведения в духе видеомagneтофона. В меню Control имеются команды для воспроизведения. Можно также экспортировать ролик в файл и просмотреть его во Flash-плеере.

### Чтобы воспользоваться пультом управления:

1. Выберите из главного меню пункт **Window** ⇒ **Toolbars** ⇒ **Controller** (Окно ⇒ Панели инструментов ⇒ Пульт управления) (рис. 8.28). Открывается окно со стандартными магнитофонными кнопками.
2. В окне пульта управления нажмите кнопку, соответствующую какой-нибудь команде (рис. 8.29).

**С** Если вы не любите загромождать рабочий стол лишними плавающими окнами, то имейте в виду, что все функции пульта управления имеются и в меню **Control** (Управление).

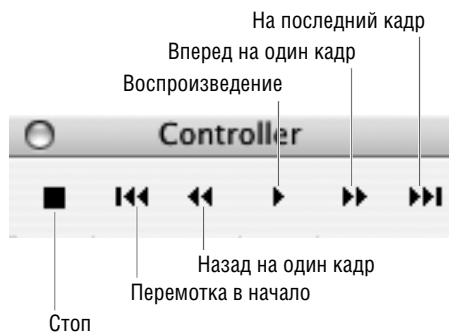
### Чтобы последовательно просмотреть все кадры:

1. Выберите на временной шкале кадр 1.
2. Из меню **Control** (рис. 8.30) выберите пункт **Step Forward One Frame** (Сделать шаг вперед на один кадр) или нажмите точку (.). Flash переходит к следующему кадру.
3. Выберите пункт **Control** ⇒ **Step Backward One Frame** (Сделать шаг назад на один кадр) или нажмите запятую (,). Flash переходит к предыдущему кадру.

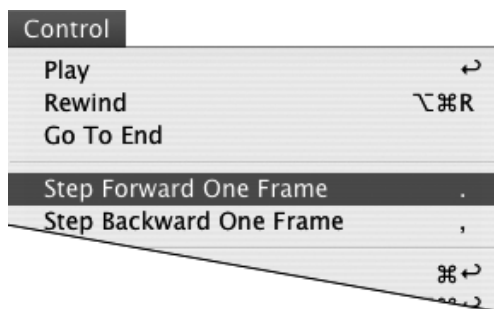
**С** Можно быстро перематывать ролик в любом направлении. Для этого буксируйте точку воспроизведения вперед или назад по временной шкале. Flash отображает на арене содержимое того кадра, на который указывает точка воспроизведения.



**Рис. 8.28.** Чтобы открыть окно пульта управления, выберите из меню пункт Window ⇒ Toolbars ⇒ Controller.



**Рис. 8.29.** В окне пульта управления имеются привычные магнитофонные кнопки для управления воспроизведением Flash-роликов.



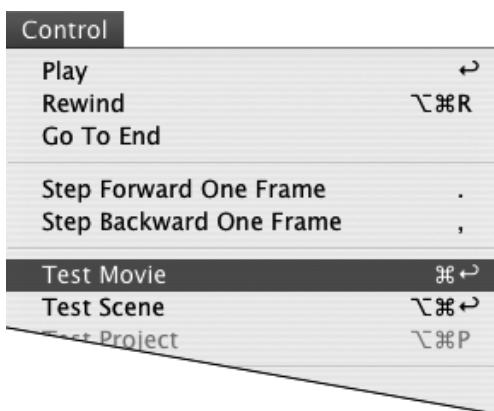
**Рис. 8.30.** В меню Control имеются команды для воспроизведения ролика целиком и покадрового просмотра.

Control	
Play	Enter
Rewind	Ctrl+Alt+R
Go To End	
Step Forward One Frame	.
Step Backward One Frame	,
Test Movie	Ctrl+Enter
Test Scene	Ctrl+Alt+Enter
Test Project	Ctrl+Alt+P
Delete ASO Files	
Delete ASO Files and Test Movie	
Loop Playback	
Play All Scenes	
Enable Simple Frame Actions	Ctrl+Alt+F
Enable Simple Buttons	Ctrl+Alt+B
✓ Enable Live Preview	
Mute Sounds	Ctrl+Alt+M

**Рис. 8.31.** В режиме Control ⇒ Loop Playback команда Play прокручивает ролик по кругу.

## Чтобы просмотреть все кадры в редакторе Flash:

- Чтобы проиграть кадры один раз, выполните команду **Control ⇒ Play** (Управление ⇒ Воспроизвести) или нажмите **Enter**. Flash будет показывать кадры по порядку, начиная с текущего и до конца ролика. В течение этого времени место команды **Play** в меню **Control** занимает команда **Stop**, которая позволяет в любой момент остановить воспроизведение.

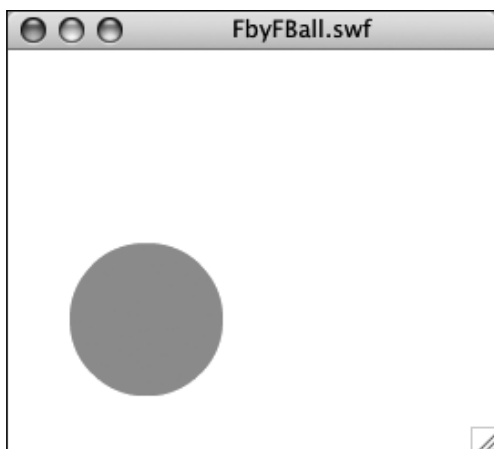


**С** Для непрерывного воспроизведения выполните команду **Control ⇒ Loop Playback** (Управление ⇒ Цикл воспроизведения) (рис. 8.31). Теперь команда **Play** будет прокручивать ролик по кругу, пока вы не выполните команду **Stop**.

## Чтобы просмотреть ролик в Flash-плеере:

- Выполните команду **Control ⇒ Test Movie** (Управление ⇒ Тестировать ролик) или нажмите комбинацию клавиш **⌘-Enter** (Mac) либо **Ctrl+Enter** (Windows) (рис. 8.32).

Flash экспортирует ролик в SWF-файл и открывает его в отдельном окне. SWF-файл сохраняется в той же папке, что и исходный Flash-файл, и имеет такое же имя, но расширение SWF вместо FLA.



**Рис. 8.32.** Выполните команду Control ⇒ Test Movie, чтобы просмотреть свой ролик в Flash-плеере.

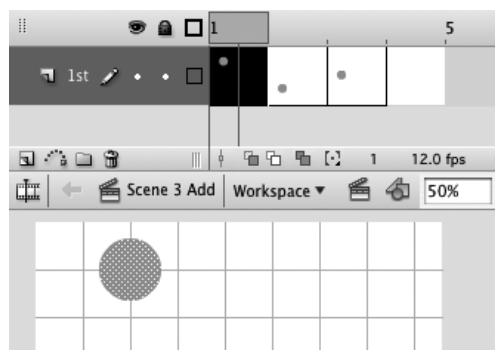
## Сглаживание анимации за счет добавления опорных кадров

Трехкадровая анимация прыгающего мяча, созданная в предыдущем упражнении, — довольно грубый пример, мяч движется рывками и слишком быстро. Чтобы сделать движение более плавным, нужно создать дополнительные кадры, фиксирующие положение мяча в разные моменты.

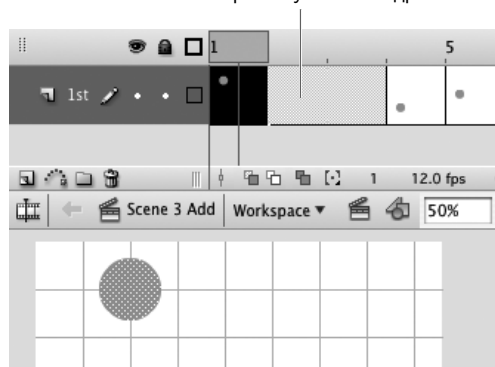
В предыдущем упражнении мяч двигался по арене сверху вниз с одним промежуточным положением. А сейчас мы увеличим число шагов до трех.

### Чтобы добавить опорные кадры в существующую анимацию:

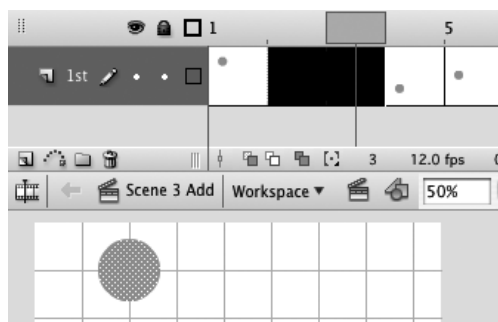
1. На временной шкале с тремя кадрами анимации прыгающего мяча выберите опорный кадр 1.
2. Дважды выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame**. Flash создаст новые промежуточные кадры в позициях 2 и 3, а опорные кадры, на которых мяч находится в середине и внизу арены, переместит в позиции 4 и 5 (рис. 8.33).
3. Выберите на временной шкале кадры 2 и 3.



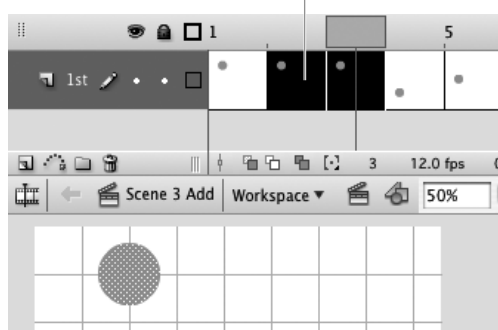
Новые промежуточные кадры



**Рис. 8.33.** Когда выбран опорный кадр 1, двукратное выполнение команды **Insert** ⇒ **Timeline** ⇒ **Frame** вставляет два промежуточных кадра после первого опорного и сдвигает следующий опорный кадр (мяч внизу арены) из позиции 2 в позицию 4.

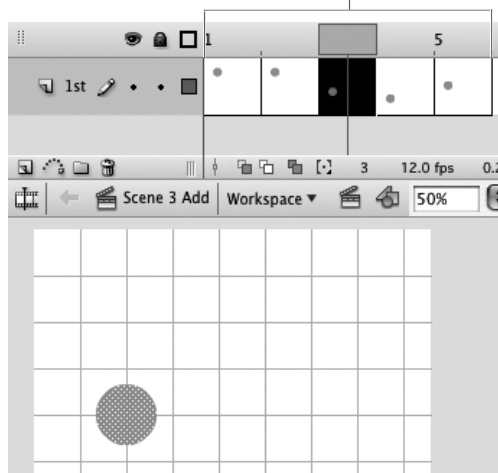


Дубликаты опорного кадра 1



**Рис. 8.34.** Команда **Modify** ⇒ **Timeline** ⇒ **Convert to Keyframes** преобразует выбранные промежуточные кадры в опорные, дублируя содержимое предшествующего опорного кадра.

Предварительный просмотр кадров в контексте



**Рис. 8.35.** Переместите мяч в опорных кадрах 2 и 3, чтобы первый отскок стал более плавным.

4. Выполните команду **Modify** ⇒ **Timeline** ⇒ **Convert to Keyframes**.

Flash преобразует промежуточные кадры в опорные, продублировав в них содержимое кадра 1 (рис. 8.34).

5. Выберите на временной шкале опорный кадр 2 и измените положение мяча на арене. Чтобы лучше видеть, куда поместить мяч, можете воспользоваться сеткой; расположите мяч, отступив примерно на треть высоты от верхнего края арены.
6. Выберите на временной шкале опорный кадр 3 и измените положение мяча на арене (рис. 8.35). Отступите примерно на две трети высоты от верхнего края арены.
7. Протестируйте получившуюся анимацию любым из описанных выше способов. Первая половина движения стала более плавной. Добавьте еще кадры, чтобы сделать плавной и вторую половину.

## Использование восковок

В предыдущем разделе вы меняли положение круга в попытке имитировать плавное движение прыгающего мяча. Чтобы облегчить решение подобных задач, Flash предлагает механизм восковок, позволяющий посмотреть на круг в контексте окружающих кадров.

В этом режиме на текущий кадр накладываются освещенные или контурные изображения соседних кадров. Сколько брать соседних кадров, решаете вы сами. Кнопки для включения и выключения различных типов восковок находятся в нижней части панели временной шкалы, в ее полосе состояния.

### Чтобы включить показ восковок:

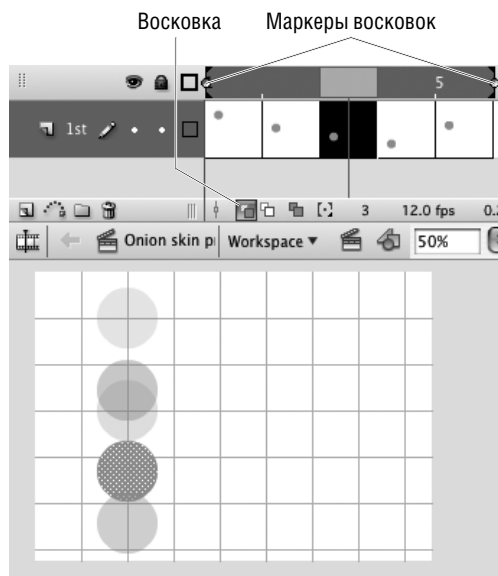
- Нажмите кнопку **Onion Skin** (Шлейф) в полосе состояния временной шкалы.

На арене появляется освещенное содержимое соседних кадров (рис. 8.36). Редактировать освещенную графику нельзя, доступны лишь полноцветные изображения в текущем кадре.

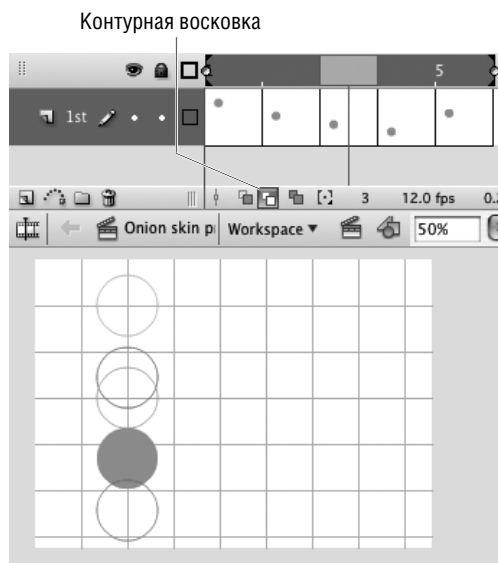
### Чтобы выключить показ контурных восковок:

- Нажмите кнопку **Onion Skin Outlines** (Контур шлейфа) в полосе состояния временной шкалы.

На арене появляются контуры содержимого соседних кадров (рис. 8.37). Редактировать контурную графику нельзя, доступны лишь сплошные изображения в текущем кадре.



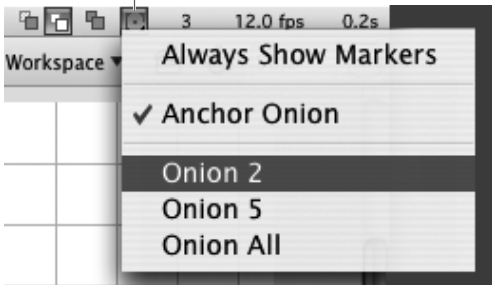
**Рис. 8.36.** В режиме показа восковок Flash отображает содержимое нескольких кадров, осветляя все, что не принадлежит текущему кадру. Маркеры восковок на временной шкале показывают, сколько соседних кадров будет взято для этой цели.



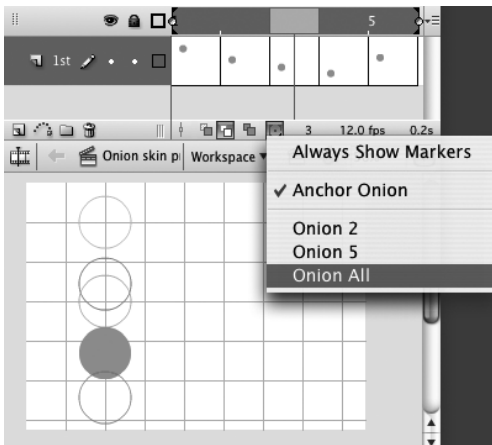
**Рис. 8.37.** В режиме показа контурных восковок Flash отображает содержимое нескольких кадров, но для всего, что не принадлежит текущему кадру, показываются только контуры.



Кнопка «Модифицировать маркеры восковок»



**Рис. 8.38.** Выпадающее меню *Modify Onion Markers* позволяет задавать число кадров, накладываемых на текущий в режиме восковок.



**Рис. 8.39.** Если в меню *Modify Onion Markers* выбран пункт *Onion All*, то Flash пометит маркерами восковок все кадры на временной шкале.

### Чтобы задать число соседних кадров, показываемых в режиме восковок:

1. Нажмите кнопку **Modify Onion Markers** (Изменить маркеры шлейфа) в полосе состояния временной шкалы. В выпадающем меню есть команды для установки числа маркеров (рис. 8.38).
2. Чтобы задать число соседних кадров, отображаемых поверх текущего в режиме восковок, выполните одно из следующих действий:
  - чтобы взять по два кадра с каждой стороны от текущего, выберите пункт **Onion 2** (Шлейф 2);
  - чтобы взять по пять кадров с каждой стороны от текущего, выберите пункт **Onion 5** (Шлейф 5);
  - чтобы включить все кадры ролика, выберите пункт **Onion All** (Все кадры в шлейфа) (рис. 8.39).

При перемещении точки воспроизведения по временной шкале Flash сдвигает и маркеры восковок. На текущий кадр всегда накладываются восковки как предшествующих, так и последующих кадров.

**C**

В режиме восковок не показывается содержимое заблокированных слоев.

**C**

Чтобы уменьшить или увеличить число соседних кадров, накладываемых в режиме восковок, буксируйте маркеры восковок вдоль временной шкалы.

**C**

Можно предотвратить перемещение маркеров восковок при выборе нового кадра на временной шкале. Установите маркеры так, чтобы они помечали те кадры, которые вы хотите просматривать вместе. В меню модификации маркеров восковок выберите пункт **Anchor Onion** (Привязка шлейфа). Пока выбранный кадр не выходит за пределы зафиксированного диапазона, для всех входящих в этот диапазон кадров будет действовать режим восковок.

## Редактирование нескольких кадров

Решив изменить местоположение анимированного элемента, вы должны сделать это во всех опорных кадрах, где он присутствует. Перемещать элементы по отдельности в каждом кадре не только утомительно, но и опасно. Стоит забыть один-единственный кадр, и элементы окажутся рассинхронизированными. Flash решает эту проблему, позволяя перемещать элементы сразу в нескольких кадрах. Те же маркеры, что обозначают включение кадров в шлейф восковок, показывают, что эти кадры разрешено изменять одновременно в режиме редактирования нескольких кадров.

### Чтобы переместить анимированную графику на арене:

1. Откройте документ с покадровой анимацией прыгающего мяча.
2. В полосе состояния временной шкалы установите режим редактирования нескольких кадров (рис. 8.40).

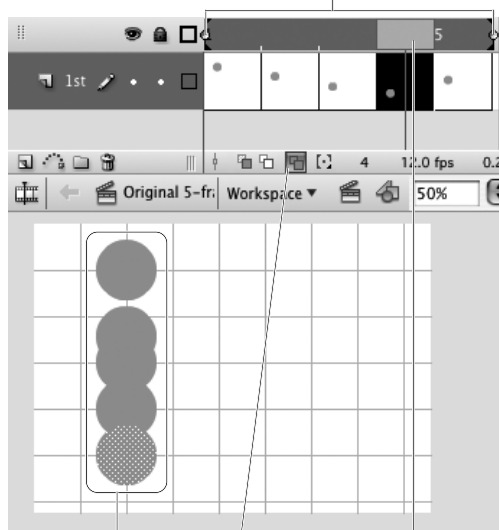
Flash отображает все графические объекты из всех кадров, помеченных маркерами восковок, и позволяет их редактировать.

3. Из меню **Modify Onion Markers** выберите пункт **Onion All**.

Теперь мяч виден на каждом этапе движения, и любой этап можно редактировать.

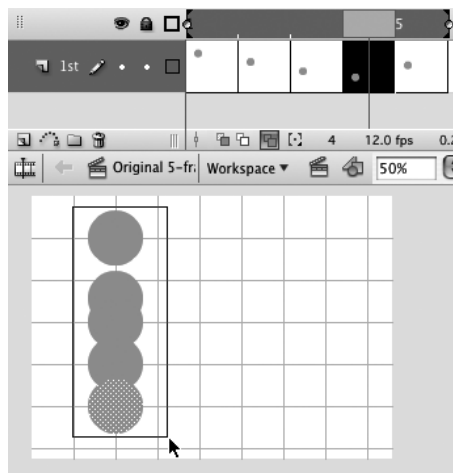
4. С помощью инструмента «Стрелка» нарисуйте прямоугольник, охватывающий все видимые на арене мячи (рис. 8.41).

Кадры, доступные для редактирования

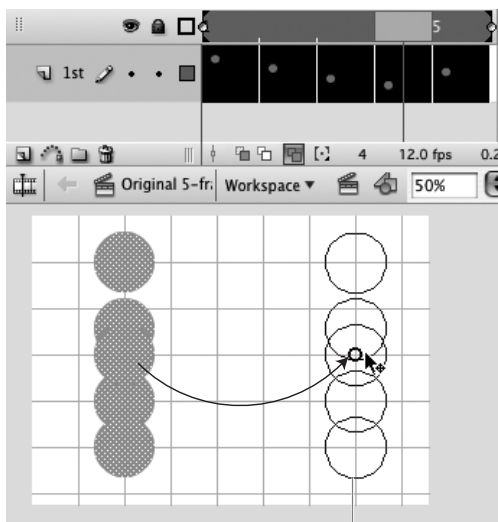


Объекты, доступные для редактирования      Кнопка «Редактировать несколько кадров»      Текущий кадр

**Рис. 8.40.** В режиме редактирования нескольких кадров Flash отображает и позволяет редактировать все графические объекты из кадров, помеченных маркерами восковок. Это дает возможность одновременно переместить всю анимированную графику во всех опорных кадрах.



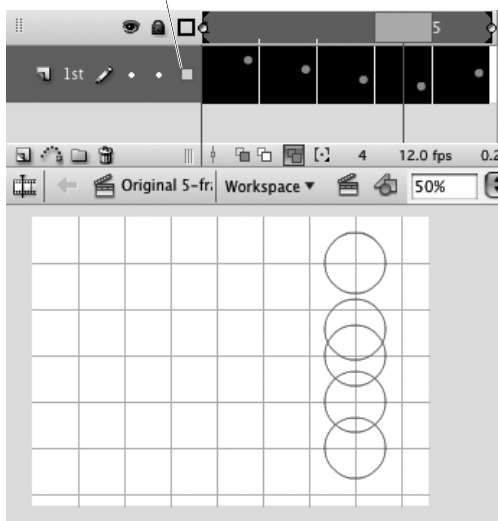
**Рис. 8.41.** В режиме редактирования нескольких кадров можно воспользоваться инструментом «Стрелка» для выбора графических объектов в любом кадре, помеченном маркерами восковок.



При перетаскивании выбранных графических элементов отображаются контуры, отмечающие новое положение

**Рис. 8.42.** В режиме редактирования нескольких кадров можно одной операцией переместить анимированную графику во всех опорных кадрах.

Переключатель режима показа контуров



**Рис. 8.43.** Включите режим показа контуров, чтобы было удобнее работать с графическими объектами, взятыми из нескольких кадров.

5. Перетащите выбранные объекты к противоположному краю арены (рис. 8.42).

Всего несколько шагов — и вы изменили положение прыгающего мяча. (А представьте, сколько пришлось бы потрудиться, чтобы по отдельности выбрать каждый опорный кадр, переместить в этом кадре круг, выбрать следующий опорный кадр, точно совместить круги в обоих кадрах и т.д.).

**С**

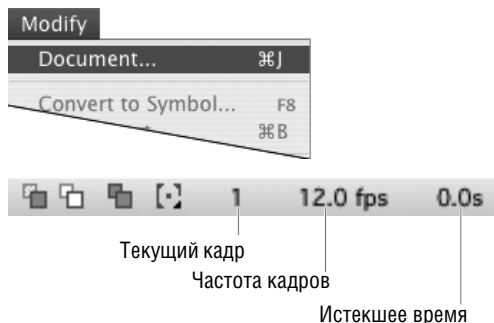
В режиме редактирования нескольких кадров Flash уже не отображает восксовки для опорных кадров, однако в промежуточных кадрах видны и восксовки, и интерполированное содержимое. Если вам не нравится видеть на арене сплошные формы из нескольких кадров, можете включить режим показа контуров в области свойств слоя на панели временной шкалы (рис. 8.43).

## Задание частоты кадров

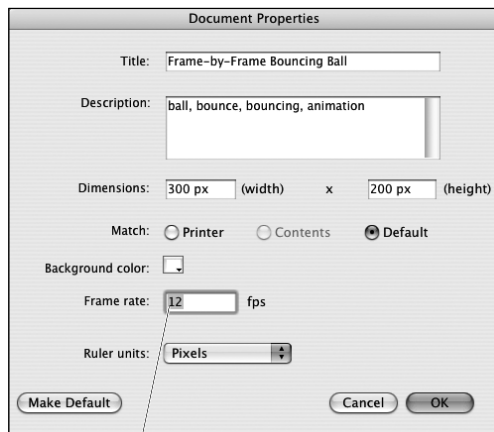
Во Flash частота кадров задается сразу для всего ролика. Делается это в окне свойств документа.

### Чтобы задать частоту кадров:

1. Чтобы открыть окно свойств документа, выполните любое из следующих действий:
  - выберите из меню пункт **Modify** ⇒ **Document** (Изменить ⇒ Документ) или нажмите комбинацию клавиш **⌘-J** (Mac) либо **Ctrl-J** (Windows);
  - в полосе состояния временной шкалы дважды щелкните по числу, равному частоте кадров (рис. 8.44).
2. В окне свойств документа введите значение в поле Frame Rate (Частота кадров) (рис. 8.45).
3. Нажмите **ОК**.



**Рис. 8.44.** Для открытия окна свойств документа выберите из меню пункт **Modify** ⇒ **Document** (верхний рисунок) дважды щелкните по числу, равному частоте кадров, на временной шкале (нижний рисунок).



**Рис. 8.45.** В окне свойств документа введите значение в поле Frame Rate. По умолчанию во Flash принята частота 12 кадров/с.

### О частоте кадров

Иллюзия анимации основывается на способности человеческого мозга заполнять разрывы. Если вам демонстрируется серия быстро сменяющих друг друга изображений, то мозг будет воспринимать ее как непрерывное движение. В анимации отображать последовательность образов нужно достаточно быстро для того, чтобы убедить мозг в том, что он видит единое изображение.

Частота кадров как раз и управляет тем, насколько быстро Flash демонстрирует изображения. Если они поступают слишком быстро, то картинка будет казаться размытой. Если слишком медленно, зритель станет воспринимать каждый кадр как отдельное изображение, и ему будет казаться, что картинка дергается. Примите еще во внимание, что, работая с Flash, вы, скорее всего, планируете доставлять ролик через Интернет, поэтому характеристики системы конечного пользователя заранее не известны. Это означает, что вы, возможно, не сумеете доставлять кадры достаточно быстро.

Стандартная для фильмов частота составляет 24 кадра/с. В случае графической анимации, доставляемой через Интернет, оптимальной считается частота 12 кадров/с (это значение принимается во Flash по умолчанию). Если вы уверены, что целевая аудитория пользуется низкоскоростными соединениями, то попробуйте уменьшить частоту до 10 кадров/с. Напротив, если ожидается использование высокоскоростных соединений, увеличьте ее до 15 кадров/с. Современные широкополосные каналы связи и средства, встроенные в Flash Player 9, позволяют увеличить частоту еще больше. Если вы точно знаете, что целевая аудитория располагает такими возможностями, то попробуйте довести частоту до 24 кадров/с. В тех случаях, когда необходима очень плавная анимация или ролик содержит видео, можно даже установить частоту 30 кадров/с.

### Анимация с переменной скоростью

Хотя частота кадров в ролике постоянна, но части анимации можно ускорить или замедлить, изменяя число кадров, на которых разворачивается действие. Чтобы растянуть участок анимации во времени, добавьте дополнительные опорные или промежуточные кадры. В примере с прыгающим мячом можно замедлить падение (скажем, отведя под него пять кадров) и ускорить отскок (три кадра). В плавной покадровой анимации много опорных кадров, в каждом из которых положение мяча лишь немного отличается от предыдущего. Однако добавление опорных кадров увеличивает размер файла. Иногда для замедления действия можно обойтись и промежуточными кадрами, которые почти не влияют на размер ролика.

#### Чтобы добавить промежуточные кадры:

1. Откройте или создайте ролик, в котором мяч подпрыгивает в пяти кадрах.  
В опорном кадре 1 мяч находится у верхнего края арены, в опорных кадрах 2 и 3 он опускается, в кадре 4 находится у нижнего края арены, а в кадре 5 — посередине. (Пошаговые инструкции см. в разделе «Сглаживание анимации за счет добавления опорных кадров» выше.)
2. Выполните команду **File** ⇨ **Save As** и сделайте копию этого файла. Назовите ее, например, BounceSlower.

3. Выберите на временной шкале опорный кадр 1.
4. Щелкните кнопкой мыши с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows), чтобы получить контекстное меню кадров.
5. Выберите из него пункт **Insert Frame** (Вставить кадр).

Flash вставляет промежуточный кадр в позицию 2 и сдвигает остальные кадры вправо (рис. 8.46).

6. Повторите шаги 3-5 для второго и третьего опорного кадра.

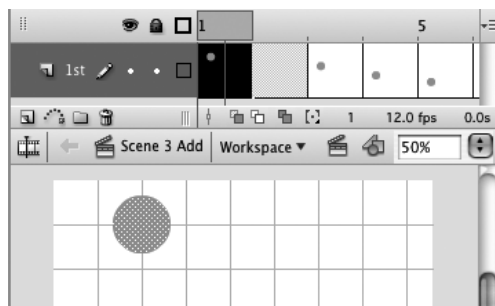
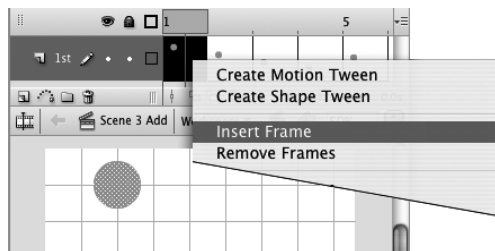
В результате опорные кадры окажутся в позициях 1, 3, 5, 7 и 8 (рис. 8.47).

7. Выполните команду **Control** ⇒ **Test Movie**.

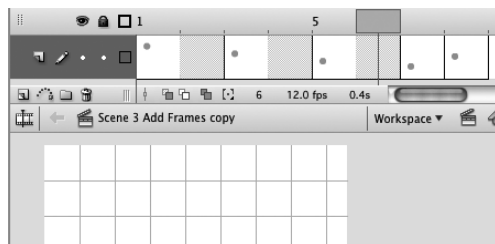
Flash экспортирует ролик в SWF-файл и открывает его в Flash-плеере. Воспроизведите сначала обычную пятикадровую анимацию прыгающего мяча, а потом только что созданную (с именем BounceSlower). Легко заметить, что анимация с добавленными промежуточными кадрами воспринимается совсем по-другому. Действие кажется замедленным.

**С**

Не забывайте, что этот пример приведен только для иллюстрации. На практике злоупотреблять подобной техникой не стоит. Если просто добавить много промежуточных кадров, то действие станет слишком медленным, и пропадет иллюзия движения.



**Рис. 8.46.** Выберите кадр и выполните команду Insert Frame из контекстного меню (верхний рисунок). Flash вставит промежуточный кадр сразу после выбранного (нижний рисунок).



**Рис. 8.47.** После того как между начальными кадрами вставлены промежуточные, первая часть анимации стала медленнее второй.

# АНИМАЦИЯ ПОСРЕДСТВОМ ИНТЕРПОЛЯЦИИ ДВИЖЕНИЯ

## 9

### Интерполяция движения и графические контейнеры

Flash требует, чтобы графические объекты, используемые в процессе интерполяции движения, были заключены в некий контейнер. Некоторые инструменты рисования создают такие контейнеры автоматически, в частности: текст, овал, все инструменты в режиме рисования объектов, а также прямоугольный и овальный примитивы. Другой способ поместить графические элементы в подходящий контейнер — сгруппировать их или преобразовать в символ.

Анимация движения путем изменения форм вручную (от кадра к кадру) — очень трудоемкое занятие. Adobe Flash CS3 Professional сокращает число рисуемых вручную кадров за счет процедуры автоматического построения промежуточных изображений. Эта процедура называется *интерполяцией* (tweening). В главе 8 вы создали трехкадровую анимацию прыгающего мяча, изменяя его положение в каждом из трех опорных кадров. Затем вы научились растягивать анимацию за счет вставки промежуточных кадров с тем же содержанием, что в предшествующем опорном. Используя интерполяцию, вы примерно также создаете опорные кадры, а Flash строит переходные изображения в виде промежуточных кадров.

Для построения промежуточных изображений Flash генерирует последовательность постепенных изменений; они достаточно просты и могут быть описаны математически. Flash применяет интерполяцию двух видов: движения и форм. В этой главе мы рассмотрим интерполяцию движения, а интерполяция форм составляет содержание главы 10.

В основу обоих видов интерполяции положен один и тот же принцип. Вы указываете начало и конец последовательности, помещая графические элементы в опорные кадры. Затем вы просите Flash описать переход между ними за данное число шагов, то есть вставить между опорными кадрами некоторое число промежуточных. Flash строит требуемую последовательность изображений с инкрементными изменениями.

## Создание прыгающего мяча с помощью интерполяции движения

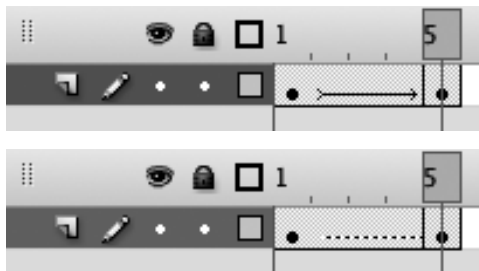
Чтобы создать интерполяцию движения, необходимо поместить подходящее графическое содержимое в начальный и конечный опорные кадры, а затем для последовательности первого опорного кадра установить значение Motion (Движение) свойства Tween (Анимация). Для интерполяции движения содержимым должен быть графический объект (объект-рисунок, примитивная форма, группа, символ или текстовое поле). Свойство Tween задается на вкладке свойств кадра в инспекторе свойств. С помощью интерполяции движения мы создадим такую же анимацию прыгающего мяча, как в главе 8, но на этот раз потребуем, чтобы положение мяча вычисляла программа.

### Какие кадры являются частью интерполяции?

На временной шкале интерполированные кадры помечены визуальными индикаторами. Flash рисует стрелку поперек автоматически сгенерированных кадров (рис. 9.1). (Отметим, что стрелка не видна, если установлен режим предварительного просмотра кадров, в том числе в контексте.)

Чтобы различить кадры, созданные в режиме интерполяции движения и в режиме интерполяции форм, Flash применяет цветовое кодирование. Если установлен режим штриховки кадров (выбирается из всплывающего меню в правом углу временной шкалы), то штриховка кадров, созданных в режиме интерполяции движения, будет голубовато-красной (Mac) или голубовато-серой (Windows). Если же режим штриховки не активен, то такие кадры будут белыми (Mac) или заполнены узором (Windows), при этом стрелка, обозначающая наличие интерполяции, станет не черной, а красной. Кадры, созданные в режиме интерполяции форм, Flash заштриховывает бледно-зеленым цветом (если режим штриховки установлен) или перечеркивает бледно-зеленой стрелкой (в противном случае).

Если кадр перечеркнут пунктирной линией, значит, была затребована интерполяция (движения или форм), но что-то случилось, и завершить ее Flash не сумел. В этом случае говорят о *битой* интерполяции.

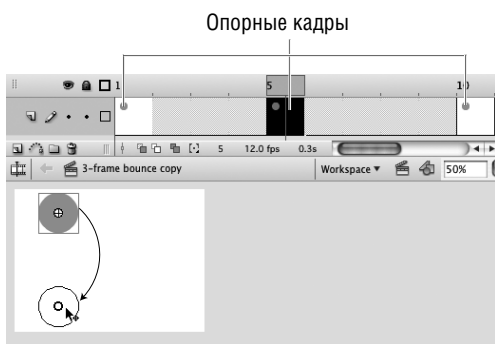


**Рис. 9.1.** Сплошная линия, перечеркивающая последовательность кадров, говорит о том, что они успешно сгенерированы в процессе интерполяции (верхний рисунок). Если же линия пунктирная, значит, при генерации возникли ошибки (нижний рисунок).

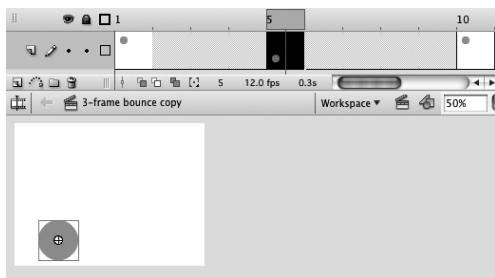




**Рис. 9.2.** Интерполируемые элементы должны находиться в контейнере. Самый лучший способ создать контейнер — воспользоваться символом. Flash может построить интерполяцию движения для любого символа: клипа, кнопки или графики.



Переместить графический объект



Графический объект в новой позиции

**Рис. 9.3.** Начальные опорные кадры для интерполяции движения создаются так же, как в случае покадровой анимации. Разница только в том, что необходимо использовать графический объект (объект-рисунок, примитивная форма, группа, символ или текстовое поле), а свойство Tween для последовательности опорного кадра устанавливать в Motion.

### Чтобы подготовить опорные кадры для интерполяции движения:

1. Создайте новый документ и назовите его MotionTweenBounce. В документе будет один слой и один опорный кадр.
2. Выберите кадр 1 на временной шкале.
3. Выберите инструмент «Овал» и задайте для него цвет No Color.
4. Нарисуйте круг у верхнего края арены. Он будет представлять мяч.
5. Выберите круг и выполните команду **Modify ⇒ Convert to Symbol** (Изменить ⇒ Преобразовать в символ).  
Открывается окно Convert to Symbol. Можно задать любой тип символа, но для этого упражнения выберите Graphic (рис. 9.2).
6. Введите имя символа, например Ball, и нажмите **OK**.
7. На временной шкале выберите кадр 5 и выполните команду **Insert ⇒ Timeline ⇒ Keyframe**. Flash создаст новый опорный кадр, в котором дублируется содержимое предшествующего.
8. Выберите кадр 10 и выполните команду **Insert ⇒ Timeline ⇒ Keyframe**.
9. Выберите кадр 5 и перетащите мяч к нижнему краю арены.

Итак, вы подготовили все для покадровой анимации, как и в главе 8. В опорном кадре 1 мяч находится в верхней точке, в опорном кадре 5 — в нижней, а в опорном кадре 10 — снова в верхней (рис. 9.3). Чтобы завершить интерполяцию, необходимо задать свойство Tween для промежуточных кадров в каждой последовательности.

**С**

Овал для этого упражнения можно было бы создать как объект-рисунок или как примитивную форму (либо создать как комбинированную форму, а потом включить ее в группу). Тогда шаги 5 и 6 можно было бы опустить. Однако эффективнее всего для интерполяции движения применять символы.

### Чтобы установить значение Motion для свойства Tween последовательности кадров:

1. Чтобы выполнить интерполяцию движения для фазы падения мяча, выберите на временной шкале любой кадр из последовательности первого опорного кадра, то есть кадр 1, 2, 3 или 4 (рис. 9.4).

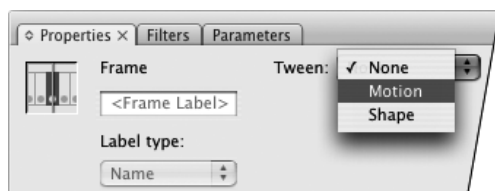
Flash автоматически выберет символ мяча на арене. Для задания интерполяции движения анимируемый символ должен быть выбран.

2. На вкладке свойств кадра в инспекторе свойств выберите из списка **Tween** (Анимация) пункт Motion (Движение) (рис. 9.5).

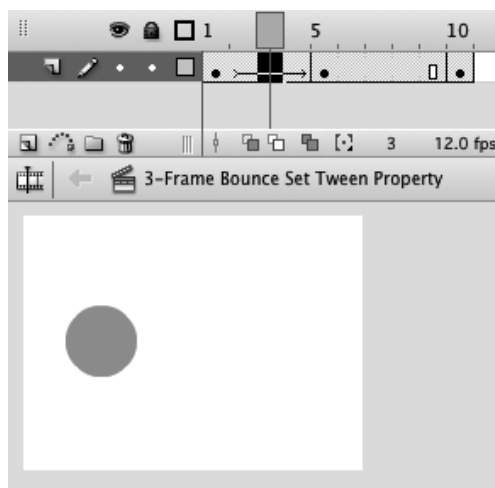
Появляется набор параметров. Flash определяет, что анимация будет занимать кадры с 1 по 4 и отображает это на временной шкале (рис. 9.6) (см. врезку «Какие кадры являются частью интерполяции» выше). В этих промежуточных кадрах больше не отображается содержимое предшествующего опорного кадра; вместо этого вы видите сгенерированные изменения. Автоматически созданное содержимое защищено, так что выбрать его невозможно (щелчок по графическому объекту в промежуточном кадре, который был создан в режиме интерполяции, — это все равно, что щелчок по пустому месту на арене). Впрочем, вы можете перетаскивать такое содержимое, но при этом промежуточный кадр будет преобразован в опорный.



**Рис. 9.4.** Для задания свойства Tween выберите любой кадр из последовательности опорного кадра.



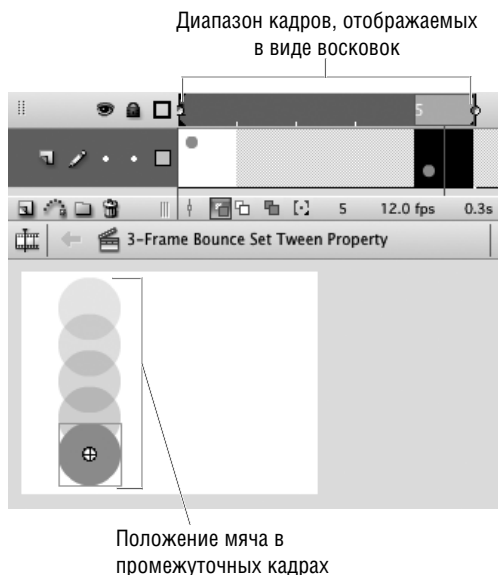
**Рис. 9.5.** Чтобы на вкладке Properties в инспекторе свойств появились параметры интерполяции движения, выберите пункт Motion из списка Tween.



**Рис. 9.6.** Flash индицирует на временной шкале, какие кадры были интерполированы. Голубая штриховка и сплошная стрелка означают, что интерполяция была завершена успешно.



**Рис. 9.7.** Для анимации прыгающего мяча нужны две последовательности интерполяции движения: одна для фазы опускания, другая для фазы отскока.



**Рис. 9.8.** Для предварительного просмотра позиций объекта на сгенерированных кадрах включите режим восковок.

3. Чтобы определить интерполяцию движения для фазы отскока мяча, выберите на временной шкале любой из кадров, принадлежащих второй последовательности, — 5, 6, 7, 8 или 9.
4. Повторите шаг 2. Flash создает другую интерполяцию для второй фазы половины движения (рис. 9.7).

**С**

Если в меню просмотра кадра (в правом углу временной шкалы) установлен режим Preview или Preview in Context, то линия со стрелкой, обозначающая интерполяцию, отсутствует. И инкрементных изменений, сгенерированных Flash, вы тоже не увидите. Но если включить режим восковок, то все промежуточные кадры появятся на арене (рис. 9.8).

**С**

Как ни странно, но, хотя вы не можете выбрать символ в промежуточном кадре, редактировать его можно. Двойной щелчок по символу в промежуточном кадре открывает его в режиме редактирования. Впрочем, для объектов-рисунков и групп этот фокус не работает; редактировать эти графические объекты можно, только находясь в опорном кадре. Если интерполяция была подготовлена с использованием примитивных форм, то двойной щелчок по примитиву в промежуточном кадре открывает диалоговое окно с предупреждением о том, что для редактирования примитива его необходимо сначала преобразовать в объект-рисунок. Если вы нажмете **ОК**, то все равно не попадете в режим редактирования, но Flash-таки преобразует примитив в последовательности начального опорного кадра в объект-рисунок. Но будьте осторожны, выполняя двойной щелчок; если вы хоть чуть-чуть сдвинете сгенерированную графику в промежуточном кадре, то Flash создаст новый опорный кадр.

## Добавление опорных кадров в интерполированную последовательность

После того как интерполяция движения успешно завершена, Flash будет добавлять новые опорные кадры, если вы измените положение сгенерированного графического объекта в промежуточном кадре. Можно добавить опорные кадры и самостоятельно, выполнив команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**.

### Чтобы добавить опорные кадры вследствие перемещения сгенерированного объекта:

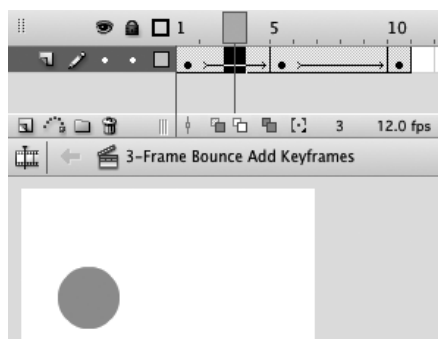
1. Создайте десятикадровую интерполяцию движения прыгающего мяча, как в предыдущем упражнении.
2. Выберите на временной шкале кадр 3. На арене вы увидите символ мяча в одной из сгенерированных Flash промежуточных позиций.
3. Выберите инструмент «Стрелка» на панели инструментов.
4. Перетащите мяч чуть вправо.

Flash вставит в позицию 3 новый опорный кадр, а предшествующую сгенерированную анимацию из пяти кадров расцепит на две (рис. 9.9). В новом опорном кадре содержится еще один экземпляр символа мяча.

### Чтобы добавить опорные кадры командой:

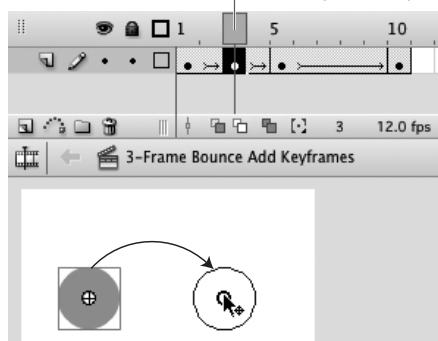
1. Продолжая работу с документом из предыдущего упражнения, выберите на временной шкале кадр 7.
2. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**.

На арене появится новый экземпляр символа в той же позиции, которую Flash сгенерировал для него в выбранном промежуточном кадре. Если хотите, можете переместить символ в другое место.

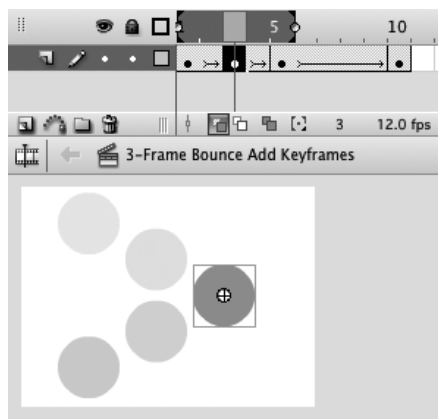


В выбранном кадре мяч находится в одной из сгенерированных позиций

Добавленный опорный кадр

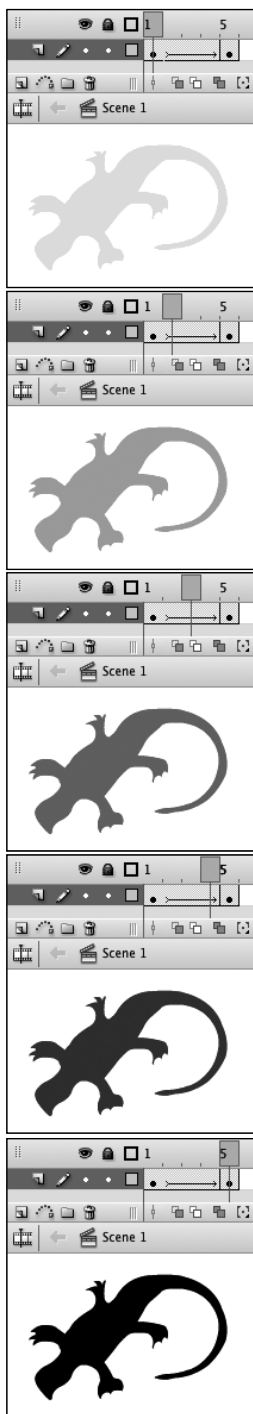


Перемещение мяча создает новый опорный кадр



Восковки других сгенерированных позиций мяча

**Рис. 9.9.** Перемещение мяча в промежуточном кадре, который входит в сгенерированную последовательность, приводит к созданию нового опорного кадра.



**Рис. 9.10.** При интерполяции движения можно изменять не только положение, но и цвет символа. В промежуточных кадрах Flash генерирует переходные цвета.

## Эффекты анимации цвета

Интерполяция не исчерпывается изменением положения элемента на арене. Можно также выполнять анимация путем изменения свойства **Color** символа. (Этот прием работает только для символов; выполнять интерполяцию для других графических объектов можно, но не с помощью цвета.)

### Чтобы менять цвет символа во времени:

1. В новом документе или в новом слое существующего документа поместите на арену экземпляр какого-нибудь символа в опорном кадре 1.
2. Выберите на временной шкале кадр 5 и выполните команду **Insert** ⇨ **Timeline** ⇨ **Keyframe**. Flash продублирует содержимое кадра 1 в новом опорном кадре.
3. Сделав кадр 5 текущим, выберите символ.
4. Чтобы изменить цвет символа, воспользуйтесь элементом управления цветом на вкладке **Properties** в инспекторе свойств (подробнее о задании цвета символа см. главу 7).
5. Выберите любой кадр в последовательности первого опорного кадра (1, 2, 3 или 4).
6. На вкладке **Properties** в инспекторе свойств выберите пункт **Motion** из списка **Tween** (Анимация). Flash перекрасит графический объект во всех трех промежуточных кадрах (рис. 9.10).

**С**

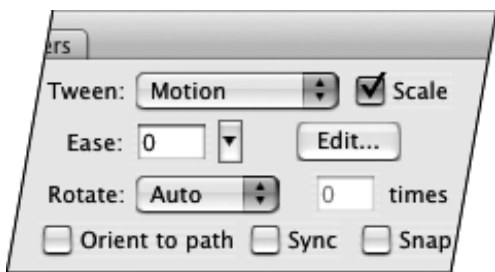
Можно анимировать изменение прозрачности символа, чтобы заставить его постепенно проявляться и исчезать.

## Анимация графики с изменяющимся размером

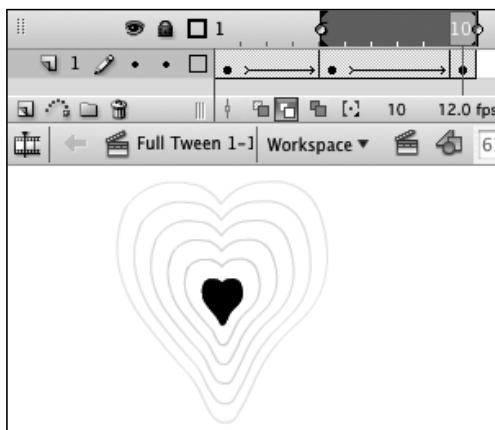
Можно анимировать изменения размера графического объекта при переходе от одного опорного кадра к следующему, выполняя интерполяцию свойств Horizontal Scale или Vertical Scale. Чтобы объект увеличивался или уменьшался, необходимо отметить флажок **Scale** (Масштаб) на вкладке свойств кадра в инспекторе свойств. Анимировать изменение размера можно для графических объектов любого типа, но для простоты мы ограничимся символами.

### Чтобы создать анимацию увеличивающегося или уменьшающегося объекта:

1. В новом документе или в новом слое существующего документа поместите на арену экземпляр какого-нибудь символа в опорном кадре 1 (о создании и использовании символов см. главу 7).
2. Чтобы создать опорный кадр, отмечающий конец фазы увеличения, выберите на временной шкале кадр 5, а затем выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**.  
Flash продублирует символ из опорного кадра 1 в новом опорном кадре.
3. Выберите любой кадр в последовательности первого опорного кадра (1, 2, 3 или 4).
4. Задайте свойство Tween этой последовательности равным Motion. В кадрах 2-4 появляется стрелка и цветовое кодирование, свидетельствующие о том, что они сгенерированы.
5. Установив точку воспроизведения на кадр 5, выберите экземпляр символа и увеличьте его (о том, как изменять размеры графических объектов, см. главу 4).



**Рис. 9.11.** Чтобы увеличить размер графического объекта равными шагами, установите значение Motion свойства Tween и отметьте флажок Scale. Эти свойства задаются на вкладке свойств кадра в инспекторе свойств.



**Рис. 9.12.** Чтобы анимировать уменьшение объекта, сделайте его размер в конечном опорном кадре меньше, чем в начальном. Посмотреть, какие размеры Flash сгенерировал в промежуточных кадрах, можно, включив показ восковок.

6. Выберите на временной шкале любой кадр в последовательности первого опорного кадра (1, 2, 3 или 4).

7. Отметьте флажок **Scale** (Масштаб) на вкладке **Properties** в инспекторе свойств (по умолчанию он уже отмечен) (рис. 9.11).

В кадрах с 1 по 5 Flash будет увеличивать размер этого экземпляра равными шагами.

8. Чтобы добавить опорный кадр, завершающий фазу уменьшения объекта, выберите кадр 10 и нажмите клавишу **F6**.

Flash продублирует символ из опорного кадра 5 в новом опорном кадре.

9. Выберите на временной шкале любой кадр в последовательности второго опорного кадра (5, 6, 7, 8 или 9).

10. Задайте свойство Tween этой последовательности равным Motion (см. шаг 4). В кадрах 5-9 появляется стрелка и цветовое кодирование, свидетельствующие о том, что они сгенерированы. Флажок **Scale** уже отмечен.

11. Установив точку воспроизведения на кадр 10, выберите экземпляр символа и уменьшите его.

Flash сгенерирует последовательность кадров, в котором размер этого экземпляра уменьшается пятью равными шагами (рис. 9.12).

**С**

Если вы не изменяли настроек на вкладке **Properties** в инспекторе свойств, то флажок **Scale** так и остается выбранным, и Flash будет обновлять сгенерированную последовательность всякий раз, как изменяется содержимое любого из ее опорных кадров. Чтобы уточнить размер масштабируемой графики, необязательно держать вкладку свойств кадра открытой.

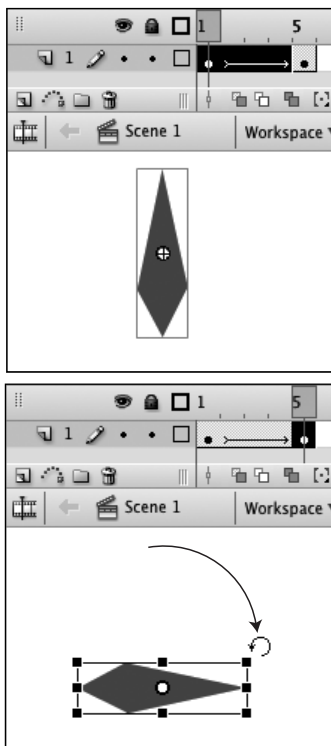
## Вращение графических элементов

Для анимации крутящегося объекта нужно выполнить интерполяцию свойства Rotation (Поворот). Если интерполяция движения включает вращение, то недостаточно просто создать начальный и конечный опорный кадр. Нужно еще задать направление вращения и число оборотов. Анимировать вращение можно для графических объектов любого типа, но для простоты мы ограничимся символами.

### Чтобы повернуть графический объект на угол меньше 360 градусов:

1. В новом документе или в новом слое существующего документа поместите на арену экземпляр какого-нибудь символа в опорном кадре 1.

Можно создать новый символ или воспользоваться существующим. Возьмите такую форму, которая выглядит по-разному на разных стадиях вращения, например, треугольник или стрелку (о создании символов см. главу 7).



**Рис. 9.13.** Чтобы подготовить интерполяцию вращением, в завершающем опорном кадре поверните элемент так, чтобы он занял конечное положение.

### Описание вращения с помощью всего двух опорных кадров

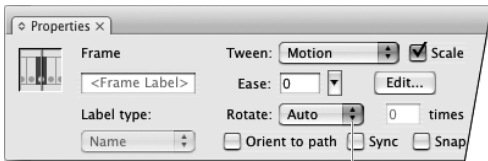
Интерполировать вращающийся объект так же просто, как в предыдущих случаях, не получится, поскольку точно описать вращение с помощью всего двух опорных кадров нельзя.

Попробуйте, например, мысленно повернуть стрелку компаса на 180 градусов из положения «север» в положение «юг». В начальном опорном кадре стрелка указывает вверх, в конечном — вниз. Но каким образом произошел этот переход?

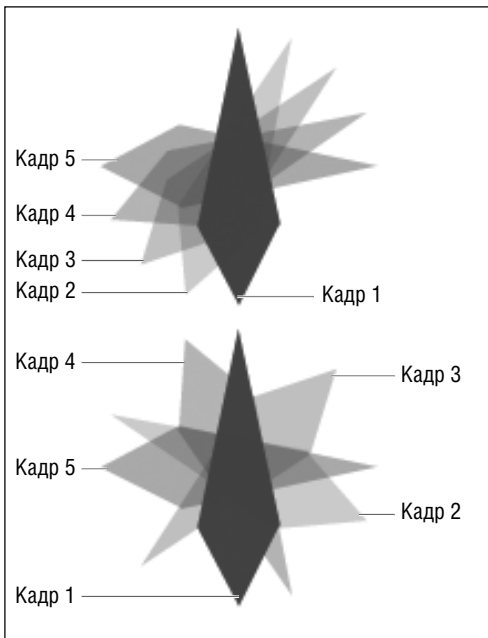
Flash предоставляет три варианта: повернуть стрелку по часовой стрелке, против часовой стрелки или симметрично отразить ее относительно горизонтальной оси. Попытка описать полный оборот стрелки всего двумя кадрами оказывается еще хуже, потому что начальный и конечный кадры одинаковы.

Чтобы однозначно описать движение, можно было бы создать ряд опорных кадров, так что в каждом последующем стрелка повернута на несколько градусов относительно предыдущего. Но это утомительный способ. К счастью на вкладке свойств кадра в инспекторе свойств есть возможность задать дополнительную информацию об интерполяции вращения, которая все же позволяет ограничиться двумя опорными кадрами.





**Рис. 9.14.** Список Rotate на вкладке Properties в инспекторе свойств позволяет задать направление вращения объекта.



**Рис. 9.15.** Чтобы сгенерировать анимацию, включающую вращение, необходимо задать его направление. Можно выбрать режим Auto, в котором Flash автоматически выберет направление, требующее минимальных изменений; в этом случае анимация будет наиболее плавной. Сравните угол изменения в каждом кадре при повороте из положения 12 часов в положение 3 часа по часовой стрелке (верхний рисунок) и против часовой стрелки (нижний рисунок).

2. На временной шкале выберите кадр 5 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**.

Flash продублирует символ из опорного кадра 1 в новом опорном кадре.

3. В пятом кадре поверните символ на арене на 90 градусов по часовой стрелке (рис. 9.13). (Подробнее о том, как поворачивать элементы, см. в главе 4.)
4. Выберите на временной шкале любой кадр в последовательности первого опорного кадра (1, 2, 3 или 4).
5. На вкладке **Properties** в инспекторе свойств выберите из списка Tween значение Motion. Появляются параметры интерполяции движения.
6. В списке **Rotation** (Поворот) выберите один из следующих пунктов:
  - для вращения объекта на наименьший угол выберите пункт Auto (Авто) (рис. 9.14);
  - для вращения объекта по часовой стрелке выберите пункт CW (По час.);
  - для вращения объекта против часовой стрелки выберите пункт CCW (Против час.);
7. Чтобы повернуть на угол меньше 360 градусов, в поле Times (раз) справа от Rotate введите 0.

Flash генерирует промежуточные кадры, так чтобы объект вращался вокруг точки трансформации. На каждом промежуточном кадре объект немного повернут относительно предыдущего положения (рис. 9.15).

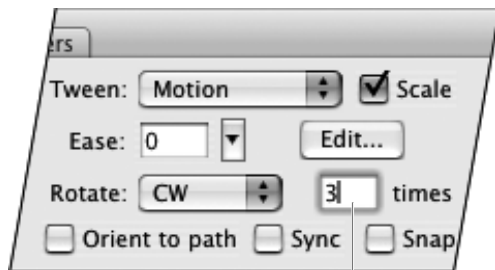
# Чтобы повернуть графический объект на 360 градусов:

1. Выполнив шаги 1 и 2 из предыдущего упражнения, создайте последовательность из пяти кадров с одинаковыми опорными кадрами в позициях 1 и 5.

Перемещать графические объекты не нужно, так как после поворота на 360 градусов начальный и конечный кадры будут выглядеть одинаково.

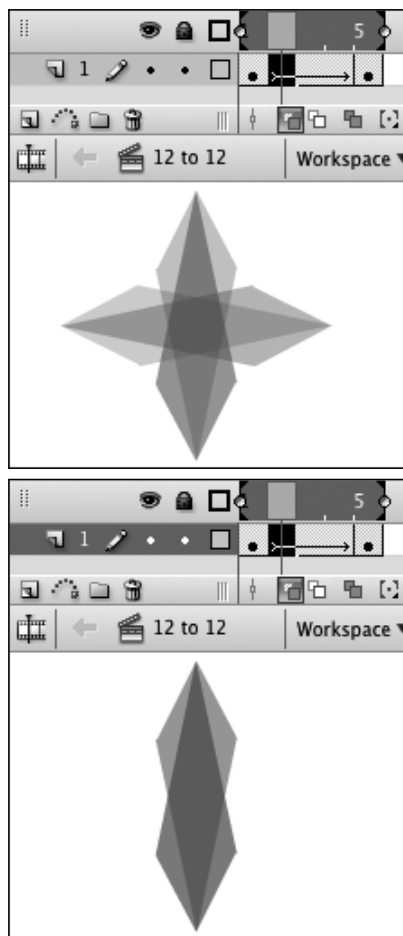
2. Выберите на временной шкале любой кадр в последовательности первого опорного кадра (1, 2, 3 или 4).
3. Установите значение Motion свойства Tween (см. шаг 5 предыдущего упражнения).
4. В списке Rotate выберите направление вращения.
5. В поле Times справа от списка Rotate введите число оборотов (рис. 9.16).

Значение в этом поле определяет, как Flash будет генерировать промежуточные кадры. Положения объекта выбираются так, чтобы он обернулся заданное число раз в заданном числе промежуточных кадров (рис. 9.17).

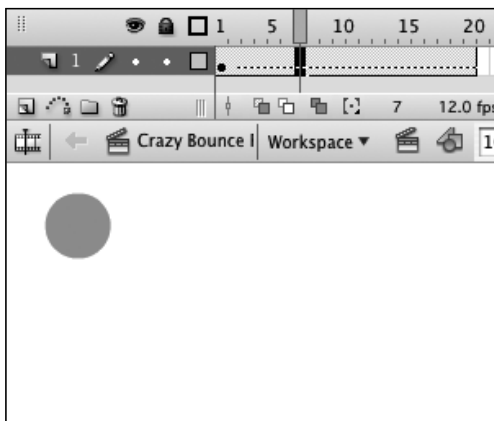


Ввести число оборотов

**Рис. 9.16.** На вкладке Properties в инспекторе свойств можно задать число оборотов.



**Рис. 9.17.** Сравните один оборот (верхний рисунок) с двумя (нижний рисунок) в одном и том же числе кадров.



**Рис. 9.18.** Пунктирная линия на временной шкале говорит о том, что эти 20 кадров сгенерированы, но не вполне успешно. Отсутствует завершающий опорный кадр.

## Перемещение графических элементов по прямой

В предыдущих упражнениях вы создали послушный мячик, который прыгает только вверх и вниз. Чтобы мячик двигался по такой замысловатой траектории, как в пинг-понге, надо увеличить число опорных кадров и расположить на них мяч в разных местах. Из одного места в другое мяч перемещается по прямой, но выглядит это динамичнее. Чтобы мячик метался быстро, перемещайте его на большее расстояние в малом числе кадров. Если хотите замедлить движение, то либо уменьшите расстояние, либо увеличьте число промежуточных кадров.

### Чтобы переместить мяч из одной точки в другую:

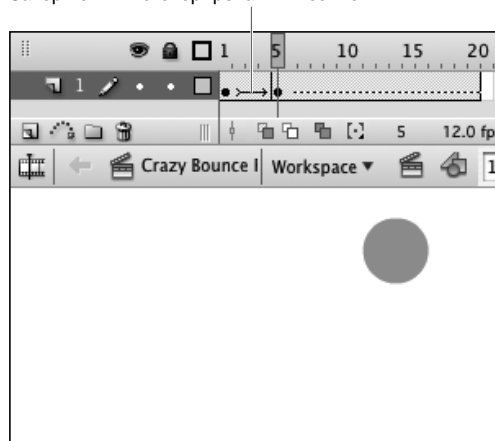
1. В новом документе или в новом слое существующего документа поместите на арену экземпляр символа мяча в опорном кадре 1.
2. На временной шкале выберите кадр 20 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame**. Flash создаст 19 промежуточных кадров.
3. На временной шкале выберите любой кадр из последовательности опорного кадра (от первого до двадцатого).
4. На вкладке **Properties** в инспекторе свойств выберите из списка **Tween** значение **Motion**. Flash перечеркивает кадры 1–20 пунктирной линией, говорящей о том, что интерполяция не завершена (рис. 9.18). Предстоит еще создать опорные кадры, описывающие движение мяча.
5. Подведите точку воспроизведения к кадру 5.

6. Перетащите мяч в другое место на арене. Flash создаст новый опорный кадр в позиции 5 и завершит интерполяцию для кадров 1–4 (рис. 9.19).
7. Подведите точку воспроизведения к кадру 10.
8. Снова перетащите мяч на арене в другое место. Flash создаст новый опорный кадр в позиции 10 и завершит интерполяцию для кадров 5–9.
9. Повторите описанную процедуру для кадров 15 и 20. Получился мяч, мечущийся из стороны в сторону, как бешеный (рис. 9.20).
10. Добавьте еще кадров, для чего выберите кадры с 30 по 40 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame**.

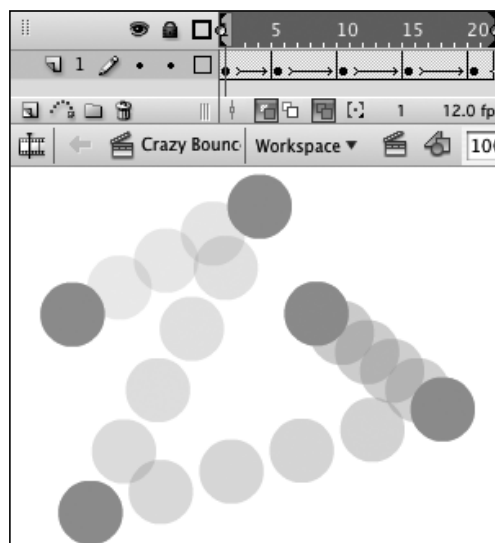
Flash применит заданные ранее свойства интерполяции движения к новым кадрам. Теперь можно определить опорные кадры, как было описано выше. Не забудьте только, что весь ряд должен завершаться опорным кадром. Если кадров получилось слишком много, лишние можно удалить

11. Чтобы завершить интерполяцию, выберите последний опорный кадр.
12. На вкладке **Properties** в инспекторе свойств выберите в списке **Tween** значение **None**. Если этого не сделать, что у всех добавленных впоследствии кадров свойство Tween будет равно Motion, что может привести к неожиданным результатам.

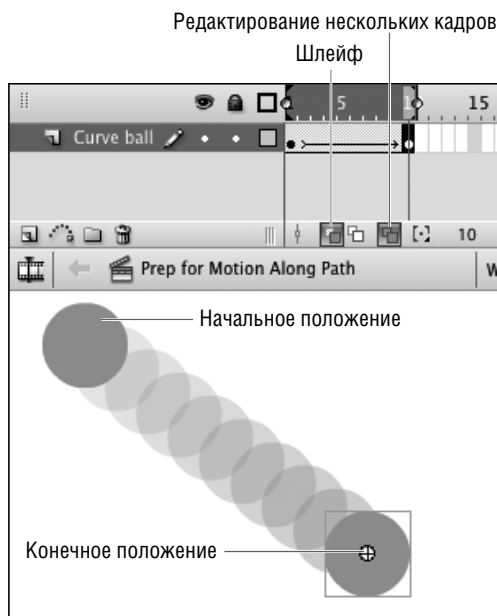
Завершенный сгенерированный сегмент



**Рис. 9.19.** Когда мяч занимает разные положения в разных кадрах одной анимации, Flash создает опорные кадры и генерирует промежуточные.

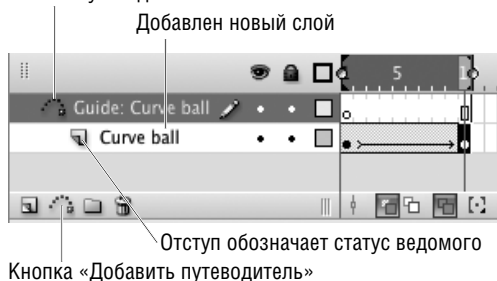


**Рис. 9.20.** Сцепляя несколько интерполяций, можно анимировать графический объект, перемещающийся от одной точки к другой.



**Рис. 9.21.** Первый шаг при подготовке к анимации вдоль пути заключается в том, чтобы определить начальное и конечное положение объекта. В данном случае объект перемещается между крайними положениями по прямой. (Чтобы видеть все компоненты интерполяции, включены режимы показа шлейфа восковок и редактирования нескольких кадров.)

Значок путевода



**Рис. 9.22.** Кнопка Add Motion Guide вставляет новый слой-путеводитель над выбранным. По умолчанию имя этого слоя включает имя слоя, который был выбран в момент создания путевода. Слой, содержащий анимируемую графику, отображается с отступом и связывается с путеводителем. Этот связанный слой во Flash называется ведомым.

## Перемещение графических элементов вдоль пути

В предыдущем упражнении мяч перемещался по арене короткими скачками от точки к точке. Иногда мяч действительно ведет себя таким образом, но чаще требуется более плавное движение, когда траектория составлена из дуг, а не отрезков прямых. Такого эффекта можно достичь, сцепляя много опорных кадров с перемещением по прямой, но Flash предлагает более эффективный способ — путевода. Путеводитель описывает точный путь, вдоль которого должен перемещаться графический объект.

### Чтобы добавить слой-путеводитель:

1. Создайте новый документ, содержащий интерполяцию движения из 10 кадров. В первом опорном кадре поместите графический объект, который предстоит анимировать, в левый верхний угол арены. В последнем кадре поместите такой же объект в правый нижний угол арены. Документ должен выглядеть примерно так, как показано на рис. 9.21.
2. Выберите слой, содержащий графический объект, который вы хотите переместить вдоль пути.
3. Нажмите кнопку **Add Motion Guide** (Добавить путевода), которая расположена в полосе состояния временной шкалы.

Flash добавит слой-путеводитель прямо над выбранным и присвоит ему по умолчанию имя Guide, за которым следует имя выбранного слоя (рис. 9.22). Слева от имени слоя отображается значок путевода. Слой, присоединенный к путеводителю, Flash отображает с отступом.

4. На панели инструментов выберите инструмент, который создает комбинированные формы или объекты-рисунки, например, карандаш.
5. На временной шкале выберите слой-путеводитель.
6. Проведите на арене линию, вдоль которой должен двигаться графический объект (рис. 9.23).

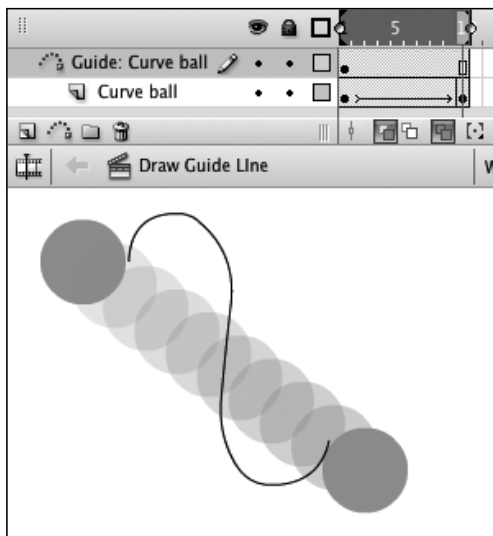
Пути на слое-путеводителе описывают пути движения графических объектов на связанных с ним обычных слоях.

7. Включите режим привязки к объектам (**View** ⇒ **Snapping** ⇒ **Snap to Objects**).

Чтобы Flash мог перемещать элемент вдоль пути, центр точки трансформации элемента (белый кружочек внутри символа или группы) должен располагаться на пути. В режиме привязки к объектам будет проще точно позиционировать анимируемую графику. (Подробнее о точках трансформации см. главы 4 и 7.)

8. В опорном кадре 1 с помощью инструмента «Стрелка» перетащите анимируемый объект за его точку трансформации, так чтобы последняя совместилась с началом пути движения.

В процессе перетаскивания индикатор совмещения слегка увеличивается, когда оказывается близко к какому-нибудь объекту, например, когда точка, за которую вы тянете, оказывается на путеводителе.

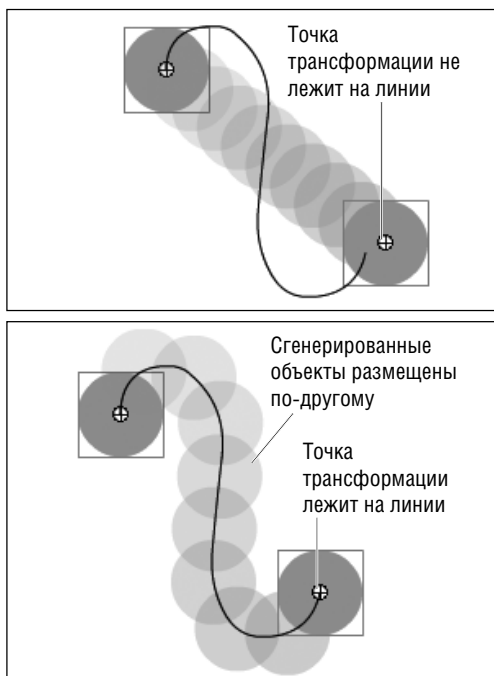


**Рис. 9.23.** Линия на слое-путеводителе описывает путь, вдоль которого графический объект будет перемещаться в связанном слое.

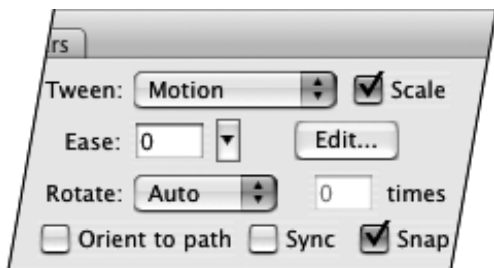
### О путеводителях

Путеводитель представляет собой графический объект, созданный в специальном слое отдельно от других. Он определяет путь движения анимированной графики. Один слой-путеводитель может управлять элементами в разных связанных с ним слоях. Связанные слои определяются как ведомые в диалоговом окне свойств слоя.

Если вы хотите, чтобы разные элементы двигались вдоль разных путей, создайте несколько слоев-путеводителей в одном документе. Каждый путеводитель управляет поведением графических объектов, которые находятся в связанных с ним слоях.



**Рис. 9.24.** Для перемещения вдоль пути точка трансформации анимируемого объекта должна находиться точно на путеводителе. (Для символов и групп точка трансформации представляется белым кружком. Для объектов-рисунков и примитивов точка трансформации появляется лишь в том случае, если вы переместили ее из центра объекта, где она располагается по умолчанию.)



**Рис. 9.25.** Если свойство Tween кадра равно Motion, то на вкладке Properties в инспекторе свойств присутствует флажок Snap. Отметьте его для слоя, связанного с путеводителем, чтобы Flash автоматически располагал точку трансформации любого графического объекта в этом слое на путеводителе. Принудительная привязка гарантирует правильность генерирования анимации вдоль пути.

9. В опорном кадре 10 перетащите анимируемый графический объект, так чтобы его точка трансформации находилась точно в конце пути движения.

Flash перерисовывает промежуточные кадры, так что объект на них перемещается вдоль заданного пути (рис. 9.24). В каждом таком кадре центр объекта находится на пути движения. В готовом ролике сам путь не виден.

**С** Нарисовав путь движения, заблокируйте слой-путеводитель, чтобы случайно не изменить его, когда будете привязывать к нему графику.

**С** Для каждого опорного кадра, содержащего анимируемую графику, отметьте на вкладке **Properties** в инспекторе свойств флажок **Snap** (Привязать), чтобы помочь Flash правильно центрировать графику в конце пути (рис. 9.25).

**С** Для создания пути подходит большинство инструментов рисования: линия, карандаш, овал, прямоугольник, многоугольник, звезда и кисть. Для инструментов можно установить как режим рисования комбинированных форм, так и режим рисования объектов. Однако пути, созданные инструментами «Прямоугольный примитив» и «Овальный примитив», в качестве путеводителей выступать не могут.

**С** Обычный направляющий слой можно преобразовать в путеводитель, если изменить положение слоя, находящегося прямо под ним. Смотрите на область предварительного просмотра и на значок в направляющем слое; по мере буксировки нижнего слоя вправо значок принимает вид составленной из точек дуги, которая и обозначает путеводитель (подробнее о направляющих слоях см. главу 6).

### Чтобы создать второй ведомый слой:

1. В документе, который вы создали в предыдущем упражнении, выберите ведомый слой (тот, в котором есть кружочек).
2. Для добавления еще одного слоя выполните любое из следующих действий:
  - выполните команду **Insert** ⇒ **Timeline** ⇒ **Layer**;
  - нажмите кнопку **Insert Layer** на временной шкале.

Flash добавит новый слой с отступом (ведомый) над выбранным слоем (рис. 9.26). Анимлируемые объекты в этом слое станут двигаться вдоль путеводителя после того, как вы их правильно расположите.

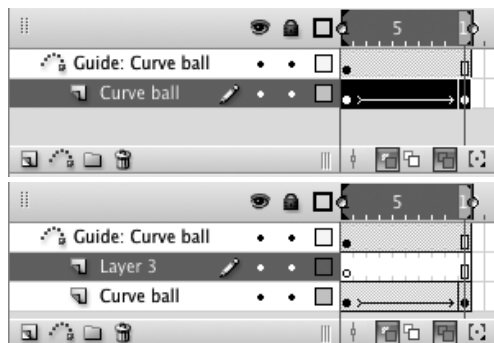
### Чтобы добавить второй слой-путеводитель:

1. Создайте документ, содержащий хотя бы один обычный слой с автоматически генерируемой интерполяцией движения и управляющий им слой-путеводитель.
  2. Выберите обычный слой на временной шкале.
  3. Нажмите кнопку **Add Motion Guide**.
- Flash добавит еще один слой-путеводитель над выбранным и свяжет его с выбранным слоем. Выполните те же шаги, что в предыдущем упражнении, чтобы нарисовать второй путь и правильно позиционировать анимируемый элемент.
4. Протестируйте ролик.

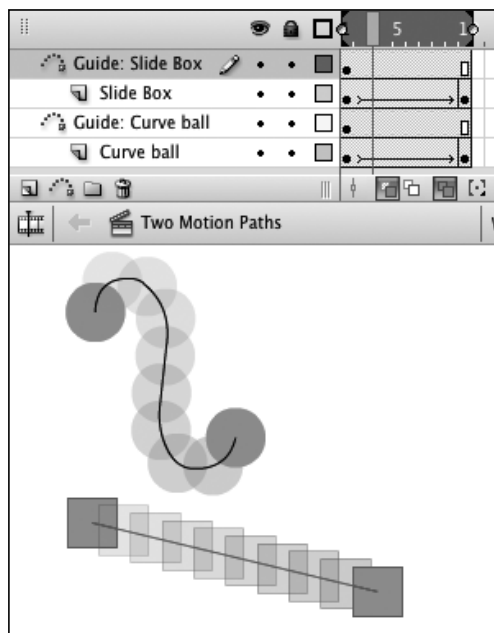
Два анимированных объекта одновременно движутся вдоль разных путей (рис. 9.27).

**С**

Чтобы быстро преобразовать существующий слой в ведомый, перетащите его под слой-путеводитель или под любой из связанных с ним слоев.

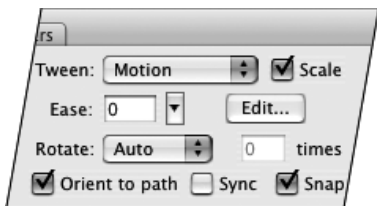


**Рис. 9.26.** Если выбран ведомый слой (верхний рисунок), то нажатие кнопки **Insert Layer** создает еще один ведомый слой под путеводителем (нижний рисунок).

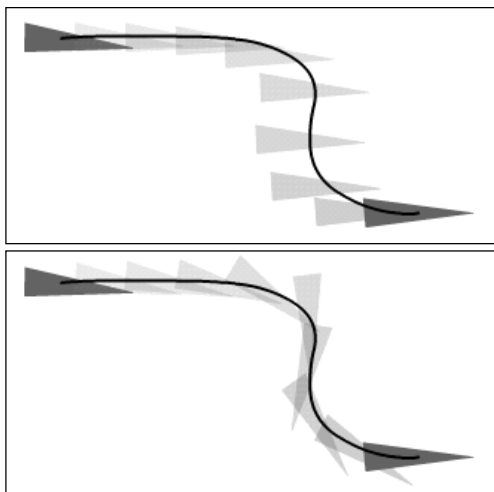


**Рис. 9.27.** Добавьте несколько слоев-путеводителей, чтобы заставить несколько графических объектов одновременно двигаться вдоль разных путей.





**Рис. 9.28.** На вкладке Properties в инспекторе свойств выберите значение Motion свойства Tween и отметьте флажок Orient to Path, чтобы Flash ориентировал анимируемый объект в направлении движения.



**Рис. 9.29.** Если стрелка не ориентирована относительно пути, то при движении она остается параллельной нижнему краю арены (верхний рисунок). Если же затребована ориентация относительно пути, то стрелка поворачивается, следуя изгибам пути (нижний рисунок).

## Ориентация графических элементов относительно пути движения

По умолчанию ориентация анимированного графического объекта не изменяется (объект не вращается) даже тогда, когда он движется вдоль пути. Чтобы движение получилось более естественным, можно заставить анимируемый объект поворачиваться, сохраняя ориентацию относительно пути в каждом промежуточном кадре.

### Чтобы ориентировать графический элемент относительно пути:

1. Создайте десятикадровую интерполяцию движения вдоль путеводителя. Для этого выполните те же шаги, что в первом упражнении из раздела «Перемещение графики вдоль пути». Но на этот раз анимируйте не круг, а треугольник или изображение какого-нибудь животного.
2. Включите показ шлейфа восковок, чтобы видеть, как объект перемещается, не меняя ориентации.
3. В слое, содержащем анимируемый объект, выберите опорный кадр 1.
4. На вкладке свойств этого кадра в инспекторе свойств отметьте флажок **Orient to Path** (Ориентировать по пути) (рис. 9.28).

Flash сгенерирует интерполяцию заново. В промежуточных кадрах объект будет поворачиваться, следуя изгибам пути (рис. 9.29).

### Зачем ориентировать относительно пути?

Представьте себе официанта, который идет с полным подносом в переполненном зале. Он то поднимает, то опускает поднос, обнося препятствия, но при этом держит его параллельно полу, чтобы ничего не пролить. Именно так работает интерполяция, если анимируемый объект не изменяет ориентацию относительно пути. Точка трансформации объекта в каждом кадре оказывается в новом месте пути, но объект не поворачивается. В случае мяча это выглядит приемлемо, но для некруглых графических объектов результат оказывается неестественным.

Хотелось бы, чтобы животное смотрело в том направлении, куда движется, поворачиваясь в соответствии с изгибами пути. А самолет, выполняющий петлю, должен быть расположен по касательной к траектории, а не перпендикулярно ей. В режиме ориентации относительно пути графический объект поворачивается вместе с изменением направления пути. Тем самым создается впечатление, что объект всегда «смотрит» вдоль пути.

С

Чтобы в режиме ориентации относительно пути объект выглядел максимально естественно, нужно расположить его под правильным углом в начальном и конечном кадре интерполяции (рис. 9.30).

С

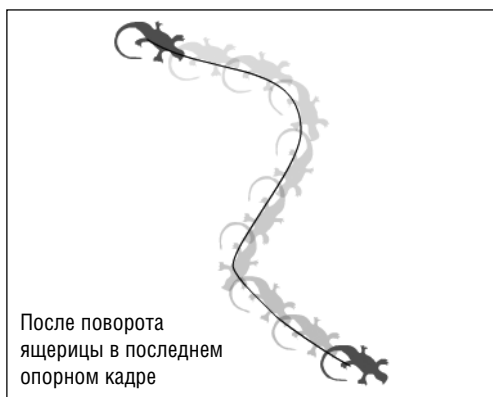
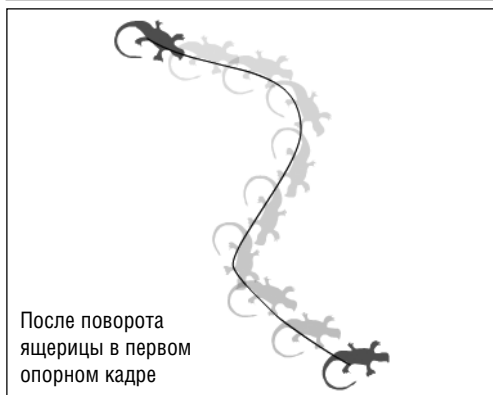
Если, несмотря на предыдущий совет, ориентация в каких-то точках выглядит странно, просмотрите все промежуточные кадры по одному. Обнаружив кадр, в котором Flash неудачно расположил графический объект, вы можете внести коррективы. Выберите промежуточный кадр на временной шкале и выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**. В новом опорном кадре, который создаст Flash, выберите объект и вручную поверните его в соответствии с направлением путеводаителя в этой точке. Flash перерисует остальные промежуточные кадры.

С

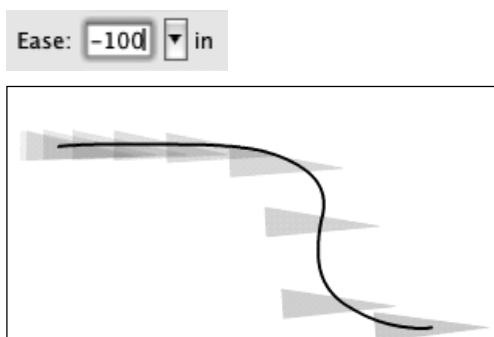
Следуя предыдущему совету, включите показ шлейфа восковок. Тогда вы сможете видеть, как корректировка отразилась на ориентации объекта в каждом сгенерированном кадре.

С

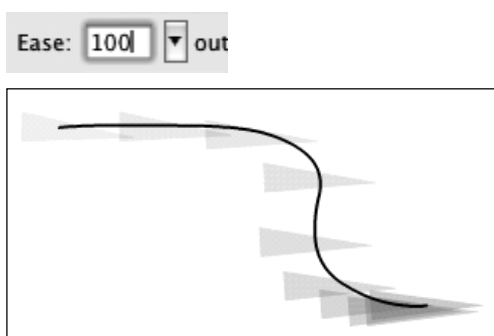
Иногда для более естественной ориентации объекта относительно пути полезно слегка сместить точку трансформации. По умолчанию точка трансформации ящерицы на рис. 9.30 находится в центре охватывающего прямоугольника. Но поскольку хвост ящерицы сильно изогнут, эта точка оказывается вне ее туловища. Если переместить точку трансформации в середину туловища ящерицы, то движение будет больше походить на реальное. (Чтобы изменить положение точки трансформации графического объекта или экземпляра символа, воспользуйтесь инструментом «Свободное преобразование».)



**Рис. 9.30.** Здесь можно было бы ориентировать ящерицу вдоль пути и получше (верхний рисунок). Для этого следует правильно расположить ее в первом (средний рисунок) и последнем (нижний рисунок) опорном кадре.



**Рис. 9.31.** Отрицательное значение в поле Ease (верхний рисунок) приводит к тому, что изменения в начальных кадрах последовательности меньше, а в конечных — больше (нижний рисунок). Создается впечатление, что анимация начинается медленно, а затем ускоряется.



**Рис. 9.32.** Положительное значение в поле Ease (верхний рисунок) приводит к тому, что изменения в начальных кадрах последовательности больше, а в конечных — меньше (нижний рисунок). Создается впечатление, что анимация начинается быстро, а затем замедляется.

## Изменение скорости анимации

В главе 8 вы научились изменять скорость движения анимированного объекта, вставляя больше или меньше промежуточных кадров. Но в случае интерполяции этот способ не годится, так как Flash распределяет движение равномерно по заранее заданному числу кадров. Однако замедлить анимацию в начале или в конце сгенерированной последовательности можно с помощью свойства Ease (Замедлить).

### Чтобы анимация начиналась медленно, а затем ускорялась:

1. Создайте десятикадровую интерполяцию движения графического объекта вдоль пути.
2. Выберите на временной шкале любой промежуточный кадр (в позиции 1–9).
3. На вкладке **Properties** в инспекторе свойств введите отрицательное значение в поле Ease (рис. 9.31).
4. Нажмите **Enter**.

Справа от поля появляется слово *in* (в). В этом случае анимация начинается медленно и постепенно ускоряется. Чем меньше значение Ease, тем больше ускорение.

### Чтобы анимация начиналась быстро, а затем замедлялась:

1. Выполните шаги 1 и 2 из предыдущего упражнения.
2. На вкладке **Properties** в инспекторе свойств введите положительное значение в поле Ease.
3. Нажмите **Enter**.

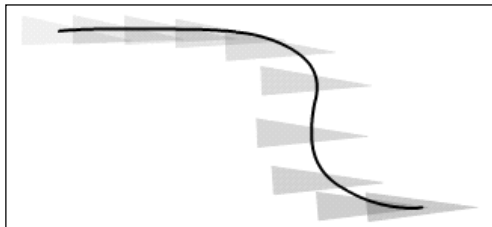
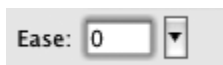
Справа от поля появляется слово *out* (из). В этом случае анимация начинается быстро и постепенно замедляется. Чем больше значение Ease, тем больше ускорение (рис. 9.32).

## С

Если значение Ease равно 0, то скорость анимации постоянна (рис. 9.33).

### Чтобы изменять скорость по своему вкусу:

1. Создайте интерполяцию движения, в которой изменяются сразу несколько свойств анимируемого объекта. Например, нарисуйте инструментом «Овал» желтый круг и преобразуйте его в символ, который назовите Sun (Солнце). На основе символа Sun создайте интерполяцию из 30 кадров. В опорном кадре 1 разместите символ у верхнего края арены, а в кадре 30 — у нижнего и при этом сделайте его вдвое больше, а цвет измените на темно-оранжевый.
2. Выберите кадр в любой из позиций от 1 до 29.
3. Чтобы получить доступ к параметрам, определяющим скорость анимации, нажмите кнопку **Edit** (Редактировать) справа от поля Ease (рис. 9.34).



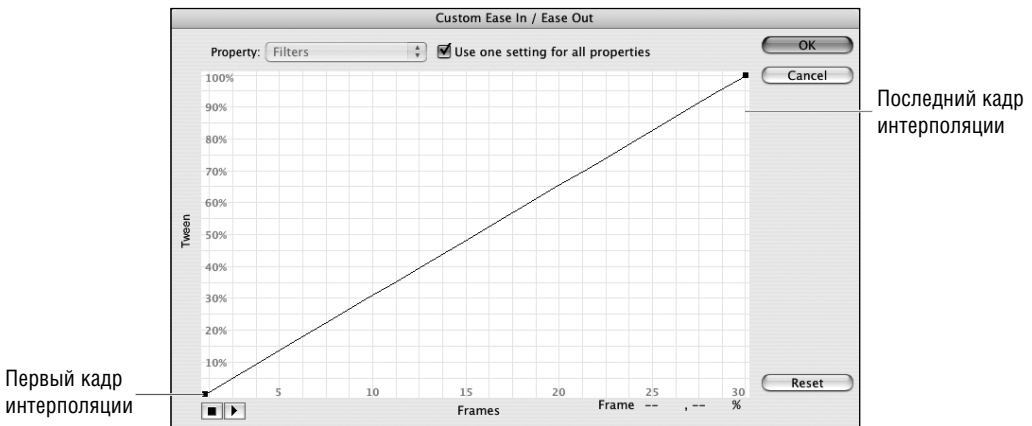
**Рис. 9.33.** Если Ease равно 0 (верхний рисунок), то изменения распределяются равномерно по всей длине последовательности промежуточных кадров (нижний рисунок). Поэтому кажется, что скорость анимации постоянна.



**Рис. 9.34.** Для последовательности опорного кадра, в котором свойство Tween равно Motion, можно самостоятельно задавать параметры скорости анимации. Для этого нажмите кнопку Edit на вкладке свойств кадра.

Открывается окно Custom Ease In/Ease Out (рис. 9.35). В нем зависимость скорости изменения выбранного свойства от времени представлена в виде графика. На горизонтальной оси показаны кадры, а на вертикальной — величина изменения. Если значение в поле Ease не было задано, то на графике мы видим прямую, которая соединяет управляющие точки, соответствующие первому кадру (в котором 0% изменений) и последнему кадру (100% изменений). График можно отредактировать, задавая процентную долю изменений любого из пяти анимируемых свойств (положение, угол поворота, масштаб, цвет и фильтры) в каждом кадре.

4. Чтобы заданная величина изменений применялась сразу ко всем пяти свойствам, отметьте флажок **Use one setting for all properties** (Использовать один параметр для всех свойств).



**Рис. 9.35.** В диалоговом окне Custom Ease In/Ease Out можно настроить скорость анимации с точностью до кадра, изменяя форму графика. По умолчанию скорость постоянна, причем первому кадру соответствует 0%, а последнему — 100%.

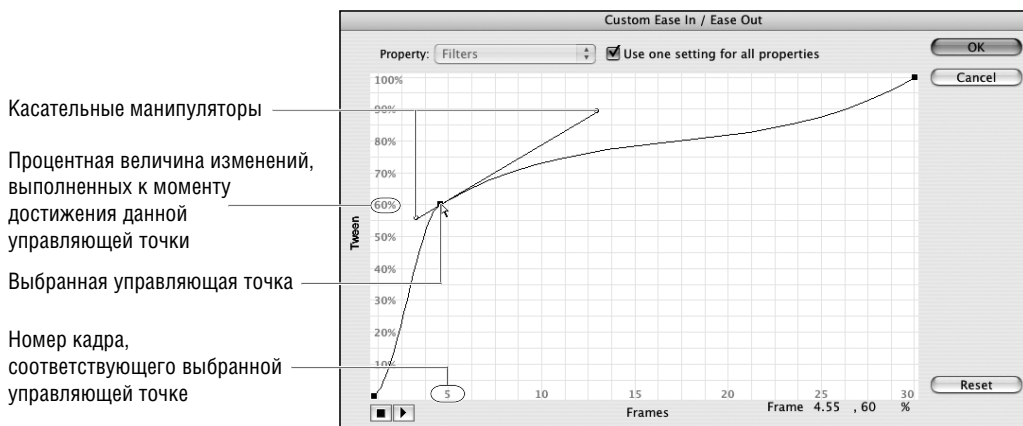
- Чтобы модифицировать скорость изменения, щелкните по кривой в точке, соответствующей кадру, где скорость должна отличаться от предлагаемой по умолчанию.

Flash добавит управляющую точку с касательными манипуляторами; они работают так же, как узлы и манипуляторы Безье, позволяя изменять форму кривой (рис. 9.36).

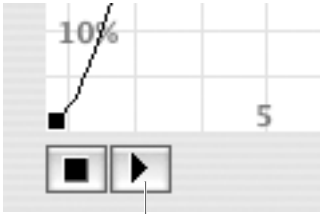
- Буксируйте управляющую точку вертикально, чтобы задать процент изменений в текущем кадре; пользуйтесь манипуляторами для точной настройки скорости изменений по обе стороны от управляющей точки.

При буксировке вверх кривая становится круче (изменения происходят быстрее), а при буксировке вниз — более пологой (изменения происходят медленнее). Для управления формой кривой в начальной и конечной точке можно использовать касательные манипуляторы.

- Повторяйте шаги 5 и 6, пока не добьетесь нужной скорости изменений в интересующих вас кадрах.
- Нажмите ОК.

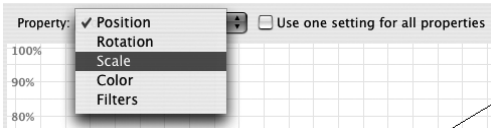


**Рис. 9.36.** Щелкните по графику в диалоговом окне Custom Ease In/Ease Out, чтобы добавить управляющую точку. Смещение управляющей точки по вертикали увеличивает или уменьшает величину изменения в текущем кадре. Касательные манипуляторы позволяют управлять наклоном кривой: чем кривая круче, тем больше скорость изменений.



Воспроизвести

**Рис. 9.37.** Чтобы увидеть, к чему привела модификация кривой скорости изменений, нажмите кнопку Play в окне Custom Ease In/Ease Out.



**Рис. 9.38.** Сброс флажка Use one setting for all properties делает доступным список Property. Выберите свойство, для которого хотите настроить скорость изменения.

**С**

Для предварительного просмотра того, что получилось, нажмите кнопку **Play** (Воспроизвести) в левом нижнем углу окна Custom Ease In/Ease Out (рис. 9.37). Интерполяция воспроизводится в окне документа Flash. Чтобы увидеть окно документа, возможно, придется сдвинуть диалоговое окно.

**С**

Если в поле Ease уже было задано какое-то значение, то оно отразится на графике. Тем самым простую настройку скорости анимации можно использовать как основу для задания более специализированного поведения. Но имейте в виду: если вы настроите нестандартную скорость, а потом введете значение в поле Ease, то оно и будет действовать, а все ваши настройки пропадут.

### Чтобы независимо изменять скорость анимации различных свойств:

1. Выполните шаги 1–3 из предыдущего упражнения.
2. Сбросьте флажок **Use one setting for all properties**.
3. В списке **Property** (Свойство) (рис. 9.38) выберите одно из следующих значений:

**Position** (Положение) управляет скоростью перемещения анимируемого объекта по арене.

**Rotation** (Поворот) управляет скоростью вращения анимируемого объекта (путем задания свойства Rotation на вкладке свойств кадра).

**Scale** (Масштаб) управляет скоростью изменения размера анимируемого объекта.

**Color** (Цвет) управляет скоростью изменения цвета экземпляра символа (свойство Color).

**Filters** (Фильтры) управляет скоростью применения эффектов фильтров.

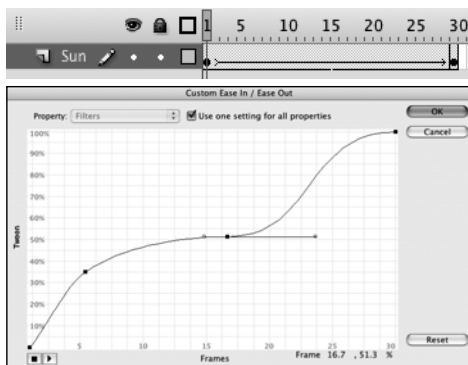
4. Измените параметры так же, как в шагах 5–7 из предыдущего упражнения.
5. Повторите шаги 3–4 для каждого интересующего вас свойства.
6. Нажмите ОК.

**С**

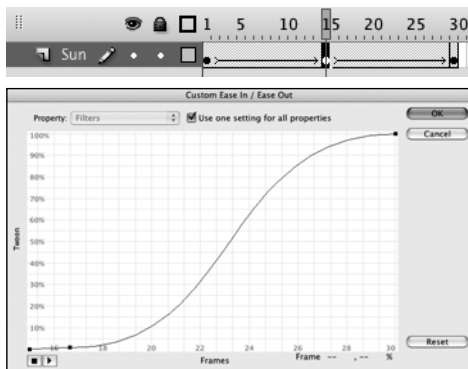
Если в последовательность сгенерированных кадров со специально настроенной скоростью анимации вставить новые опорные кадры, то Flash попытается по возможности сохранить настройки для вновь образованной последовательности (рис. 9.39). Однако Flash не может просто разбить кривую изменения скорости на меньшие сегменты. В каждом новом опорном кадре процент изменения для следующей за ним последовательности должен быть равен 0, а в последнем кадре последовательности процент изменений должен составлять 100. Нажмите кнопку **Edit** на вкладке свойств кадра, чтобы просмотреть и, возможно, более точно настроить получившиеся параметры.

**С**

Чтобы применить одну и ту же кривую к нескольким, но не ко всем свойствам, воспользуйтесь стандартными командами копирования и вставки. Настройте кривую для первого свойства и нажмите **⌘-C** (Mac) либо **Ctrl-C** (Windows). Затем выберите другое свойство из списка **Property** и нажмите **⌘-V** (Mac) либо **Ctrl-V** (Windows), чтобы вставить только что скопированную кривую.



Исходная кривая скорости изменений



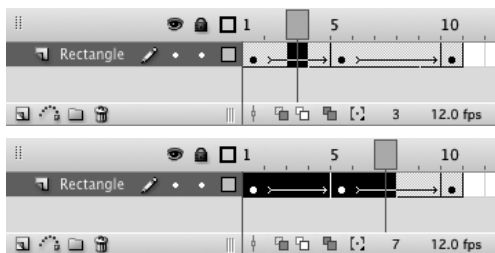
Кривая скорости изменений после добавления опорного кадра

**Рис. 9.39.** Если в последовательность сгенерированных кадров со специально настроенной скоростью анимации вставить новые опорные кадры, то Flash попытается по возможности сохранить настройки для вновь образованной последовательности.

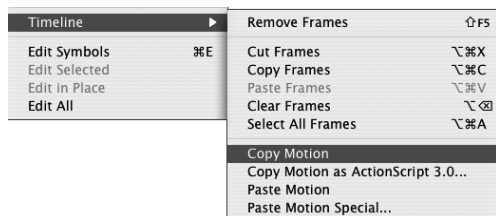
### Модификация движения с помощью настройки скорости анимации

Настраивание скорости анимации можно использовать для варьирования стандартной интерполяции, например, чтобы заставить объект несколько раз качнуться туда-сюда несмотря на то, что интерполяция определена всего двумя опорными кадрами, между которыми объект перемещается по прямой. Представьте себе интерполяцию из 20 кадров; в первом объект находится у левого края арены, а в последнем — у правого края. Теперь настроим скорость изменения свойства **Position** в последовательности первого опорного кадра. Добавим управляющую точку в кадре 5 и поднимем ее вертикально, так чтобы 50% всего изменения происходило в этом кадре. Затем добавим управляющую точку в кадре 10 и расположим ее так, чтобы на этот кадр приходилось 25% изменения. В результате прямоугольник дойдет до середины арены к кадру 5, потом между кадрами 5 и 10 вернется назад и снова отправится вперед, пока к кадру 20 не достигнет правого края арены.





**Рис. 9.40.** Чтобы скопировать информацию об одном интерполированном движении, укажите на временной шкале любой кадр в соответствующей сгенерированной последовательности (верхний рисунок). Чтобы скопировать информацию о нескольких движениях, выберите все соответствующие последовательности (нижний рисунок). Они должны быть смежными, но последнюю не обязательно выбирать целиком.



**Рис. 9.41.** Команда **Edit** ⇒ **Timeline** ⇒ **Copy Motion** копирует свойства, описывающие одну или несколько интерполяций, занимающих выбранный набор кадров.

### О копировании движения

Команда **Copy Motion** копирует информацию, которая полностью описывает сгенерированную последовательность: изменения в свойствах анимируемого объекта (положение, поворот, масштаб и т.п.), протяженность интерполяции (число занимаемых кадров) и другие настройки (скажем, скорость анимации). Команда **Paste Motion** применяет эту информацию к другому графическому объекту, чтобы его свойства изменялись точно так же, в таком же числе кадров и с теми же параметрами интерполяции. Команда **Paste Motion Special** избирательно применяет изменения к свойствам анимируемого объекта. Команда **Copy Motion as ActionScript 3.0** транслирует интерполяцию в набор команд на языке ActionScript 3.0.

## Копирование интерполированного движения

Во Flash CS3 появилось три новых команды, позволяющих сократить объем работы по созданию повторяющихся анимаций. Это команды **Copy Motion** (Копировать движение), **Paste Motion** (Вставить движение) и **Paste Motion Special** (Вставить специальное движение), которые упрощают создание нескольких объектов, выполняющих одно и то же «движение».

### Чтобы скопировать свойства интерполированного движения:

1. Воспользуйтесь любым из рассмотренных в этой главе способов, чтобы создать интерполяцию движения символа в одном слое. Интерполяция может состоять из одной или нескольких последовательностей опорного кадра.
2. Чтобы выбрать свойства интерполированного движения, выполните одно из следующих действий:

- для выбора свойств одной интерполяции укажите на временной шкале любой кадр в последовательности, содержащей нужное движение (рис. 9.40);
- для выбора свойств нескольких интерполяций укажите на временной шкале те последовательности, которые содержат нужное движение; они должны быть смежными.

3. Выполните команду **Edit** ⇒ **Timeline** ⇒ **Copy Motion** (Редактирование ⇒ Временная шкала ⇒ Копировать движение) (рис. 9.41). Flash копирует изменения в свойствах, определяющих движения в каждой из выбранных (или частично выбранных) последовательностей: протяженность, координаты *x* и *y*, поворот и перекокс, масштабы по осям *x* и *y*, цвет, настройки скорости анимации, фильтры и наложения (подробнее о фильтрах и наложениях см. главу 11).

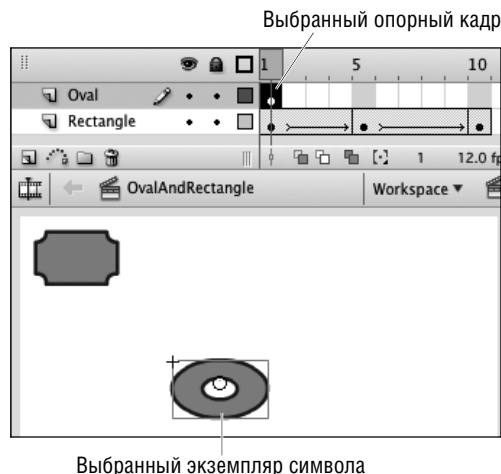
# Чтобы применить свойства интерполированного движения к другому символу:

1. Чтобы выбрать конечный экземпляр символа, выполните одно из следующих действий:

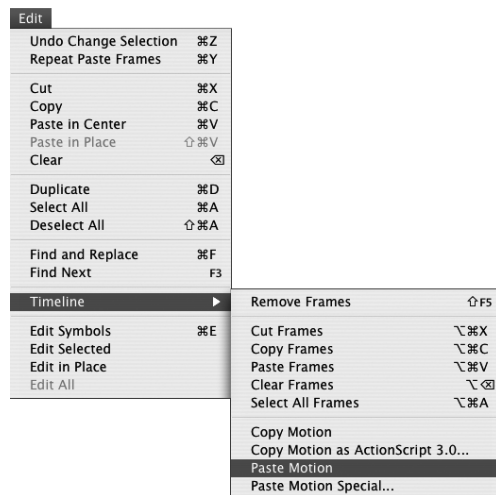
- выберите на арене экземпляр символа, к которому намереваетесь применить скопированные свойства интерполяции (рис. 9.42);
- выберите на временной шкале опорный кадр, содержащий экземпляр символа, к которому намереваетесь применить скопированные свойства интерполяции.

Символ должен быть изолирован в своем собственном слое.

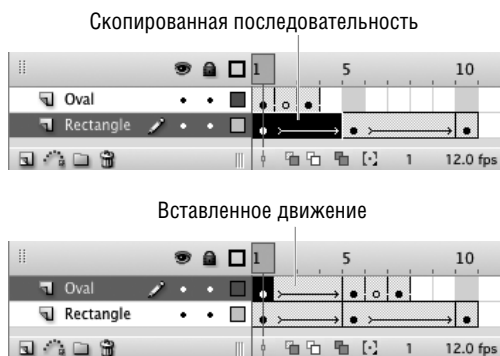
2. Выполните команду **Edit ⇒ Timeline ⇒ Paste Motion** (Редактирование ⇒ Временная шкала ⇒ Вставить движение) (рис. 9.43).



**Рис. 9.42.** Чтобы применить скопированные свойства интерполяции к новому экземпляру символа, выберите этот символ на арене или его опорный кадр на временной шкале.



**Рис. 9.43.** Команда **Edit ⇒ Timeline ⇒ Paste Motion** создает интерполяцию с теми свойствами, которые были скопированы ранее. Flash вставляет одну или несколько последовательностей опорного кадра перед выбранным на временной шкале кадром, а сам этот кадр не изменяет.



**Рис. 9.44.** Команда Paste Motion вставляет скопированное движение на временную шкалу для выбранного символа. Она сохраняет невыбранные опорные и промежуточные кадры, сдвигая их вправо по временной шкале.

Flash вставляет на временную шкалу то же количество последовательностей, что было скопировано перед этим. Выбранный опорный кадр и все следующие за ним сдвигаются вправо по временной шкале (рис. 9.44).

### 3. Протестируйте ролик.

Символ, к которому применены скопированные свойства интерполяции, теперь двигается и изменяется точно так же, как исходный.

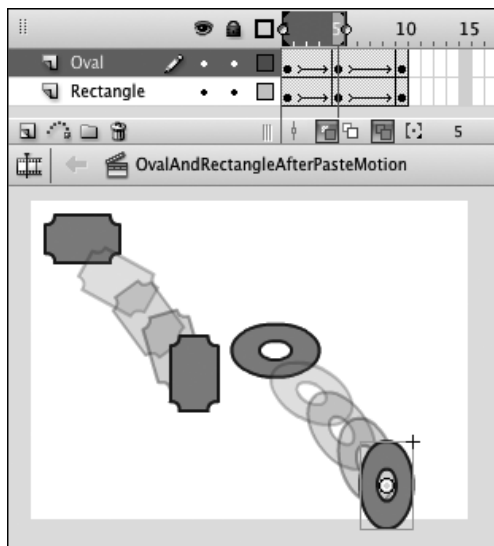
**С**

В предыдущих упражнениях описаны копирование и вставка свойств интерполированного движения символа, но Flash может также скопировать свойства интерполяции любого графического объекта и применить к другому объекту. Графический объект должен отвечать условиям, необходимым для интерполяции движения; он должен находиться в контейнере и быть единственным объектом в своем слое. При вставке свойства скопированной интерполяции объект-получатель должен обладать способностью анимировать это свойство. Можно скопировать движение, перемещающее символ по арене, и применить его к текстовому полю. Но нельзя, к примеру, скопировать изменения свойства Color символа и применить это изменение к объекту-рисунку или примитиву.

### Вставленное движение: все относительно (почти)

При вставке движения почти все параметры применяются к графическим объектам относительно. Предположим, что исходным графическим объектом является прямоугольный примитив, расположенный в левом верхнем углу арены, который перемещается по диагонали в центр. Если скопировать это движение и применить его к овальному примитиву, который расположен в центре арены, то он будет двигаться вниз и вправо по диагонали и остановится вблизи правого нижнего угла (рис. 9.45). Если вы хотите в точности повторить движение из левого верхнего угла в центр, то должны сначала поместить конечный объект в то же начальное положение, что и исходный.

Команда **Paste Motion Special** позволяет выбрать те аспекты скопированного движения, которые следует применить к конечному объекту. Отметив или сбросив различные флажки, вы можете заставить Flash применить (или пропустить) скопированные изменения следующих свойств: координаты  $x$  и  $y$ , масштаб по горизонтали и по вертикали, поворот и перекося, цвет, фильтры и наложение. Кроме того, можно отменить изменения масштаба и угла поворота/перекося, которые были произведены над конечным объектом. Пусть, к примеру, конечный объект следовало бы повернуть на 90 градусов по часовой стрелке. Тогда, отметив флажок **Override Target Rotation and Skew Properties** (Изменить конечные свойства поворота и наклона), вы потребуете, чтобы Flash применил скопированное движение к объекту в исходном состоянии, как если бы он вообще не поворачивался (см. упражнение «Чтобы избирательно применить скопированные свойства движения»).



**Рис. 9.45.** В результате применения движения прямоугольника к овалу Flash воспроизводит перемещение вниз по диагонали. Овал проходит то же расстояние и поворачивается на тот же угол, но относительно собственной начальной позиции, а не начальной позиции прямоугольника.



**Рис. 9.46.** Для избирательного применения анимации к свойствам выполните команду **Edit** ⇒ **Timeline** ⇒ **Paste Motion Special**.



**Рис. 9.47.** В диалоговом окне **Paste Motion Special** отметьте флажки, соответствующие свойствам, к которым вы хотите применить скопированную анимацию.

### Чтобы избирательно применить скопированные свойства движения:

1. Выполните шаг 1 из предыдущего упражнения.
2. Выполните команду **Edit** ⇒ **Timeline** ⇒ **Paste Motion Special** (Редактирование ⇒ Временная шкала ⇒ Вставить движение) (рис. 9.46).
3. Чтобы применить отдельные свойства скопированного движения, отметьте следующие флажки в любой комбинации (рис. 9.47):

- для копирования движения по горизонтали отметьте флажок **X Position** (Положение по оси X);
- для копирования движения по вертикали отметьте флажок **Y Position** (Положение по оси Y);
- для копирования изменений размера вдоль горизонтальной оси отметьте флажок **Horizontal Scale** (Масштаб по горизонтали);
- для копирования изменений размера вдоль вертикальной оси отметьте флажок **Vertical Scale** (Масштаб по вертикали);
- для копирования изменений угла поворота и перекоса отметьте флажок **Vertical Scale** (Вращение и наклон);
- для копирования изменений свойства **Color** экземпляра символа, отметьте флажок **Color** (Цвет). (Это свойство имеется только у экземпляров символов, см. главу 7);
- для копирования настроек фильтров, изменений этих настроек и состояний включен/выключен отметьте флажок **Filters**. Фильтры можно применять только к символам-клипам, символам-кнопкам и текстовым полям;
- чтобы скопировать режим наложения из исходного символам-клипа в новый экземпляр символа, отметьте флажок **Blend Mode** (Режим наложения). (Это свойство имеется только у символов-клипов.);

- чтобы применить изменения масштаба к выбранному графическому объекту относительно его исходного размера (игнорируя масштабирование, примененное к объекту на арене), отметьте флажок **Override target scale properties** (Изменить конечные свойства масштаба). В противном случае Flash применит изменения масштаба к текущему размеру объекта на арене (рис. 9.48).
- чтобы применить изменения угла поворота и перекося к выбранному графическому объекту относительно его исходной ориентации (игнорируя повороты и перекосы, примененные к объекту на арене), отметьте флажок **Override target rotation and skew properties** (Изменить конечные свойства поворота и наклона). В противном случае Flash применит изменения угла поворота и перекося к текущей ориентации объекта на арене.

Исходный символ



Модифицированный символ



Применить с учетом выполненных ранее масштабирования и вращения/перекося



Отменить выполненные ранее масштабирование и вращение/перекос



**Рис. 9.48.** Когда команда **Paste Motion Special** применяется к графическому объекту, который был модифицирован путем масштабирования и/или вращения и перекося, то необходимо решить, применять ли скопированное движение к модифицированному или исходному объекту. Для этого служат флажки в диалоговом окне **Paste Motion Special**. На этом рисунке одно и то же движение применяется к двум экземплярам модифицированного символа. Для левого символа флажки **Override target scale properties** и **Override target rotation and skew properties** сброшены, для правого — отмечены.

## С

Можно также скопировать свойства интерполяции в форме, которая затем вставляется на панель действий или в **ActionScript**-файл. Flash транслирует анимацию, созданную с помощью графических инструментов и интерполяции движения, в команды на языке **ActionScript**. Дополнительную информацию о команде **Copy Motion as ActionScript 3.0** см. в главе 13.

# АНИМАЦИЯ ПОСРЕДСТВОМ ИНТЕРПОЛЯЦИИ ФОРМ

# 10

## Чем интерполяция форм отличается от интерполяции движения?

Важное отличие между двумя видами интерполяции состоит в том, что в случае интерполяции движения графическое содержимое должно быть заключено в контейнер (этому условию удовлетворяют объекты-рисунки, примитивные формы, группы, символы и текстовые поля), тогда как для интерполяции форм требуется, чтобы графика была редактируемой. Комбинированные формы и объекты-рисунки отвечают этому требованию, примитивы — тоже, но с тем ограничением, что при изменении пути должны сохраняться определяющие характеристики примитива (подробнее о редактировании примитивов см. главу 4). Интерполируемые формы могут перемещаться вдоль прямой линии, но не могут автоматически следовать за путеводителем и поворачиваться на заданное число оборотов.

Интерполяция форм в Adobe Flash CS3 Professional работает во многом так же, как интерполяция движения. Сначала подготавливается последовательность опорного кадра (графическое содержимое начального и конечного кадров в ней должны отличаться), а затем ее свойству Tween присваивается значение Shape. Flash заполняет промежуточные кадры переходными изображениями, преобразуя начальную форму в конечную. В процессе интерполяции форм можно анимировать изменения тех же свойств, что и при интерполяции движения (размер, цвет, положение и т.д.), но дополнительно изменяется еще и путь, который определяет «форму» графического содержимого.

Flash может анимировать и более одной формы в слое, но результаты при этом могут оказаться непредсказуемыми. Когда в слое выбрано несколько форм, нельзя указать, какая начальная форма в какую конечную должна перейти. Ограничившись единственной формой в слое, вы сможете избежать неоднозначности.

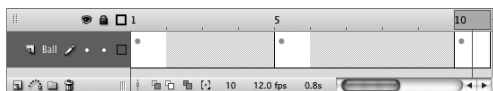
Поскольку в упражнениях из этой главы предстоит задавать свойство Tween, оставьте вкладку **Properties** в инспекторе свойств открытой. Если явно не оговорено противное, во всех упражнениях можно использовать комбинированные формы, объекты-рисунки или примитивы.

## Создание прыгающего мяча с помощью интерполяции форм

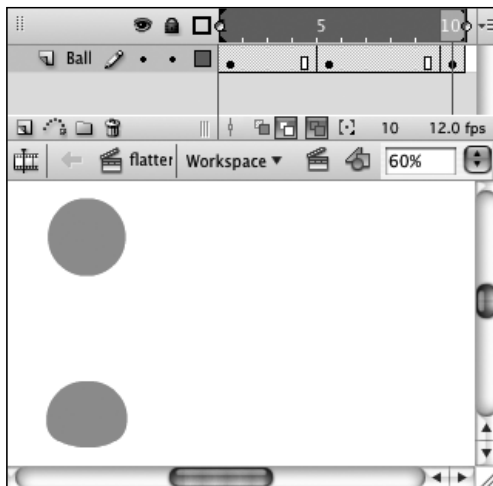
Хотя посредством интерполяции форм и можно анимировать различные свойства графического объекта — цвет, размер, положение, — все же основной функцией остается преобразование одной формы в другую. Можно было бы с помощью интерполяции форм повторить анимацию прыгающего мяча, созданную в главах 8 и 9, но интереснее сплющить мяч в момент удара о землю.

### Чтобы определить интерполяцию форм с помощью вкладки свойств кадра в инспекторе свойств:

1. В новом документе Flash или в новом слое существующего документа установите точку воспроизведения на кадр 1.
2. Выберите инструмент «Овал». Можно задать для него как комбинированный режим, так и режим рисования объектов. Не пользуйтесь инструментом «Овальный примитив».
3. Задайте цвет обводки No Color (Без цвета).
4. Нарисуйте круг у верхнего края арены. Он будет представлять мяч.
5. Выберите кадр 5 на временной шкале и выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**. Flash создаст новый опорный кадр, дублирующий содержимое предыдущего.
6. Выберите кадр 10 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe** (рис. 10.1).
7. В кадре 5 выберите мяч и перетащите его к нижнему краю арены.
8. С помощью инструментов рисования сплющите овал в кадре 5, придав ему форму, показанную на рис. 10.2.

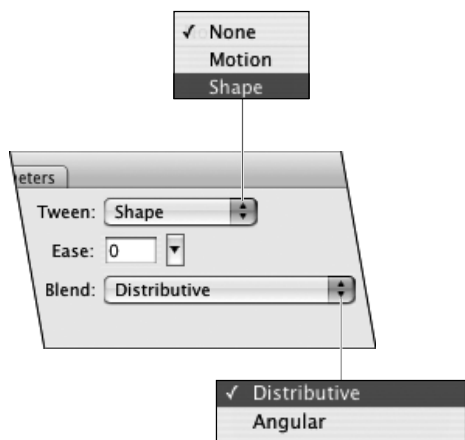


**Рис. 10.1.** Для подготовки прыгающего мяча к анимации посредством интерполяции формы, создайте опорные кадры 1, 5 и 10 так же, как в предыдущей главе.

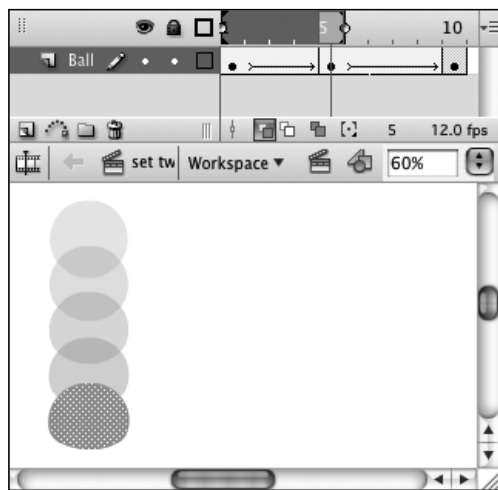


**Рис. 10.2.** Благодаря изменению формы мяча у нижнего края арены, движение будет выглядеть более естественным. Мяч сплющивается при контакте с твердой поверхностью, например, с полом. (Чтобы увидеть начальный и конечный кадры одновременно, включите показ шлейфа восковок.)





**Рис. 10.3.** При выборе пункта Shape из списка Tween на вкладке свойств кадра появляются параметры интерполяции форм.



Интерполированная последовательность при движении вниз

**Рис. 10.4.** Включив показ шлейфа восковок, вы увидите, какие промежуточные кадры Flash сгенерировал для интерполяции формы. Анимация прыгающего мяча похожа на ту, что была построена с помощью интерполяции движения. Но в данном случае изменение формы объекта создает иллюзию удара о твердую поверхность.

9. Чтобы описать интерполяцию формы для фазы опускания мяча, выберите на временной шкале любой кадр в первой последовательности (1, 2, 3 или 4). Обратите внимание, что мяч выбирается автоматически. При определении интерполяции форм анимируемый элемент должен быть выбран.

10. На вкладке **Properties** в инспекторе свойств выберите пункт Shape (Форма) из списка **Tween** (Анимация).

Появляются параметры интерполяции форм (рис. 10.3).

Flash создает переходные формы в кадрах 1–4 и помечает их цветом на временной шкале. Если был установлен режим штриховки (из меню просмотра кадров в правой части панели редактирования), то кадры, относящиеся к интерполяции формы, заштриховываются светло-зеленым цветом. Если этот режим выключен, то кадры будут белыми (Mac) или заполнены узором (Windows), но стрелочка, обозначающая наличие интерполяции, меняет цвет с черного на зеленый.

11. В списке **Blend** (Смешать) выберите один из следующих пунктов:

- чтобы сохранить резкие углы и прямые линии при преобразовании одной формы в другую, установите режим Angular (Угловой);
- для сглаживания промежуточных форм установите режим Distributive (Дистрибутив).

12. Для определения интерполяции фазы отскока выберите любой кадр из второй последовательности (5, 6, 7, 8 или 9).

13. Повторите шаги 10–12.

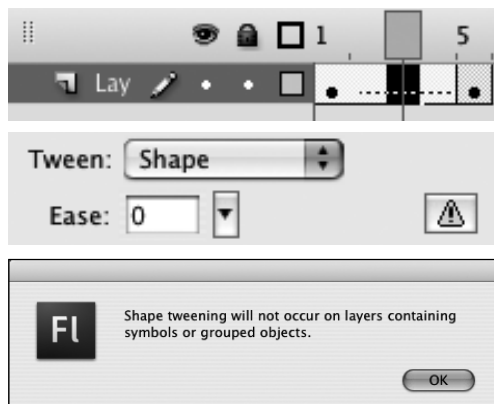
Flash создаст вторую часть траектории мяча, используя другую интерполяцию (рис. 10.4).

C

Flash не мешает вам определить интерполяцию форм для кадров, содержащих сгруппированные формы или символы, но работать такая интерполяция не будет. Flash предупреждает об этом, перечеркивая соответствующие кадры на временной шкале пунктирной линией. При выборе таких кадров на вкладке **Properties** в инспекторе свойств появляется значок предупреждения (рис. 10.5). Увидев его, разберитесь, что находится на арене. Если объект, который вы хотите анимировать, представляет собой группу или символ, воспользуйтесь интерполяцией движения. Или расщепите такой объект на составные части (выберите его и выполните команду **Modify** ⇒ **Break Apart**). Если в том же слое, что и редактируемая форма, которую вы собираетесь анимировать, находится какой-то символ или группа, перенесите мешающий объект в отдельный слой (выберите его и выполните команду **Modify** ⇒ **Timeline** ⇒ **Distribute to Layers**).

C

Отметим, что настройка скорости анимации для интерполяции форм недоступна, это можно делать только, если интерполируется движение.



**Рис. 10.5.** Если в кадрах, предназначенных для интерполяции движения, имеются группы или символы, Flash перечеркивает всю последовательность пунктирной линией, обозначающей, что интерполяция невозможна (верхний рисунок). При этом на вкладке свойств кадра появляется значок предупреждения (восклицательный знак). Щелкнув по нему, вы получите диалоговое окно с сообщением (нижний рисунок).

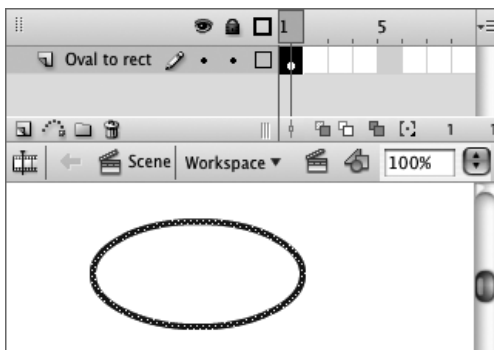


Рис. 10.6. Нарисуйте овал в первом опорном кадре, составляющем интерполяцию формы.

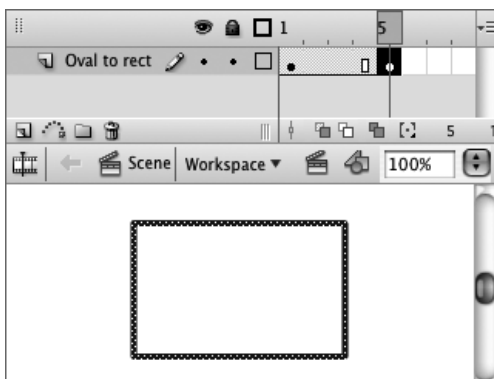


Рис. 10.7. Нарисуйте прямоугольник во втором опорном кадре, составляющем интерполяцию формы.

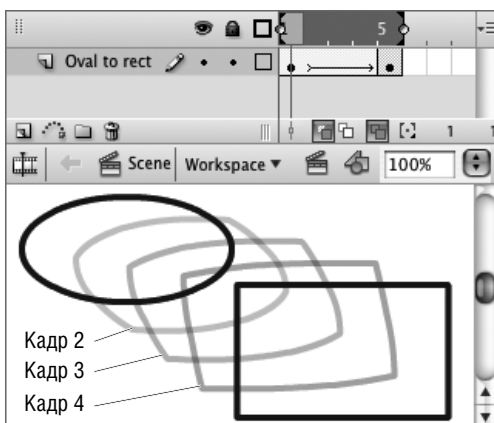


Рис. 10.8. Если указано, что кадры 1-4 составляют интерполяцию формы, то Flash создает три промежуточных формы для трансформации овала в прямоугольник. Чтобы увидеть этапы трансформации, включите показ шлейфа восков.

## Трансформация простых линий и заливок

Flash может трансформировать как заливки, так и обводки. В настоящем разделе мы научимся делать то и другое.

### Чтобы трансформировать овал в прямоугольник:

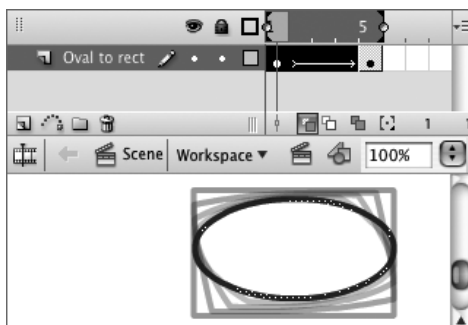
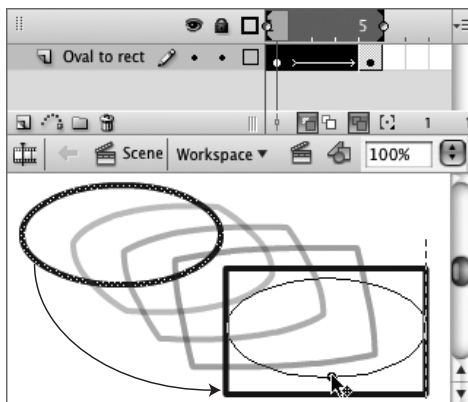
1. В новом документе или в новом слое нарисуйте контур овала в кадре 1 на арене (рис. 10.6).
2. Выберите на временной шкале кадр 5 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Blank Keyframe**. Flash вставит опорный кадр и удалит все, что находится на арене.
3. В кадре 5 на арене нарисуйте контур прямоугольника (рис. 10.7). Необязательно располагать прямоугольник там же, где был овал, это можно будет подкорректировать позже.
4. Выберите любой кадр в последовательности опорного кадра (1, 2, 3 или 4).
5. На вкладке **Properties** в инспекторе свойств выберите пункт **Shape** из списка **Tween**.

Flash трансформирует овал в прямоугольник за три шага — по одному в каждом промежуточном кадре (рис. 10.8).

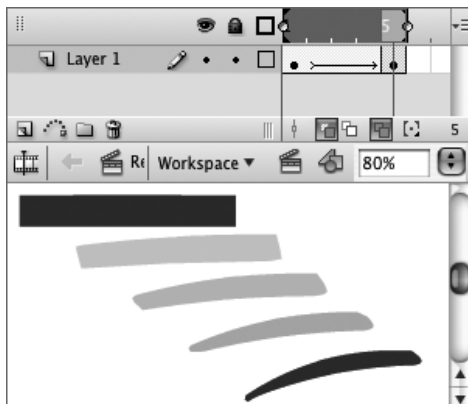
6. Чтобы совместить овал с прямоугольником, нажмите кнопку **Onion Skin** или **Onion Skin Outlines** в полосе состояния временной шкалы. Flash покажет все промежуточные кадры.
7. Поместите точку воспроизведения в кадр 1.
8. Переместите овал на арене, так чтобы он совместился с прямоугольником (рис. 10.9). Овал трансформируется в прямоугольник, оставаясь на том же месте.

### Чтобы трансформировать прямоугольник в произвольную форму:

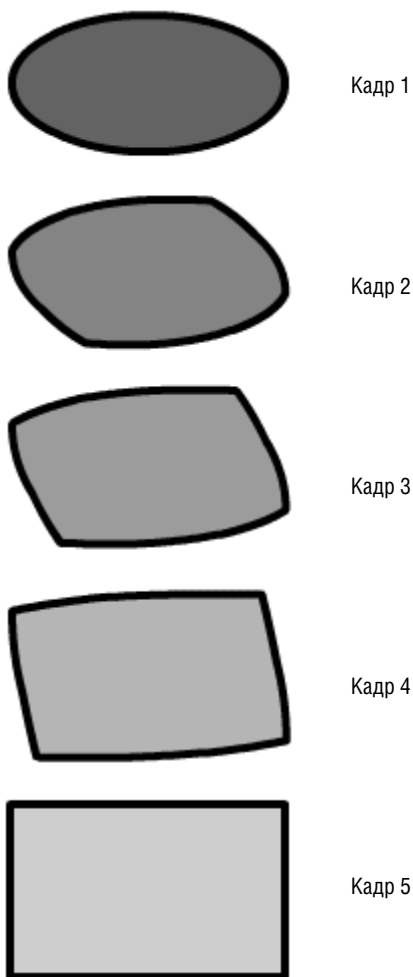
1. В новом документе или в новом слое нарисуйте прямоугольную заливку в кадре 1 на арене.
2. Выберите на временной шкале кадр 5 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Blank Keyframe**.
3. Воспользовавшись кистью, нарисуйте на арене произвольную форму в кадре 5. Не надо делать форму слишком сложной, одного пятна или мазка с плавными очертаниями хватит.
4. Выберите любой кадр в последовательности опорного кадра (1, 2, 3 или 4).
5. Установите свойство Tween равным Shape (см. шаг 5 из предыдущего упражнения). Flash трансформирует прямоугольник в произвольную форму за три шага — по одному в каждом промежуточном кадре (рис. 10.10).



**Рис. 10.9.** Включите показ восковок, чтобы точнее позиционировать формы в опорных кадрах. В данном случае, выбрав кадр 1, вы можете перетащить овал, так чтобы его центр совпал с центром прямоугольника (верхний рисунок). В результате овал будет трансформироваться в прямоугольник, оставаясь на месте (нижний рисунок).



**Рис. 10.10.** Flash трансформирует прямоугольник в произвольную форму с помощью интерполяции форм.



**Рис. 10.11.** Flash трансформирует форму в пяти кадрах. В процессе интерполяции меняется не только форма, но и цвет объекта (от темного к светлому).

## Интерполяция нескольких форм

При интерполяции движения Flash может анимировать только один объект. А при интерполяции форм в слое может находиться несколько форм. Правда, результат может получиться странным. Чем меньше форм и чем они проще, тем надежнее выполняется интерполяция нескольких форм. Чтобы результат был полностью предсказуем, ограничьтесь единственной формой в слое. Иногда, впрочем, возникает необходимость оставить в слое две формы: заливку и ее обводку. Если трансформация не слишком сложна, то Flash сумеет справиться с обеими.

### Чтобы трансформировать заливку вместе с обводкой:

1. Выполните шаги, описанные в предыдущих упражнениях, чтобы создать трансформацию овала в прямоугольник.
2. Залейте формы разными цветами.

Flash трансформирует и заливку, и обводку, меняя заодно и цвет (рис. 10.11).

C

Можно анимировать исчезновение: делать обводку (или заливку, или форму целиком) все светлее и светлее, пока от нее ничего не останется. Для этого сделайте конечный цвет обводки (и/или заливки) таким же, как у фона, или полностью прозрачным (с коэффициентом прозрачности 0).

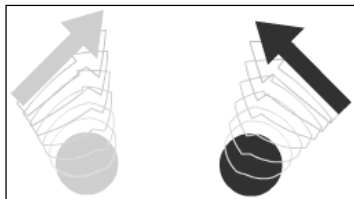
C

Чтобы гарантировать полное исчезновение элемента в предыдущем совете (на некоторых мониторах с малым количеством цветов оттенки и прозрачность могут отображаться некорректно), выберите кадр, следующий за последним опорным кадром в интерполированной последовательности. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**, которая дублирует содержимое предшествующего опорного кадра; затем удалите в нем обводку или заливку, которая должна исчезнуть. Если вы хотите, чтобы обводка и заливка исчезли одновременно, выполните команду **Insert** ⇒ **Timeline** ⇒ **Blank Keyframe**.

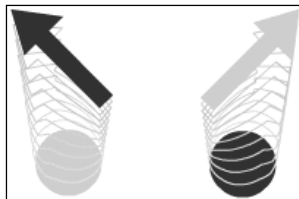
### Когда интерполяция нескольких форм работает не так, как надо

Если вы интерполируете стационарные элементы, то, скорее всего, наличие нескольких форм в одном слое не вызовет неприятностей. Но если элементы активно перемещаются, то Flash может не понять, куда какая форма движется. Даже если вы хотите, чтобы пути двух форм пересекались, Flash создаст кратчайший маршрут от начальной формы к конечной. Эта проблема иллюстрируется на рис. 10.12.

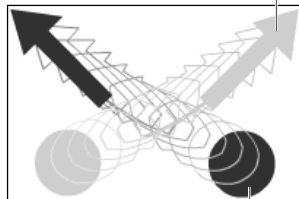
Все объекты принадлежат одному слою



Все объекты принадлежат одному слою

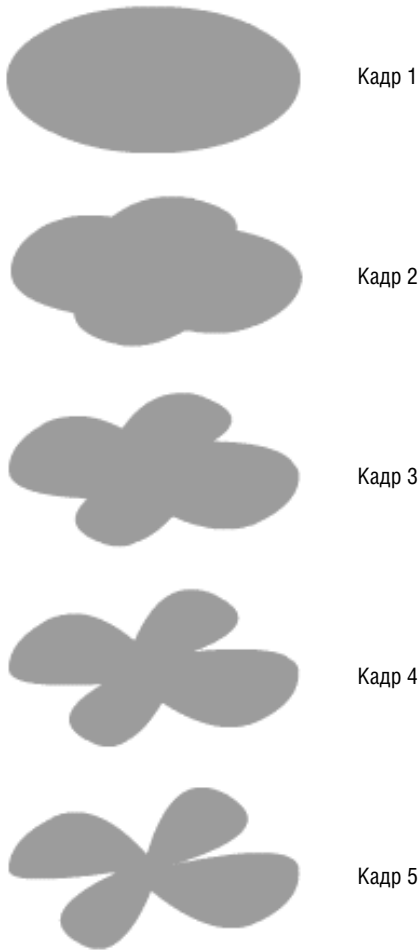


Светлый кружок и светлая стрелка в одном слое



Темный кружок и темная стрелка в другом слое

**Рис. 10.12.** Если пути нескольких форм в одном слое не пересекаются, интерполяция работает нормально. На левом рисунке оба объекта принадлежат одному слою, и светлый кружок трансформируется в стрелку без каких бы то ни было проблем. На среднем рисунке оба объекта тоже находятся в одном слое, но Flash преобразует светлый кружок в темную стрелку, а темный — в светлую, так как это кратчайший маршрут. Если вы хотите, чтобы пути перекрещивались, то нужно поместить каждый объект в отдельный слой, как показано на правом рисунке.



**Рис. 10.13.** Flash справляется с трансформацией овала в простой цветок без контрольных точек.

## Трансформация простой формы в сложную

Чем сложнее анимируемая форма, тем труднее Flash получить ожидаемый результат. Можно немного помочь, применяя контрольные точки — маркеры, с помощью которых обозначаются соответственные точки на путях исходной и конечной форм.

### Чтобы интерполировать сложную форму:

1. В новом документе или в новом слое нарисуйте овал без обводки в кадре 1 на арене.
2. Выберите кадр 5 на временной шкале и выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**. Flash продублирует содержимое опорного кадра 1 в кадре 5.
3. Выберите любой кадр в последовательности опорного кадра (1, 2, 3 или 4).
4. На вкладке **Properties** в инспекторе свойств выберите из списка **Tween** пункт **Shape**.
5. Поместите точку воспроизведения в кадр 5.
6. С помощью инструмента «Стрелка» или пера и инструмента «Спецвыделение» отбуксируйте четыре угловые точки к центру овала, так чтобы получился цветок с четырьмя лепестками. (Дополнительную информацию о редактировании форм см. в главе 4.)
7. Протестируйте ролик.

Flash хорошо справился с этой интерполяцией (рис. 10.13). Достаточно очевидно, какие точки овала следует смещать внутрь, чтобы образовались лепестки. Но если продолжить модификацию формы, то Flash уже не сможет понять, как из одной создать другую. Вот тут-то и пригодятся контрольные точки.

### Чтобы воспользоваться контрольными точками:

1. Взяв за основу анимацию, созданную в предыдущем упражнении, выберите на временной шкале кадр 10 и выполните команду **Insert ⇒ Timeline ⇒ Keyframe**.

2. В кадре 10 добавьте к цветку стебель.

Измените контур цветка с помощью инструмента «Стрелка» или пера и инструмента «Спецвыделение». Или нарисуйте стебель кистью того же цвета, что и сам цветок.

3. Задайте интерполяцию для кадров 5-9.

4. Протестируйте ролик.

Наличия стебля усложняет задачу, и Flash не может создать гладкую и естественную интерполяцию (рис. 10.14).

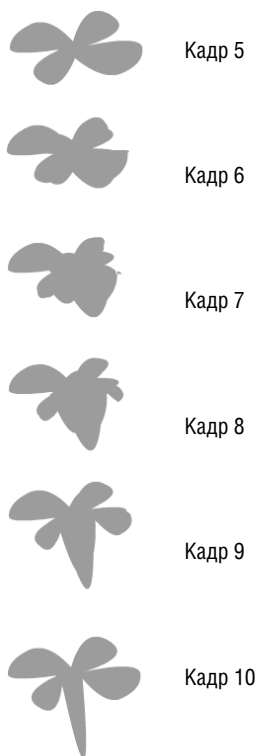
5. Перед тем как начать расставлять контрольные точки, поместите точку воспроизведения в опорный кадр 5 (начальный кадр этой интерполяции).

6. Выполните команду **Modify ⇒ Shape ⇒ Add Shape Hint** (Изменить ⇒ Форма ⇒ Добавить контрольную точку) или нажмите комбинацию клавиш **Shift-⌘-H** (Mac) либо **Ctrl-Shift-H** (Windows) (рис. 10.15).

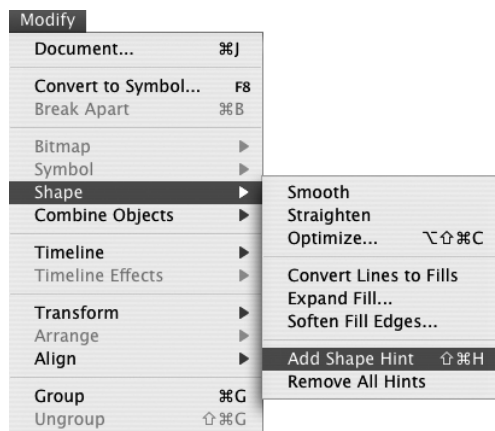
Flash помещает контрольную точку — красный кружочек с буквой, начиная с *a*, — в центр объекта в текущем кадре. Вам предстоит переместить эту точку в проблемное место на пути формы.

7. С помощью инструмента «Стрелка» перетаскивайте контрольную точку туда, где Flash испытывает трудности.

В процессе перетаскивания Flash отмечает текущее положение контрольной точки кружком, который становится темнее и больше, когда попадает на путь. Не старайтесь поставить контрольную точку абсолютно точно, важно лишь, чтобы она лежала на пути. Уточнить местоположение можно будет и позже.

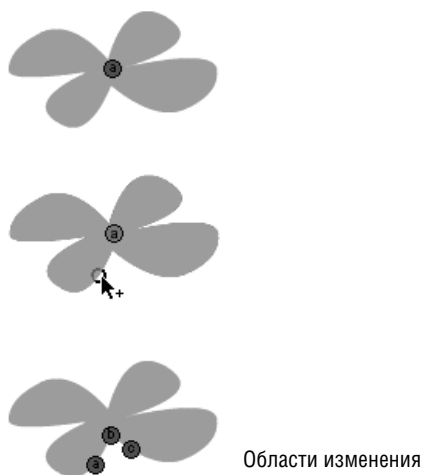


**Рис. 10.14.** Добавление к цветку стебля оказывается слишком сложно для Flash, построить гладкую интерполяцию не удастся. Особенно плохо выглядят кадры 7 и 8.

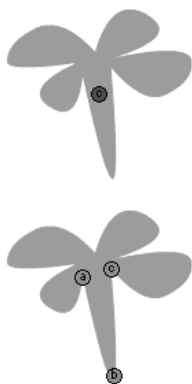


**Рис. 10.15.** Команда **Modify ⇒ Shape ⇒ Add Shape Hint** активирует расстановку маркеров, которые помогают Flash связать исходную форму с конечной.





**Рис. 10.16.** Начинайте расставлять контрольные точки в первом опорном кадре интерполированной последовательности. Flash помещает точки в центр объекта (верхний рисунок), а вы должны перетаскать их в нужные места (средний рисунок). Расставьте контрольные точки в алфавитном порядке вдоль пути объекта, помещая их туда, где происходят наиболее важные изменения (нижний рисунок). В данном случае контрольные точки а, b и с поставлены там, где от цветка отходит стебель.



**Рис. 10.17.** Чтобы завершить расстановку контрольных точек, выберите второй опорный кадр последовательности. Flash собирает все контрольные точки, соответствующие тем, что были расставлены в предшествующем кадре, в центре объекта (верхний рисунок). А вам предстоит перетаскать их в стратегически важные места на пути конечной формы (нижний рисунок).

8. Повторяйте шаги 6 и 7, пока не расставите контрольные точки во всех проблемных местах формы в кадре 5 (рис. 10.16).

При добавлении каждой контрольной точки появляется еще один красный маркер с буквой. Расставлять маркеры произвольно нельзя, они должны следовать друг за другом вдоль контура в алфавитном порядке. (Для Flash лучше всего, если вы будете расставлять контрольные точки против часовой стрелки, начиная с левого верхнего угла формы, но, если хотите, можете расставлять и по часовой стрелке.)

9. Поместите точку воспроизведения в опорный кадр 10.

Flash уже добавил контрольные точки в этом кадре, все они скопились в центре формы.

10. С помощью инструмента «Стрелка» растащите контрольные точки по своим местам на пути новой формы.

Сохраняйте тот же порядок (по или против часовой стрелки), как на шаге 8 (рис. 10.17). Когда маркер оказывается на пути конечной формы, он из красного становится зеленым; если в этот момент взглянуть на начальный кадр последовательности, то обнаружится, что те маркеры, которым установлено соответствие, стали желтыми.

11. Чтобы оценить плоды своих усилий, воспроизведите ролик.

12. Чтобы уточнить позиции контрольных точек, выберите один из опорных кадров и включите режим показа восковок.

Расставьте восковые маркеры так, чтобы охватить все кадры интерполированной последовательности. Там, где на восковках обнаруживаются неудачно поставленные контрольные точки, нужно подкорректировать соответствие между начальным и конечным кадрами (рис. 10.18). При перемещении контрольных точек изменяются промежуточные кадры. Возможно, придется подправить контрольные точки в обоих опорных кадрах. Если добиться гладкой интерполяции все равно не удастся, попробуйте увеличить число контрольных точек.

C

Чтобы удалить одну контрольную точку, сделайте начальный опорный кадр текущим. Выберите ненужную контрольную точку и перетащите ее за пределы окна документа. Или щелкните кнопкой мыши с нажатой клавишей **Control** (Mac) либо правой кнопкой мыши (Windows) по контрольной точке и выберите из контекстного меню команду **Remove Hint** (Удалить контрольную точку).

C

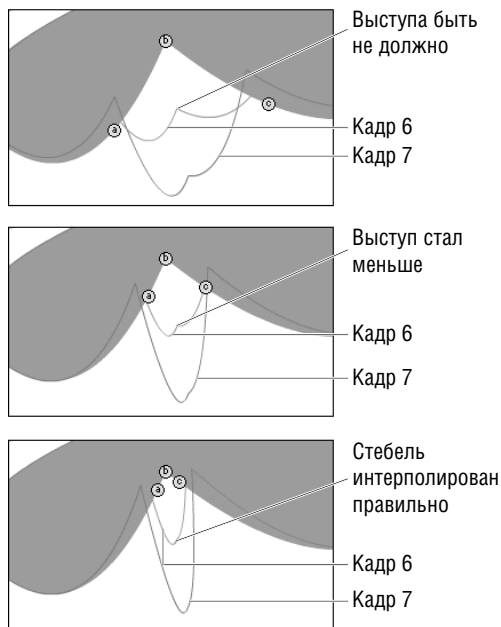
Чтобы удалить сразу все контрольные точки, сделайте начальный опорный кадр текущим и выполните команду **Modify** ⇒ **Shape** ⇒ **Remove All Hints** (Изменить ⇒ Форма ⇒ Удалить все контрольные точки). Или щелкните кнопкой мыши, удерживая клавишу **⌘** (Mac) либо **Ctrl** (Windows) по любой контрольной точке и выберите команду **Remove All Hints** (Удалить все контрольные точки).

C

При перемещении контрольных точек восковки не всегда обновляются правильно. Щелчок по пустому месту на арене заставит Flash перерисовать все восковки.

C

Если никак не удастся создать гладкую интерполяцию с помощью одних лишь контрольных точек, разбейте последовательность на более мелкие кусочки, вставив опорные кадры в те места, где трансформация происходит не так, как вы ожидаете. В этих кадрах нарисуйте формы самостоятельно.



**Рис. 10.18.** Иногда трудно точно сопоставить контрольные точки в двух опорных кадрах после начальной расстановки. Расставив маркеры в первом и последнем кадре, включите показ восковок, чтобы посмотреть, где нужно что-то подкорректировать. После начальной расстановки у растущего стебля появляется выступ (верхний рисунок). После сближения точек ситуация улучшается (средний рисунок). А когда интерполяция, отображаемая на восковках, станет гладкой, работу можно считать законченной.

### Проблемы при интерполяции примитивов

Хотя интерполяция овальных и прямоугольных примитивов возможна, результат, вероятно, покажется вам обескураживающим. Когда вы меняете форму примитива на арене с помощью управляющих точек, возникает ощущение движения. Потяните за точку, управляющую начальным или конечным углом овального примитива, и вы увидите, что радиус вращается как минутная стрелка часов. Когда Flash перерисовывает форму, вам кажется, что происходит вращение, как будто вы раскрываете и закрываете веер. Хотелось бы надеяться, что при интерполяции овального примитива это ощущение движения не исчезнет. Однако Flash в этом случае не повторяет манипуляции с управляющими точками, а анимирует изменения путей комбинированных форм, которые лежат в основе примитивов (рис. 10.19). Вы можете добавить контрольные точки, чтобы достичь желаемого эффекта или, по крайней мере, сделать переход более естественным. Но, скорее всего, примитив придется разбить на части и добавить опорные кадры. Либо применить комбинацию покадровой анимации с интерполяцией движения и форм, только так удастся воспроизвести искомое ощущение движения.

Интерполяция  
формы без  
контрольных  
точек



Интерполяция  
формы с  
контрольными  
точками



**Рис. 10.19.** Манипуляции управляющими точками для изменения начального и конечного угла овального примитива сводятся к их буксировке по или против часовой стрелки. Flash не умеет анимировать такое вращательное движение в рамках одной интерполяции формы. Flash может анимировать лишь изменения пути, в данном случае от полного овала к овалу с вырезанным сектором (верхний рисунок). Контрольные точки могут поспособствовать созданию более гладкого перехода от первой формы ко второй (нижний рисунок), но, чтобы создать ощущения раскрытия веера, их одних недостаточно.

## Создание форм, которые движутся при изменении

Невозможно создать интерполяцию формы вдоль пути, но можно перемещать формы на арене по прямой. Для этого нужно менять положение элементов на арене в разных опорных кадрах.

### Чтобы перемещать объекты в процессе интерполяции форм:

1. В новом документе или в новом слое выберите кадр 20 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame**. Flash вставит пустые промежуточные кадры в позиции от 2 до 20.
2. Выберите на временной шкале любой кадр из диапазона 2–20. Отметим, что для выбора кадра по нему нужно щелкнуть, одного лишь позиционирования точки воспроизведения недостаточно.
3. На вкладке **Properties** в инспекторе свойств выберите пункт **Shape** из списка **Tween**.

Хотя на арене пока нет ни одной формы, Flash позволяет задать режим интерполяции форм. На временной шкале кадры окажутся перечеркнутыми пунктирной линией, означающей, что интерполяция не завершена (рис. 10.20). Теперь можно добавлять опорные кадры и формы.

4. Вставьте пустые опорные кадры в позиции 5, 10, 15 и 20.

Flash создаст четыре интерполированные последовательности (рис. 10.21). Пока все они «битые», так как опорные кадры пусты.



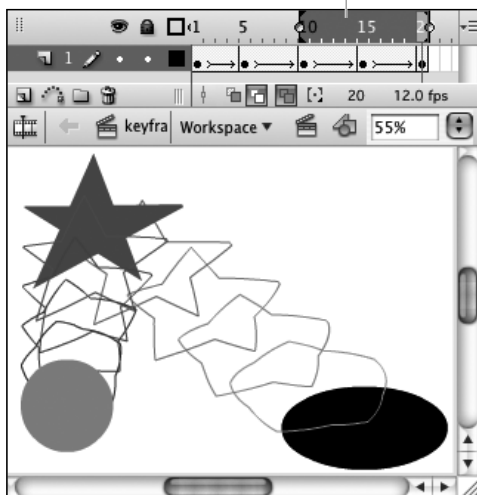
**Рис. 10.20.** Чтобы избавиться от многократной установки свойства **Tween** на вкладке свойств кадра, можно установить режим интерполяции форм сразу для диапазона кадров, а потом вставить в него опорные кадры и добавить формы. Режим интерполяции форм устанавливается несмотря на то, что никакого содержимого, которое можно было бы анимировать, еще нет.



**Рис. 10.21.** При вставке опорных кадров в длинную интерполированную последовательность Flash разбивает ее на более мелкие кусочки. Пока в опорных кадрах не появится содержимое, каждый такой фрагмент будет перечеркнут пунктирной линией, означающей, что интерполяция не завершена.



Вторая половина интерполированной  
последовательности



**Рис. 10.22.** Поместите разные формы в разных местах арены в каждом опорном кадре. Flash сгенерирует промежуточные шаги, необходимые для трансформации форм с одновременным перемещением их по арене. Воспроизведите ролик или включите показ восковок, чтобы понаблюдать за изменением форм. (На рисунке также включен режим редактирования нескольких кадров, чтобы были видны формы в опорных кадрах.)

5. Для завершения интерполяции нарисуйте в каждом кадре разные формы и разместите их в разных углах арены.

Например, в кадре 1 нарисуйте круговую заливку в левом нижнем углу, в кадре 5 — прямоугольную заливку в правом верхнем углу, в кадре 10 — овал в правом нижнем углу, в кадре 15 — звезду в левом верхнем углу, а в кадре 20 продублируйте содержимое кадра 1. Для разнообразия нарисуйте все объекты разными цветами. По мере добавления содержимого в опорные кадры Flash рисует на временной шкале соответствующие стрелки.

6. Протестируйте ролик.

Вы увидите, как графический объект пляшет по арене, изменяя форму (рис. 10.22).

## С

Хотя в режиме интерполяции форм объекты не могут перемещаться вдоль пути, как при интерполяции движения, но можно добиться желаемого эффекта, создав дополнительные опорные кадры. Сначала установите режим интерполяции форм. На временной шкале выберите весь диапазон кадров, в которых разместится интерполированная последовательность. Выполните команду **Modify ⇒ Timeline ⇒ Convert to Keyframes**. Flash преобразует все промежуточные кадры (с переходным содержанием) в опорные. Теперь можно разместить в опорных кадрах объекты в нужных местах. Чтобы эмулировать путеводитель, создайте обычный направляющий слой (см. главу 6) и нарисуйте на нем путь, вдоль которого должна происходить трансформация. Выполните команду **View ⇒ Snapping ⇒ Snap to Objects** (Просмотр ⇒ Привязка ⇒ Привязка к объектам). Расставьте формы в каждом кадре по местам. Когда форма окажется близко к пути, нарисованном в направляющем слое, Flash принудительно перенесет ее на путь.

## С

Можно поступить и по-другому. Чтобы анимировать изменение формы вдоль пути, создайте интерполяцию формы, сохраните получившуюся анимацию в символе-клипе, затем создайте интерполяцию движения, применив этот символ и путеводитель. О том, как сохранять анимацию в виде символа-клипа, вы узнаете в главе 11.

# БОЛЕЕ СЛОЖНЫЕ ЗАДАЧИ АНИМАЦИИ

---

11

Вы научились манипулировать формами, находящимися в одном слое, с одновременной анимацией. Однако программа Adobe Flash CS3 Professional способна решать и куда более хитрые анимационные задачи. Для создания сложных анимированных роликов нужно уметь работать с несколькими формами и несколькими слоями. Иногда для организации продолжительной анимации имеет смысл задействовать даже несколько сцен. В этой главе вы научитесь работать с несколькими слоями на временной шкале, раскладывать анимации по разным слоям с целью создания более сложного движения и сохранять анимации в виде повторно используемых элементов — графических символов или клипов. Овладев этими приемами, вы сможете добиться по-настоящему жизненных анимаций.

## Сцены

Если временная шкала представляет собой оглавление вашего ролика, то сцены можно упорядочить главам. Проект Flash с большим объемом анимации может включать сотни кадров. Сцены позволяют разбить это множество на более мелкие части. В ролике, созданном из обычного Flash-документа, сцены воспроизводятся по порядку, если только вы не включили интерактивные средства, позволяющие изменять естественный порядок сцен. (Дополнительную информацию об интерактивных возможностях в роликах Flash см. в главах 12 и 13). С помощью панели сцен можно посмотреть, какие сцены есть в ролике, создать новые, удалить старые и провести реорганизацию.

### Чтобы получить доступ к панели сцен:

- Если панель сцен еще не открыта, выберите из главного меню пункт **Window** ⇒ **Other Panels** ⇒ **Scene** (Окно ⇒ Другие панели ⇒ Сцена).

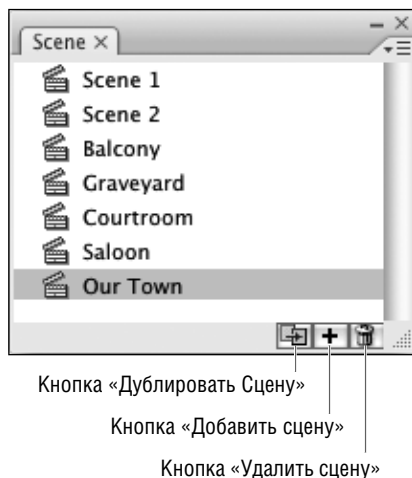
Появится панель сцен. В новом документе на панели присутствует только сцена по умолчанию с именем Scene 1 (Сцена 1). При добавлении новых сцен в ролик они появляются на панели в порядке создания (рис. 11.1).

### Чтобы добавить сцену:

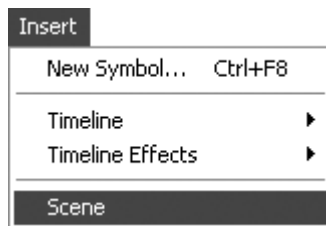
Выполните одно из следующих действий:

- выберите из меню пункт **Insert** ⇒ **Scene** (Вставить ⇒ Сцена) (рис. 11.2);
- нажмите кнопку **Add Scene** (Добавить сцену) на панели сцен.

Flash добавит новую сцену, присвоив ей по умолчанию имя Scene 2.



**Рис. 11.1.** На панели сцен перечислены все сцены в ролике. Также имеются кнопки для добавления, дублирования и удаления сцен.



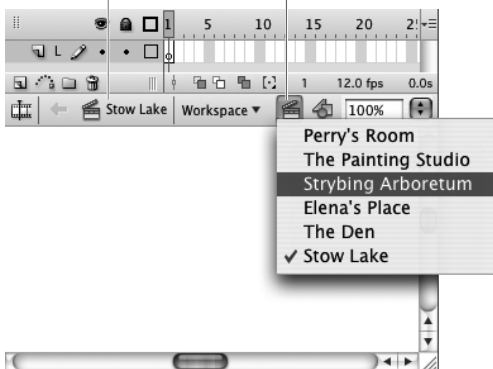
Кнопка «Добавить сцену» на панели сцен



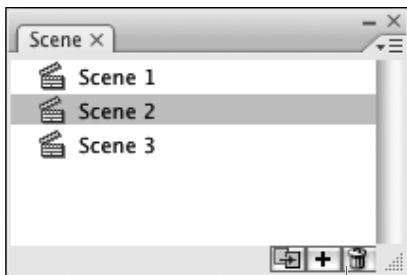
**Рис. 11.2.** Чтобы добавить новую сцену в документ, выполните команду **Insert** ⇒ **Scene** (верхний рисунок) или нажмите кнопку **Add Scene** на панели сцен (нижний рисунок).



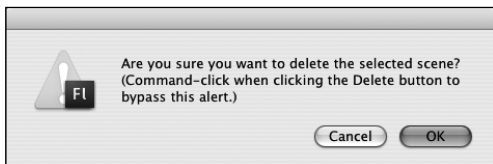
Текущая сцена      Список «Редактирование сцен»



**Рис. 11.3.** На панели редактирования отображается имя текущей сцены. Для быстрого переключения с одной сцены на другую выберите имя новой сцены из списка Edit Scene.



Нажмите, чтобы удалить выбранную сцену



**Рис. 11.4.** При удалении сцены (верхний рисунок) Flash просит подтвердить решение (нижний рисунок).

## Чтобы выбрать сцену для редактирования:

Выполните одно из следующих действий:

- Нажмите кнопку **Edit Scene** (Изменить сцену) на панели сцен.  
Появляется текущий список сцен, выберите из него нужную.
- Выберите сцену из прокручиваемого списка на панели сцен.

Flash отображает выбранную сцену на арене, помещает ее имя в поле текущей сцены на панели редактирования и помечает флажком имя этой сцены в раскрывающемся списке **Edit Scene** (рис. 11.3).

## Чтобы удалить выбранную сцену:

1. Нажмите кнопку **Delete Scene** (Удалить сцену) на панели сцен.  
Открывается диалоговое окно, в котором вас просят подтвердить решение (рис. 11.4).
2. Нажмите **OK**.

Flash удаляет сцену, убирая ее имя из раскрываемого списка **Edit Scene** и из прокручиваемого списка на панели сцен.

**C**

Если вы не хотите видеть окно подтверждения при удалении сцены, щелкните по кнопке **Delete Scene**, удерживая клавишу **⌘** (Mac) либо **Ctrl** (Windows).

### Чтобы изменить порядок сцен:

- На панели сцен перетащите выбранную сцену вверх или вниз по списку.

Flash двигает разделительную черту вслед за мышью. В Windows черта показывает новое положение сцены, в Mac она подчеркивает имя сцены, которая окажется *следующей* за перемещаемой (рис. 11.5).

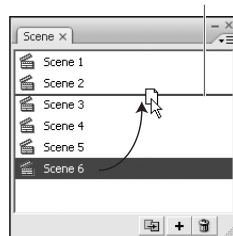
### Чтобы переименовать сцену:

1. Дважды щелкните по имени сцены, которую хотите переименовать. Поле имени становится доступным для ввода.
2. Введите новое имя сцены.
3. Нажмите **Enter**.

или

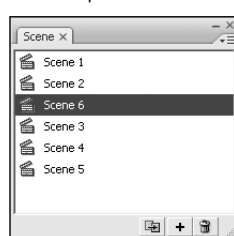
Щелкните вне поля ввода.

Windows



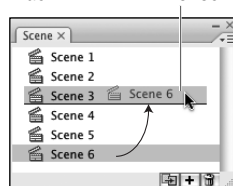
Тяните, чтобы переместить сцену

Новое положение сцены

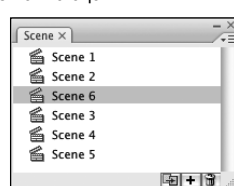


После изменения порядка сцен

Mac



Новое положение сцены



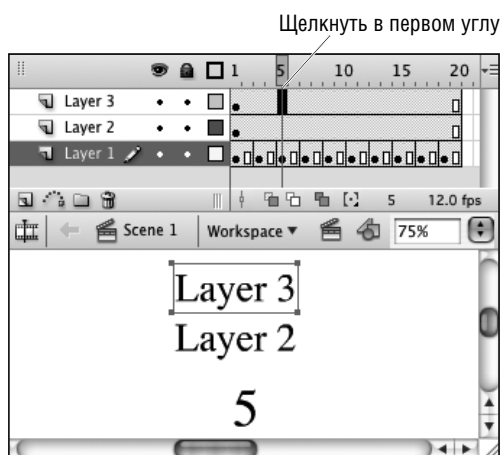
**Рис. 11.5.** Перетаскивая имя сцены на панели сцен, вы изменяете порядок следования сцен. В Windows горизонтальная черта показывает новую позицию сцены; в Mac она подчеркивает ту сцену, которая окажется после перетаскиваемой. Поместив сцену в нужное место, отпустите кнопку мыши.

### Подводные камни при работе со сценами

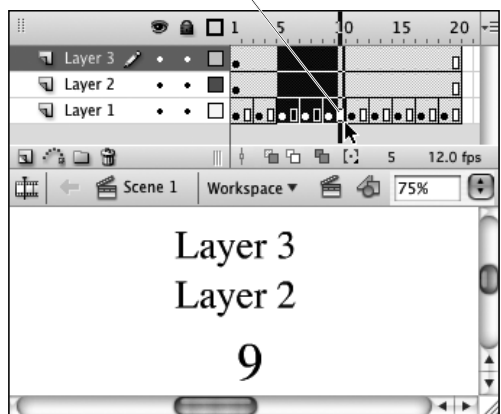
Сцена во Flash — это средство организации контента на этапе разработки; во время выполнения такого понятия не существует. При создании документа каждая сцена ведет себя как автономный ролик, но в ходе публикации Flash объединяет их в непрерывный набор кадров. (Представьте себе документ с двумя сценами: в первой находятся кадры 1–10 и во второй кадры с такими же номерами. В готовом ролике кадры будут занимать позиции с 1 по 20.)

Тот факт, что каждая сцена в некотором смысле является началом нового ролика, несколько затрудняет поддержание непрерывности действия. Для интерактивных роликов, в которых используются переменные, сцены могут оказаться непригодным инструментом. В таких случаях для организации анимации придется либо работать с одним длинным роликом, либо включить в ролик другие ролики или клипы.

Отсутствие сцен в опубликованном ролике создает и другие проблемы. Например, нужно избегать присвоения одинаковых меток кадрам из разных сцен. В противном случае не будут работать сценарии, которые ищут кадр по метке и выводят его содержимое.



Буксировать в противоположный угол



**Рис. 11.6.** В режиме одиночного выбора буксируете мышью, охватывая кадры в разных слоях (верхний рисунок), чтобы сделать их выбранными (нижний рисунок).

## Манипулирование кадрами в нескольких слоях

По мере усложнения анимации в документ приходится добавлять новые слои. Кадры можно перемещать из одного слоя в другой с помощью операций копирования, вырезания и вставки. Можно вставлять опорные кадры, в том числе пустые, в выбранный диапазон кадров или слоев.

### Чтобы выбрать и скопировать кадры из нескольких слоев:

1. Создайте документ с тремя слоями по 20 кадров в каждом.

Нарисуйте что-нибудь в кадрах, чтобы понимать, что происходит во время операции. Например, с помощью инструмента «Текст» поместите в каждый второй кадр в слое 1 его номер, а в кадры из слоев 2 и 3 — текстовый блок с именем слоя.

2. Чтобы выбрать кадры в слое 3 на временной шкале, выполните одно из следующих действий:

- в режиме одиночного выбора кадра буксируйте мышью, как будто хотите нарисовать прямоугольник, охватывающий кадры с 5 по 10 во всех трех слоях;
- в групповом режиме для выбора диапазона кадров буксируйте мышью с нажатой клавишей **⌘** (Mac) либо **Ctrl** (Windows).

Flash подсвечивает выбранные кадры (рис. 11.6).

3. Выполните команду **Edit ⇒ Timeline ⇒ Copy Frames** (Редактирование ⇒ Временная шкала ⇒ Копировать кадры).

Flash скопирует информацию о выбранных кадрах и слоях в буфер обмена.

## C

Чтобы выбрать блок кадров сразу в нескольких слоях без буксировки, в режиме одиночного выбора щелкните по кадру в любом из четырех углов блока. Затем щелкните в противоположном углу с нажатой клавишей **Shift**. Flash выбирает все кадры в определенном таким образом прямоугольнике (рис. 11.7). В режиме группового выбора щелкните в одном углу с нажатой клавишей **⌘** (Mac) либо **Ctrl** (Windows), а потом в противоположном — с нажатыми клавишами **Shift-⌘** либо **Ctrl-Shift** (Windows).

### Чтобы заменить содержимое кадров, выбранных в нескольких слоях:

1. В документе из предыдущего упражнения выберите кадры 15–20 во всех трех слоях.
2. Выполните команду **Edit ⇒ Timeline ⇒ Paste Frames** (Редактирование ⇒ Временная шкала ⇒ Вставить кадры).

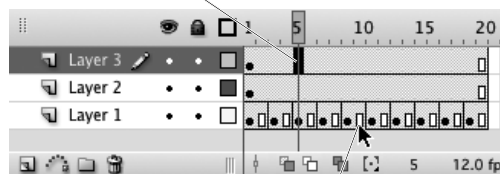
Flash вставит скопированные кадры 5–10 на место кадров 15–20 в каждом из трех слоев. Теперь в слое 1 на арене в кадре 15 будет число 5, в кадре 17 — число 7, а в кадре 19 — число 9.

### Чтобы вставить содержимое выбранных из нескольких слоев кадров в пустые кадры:

1. В том же документе выберите кадр 21 во всех трех слоях.
2. Выполните команду **Edit ⇒ Timeline ⇒ Paste Frames**.

Flash вставит скопированные кадры 5–10 на место протокадров 21–26 в каждом из трех слоев (рис. 11.8). Теперь в слое 1 в кадре 21 будет число 5, в кадре 23 — число 7, а в кадре 25 — число 9.

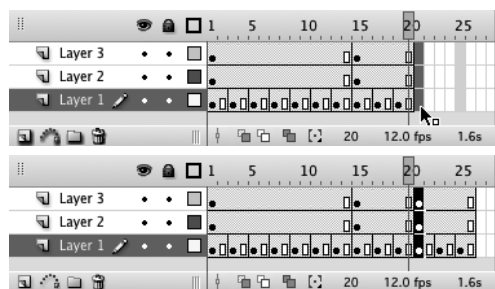
Выбрав один угол



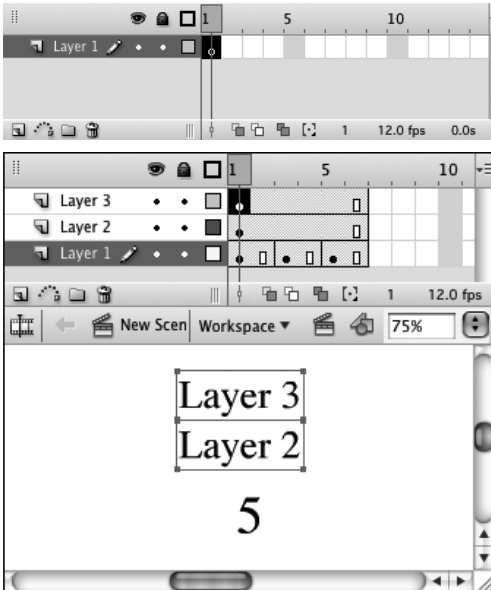
Подготовьтесь к щелчку в противоположном углу  
После щелчка с нажатой клавишей Shift



**Рис. 11.7.** Чтобы выбрать блок кадров без буксировки, в режиме одиночного выбора щелкните в любом углу блока. Затем щелкните в противоположном углу с нажатой клавишей **Shift**. В режиме группового выбора щелкните в одном углу с нажатой клавишей **⌘** (Mac) либо **Ctrl** (Windows), а потом в противоположном — с нажатыми клавишами **Shift-⌘** либо **Ctrl-Shift** (Windows).



**Рис. 11.8.** Вставка нескольких кадров из нескольких слоев в конец области определенных кадров (верхний рисунок) приводит к расширению временной шкалы вправо.



**Рис. 11.9.** Когда скопированные из нескольких слоев кадры вставляются в первый кадр на новой сцене (верхний рисунок), Flash автоматически создает необходимые дополнительные слои и кадры (нижний рисунок).

### Чтобы вставить выбранные из нескольких слоев кадры в другую сцену:

1. В том же документе вставьте новую сцену, как было описано в первом разделе настоящей главы. По умолчанию в новой сцене есть один слой с одним опорным кадром.
2. Выберите опорный кадр 1.
3. Выполните команду **Edit** ⇒ **Timeline** ⇒ **Paste Frames**.

Flash вставляет скопированные кадры из первой сцены во вторую (кадры 5–10 из слоев 1–3). При этом слои 2 и 3 автоматически добавляются, равно как и кадры 1–6 в каждом слое (рис. 11.9). Теперь в слое 1 число 5 находится в кадре 1, число 7 — в кадре 3, а число 9 — в кадре 5.

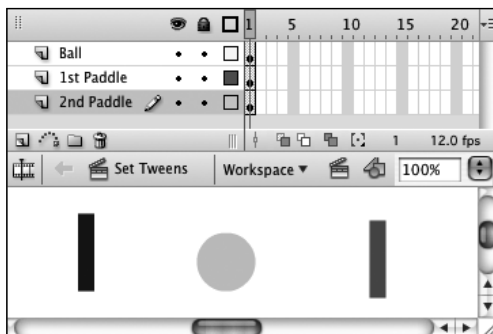
## Анимация с помощью нескольких интерполяций движения

В главе 9 было сказано, что Flash может интерполировать движение только одного объекта в слое. Для интерполяции сразу нескольких объектов необходимо поместить каждый в отдельный слой. Включив показ восковок и режим редактирования нескольких кадров, вы сможете выровнять объекты в каждый момент времени. Чтобы почувствовать, как выглядит интерполяция нескольких объектов, попробуйте объединить три простых анимации, чтобы имитировать игру в пинг-понг. В одном слое будет находиться мячик, а в двух других — ракетки.

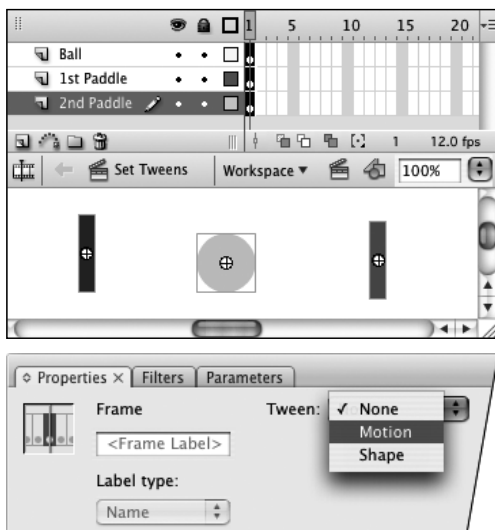
### Чтобы подготовить графические объекты в разных слоях:

1. Откройте новый документ и добавьте в него два слоя.
2. Переименуйте слой.  
Верхний слой назовите *Ball*, следующий за ним — *1st Paddle*, а нижний — *2nd Paddle*. Такие имена помогут следить за элементами и их положениями.
3. В качестве графических объектов возьмите символы (см. главу 7).

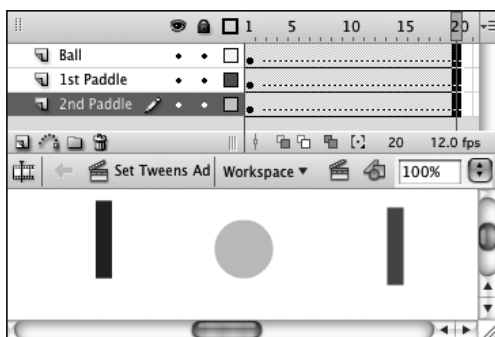
В слое *Ball* нарисуйте на арене овал, в слое *1st Paddle* — прямоугольник, который затем скопируйте в слой *2nd Paddle*. Все формы закрасьте разными цветами. Получившаяся картина должна выглядеть примерно так, как на рис. 11.10.



**Рис. 11.10.** Чтобы интерполировать движение сразу нескольких объектов, нужно поместить каждый в отдельный слой, как на этом рисунке. Осмысленные имена слоев помогают следить за тем, что где находится.



**Рис. 11.11.** Если выбраны кадры из нескольких слоев (верхний рисунок), то, задав значение Motion свойства Tween (нижний рисунок), вы подготовите интерполяцию движения во всех этих слоях.



**Рис. 11.12.** Для кадров во всех трех слоях свойство Tween равно Motion, но интерполяция не завершена, о чем свидетельствуют пунктирные линии. Нужно еще добавить опорные кадры и содержимое.

### Чтобы подготовить интерполяцию во всех слоях одной командой:

1. В документе из предыдущего упражнения выберите на арене опорный кадр 1 во всех трех слоях.
2. На вкладке **Properties** в инспекторе свойств выберите из списка **Tween** пункт **Motion**.  
Flash установит для всех кадров режим интерполяции движения (рис. 11.11).
3. Выберите на временной шкале кадр 20 во всех трех слоях.
4. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame**.

Flash интерполирует движение до кадра 20 во всех трех слоях. Пунктирная линия говорит о том, что интерполяция не завершена (рис. 11.12).

Переместите символы и создайте опорные кадры, чтобы завершить обе интерполяции.

**С**

В настоящий момент для всех слоев в документе задано значение Motion свойства Tween. Если добавить новые кадры, то у них это свойство тоже будет равно Motion. Чтобы полностью завершить интерполяцию, не забудьте сделать свойство Tween для последнего опорного кадра в каждом слое равным None.

### Чтобы скорректировать позиции интерполированных объектов:

1. В документе из предыдущего упражнения поместите точку воспроизведения в кадр 5.
2. Нарисуйте на арене мячик приблизительно в том месте, где он должен соприкоснуться с ракеткой в момент первого удара.

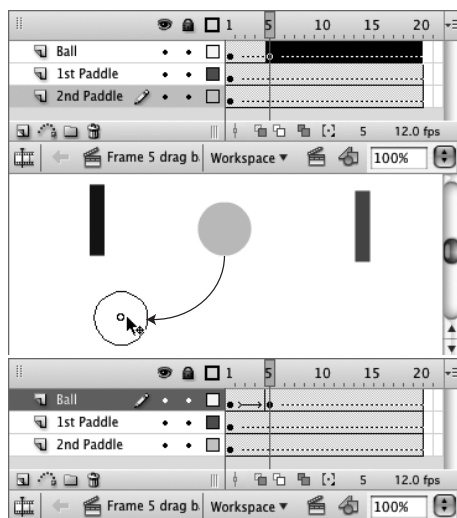
Flash сделает слой Ball активным и создаст в позиции 5 опорный кадр для нового положения мяча (рис. 11.13). Интерполяция движения между кадрами 1 и 5 в слое Ball завершается, а в остальных кадрах остается «битой».

3. Измените положение первой ракетки на арене, так чтобы она соприкасалась с мячом для нанесения первого удара.

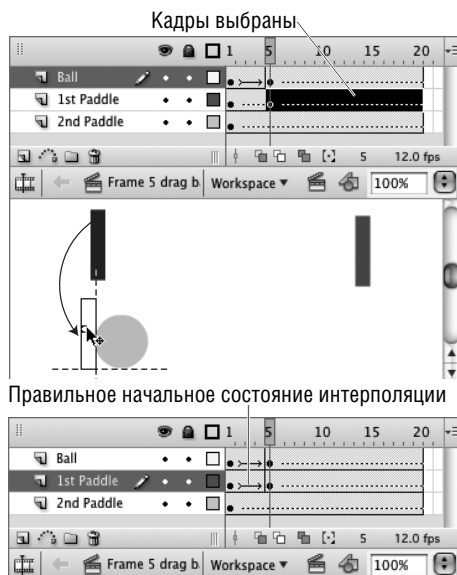
Flash сделает слой 1st Paddle активным и создаст в позиции 5 опорный кадр для нового положения ракетки (рис. 11.14).

4. Поместите точку воспроизведения в кадр 10 на временной шкале.
5. Нарисуйте на арене мячик приблизительно в том месте, где он должен соприкоснуться с ракеткой в момент второго удара.

Flash сделает слой Ball активным и создаст в позиции 10 опорный кадр для нового положения мяча. Интерполяция движения между кадрами 5 и 10 в слое Ball завершается.

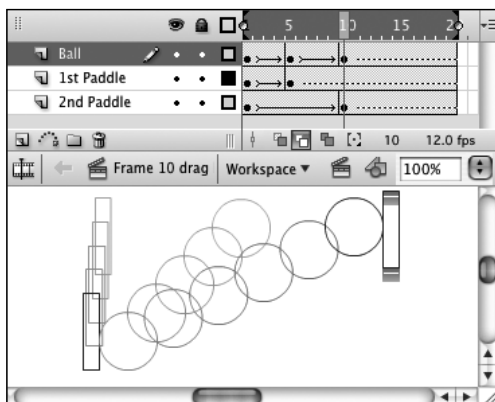


**Рис. 11.13.** Перемещение графики в кадре, определенном как часть интерполированного движения, заставляет Flash сделать содержащий эту графику слой активным (верхний рисунок). В этом слое создается опорный кадр, описывающий новое положение графики, и тем самым завершается одна интерполяция (нижний рисунок).

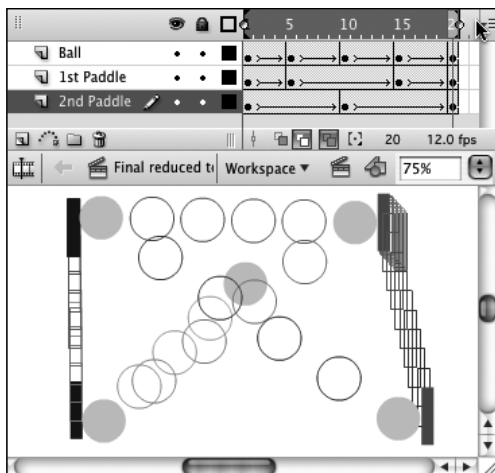


**Рис. 11.14.** После перемещения ракетки (верхний рисунок) создается впечатление, что Flash выбрал неправильный набор кадров. Однако, как только вы отпустите кнопку мыши, интерполяция будет выполнена именно в кадрах 1–4 (нижний рисунок).





**Рис. 11.15.** Перемещение элемента в другом кадре создает еще одну интерполяцию. Кажется, что правая ракетка движется медленнее левой, поскольку для правой Flash сгенерировал последовательность из 10 кадров, а для левой — только из пяти. (Чтобы были видны интерполированные формы, включен показ шлейфа восковок.)



**Рис. 11.16.** Включив режимы показа контуров восковок и редактирования нескольких кадров, легко подкорректировать расположение объектов. Здесь правая ракетка движется не по прямой. Если вам это не нравится, переместите ракетки в первом и последнем кадрах, так чтобы одна находилась точно над другой, а затем переместите мячик, чтобы он соприкасался с обеими ракетками.

6. Измените положение второй ракетки на арене, так чтобы она соприкасалась с мячом для нанесения второго удара.

Flash сделает слой 2nd Paddle активным и создаст в позиции 10 опорный кадр для нового положения ракетки (рис. 11.15).

7. Повторив шаги 1–6, создайте опорные кадры 15 и 20, в которых мяч будет ударяться о ракетки еще раз.
8. Протестируйте ролик, чтобы увидеть получившуюся анимацию в действии.
9. Установите режимы показа контуров восковок и редактирования нескольких кадров, чтобы подкорректировать движение (рис. 11.16).

Хотя в режиме редактирования нескольких кадров не показываются восковки опорных кадров, для интерполированной графики они все же видны.

**С**

После того как набор кадров сделан частью интерполяции, любое изменение объекта в каком-то из них заставляет Flash создать новый опорный кадр. Достаточно просто нажать кнопку мыши и удерживать ее секунду-другую. Чтобы случайно не изменить положение объектов и не создать тем самым лишний опорный кадр, заблокируйте или скройте слои, с которыми не работаете.

**С**

Чтобы расположить объекты на арене точнее, чем позволяет буксировка, выберите элемент и на его вкладке **Properties** в инспекторе свойств или на панели информации задайте координаты *x* и *y*.

## С

Если вам нравится использовать режим привязки с выравниванием (принят по умолчанию) для позиционирования объектов, то вы, наверное, заметили, что при выполнении предыдущего упражнения направляющие иногда не появлялись, когда вы перетаскивали графику в промежуточных кадрах, для которых свойство Tween равно Motion. Механизм направля-

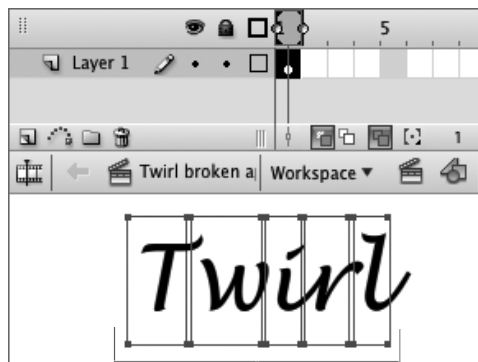
ющих работает оптимально, когда все задействованные объекты находятся в опорных кадрах. Если направляющие никак не появляются, то для начала перетащите объекты примерно в те места, где они должны находиться. После того как Flash создаст новые опорные кадры, направляющие появятся, и вы сможете уточнить местоположение объектов.

### Интерполяция текста

Интерполяцию движения можно применить и к отдельным символам внутри фрагмента текста. Создав подлежащий анимации текст, выберите его и выполните команду **Modify ⇒ Break Apart** (Изменить ⇒ Разбить). Она помещает каждый символ в отдельное текстовое поле. Затем выберите все символы и выполните команду **Modify ⇒ Timeline ⇒ Distribute to Layers** (Изменить ⇒ Временная шкала ⇒ Распределить по слоям). Каждый символ окажется в своем слое. Теперь можно воспользоваться любым приемом анимации из описанных в главах 8, 9 или еще раньше и анимировать отдельные символы текста (рис. 11.17).

Для трансформации формы литер потребуется интерполяция форм. Следовательно, литеры придется преобразовать из редактируемых текстовых элементов в редактируемые графические объекты. Выберите одно или несколько текстовых полей, содержащих отдельные литеры, и выполните команду **Modify ⇒ Break Apart**. Визуально литеры не изменились, но теперь они представляют собой элементарные формы, которые можно модифицировать, применяя инструменты рисования и интерполяцию форм. Интерполированные литеры могут выглядеть весьма замысловато. Не забывайте применять контрольные точки (см. главу 10), чтобы помочь Flash корректно выполнить переход.

Текст, распределенный по слоям

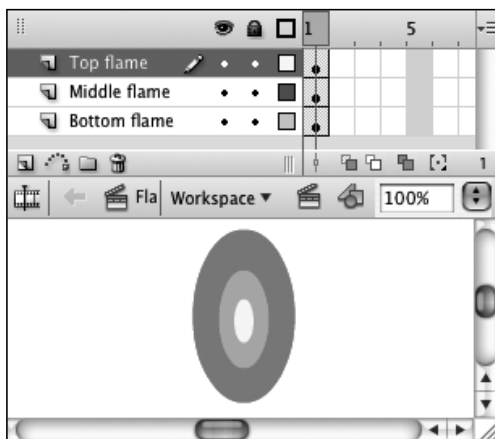


Расщепленный текст, по одному символу в поле

Каждая литера – независимая интерполяция движения



**Рис. 11.17.** С помощью команды **Modify ⇒ Break Apart** в сочетании с **Modify ⇒ Timeline ⇒ Distribute to Layers** можно быстро подготовить интерполяцию для анимации отдельных текстовых символов. Каждая литера в анимированном тексте — это независимая интерполяция.



**Рис. 11.18.** Создайте все части многокомпонентной интерполяции в разных слоях. Назовите слои так, чтобы понимать, что где находится.

### Когда следует разносить один элемент по нескольким слоям?

Часто элемент, который представляется вам единым целым, во Flash состоит из нескольких форм. Хороший пример — пламя свечи. Чтобы имитировать мерцание горячей свечи, можно было бы представить пламя с помощью трех оттенков оранжевого цвета, а затем анимировать изменения цвета и формы.

Естественно поместить все фрагменты пламени в один слой, особенно если вы создаете комбинированные формы и хотите сразу видеть, как они взаимодействуют. К сожалению, Flash с трудом интерполирует несколько форм (будь то комбинированные формы или объекты-рисунки) в одном слое. Лучше создать грубые наброски каждого фрагмента в отдельном слое, а затем подкорректировать их. А можно создать все формы в одном слое, а затем выбрать их и разнести по слоям командой **Insert ⇒ Timeline ⇒ Distribute to Layers**. В этом случае Flash должен будет интерполировать в каждом слое только одну форму, что положительно отразится на результате.

## Анимация с помощью интерполяции нескольких форм

Приступая к интерполяции сложных форм, важно помнить, что Flash ведет себя наиболее предсказуемо, когда в слое всего одна форма. В следующих упражнениях мы создадим многокомпонентную, многослойную графику и построим интерполяцию для всего пакета.

### Чтобы интерполировать формы в отдельных слоях:

1. Откройте новый документ и добавьте в него два слоя.
2. Назовите слои *Top Flame*, *Middle Flame* и *Bottom Flame* (верхняя, средняя и нижняя часть пламени). Осмысленные имена помогут следить за объектами и их перемещениями.
3. Создайте формы.

Нарисуйте на арене три концентрических овала без обводок. В слое *Bottom Flame* пусть будет большой овал, в слое *Middle Flame* — овал поменьше, а в слое *Top Flame* — самый маленький. Покрасьте овалы в разные цвета. Должно получиться нечто похожее на рис. 11.18.

4. Выберите во всех слоях кадр 5.
5. Выполните команду **Insert ⇒ Timeline ⇒ Keyframe**.

Flash создает в этой позиции в каждом слое опорный кадр с тем же содержимым, что в кадре 1.

6. Выберите любой из кадров в последовательности опорного кадра 1 (т.е. 1, 2, 3 или 4) во всех трех слоях.

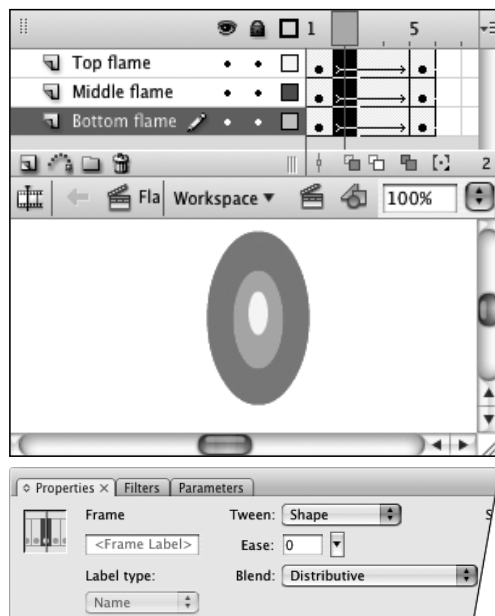
7. На вкладке **Properties** в инспекторе свойств выберите из списка **Tween** пункт **Shape**.

Flash установит для кадров 1–4 во всех слоях режим интерполяции форм (рис. 11.19). Чтобы создать эффект мерцания, форму овалов в опорном кадре 5, придется изменить.

8. Поместите точку воспроизведения в кадр 5.  
9. Измените форму овалов на арене, чтобы они больше напоминали пламя.  
10. Протестируйте ролик.

Flash выполняет интерполяцию отдельно в каждом слое. Для сравнения попробуйте создать овал и пламя в одном слое, а потом интерполировать переход одного в другое (рис. 11.20).

11. Включите показ контуров восковок и режим редактирования нескольких кадров, затем переместите объекты, составляющие пламя, добиваясь максимального правдоподобия.



**Рис. 11.19.** Выбрав несколько кадров, можно задать для всех один и тот же метод интерполяции. Для этого достаточно установить свойство Tween на вкладке Properties в инспекторе свойств.



**Рис. 11.20.** Неважно, чем вы пользуетесь — комбинированными формами или объектами-рисунками, — лишь бы эти три фрагмента пламени были помещены в разные слои (левый рисунок). Flash неплохо выполняет интерполяцию даже без контрольных точек. Если же все три формы находятся в одном слое (правый рисунок), то интерполяция оказывается неубедительной.

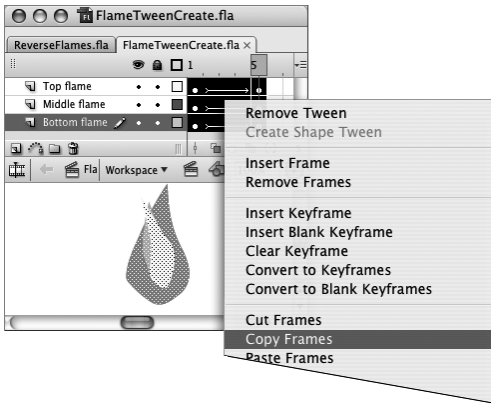


Рис. 11.21. Контекстное меню позволяет скопировать все выбранные кадры одной командой.

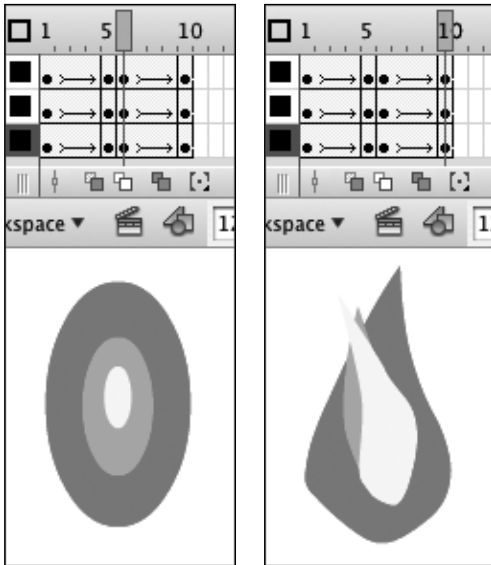


Рис. 11.22. После вставки скопированных кадров вторая последовательность начинается с овального пламени (левый рисунок) и заканчивается высоким мерцающим пламенем (правый рисунок).

## Изменение порядка кадров

Иногда можно сэкономить время, создав лишь половину анимации и позволив Flash сделать остальное. Посмотрите, как вспыхивает и опадает пламя свечи. Фаза опадания противоположна фазе вспыхивания. Поэтому вы можете анимировать только вспыхивание, а затем изменить порядок кадров на обратный.

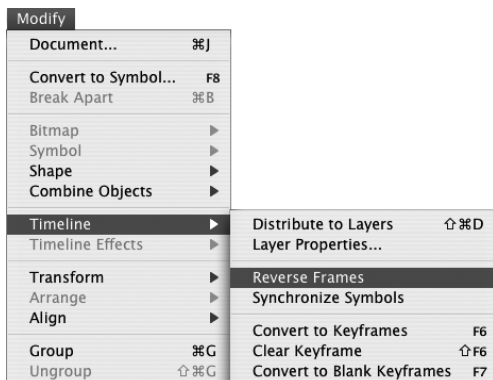
### Чтобы обратить порядок кадров:

1. Откройте документ, созданный в предыдущем разделе.
- Этот ролик занимает пять кадров в трех слоях. На первом опорном кадре пламя представлено в виде трех concentric овалов, а на последнем — вытянутое, мерцающее.
2. Выберите на временной шкале все кадры во всех слоях.
3. Щелкните мышью с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows) по любому из выбранных кадров и выберите команду **Copy Frames** (Копировать кадры) из контекстного меню (рис. 11.21).
4. Выберите на временной шкале кадр 6 во всех трех слоях.
5. Щелкните мышью с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows) по любому из выбранных кадров и выберите команду **Paste Frames** (Вставить кадры) из контекстного меню.

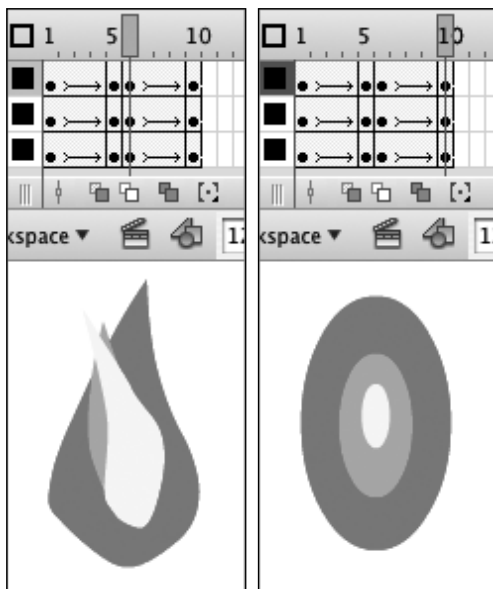
Теперь ролик содержит две одинаковые смежные последовательности кадров, на которых пламя вспыхивает (рис. 11.22).

6. Выберите кадры 6–10 во всех трех слоях.
7. Выполните команду **Modify** ⇒ **Timeline** ⇒ **Reverse Frames** (Изменить ⇒ Временная шкала ⇒ Обратить кадры) (рис. 11.23).

Flash меняет порядок кадров во второй последовательности на противоположный, так что в начале пламя высокое и мерцающее, а в конце — овальное (рис. 11.24).



**Рис. 11.23.** Команда **Modify** ⇒ **Timeline** ⇒ **Reverse Frames** обращает порядок выбранных кадров. Она применяется, чтобы воспроизвести выбранную интерполяцию в обратном направлении.



**Рис. 11.24.** После обращения кадров вторая последовательность начинается с мерцающего пламени (левый рисунок), а заканчивается овальным (правый рисунок).

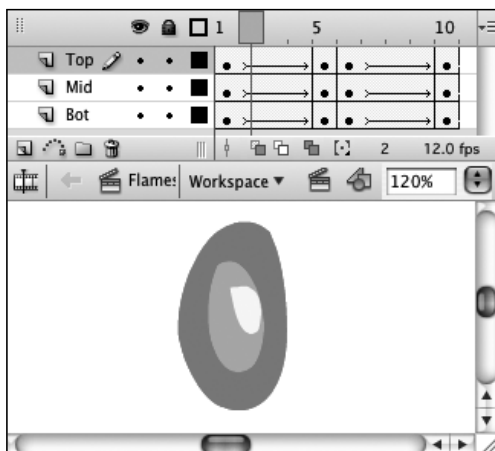


Рис. 11.25. Интерполяция, которую Flash сгенерировал в кадре 2, оставляет желать лучшего.

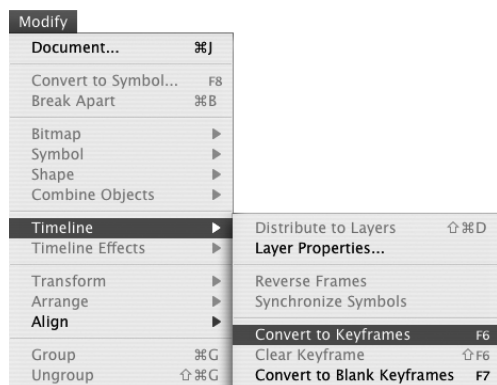


Рис. 11.26. Чтобы создать опорный кадр, в котором можно подкорректировать многослойную интерполяцию, выполните команду **Modify** ⇒ **Timeline** ⇒ **Convert to Keyframes**.

## Комбинирование интерполяции с покадровой анимацией

Не всегда можно рассчитывать, что Flash создаст промежуточные кадры именно так, как вы хотите. Особенно этот относится к интерполяции форм. Но можно комбинировать автоматическую интерполяцию с покадровой анимацией, давая Flash возможность делать то, что он может. Или позволить Flash создать набросок анимации, а затем вставить опорные кадры, чтобы уточнить движение. Flash помогает в этом, преобразуя неизменяемые промежуточные кадры в опорные, которые можно редактировать по собственному усмотрению.

В предыдущем разделе вы создали грубую версию мерцающего пламени. А теперь мы ее улучшим.

### Чтобы преобразовать промежуточные кадры в опорные:

1. Откройте документ, созданный в предыдущем разделе.
2. Поместите точку воспроизведения в кадр 2, щелкнув в точке, расположенной в этой позиции над всеми слоями, или просто перетаскив точку воспроизведения.

Первый шаг сгенерированной интерполяции не особенно впечатляет: кажется, что центральная часть пламени слишком далеко смещена в сторону (рис. 11.25). Поскольку кадр 2 промежуточный, редактировать его нельзя. Можно попытаться улучшить движение, добавив контрольные точки, или же создать новый опорный кадр, в котором уточнить анимацию.

3. Для преобразования промежуточного кадра 2 в опорный выберите его во всех трех слоях.
4. Выполните команду **Modify** ⇒ **Timeline** ⇒ **Convert to Keyframes** (Изменить ⇒ Временная шкала ⇒ Преобразовать в ключевые кадры) или нажмите **F6** (рис. 11.26).

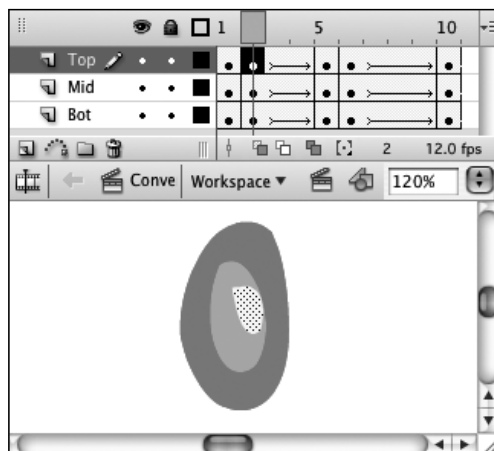
Flash преобразует промежуточный кадр 2 в опорный и копирует в него переходные формы, сгенерированные в процессе интерполяции. Теперь вы можете их усовершенствовать (рис. 11.27). Если хотите получить более гладкое движение, увеличьте число кадров между опорными кадрами 1 и 2.

5. Чтобы вставить дополнительные промежуточные кадры, поместите точку воспроизведения в опорный кадр 1.
6. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame** или нажмите F5.

Flash вставит новые промежуточные кадры во все слои. Команду **Insert** ⇒ **Timeline** ⇒ **Frame** можно повторять, пока не добьетесь желаемого результата (рис. 11.28). Вставленные кадры наследуют свойство Tween от опорного кадра 1. Теперь можно посмотреть, как новые кадры повлияли на интерполяцию, и повторить всю процедуру, если где-то еще остались шероховатости.

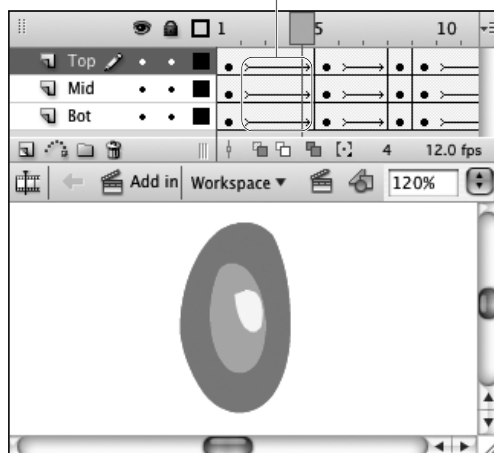
**С** На шаге 4 предыдущего упражнения, когда промежуточные кадры преобразуются в опорные, Flash создает все новые формы как комбинированные (даже если в предшествующем опорном кадре использовались объекты-рисунки или примитивы), что может привести к нежелательным последствиям, если формы принадлежат одному слою. Это еще одна причина для того, чтобы размещать интерполируемые формы в разных слоях.

**С** Flash может изменять только один цвет в каждой интерполированной последовательности. Чтобы ускорить процесс изменения нескольких цветов, подготовьте одну протяженную интерполяцию (все равно — движения или форм), в которой происходит переход от начального цвета к конечному. Затем преобразуйте некоторые промежуточные кадры в опорные, чтобы можно было произвести дополнительные изменения цвета.



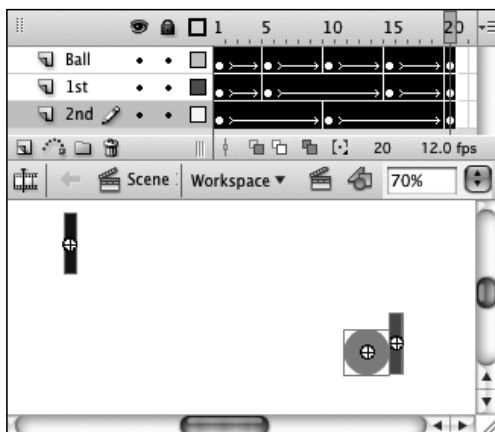
**Рис. 11.27.** Преобразовав сгенерированный промежуточный кадр 2 в опорный, вы можете изменить в нем форму пламени.

Добавленные кадры



**Рис. 11.28.** Чтобы увеличить длину новой последовательности, выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame**.





**Рис. 11.29.** Для преобразования существующей анимации в символ нужно сначала выбрать все кадры и слои, содержащие последовательность анимации.



**Рис. 11.30.** В окне Create New Symbol задается тип нового символа. Символы-клипы (фрагменты ролика) работают в собственной независимой временной шкале. Анимированные графические символы воспроизводятся синхронно с содержащим их главным роликом. В одном кадре шкалы главного ролика отображается один кадр шкалы графического символа.

## Сохранение анимаций в виде графических символов

В главе 7 вы научились сохранять плоды своих трудов в виде символов для повторного использования, уменьшая заодно размер файлов. Flash позволяет сделать то же самое для многокадровых многослойных анимационных последовательностей. Такую последовательность можно сохранить либо в виде анимированного графического символа, либо в виде символа-клипа. При повторном использовании таких символов размер файла увеличивается куда меньше, чем если бы вы заново создавали анимацию с графическими символами. Кроме того, в сложных анимациях применение символов позволяет сократить общее число кадров и слоев.

### Чтобы преобразовать анимацию в графический символ:

1. Откройте документ с анимацией игры в пинг-понг, созданный в разделе «Анимация с помощью нескольких интерполяций движения». Или создайте какую-нибудь другую многослойную анимацию. В анимации пинг-понга три слоя по 20 кадров в каждом.
2. Выберите на временной шкале все 20 кадров во всех трех слоях (рис. 11.29).
3. Выполните команду **Edit** ⇒ **Timeline** ⇒ **Copy Frames**.
4. Выполните команду **Insert** ⇒ **New Symbol** или нажмите комбинацию клавиш **⌘-F8** (Mac) либо **Ctrl-F8** (Windows). Открывается диалоговое окно Create New Symbol (рис. 11.30).
5. Введите имя символа (например, Ping-Pong-Animation).

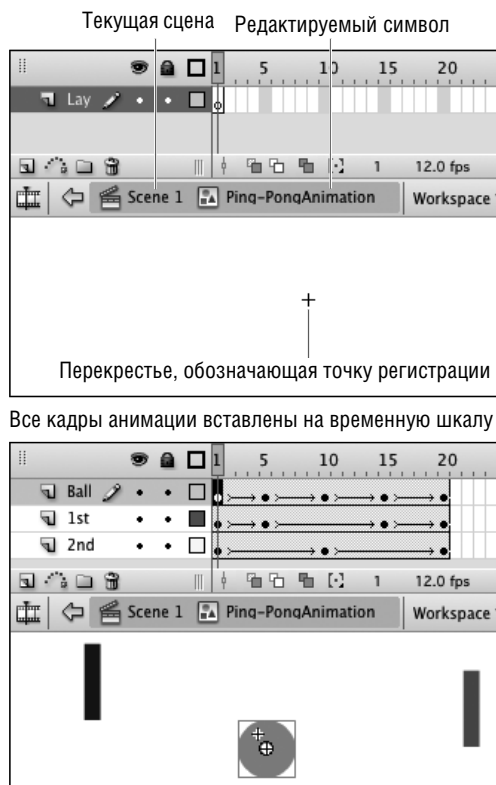
6. В качестве типа символа укажите Graphic (Графический).

7. Нажмите ОК.

Flash поместит новый символ в библиотеку и переключится в режим редактирования этого символа. Имя символа появляется на панели редактирования. По умолчанию временная шкала нового символа содержит один слой и пустой опорный кадр в позиции 1.

8. На временной шкале символа выберите кадр 1 и выполните команду **Edit ⇒ Timeline ⇒ Paste Frames**. Flash вставит 20 кадров и три слоя, скопированных из исходного ролика игры в пинг-понг на временную шкалу символа Ping-Pong Animation (рис. 11.31). Если хотите, можете подкорректировать представленную последовательность.

9. Для возврата в режим редактирования документа, выполните команду **Edit ⇒ Edit Document** (Редактирование ⇒ Редактировать документ). Flash помещает новый символ в библиотеку активного документа.



Все кадры анимации вставлены на временную шкалу

**Рис. 11.31.** При создании нового символа Flash переключается в режим редактирования символа, делая доступной временную шкалу нового символа (верхний рисунок). Чтобы создать анимированный символ, необходимо вставить все кадры анимации на временную шкалу символа (нижний рисунок).

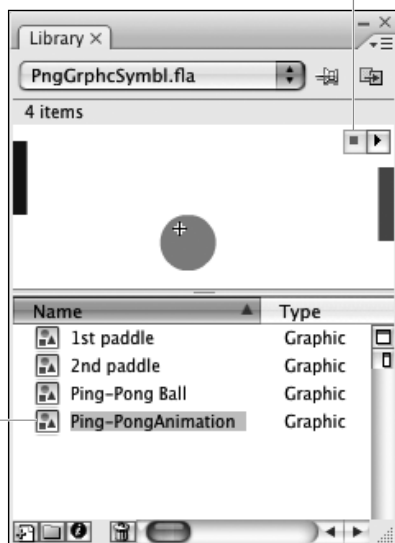
### Символы позволяют обойтись меньшим числом слоев

Вообще говоря, для интерполированных анимаций каждый символ следует помещать в отдельный слой. Например, чтобы анимировать человека, нужно создать слои для головы, туловища, каждой руки и ноги. Если анимация сложная, могут понадобиться также слои для глаз, рта, пальцев ног и рук. А если человек находится на фоне чего-то, то число слоев еще возрастет.

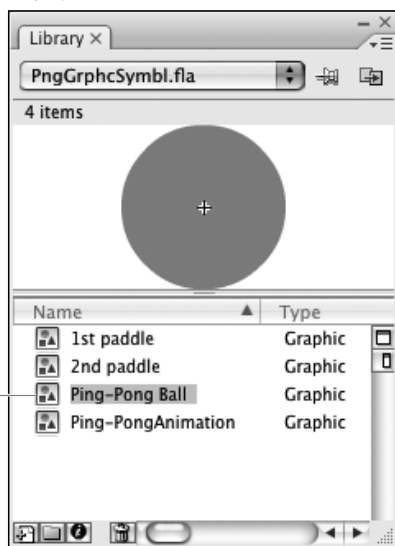
Преобразование всей анимационной последовательности в символ по сути дела схлопывает все эти слои в один объект. Этот процесс чем-то напоминает группировку. На арене символ находится в одном слое, который однако содержит все слои исходной анимации.

Анимированный графический символ

Кнопки «Воспроизвести» и «Стоп»



Статический графический символ



**Рис. 11.32.** Для анимированного графического символа (верхний рисунок) на библиотечной панели есть кнопки Stop и Play в области предварительного просмотра; больше он ничем не отличается от статического графического символа (нижний рисунок).

**С**

В списке символов на библиотечной панели анимированный графический символ выглядит точно так же, как статический; оба представлены одним и тем же значком, а в колонке Туре в обоих случаях стоит слово Graphic (Графический). Однако для анимированного символа в правом верхнем углу области предварительного просмотра есть кнопки **Play** и **Stop**, отсутствующие для статического символа (рис. 11.32). Чтобы просмотреть анимированный символ, нажмите кнопку **Play**.

**С**

При вставке нескольких кадров и слоев на временную шкалу в режиме редактирования символа перекрестье, обозначающее точку регистрации символа в целом, может оказаться в странном месте. Чтобы переместить элементы, составляющие символ, нажмите кнопку **Edit Multiple Frames** в полосе состояния, выберите режим показа всех восковок (Onion All — Все кадры в шлейфе) в меню **Modify Onion Markers** (Изменить маркеры шлейфа) и, наконец, выполните команду **Edit ⇒ Select All** (Редактирование ⇒ Выделить все), чтобы выбрать содержимое всех опорных кадров во всех слоях. Теперь можно изменить положение символа относительно перекрестья.

### Чем анимированные графические символы отличаются от символов-клипов?

Во Flash есть два вида анимированных символов: графические и клипы. Разница между ними тонкая и на первый взгляд едва уловимая. Анимированный графический символ привязывается к временной шкале включающего ролика, тогда как клип воспроизводится в независимой временной шкале. Когда точка воспроизведения на основной временной шкале перестает двигаться, анимированный графический символ не проигрывается, а символ-клип продолжает проигрываться.

Можно представлять себе кадры анимированного графического символа как стопку слайдов, в символ-клип — как киноленту. Анимированный графический символ проецирует свои слайды, по одному на кадр синхронно с кадрами включающего ролика: чтобы увидеть следующий кадр символа, нужно перейти к следующему кадру ролика. Как и у стопки слайдов, у анимированного графического символа нет звуковой дорожки. Если в ролике было звуковое сопровождение, и вы преобразовали его в символ, то звук будет потерян. Если к кнопкам присоединены сценарии на языке ActionScript 1.0 или 2.0, то интерактивность сохранится и в графическом символе (хотя все действия, управляющие временной шкалой, будут относиться к временной шкале включающего ролика, а не самого графического символа, что может приводить к путанице).

Сценарии на языке ActionScript 3.0, находившиеся в кадрах, преобразованных в графический символ, не публикуются. Действия, запрограммированные на языке ActionScript 3.0, находящиеся внутри символов, которые вошли в анимированный графический символ, переживают процедуру преобразования, но могут управлять не той временной шкалой, что вы ожидаете; это зависит от настройки. Прочие виды интерактивности после преобразования теряются. (Подробнее об интерактивности и языке ActionScript 3.0 см. главу 13). Символ-клип может проецировать все свои кадры один за другим в единственном кадре включающего ролика. У клипов есть звуковая дорожка, и они в полной мере сохраняют интерактивность. (Подробнее о звуковом сопровождении см. главу 15; об интерактивности — главы 12 и 13).

И еще одно: поскольку символы-клипы воспроизводятся согласно собственной временной шкале, они не представлены как анимации в среде разработки Flash. На арене вы видите лишь статический элемент, составляющий содержимое первого кадра клипа. Анимированные графические символы, разделяющие временную шкалу с основным роликом, показывают содержащуюся в них анимацию в среде разработки.

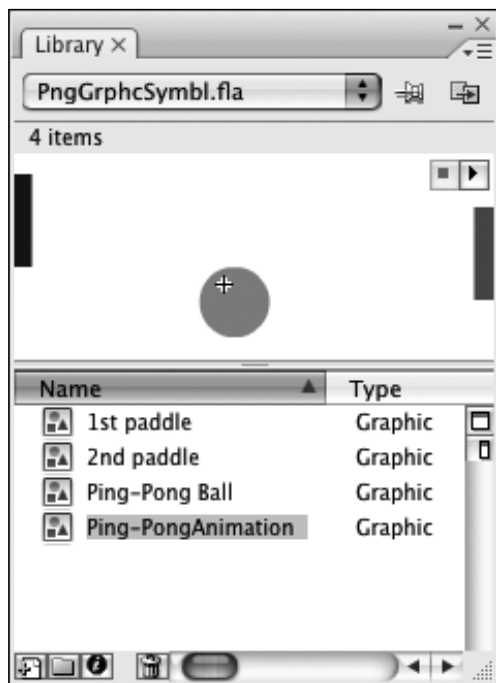


Рис. 11.33. Выберите графический символ и перетащите его в свой документ, чтобы поместить экземпляр на арену.

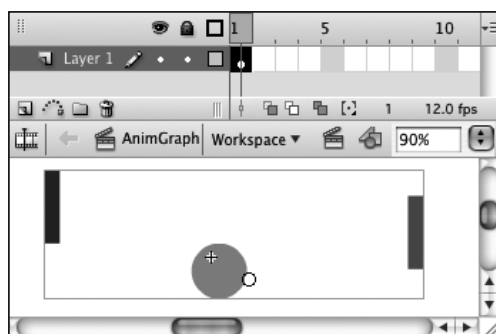


Рис. 11.34. После перетаскивания экземпляра анимированного графического символа на арену вы видите только первый его кадр, причем все графические объекты в нем выбраны. Чтобы просмотреть хранящуюся в данном символе анимацию целиком, необходимо добавить дополнительные кадры.

## Использование анимированных графических символов

Чтобы воспользоваться анимированным графическим символом, необходимо поместить его экземпляр в ролик. В том слое, куда вставляется символ, должно быть достаточно кадров для его отображения. Работать с экземплярами анимированных графических символов можно точно так же, как с любыми другими символами, — сочетать с другой графикой в том же слое, применять интерполяцию движения, изменять цвет, размер и угол поворота и т.д.

### Чтобы поместить экземпляр анимированного графического символа:

1. Откройте ролик, созданный в предыдущем разделе, и выполните команду **Insert** ⇒ **Scene** (Вставить ⇒ Сцена) для создания новой сцены.

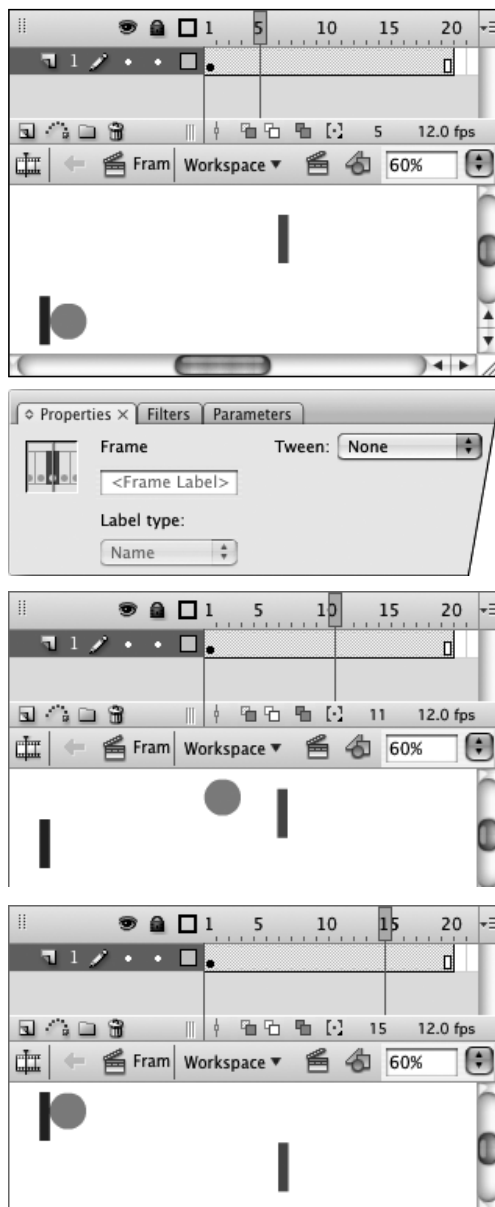
Flash отображает временную шкалу новой сцены — единственный слой с пустым опорным кадром в позиции 1. Арена пуста. Новая сцена предоставляет вам пустую арену для работы и упрощает сравнение двух анимаций: исходной (созданной непосредственно на временной шкале основного ролика) и входящей в экземпляр помещенного в ролик графического символа.

2. Перейдите на библиотечную панель. Если она закрыта, выполните команду **Window** ⇒ **Library**.
3. Выберите из библиотеки документа символ **Ping-Pong Animation**. В окне предварительного просмотра появляется первый кадр анимации (рис. 11.33).
4. Перетащите копию выбранного символа на арену.

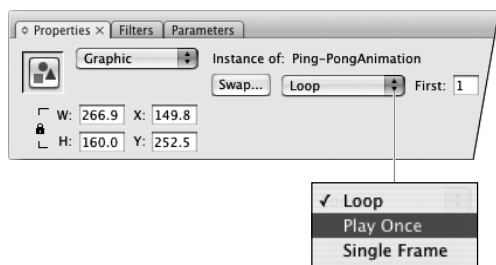
Flash помещает символ в кадр 1. В этот момент виден только первый кадр анимации (рис. 11.34). Всего анимация занимает 20 кадров, поэтому для просмотра символа целиком нужно будет добавить по меньшей мере столько же кадров.

5. Выберите на временной шкале кадр 20 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame**. Flash добавит промежуточные кадры 2–20.
6. Протестируйте ролик.

Теперь Flash может показать в ролике все кадры из анимированного графического символа. В кадре 2 ролика отображается кадр 2 символа и так далее (рис. 11.35). Если добавить в ролик менее 20 кадров, то Flash покажет лишь столько кадров символа, сколько есть кадров в ролике.



**Рис. 11.35.** Свойство Tween для кадров 1-20 равно None, тем не менее анимация показывается. Flash отображает первые 20 кадров графического символа, помещенного в опорный кадр 1. Можно сравнить символ со стопкой слайдов, каждый из которых проецируется в отдельном кадре основного ролика. Если основной ролик оказывается длиннее, то показ слайдов начинается с начала.



**Рис. 11.36.** По умолчанию анимированные графические символы синхронизированы с временной шкалой основного ролика. Если символ содержит меньше кадров, чем содержащий его слой, то воспроизведение начинается сначала. Чтобы предотвратить заикливание, выберите из списка Options for Graphics на вкладке Properties пункт Play Once.

**С**

По умолчанию Flash заикликает анимацию графических символов. Если число кадров в слое, содержащем анимированный графический символ, больше, чем необходимо для его воспроизведения, Flash начинает проигрывать символ с начала, пока все кадры не будут заполнены. Это можно запретить. Выберите экземпляр символа на арене и на вкладке **Properties** в инспекторе свойств выберите из списка **Options for Graphics** (Параметры для графики) команду **Play Once** (Воспроизвести один раз) (рис. 11.36). Можно еще назначить в качестве начального не первый кадр символа. Для этого введите номер кадра в поле First (Первый). (Можно даже потребовать, чтобы воспроизводился только один кадр анимированного символа. Для этого в списке **Options for Graphics** выберите пункт **Single Frame** и введите номер интересующего вас кадра в поле First. Этот прием позволяет приостановить анимацию символа в течение заданного числа кадров. В том месте, где показ символа следует возобновить, вставьте новый опорный кадр, поместите в него другую копию символа и установите режим циклического воспроизведения.)

### Анимированные графические символы увеличивают размер файла

Хотя символы предназначены для того, чтобы уменьшить размер файла, анимированные графические символы в этом отношении уступают символам-клипам. Для каждого промежуточного кадра, в котором отображается анимированная графика, Flash экспортирует от 12 до 15 байтов. Это справедливо даже в том случае, когда вы добавляете промежуточные кадры лишь для того, чтобы заикливать анимацию. Для воспроизведения символа клипа никакие промежуточные кадры вообще не нужны.

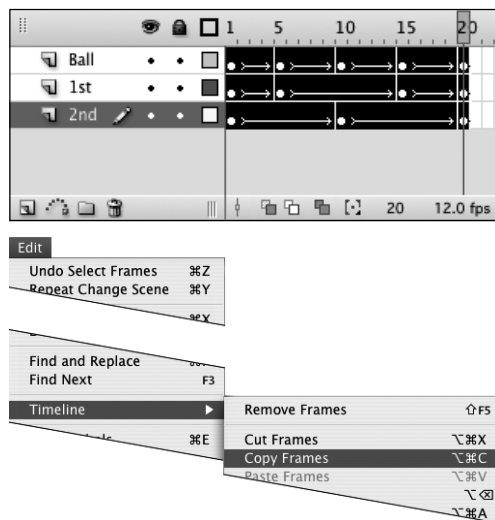
Если в анимированный символ вложен другой анимированный символ, то проблема только усугубляется, так как для каждого промежуточного кадра, принадлежащего вложенному символу, нужно те же 12–15 байтов. Повторное использование символов-клипов, в том числе вложенных, почти не увеличивает размер файла. Чтобы убедиться в этом, включите анимированный графический символ внутрь него самого и результат поместите на арену, зарезервировав последовательность из 20 кадров. Протестируйте ролик (командой **Control ⇨ Test Movie**). С помощью профилировщика пропускной способности (см. главу 17) посмотрите, сколько места занимает каждый кадр. Теперь вернитесь в документ, измените тип символа на Movie Clip и повторите тест.

## Сохранение анимации в виде символа-клипа

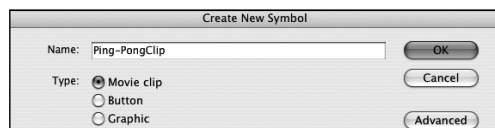
Процедура сохранения анимации в виде символа-клипа отличается от сохранения в виде анимированного графического символа только тем, что в диалоговом окне Create New Symbol в качестве типа символа задается Movie Clip.

### Чтобы преобразовать анимацию в символ-клип:

1. Откройте документ с анимацией игры в пинг-понг, созданный в разделе «Анимация с помощью интерполяции движения». Эта анимация состоит из трех слоев по 20 кадров в каждом.
2. Выберите на временной шкале все 20 кадров во всех трех слоях.
3. Выполните команду **Edit** ⇒ **Timeline** ⇒ **Copy Frames** (рис. 11.37).
4. Выполните команду **Insert** ⇒ **New Symbol** или нажмите комбинацию клавиш **⌘-F8** (Mac) либо **Ctrl-F8** (Windows). Открывается диалоговое окно Create New Symbol.
5. Введите в поле Name имя символа, например, Ping-PongClip. Flash помнит, какой тип символа вы задавали в последний раз, и по умолчанию предлагает точно такой же.
6. Задайте в качестве типа символа Movie Clip (Фрагмент ролика) (рис. 11.38).

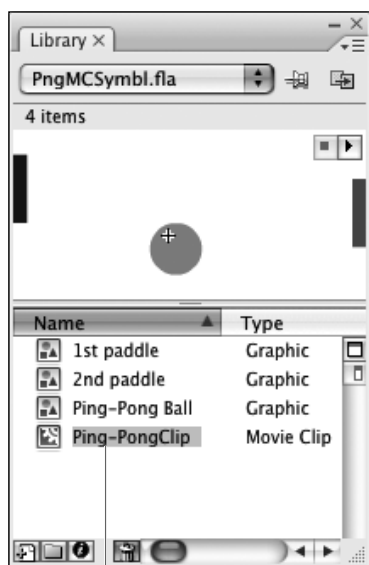
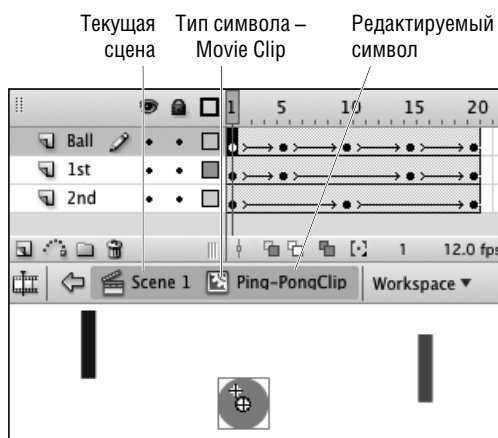


**Рис. 11.37.** Чтобы сохранить анимацию в виде символа, выберите все составляющие ее кадры и слои, а затем выполните команду **Edit** ⇒ **Timeline** ⇒ **Copy Frames**.



**Рис. 11.38.** Для определения символа с независимой временной шкалой задайте тип Movie Clip в окне Create New Symbol. Символ-клип целиком воспроизводится в одном кадре основного ролика.





Редактируемый клип

**Рис. 11.39.** После того как символу присвоены имя и тип, Flash переключается в режим редактирования символа. Значок слева от имени символа показывает, что это именно клип. Теперь можно вставить кадры анимации на временную шкалу символа.

## 7. Нажмите **ОК**.

Flash создаст новый символ на библиотечной панели и переключится в режим редактирования этого символа.

Имя символа появляется на панели редактирования. По умолчанию временная шкала нового символа содержит один слой и пустой опорный кадр в позиции 1.

## 8. На временной шкале символа выберите кадр 1 и выполните команду **Edit** ⇒ **Timeline** ⇒ **Paste Frames**. Flash вставит 20 кадров и три слоя, скопированных из исходного ролика игры в пинг-понг на временную шкалу символа Ping-PongClip (рис. 11.39). Если хотите, можете подкорректировать представленную последовательность.

## 9. Для возврата в режим редактирования документа щелкните по имени текущей сцены на панели редактирования.

**С**

Чтобы символ-клип содержал столько же кадров, сколько в уже существующем анимированном графическом символе, можно просто продублировать последний и изменить его тип. Выберите анимированный графический символ на библиотечной панели. Из меню панели выберите команду **Duplicate**. Откроется диалоговое окно **Duplicate Symbol**, в котором можно переименовать символ и изменить его тип на **Movie Clip**.

### Неискажающее масштабирование

Одна из примечательных особенностей Flash — это возможность повторного использования элементов. Но при масштабировании повторно используемого элемента он может исказиться. Допустим, вы создали прямоугольник со скругленными углами. Масштабировать его можно для разных целей; он может стать элементом графического интерфейса (кнопкой) или частью художественной анимации. Хочется надеяться, что степень скругления углов останется той же самой вне зависимости от размеров прямоугольника. Но на деле углы иногда искажаются по вертикали, а иногда по горизонтали.

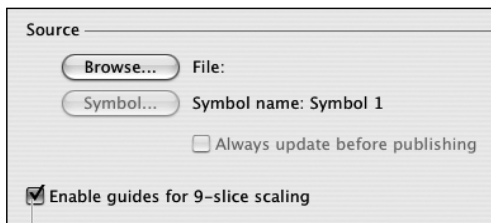
Искажение может проявиться на этапе создания содержимого документа или во время воспроизведения ролика. Последнее случается, если размеры элементов динамически изменяются сценарием, или в результате изменения размера окна браузера, в котором ролик воспроизводится. Во втором случае вы можете до некоторой степени контролировать происходящее с помощью различных параметров публикации (см. главу 17).

Во Flash CS3 при публикации для проигрывателей Flash Player 9 или 8 можно наложить ограничения на способы масштабирования символов-клипов, включив режим неискажающего (9-дольного) масштабирования (9-slice scaling). В этом режиме Flash накладывает на символ-клип сетку. Угловые ячейки сетки не масштабируются вовсе; все остальные уменьшаются или увеличиваются, как положено, после чего на результат «наклеиваются» исходные уголки. Этот способ позволяет сохранить углы элемента неизменными при любом масштабе.

Чтобы включить режим неискажающего масштабирования в момент создания символа-клипа, откройте дополнительную секцию окна свойств символа (кнопка **Advanced** (Дополнительно)) и отметьте флажок **Enable guides for 9-slice scaling** (Включить направляющие для 9-дольного масштабирования) (рис. 11.40). Если потребуется включить этот режим позже, выберите символ на библиотечной панели, выберите пункт **Properties** (Свойства) из меню панели, чтобы открыть окно свойств символа, и там отметьте флажок **Enable guides for 9-slice scaling**. (Режим неискажающего масштабирования можно включить и программно из ActionScript-сценария).



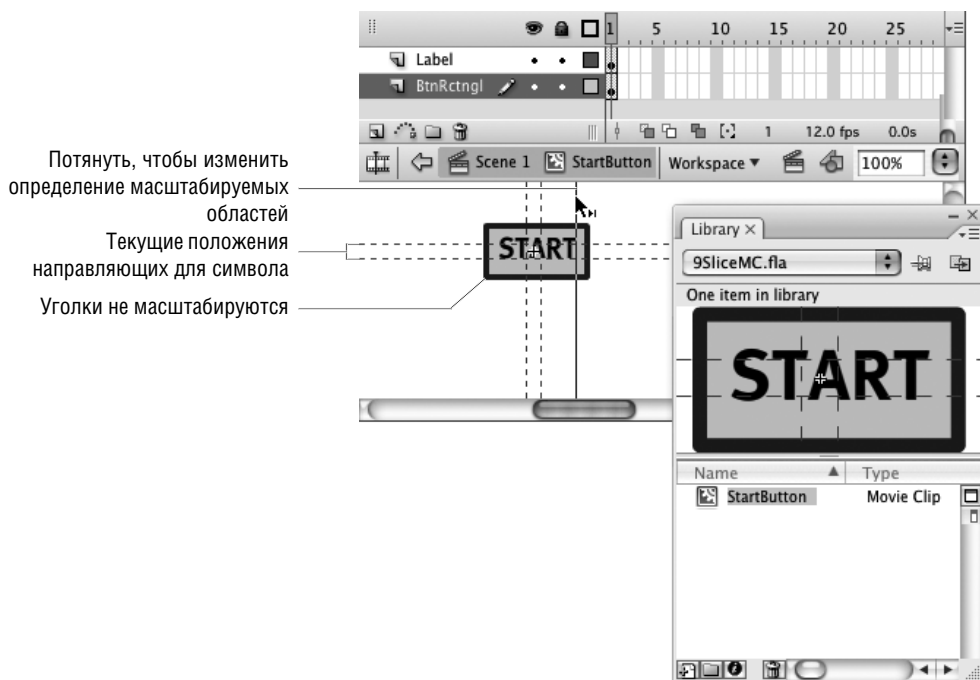
Нажмите для показа дополнительных настроек



Отметьте, чтобы включить направляющие 9-дольного масштабирования

**Рис. 11.40.** Чтобы включить режим неискажающего масштабирования для символов-клипов, откройте дополнительную секцию в окне Create New Symbol (верхний рисунок) и отметьте флажок Enable guides for 9-slice scaling (нижний рисунок).

Если режим неискажающего масштабирования включен, то в области предварительного просмотра символа на библиотечной панели, а также при редактировании символа появляются направляющие линии (рис. 11.41). В режиме редактирования символа эти линии можно буксировать, чтобы определить уголки, подходящие для конкретного изображения.



**Рис. 11.41.** Если режим неискажающего масштабирования включен, на библиотечной панели видны направляющие. В режиме редактирования символа эти направляющие появляются на арене, где их можно отбуксировать в нужное место.

Есть еще один способ избежать искажений при изменении размеров графического объекта в среде разработки. Создайте объект как примитивную форму и измените ее ширину или высоту в инспекторе свойств, на панели преобразований или на панели информации. При вводе новых значений высоты или ширины примитива Flash применяет технику неискажающего масштабирования. Но при масштабировании примитивной формы инструментом «Свободное преобразование» искажения все равно будут возникать.

## Использование символов-клипов

Чтобы воспользоваться символом-клипом, надо поместить его экземпляр на арену в разрабатываемом документе. В отличие от анимированных графических символов, символы-клипы обладают собственной временной шкалой. Клип воспроизводится непрерывно, как короткий фильм, в одном кадре основного ролика. Коль скоро клип не содержит команды останова, например пустого опорного кадра на временной шкале того слоя, который содержит клип, он продолжает проигрываться.

При разработке документа виден только первый кадр символа-клипа. Чтобы просмотреть всю анимацию в контексте остальных элементов ролика, необходимо экспортировать ролик (например, выбрав один из режимов тестирования). Просмотреть заключенную в клипе анимацию автономно можно на библиотечной панели.

### Чтобы поместить экземпляр символа-клипа:

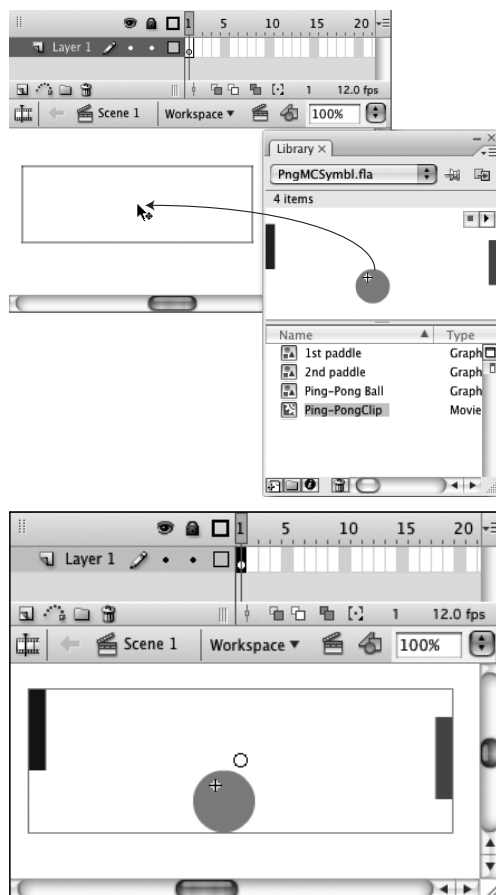
1. Продолжая работу с документом из предыдущего упражнения, выполните команду **Insert ⇒ Scene**.

Flash создаст новую сцену и покажет ее временную шкалу: единственный слой с пустым опорным кадром в позиции 1. Арена пуста.

Новая сцена предоставляет вам пустую арену для работы и упрощает сравнение двух анимаций: исходной (созданной непосредственно на временной шкале основного ролика) и входящей в экземпляр помещенного в ролик графического символа.

2. Перейдите на библиотечную панель. Если она закрыта, выполните команду **Window ⇒ Library**.
3. Выберите из библиотеки документа символ Ping-PongClip. В окне предварительного просмотра появляется первый кадр анимации.
4. Перетащите копию выбранного символа на арену.

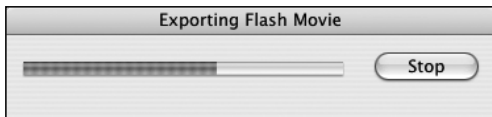
Flash помещает символ в кадр 1 (рис. 11.42). Не нужно вставлять никаких дополнительных кадров, однако, чтобы увидеть анимацию, ролик необходимо экспортировать.



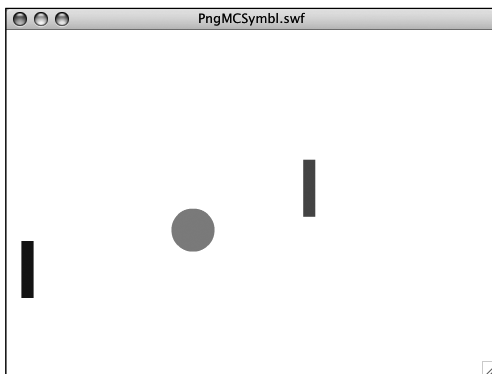
**Рис. 11.42.** Перетащите экземпляр символа-клипа с библиотечной панели на арену (верхний рисунок). Flash помещает экземпляр в опорный кадр 1 (нижний рисунок).



**Рис. 11.43.** Для предварительного просмотра одной сцены ролика выполните команду **Control ⇒ Test Scene**.



**Рис. 11.44.** В диалоговом окне **Exporting Flash Movie** имеется индикатор выполнения и кнопка для отмены экспорта.



**Рис. 11.45.** Flash-плеер воспроизводит ролик в обычном окне. Чтобы выйти из программы, закройте это окно.

### Чтобы просмотреть анимацию символа-клипа в контексте:

1. Продолжая работу с документом из предыдущего упражнения, выполните команду **Control ⇒ Test Scene** (Управление ⇒ Тестировать сцену) (рис. 11.43).

Flash экспортирует ролик в формате SWF, применяя текущие параметры публикации. (Дополнительную информацию о параметрах публикации см. в главе 17). Пока происходит экспорт, на экране присутствует окно **Exporting Flash Movie**, в котором имеется индикатор выполнения и кнопка **Stop** (Mac) либо **Cancel** (Windows), позволяющая отменить операцию (рис. 11.44). По завершении экспорта SWF-файл открывается в Flash-плеере (рис. 11.45).

2. Просмотрев ролик в тестовом режиме, закройте окно плеера.

### Кэширование растров

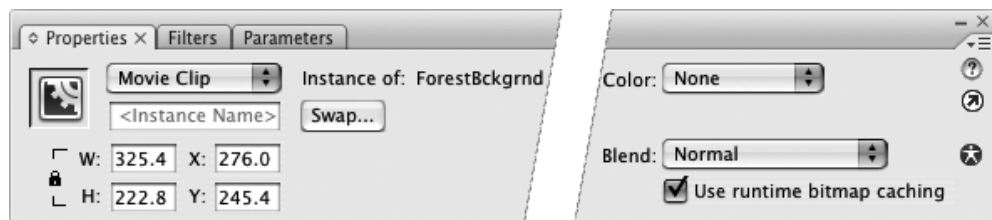
Роль Flash в Web изначально была обусловлена в том числе способностью выполнять анимацию с использованием векторной, а не растровой графики, из-за чего анимация загружалась очень быстро. Однако, если изображение сложное, но со временем изменяется не сильно, особенно когда изменения сводятся в перемещению, а не перерисовке форм, анимация на основе растровых изображения может оказаться быстрее и требовать меньше процессорного времени.

В качестве простого примера рассмотрим мультфильм, в котором по лесу, где растет много деревьев, цветов, кустов и лиан, пробираются созданные вами персонажи. Если фон остается неизменным, то можно создать фоновый растровый слой, который отображается на всем протяжении ролика, не требуя перерисовки в каждом кадре.

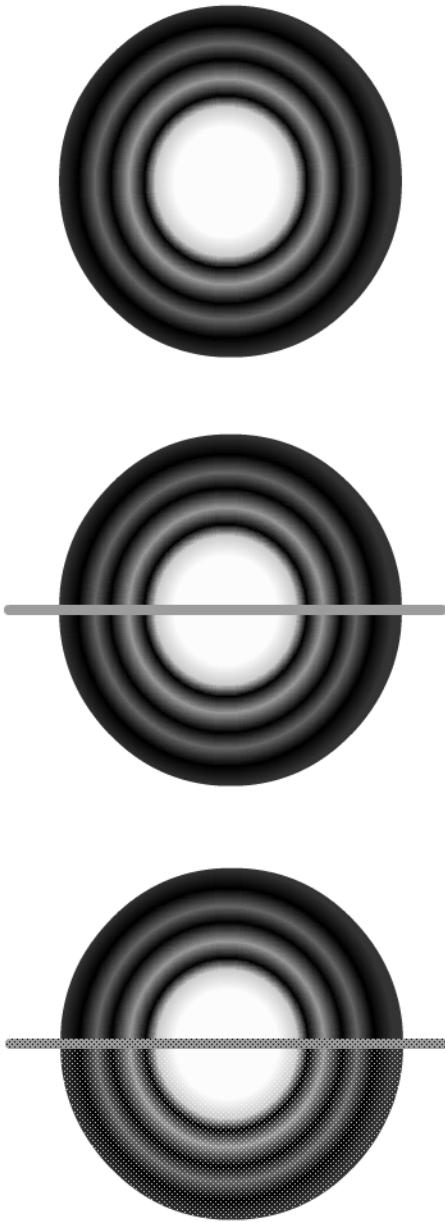
Однако у этого метода есть недостатки. Ни создать, ни изменить растровое изображение внутри Flash нельзя; для этой цели придется воспользоваться другой программой, например Adobe Fireworks или Adobe Photoshop, и импортировать созданный в ней файл. (А если понадобится что-то подправить, то процедуру редактирования во внешней программе с последующим импортом придется повторить.)

А что делать, если фон время от времени должен изменяться? Скажем, декорация гораздо шире экрана и периодически смещается вправо, чтобы стали видны ранее скрытые части. Не изменяется ничего, кроме положения фона. При использовании растрового изображения Flash может сдвигать его попиксельно, практически без замедления ролика, но создавать его все равно придется в другой программе. Если изображение векторное, то Flash будет вынужден пересчитывать все векторы при любом перемещении, что отрицательно скажется на производительности. Flash CS3 позволяет взять лучшее от обоих подходов за счет преобразования векторного изображения в растровое на этапе выполнения.

Эта техника называется *кэшированием растров*. Чтобы воспользоваться ей, подготовьте какой-нибудь сложный элемент в виде символа-клипа. На вкладке **Properties** в инспекторе свойств отметьте флажок **Use runtime bitmap caching** (Применить кэширования растрового изображения во время выполнения) (рис. 11.46). Во время выполнения Flash преобразует векторное изображение в растровое, а потом будет сдвигать его попиксельно по мере необходимости, избегая пересчета всех векторов.



**Рис. 11.46.** Если клип содержит сложную графику, не подвергаемую сложной анимации, то можно установить режим кэширования образа символа во время выполнения. Flash запоминает мгновенный снимок в виде растрового изображения и перерисовывает векторную графику только, когда что-то изменяется. Кэширование растров может заметно ускорить обработку на этапе воспроизведения ролика. Выберите экземпляр символа-клипа на арене и на его вкладке свойств отметьте флажок **Use runtime bitmap caching**.



**Рис. 11.47.** Один из способов создать радугу — залить круг многоцветным радиальным градиентом, а затем удалить нижнюю часть круга. Для удаления проще всего провести через центр круга линию (средний рисунок), затем выбрать линию и нижнюю половину круга (нижний рисунок) и нажать клавишу Delete.

## Использование анимированных масок

В главе 6 вы узнали о слоях-масках, которые позволяют скрывать и показывать объекты в слоях, расположенных под ними. Иногда самый лучший способ создать иллюзию движения — это анимировать маску, которая будет постепенно скрывать или показывать объект.

Возьмем, к примеру, линию, которая простирается от левого до правого края арены. Если создать маску, которая будет понемногу открывать эту линию, то создастся впечатление, что линия сама себя рисует. А если обратить процесс, то линия будет стираться. Создание вращающейся маски или маски, которая состоит из интерполированных форм, может дать еще более интересные эффекты. Чем лучше вы освоитесь с использованием анимированных масок для показа стационарных элементов, тем больше применений будете находить для этой техники. Чтобы попрактиковаться, попробуем анимировать маску для создания появляющейся радуги.

### Чтобы создать стационарный графический объект и движущуюся маску, которая этот объект раскрывает:

1. Создайте документ с двумя слоями. Нижний слой назовите *Rainbow* (Радуга), а верхний — *Rotating Rectangle* (Вращающийся прямоугольник).
2. В опорном кадре 1 слоя *Rainbow* нарисуйте на арене изображение радуги, пользуясь различными инструментами рисования.

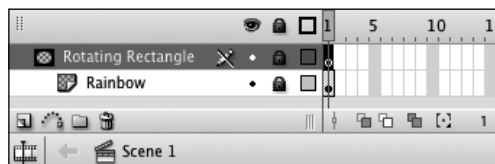
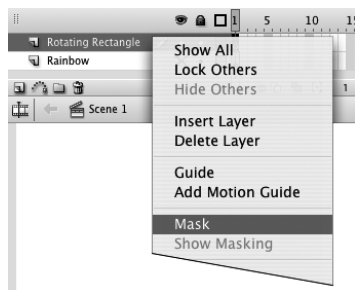
Например, можно с помощью инструмента «Овал» нарисовать идеальный круг в режиме комбинированных форм. Залейте круг радиальным градиентом, в котором отчетливо видны разные цвета спектра. Затем сотрите нижнюю часть круга (рис. 11.47). То, что останется, и будет радугой. На всякий случай преобразуйте эту форму в символ (выберите ее на арене, выполните команду **Modify** ⇨ **Convert to Symbol**, задайте тип символа *Graphic*, присвойте ему имя и нажмите **OK**), чтобы осталась копия, если вы по ошибке удалите оригинал.

3. Щелкните по имени слоя Rotating Rectangle кнопкой мыши с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows). Появится контекстное меню слоя.
4. Выберите из него пункт **Mask** (Маска).
5. Щелкните по значкам замка на панели временной шкалы для обоих слоев Rainbow и Rotating Rectangle, чтобы разблокировать их.
6. В опорном кадре 1 в слое Rotating Rectangle нарисуйте на арене прямоугольник, расположив его точно под радугой (рис. 11.49).

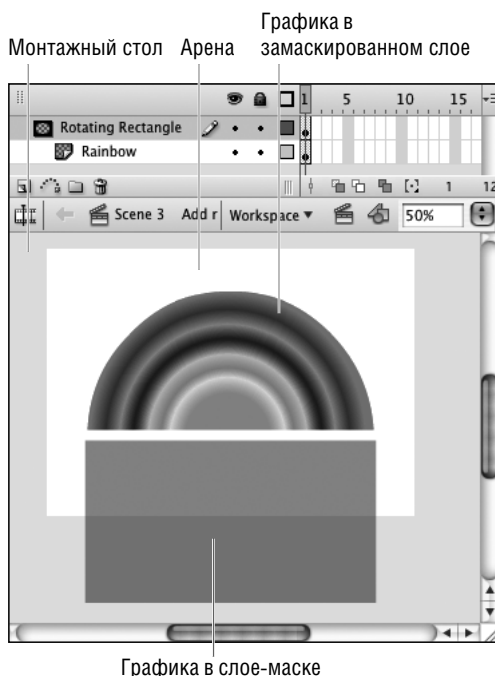
Этот прямоугольник будет маской. Все элементы связанного слоя, которые находятся под прямоугольником, видны; остальные скрыты.

Сделайте прямоугольник чуть больше радуги, чтобы маска гарантированно закрыла радугу целиком. Если взять прозрачный цвет заливки, то вы будете видеть радугу сквозь прямоугольник-маску, это поможет точно выбрать его положение. (Чтобы залить прямоугольник прозрачным цветом, выберите его, а затем на панели цветов введите процентную величину коэффициента прозрачности в поле Alpha.)

**С** Для слоя-маски можно применять все три вида анимации: покадровую, интерполяцию движения и интерполяцию форм. Можно также пользоваться анимированным графическим символом или символом-клипом. Те, кто хорошо владеет языком **ActionScript**, могут даже сделать так, что один символ-клип будет маскировать другой, или создать прозрачные маски.

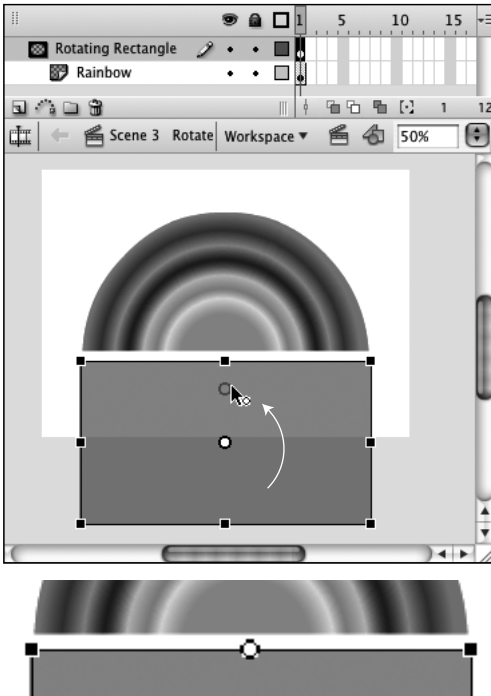


**Рис. 11.48.** Щелкните по слою с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows) и выберите пункт **Mask** из контекстного меню (верхний рисунок). Flash автоматически свяжет с маской лежащий под ней слой и заблокирует оба слоя.



**Рис. 11.49.** Чтобы создать маску, которая постепенно открывает радугу, нарисуйте прямоугольник, который закрывал бы ее целиком. Расположите прямоугольник точно под радугой, чтобы она оказалась скрытой.





**Рис. 11.50.** С помощью инструмента «Свободное преобразование» переместите центр вращения символа (или группы). Для этого перетащите кружочек, обозначающий точку трансформации. Кружочек-модификатор рядом со стрелкой показывает, где окажется новая точка трансформации. Если расположить ее в середине верхней стороны, то при вращении прямоугольник будет все время частично перекрывать радугу.

### Чтобы подготовить маску для вращательной анимации:

1. Выберите прямоугольник и выполните команду **Modify ⇒ Convert to Symbol** (Изменить ⇒ Преобразовать в символ).

Поскольку мы собираемся создать вращательную анимацию, для маски придется воспользоваться интерполяцией движения, то есть маска должна быть графическим объектом. Это может быть объект-рисунок, примитивная форма или сгруппированный элемент, но лучше всего воспользоваться символом, чтобы потом маску легче было применить повторно.

Открывается диалоговое окно **Convert to Symbol**.

2. Введите имя символа, задайте тип **Graphic** и нажмите **ОК**.
3. С помощью инструмента «Свободное преобразование» выберите на арене прямоугольник и укажите курсором на точку трансформации объекта (белый кружочек, который по умолчанию расположен в центре символа).

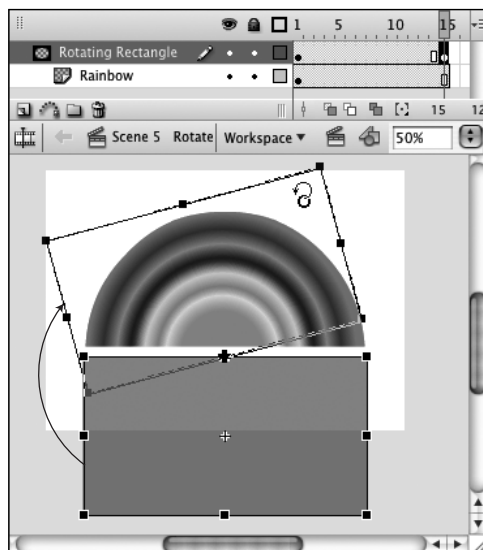
Рядом со стрелкой появляется кружочек-модификатор, означающий, что точку трансформации выбранного объекта можно перемещать.

4. Перетащите точку трансформации в середину верхней стороны прямоугольника (рис. 11.50). Теперь можно повернуть прямоугольник.

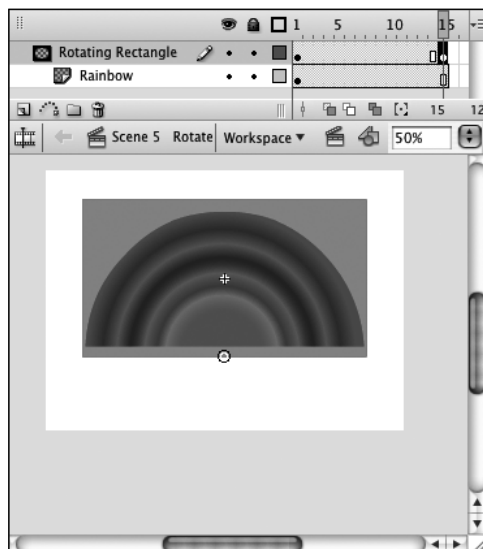
### Чтобы завершить анимацию вращающейся маски:

1. Выберите кадр 15 в слое Rainbow на временной шкале и выполните команду **Insert** ⇒ **Timeline** ⇒ **Frame**.
2. В слое Rotating Rectangle выберите кадр 15 и выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**.
3. В опорном кадре 15 выберите на арене прямоугольник и с помощью инструмента «Свободное преобразование» в режиме Rotate and Skew (Вращение и наклон) измените его положение; буксируя левый нижний угол, поверните прямоугольник так, чтобы он почти закрыл радугу (рис. 11.51). Если в режиме разработки маска закрывает радугу, то в готовом ролике часть радуги под маской наоборот будет видна.
4. В слое Rotating Rectangle выберите на временной шкале любой из кадров в последовательности опорного кадра 1 (кадры 1-14).
5. На вкладке **Properties** в инспекторе свойств выберите в списке **Tween** пункт Motion.
6. В списке **Rotate** (Поворот) выберите пункт CW (По час.).
7. В поле Times (раз) введите 0.

Тем самым вы определили интерполяцию движения, заключающуюся в повороте прямоугольника на 180 градусов таким образом, что он постепенно закрывает радугу (рис. 11.52).



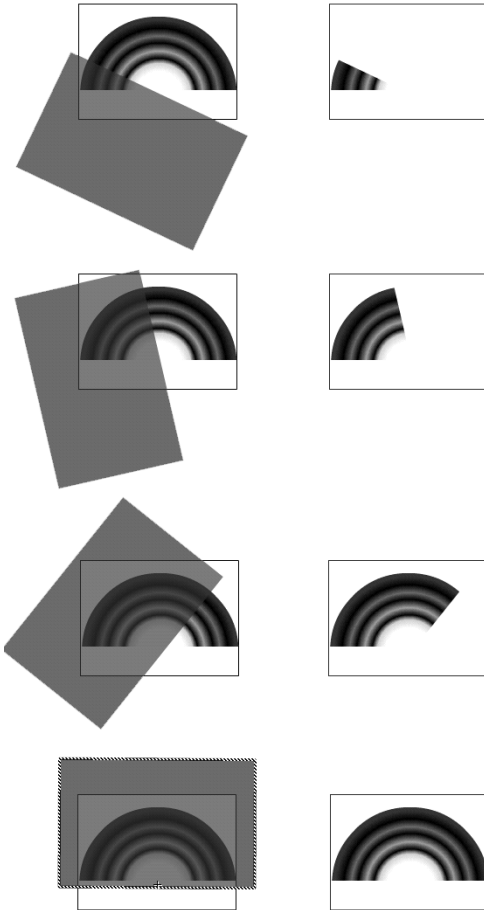
**Рис. 11.51.** В конечном опорном кадре последовательности расположите маску так, чтобы она полностью закрывала элементы, которые должны быть видны. В данном случае мы повернули прямоугольник инструментом «Свободное преобразование» в положение, где радуга оказывается под ним.



**Рис. 11.52.** После завершения интерполяции движения маска целиком закрывает радугу. Если залить маску прозрачным цветом, то вы будете видеть сквозь нее раскрываемые объекты. В готовом ролике прозрачность не повлияет на видимость объектов.

Ролик воспроизводится  
с разблокированными  
слоями

Ролик воспроизводится  
с заблокированными  
слоями



**Рис. 11.53.** При воспроизведении ролика с разблокированными слоями (левый рисунок) вы будете видеть промежуточные позиции маски. Если же слои заблокированы (правый рисунок), то поведение будет таким же, как в готовом ролике.

### Чтобы посмотреть на получившуюся анимацию:

- Выполните команду **Control** ⇒ **Test Movie** (Управление ⇒ Тестировать ролик) или **Control** ⇒ **Test Scene** (Управление ⇒ Тестировать сцену).

или

- Щелкните по значку замка на панели временной шкалы, чтобы заблокировать оба слоя; при этом вы будете видеть замаскированную радугу. Затем воспроизведите ролик, чтобы понаблюдать за ее постепенным появлением. Если радуга появилась не полностью, попробуйте увеличить прямоугольник или изменить его начальное положение. Разблокируйте оба слоя и протащите точку воспроизведения по всему ролику, чтобы видеть положение прямоугольника в каждом промежуточном кадре (рис. 11.53).

**С**

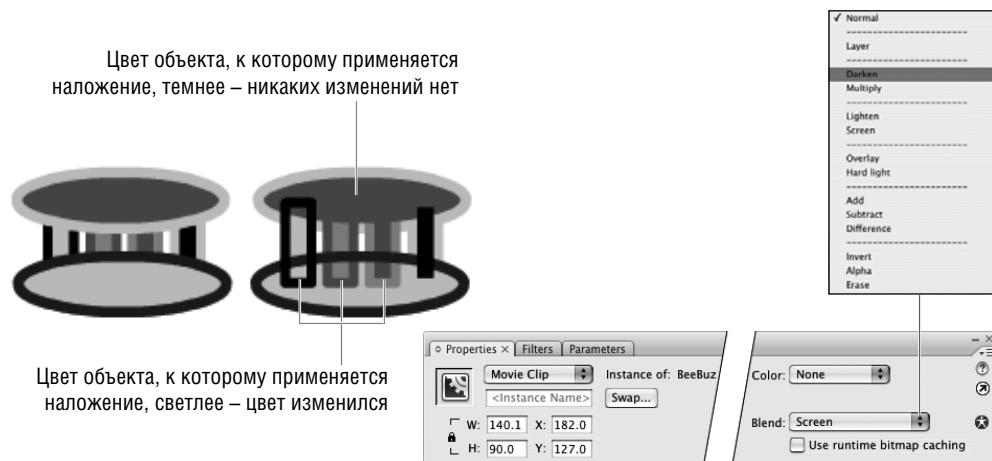
Чтобы радуга проявлялась постепенно, анимируйте изменение прозрачности. Выберите кадр 15 в слое **Rainbow** и нажмите **F6**, чтобы продублировать экземпляр символа радуги в новом опорном кадре. Выберите в слое **Rainbow** кадр 1 и на вкладке его свойств задайте тип анимации **Motion** (Движение). Оставив символ радуги выбранным, выберите в списке **Color** на вкладке в инспекторе свойств пункт **Alpha**, а в поле **Value** введите небольшое значение.

### Замечание о режимах наложения

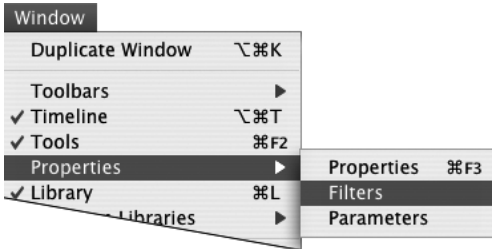
Режимы наложения позволяют управлять взаимодействием графики из различных слоев и подслоев. Наложение можно использовать для создания специальных эффектов. Техника создания сложных эффектов путем наложения выходит за рамки данной книги, но краткий обзор возможностей мы все же приведем.

Если вы знакомы с концепцией наложения в программах Adobe Fireworks или Adobe Photoshop, то понимаете, что можно сделать в этом отношении во Flash, так как механизмы очень похожи. В результате наложения перекрывающиеся изображения взаимодействуют так, что создаются новые цвета и интересные эффекты. Художники часто пользуются наложением для создания коллажей, на которых различных изображения являются частью единого целого (например, наложив изображение детского личика на изображение воздушного шарика, мы получим шарик с рожицей). Фотографы применяют наложение для улучшения или исправления дефектов в цифровых снимках (например, чтобы осветить слишком темные части фотографии).

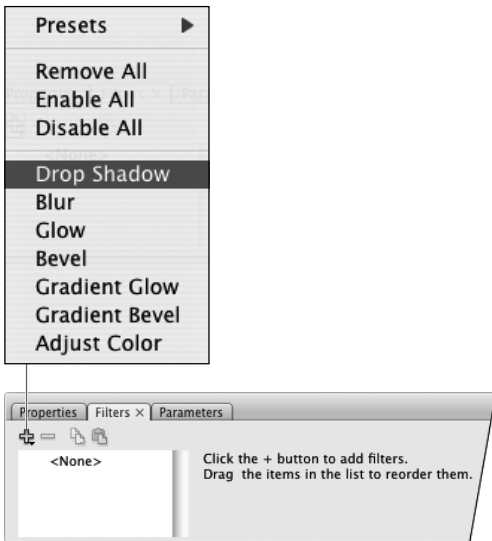
Наложение применяется к экземпляру символа-клипа на арене. Flash вычисляет новые цвета попиксельно, то есть модифицирует RGB-значение каждого пикселя клипа согласно формуле, зависящей от выбранного режима наложения. Меняет ли конкретный пиксель свой цвет, зависит от режима наложения, от цвета пикселя в клипе и от цвета пикселя, который находится в слое под клипом. Пиксели в нижних слоях могут принадлежать другим клипам, графическим элементам или самой арене (тогда их цвет будет совпадать с цветом фона арены). Режим наложения задается на вкладке свойств клипа в инспекторе свойств (рис. 11.54).



**Рис. 11.54.** Выберите на арене символ-клип, к которому хотите применить наложение; затем задайте режим в списке Blend (Смешать) (в данном случае выбран режим Darken (Затемнить) на вкладке Properties в инспекторе свойств. Оба овала представляют собой символы-клипы, которые расположены в слое поверх набора прямоугольных столбиков. На левом рисунке задан режим наложения Normal (Обычный), а на правом — Darken. Flash вычисляет цвет каждого пикселя. Там, где пиксели объекта, к которому применяется наложение (овалы), светлее, чем пиксели в нижележащем слое, Flash изменяет цвет на более темный.



**Рис. 11.55.** Чтобы оказаться на вкладке **Filters** в инспекторе свойств, выполните команду **Window** ⇒ **Properties** ⇒ **Filters**.



**Рис. 11.56.** Чтобы добавить тень к тексту, выберите на арене текстовое поле и перейдите на вкладку **Filters** в инспекторе свойств. Из меню **Add Filter** выберите пункт **Drop Shadow**.

## Использование фильтров

Во Flash CS3 имеется семь фильтров для создания специальных эффектов: тень, размытие, свечение, фаска, градиентное свечение, градиентная фаска и настройка цвета. Применять фильтры можно лишь к символам-клипам, символам-кнопкам и текстовым полям. Выберите на арене один из таких объектов и перейдите на вкладку **Filters** (Фильтры) в инспекторе свойств, где можно указать фильтр и задать его настройки. К одному объекту можно применить сразу несколько фильтров. Разрешается копировать фильтры из одного объекта в другой, а также заранее создать и сохранить комбинацию фильтров с конкретными настройками, которую впоследствии можно будет применить к объекту. Можно анимировать изменения фильтров, чтобы, например, создать переливающуюся цветами кнопку. Приемы создания сложных фильтров выходят за рамки данной книги, но базовые принципы работы со всеми типами фильтров похожи.

В упражнениях из этого раздела мы в качестве введения в фильтры Flash создадим простое текстовое поле с тенью.

### Чтобы получить доступ к панели фильтров:

- Если панель фильтров закрыта или не является активной вкладкой в инспекторе свойств, выполните команду **Window** ⇒ **Properties** ⇒ **Filters** (рис. 11.55).

### Чтобы применить фильтры к объекту:

1. Выберите на арене текстовое поле, экземпляр символа-клипа или кнопку, к которой собираетесь применить фильтр. Для целей данного упражнения пусть это будет текстовое поле.
2. На вкладке **Filters** в инспекторе свойств выберите фильтр из списка **Add Filter** (Добавить фильтр) (знак плюс). Для этого упражнения остановимся на фильтре **Drop Shadow** (Тень) (рис. 11.56).

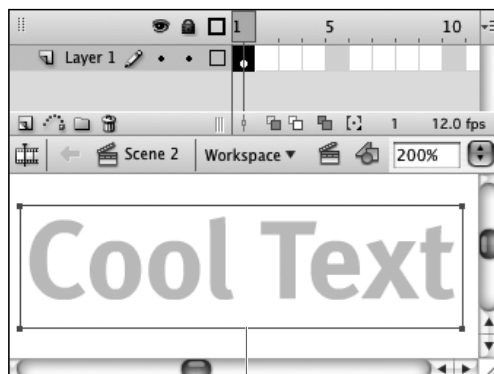
Flash применяет фильтр к объекту на арене и добавляет имя фильтра в список фильтров, связанных с этим объектом. В правой части вкладки Filters появляются элементы управления для задания параметров фильтра, которые пока имеют значения по умолчанию (рис. 11.57). Задайте нужные вам параметры эффекта.

**C**

К одному клипу, кнопке или текстовому полю можно применить несколько фильтров. Но чем больше фильтров, тем больше процессорного времени нужно для их обработки. Поэтому старайтесь не применять к одному объекту много фильтров.

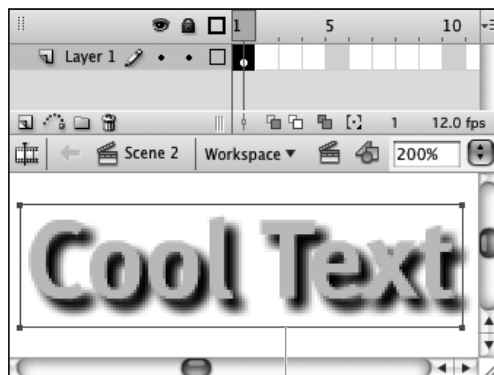
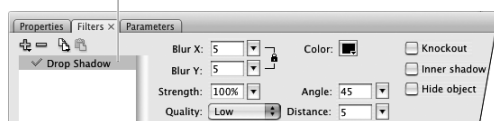
**C**

От порядка применения фильтров зависит, как будет смотреться эффект. Чтобы изменить порядок, перетаскивайте имя фильтра в списке в нужное место.



Исходный текст

Выбран фильтр «Тень»



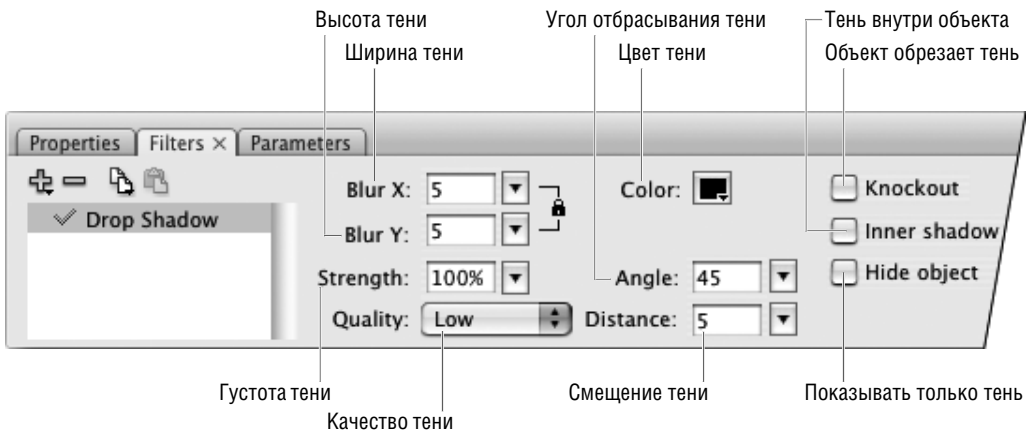
Результат применения фильтра к объекту

**Рис. 11.57.** Выберите пункт Drop Shadow из списка Add Filter, чтобы применить фильтр «Тень» к выбранному текстовому полю.

**Чтобы изменить параметры фильтра «Тень»:**

1. Выберите на арене текстовое поле, экземпляр символа-клипа или кнопку, к которой собираетесь применить фильтр. Для целей данного упражнения возьмите текстовое поле, к которому ранее применили фильтр «Тень».
2. На вкладке **Filters** в инспекторе свойств укажите в списке примененных фильтров тот, который хотите модифицировать, в данном случае — Drop Shadow.

В правой части вкладки появляются параметры выбранного фильтра (рис. 11.58).



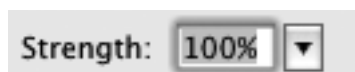
**Рис. 11.58.** Список фильтров, примененных к выбранному объекту, отображается в левой части вкладки Filters. После выбора из этого списка какого-нибудь фильтра в правой части вкладки появляются его параметры. У фильтра «Тень» десять параметров.

3. Чтобы изменить внешний вид тени, выполните любые из следующих действий:

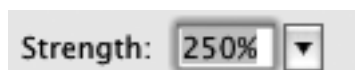
- для изменения размера тени заполните поля Blur X (Размытие X) и/или Blur Y (Размытие Y). Поле Blur X определяет ширину, а поле Blur Y — высоту тени. По умолчанию Flash изменяет эти поля одновременно, делая их одинаковыми (поля соединены линией со значком запертого замка). Чтобы задать разные значения, щелкните по значку замка (рис. 11.59);
- чтобы изменить яркость тени, введите значение от 0 до 100 в поле Strength (Сила). Чем больше значение, тем темнее тень (рис. 11.60);
- чтобы изменить цвет тени, щелкните по элементу управления Color и выберите нужный цвет в открывающемся окне образцов (о работе с элементами управления цветом см. главу 2);
- чтобы изменить расстояние от тени до объекта, введите значение от -32 до 32 в поле Distance (Расстояние). Если расстояние отрицательно, то тень располагается выше объекта, если положительно — ниже;
- чтобы изменить угол отбрасывания тени, введите значение от 0 до 360 в поле Angle (Угол). Угол измеряется в градусах. Если прибегнуть к аналогии с часами, то 0 соответствует 9 часам, 90 — 12 часам, 180 — 3 часам и т.д.



**Рис. 11.59.** Если замок заперт (верхний рисунок), то ширина и высота изменяются синхронно, оставаясь одинаковыми. Если отперт (нижний рисунок), то можно заполнять поля независимо.



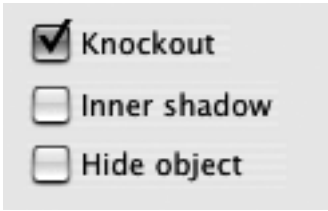
Cool Text



Cool Text

**Рис. 11.60.** Если выбран черный цвет тени, то величина 100% в поле Strength соответствует темно-серой тени; при 250% тень становится почти черной.





Cool Text

**Рис. 11.61.** Если отметить флажок Knockout, то создается эффект вырезанного трафарета.

- чтобы казалось, будто объект выступает из тени, отметьте флажок **Knockout** (Удаление фона) (рис. 11.61);
- чтобы тень была видна только внутри границ самого объекта, отметьте флажок **Inner Shadow** (Внутренняя тень);
- чтобы полностью скрыть объект, оставив только отбрасываемую им тень, отметьте флажок **Hide Object** (Скрыть объект). Эта возможность полезна, когда вы хотите точно настроить внешний вид тени, не отвлекаясь на сам объект;
- чтобы задать качество тени, выберите значение Low (Низкое), Medium (Среднее) или High (Высокое) из списка **Quality** (Качество). Максимальная скорость воспроизведения ролика достигается при низком качестве тени.

**С**

Как и в случае большинства полей ввода, для задания параметров фильтра «Тень» можно воспользоваться ползунком. Он появится, если щелкнуть по треугольничку справа от поля. Для поля Angle ползунок круговой; буксируйте его по кругу, как будто переводите стрелку настенных часов.

### Чтобы снять фильтр с объекта:

1. Выберите на арене объект, с которого хотите снять фильтр.
2. В списке примененных фильтров на вкладке Filters в инспекторе свойств выберите фильтр, который хотите отменить.
3. Нажмите кнопку удаления фильтра (знак минус) (рис. 11.62). Flash удалит фильтр из списка и перестанет применять соответствующий эффект к объекту на арене.

**C**

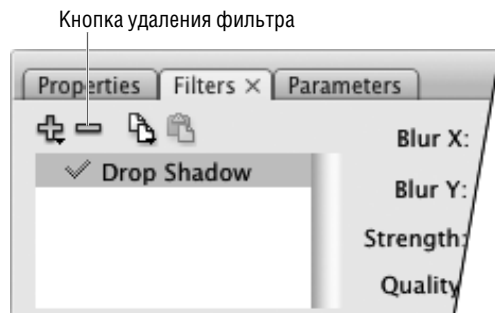
Если вы хотите только отключить фильтр, а не удалять его совсем, можете временно его деактивировать. В списке примененных фильтров на вкладке Filters в инспекторе свойств зеленой галочкой отмечены активные фильтры. Если щелкнуть по ней, она превратится в красный крестик; соответствующий фильтр деактивирован и не применяется к объекту на арене.

**C**

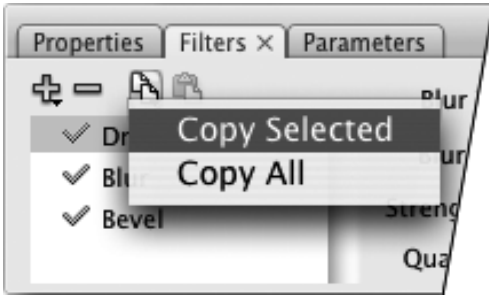
Чтобы отменить все фильтры, кроме одного, щелкните мышью с нажатой клавишей **Option** (Mac) либо **Alt** (Windows) по галочке рядом с тем фильтром, который хотите оставить. Против всех остальных фильтров в списке появятся крестики. Чтобы активировать все фильтры, щелкните мышью с нажатой клавишей **Option** (Mac) либо **Alt** (Windows) по единственной зеленой галочке.

**C**

Чтобы активировать или деактивировать вообще все примененные фильтры, выберите из списка **Add Filter** пункт **Enable All** (Включить все) или **Disable All** (Отключить все).

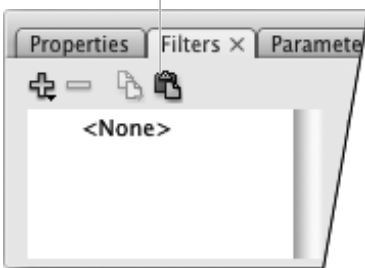


**Рис. 11.62.** Чтобы удалить фильтр, выберите его в списке примененных на вкладке Filters в инспекторе свойств и нажмите кнопку удаления.



**Рис. 11.63.** Щелкните по значку с двумя документами, чтобы открыть меню копирования фильтров. Эта функция позволяет скопировать только выбранный или все фильтры, присутствующие в списке примененных.

Кнопка вставки фильтров



**Рис. 11.64.** Нажмите кнопку вставки фильтров, чтобы применить скопированные фильтры к символу-клипу, кнопке или текстовому полю, выбранному на арене.

### Чтобы скопировать настройки фильтров из одного объекта в другой:

1. Выберите на арене объект, чьи фильтры хотите скопировать.
2. Выберите фильтр, который хотите скопировать, в списке на вкладке **Filters** в инспекторе свойств.
3. Из меню копирования фильтров (рис. 11.63) выполните одну из следующих команд:
  - чтобы скопировать только выбранный фильтр, выполните команду **Copy Selected** (Копировать выделенный);
  - чтобы скопировать все фильтры, присутствующие в списке, выполните команду **Copy All** (Копировать все);
4. Выберите на арене объект, к которому хотите применить скопированные фильтры.
5. Нажмите кнопку **Paste Filters** (Вставить фильтры) на вкладке **Filters** в инспекторе свойств (рис. 11.64). Flash применит скопированные фильтры вместе с их настройками к выбранному объекту.

**C**

Можно также сохранить настройки фильтров в библиотеке часто используемых эффектов. После того как фильтры применены к объекту и настроены, выберите этот объект на арене. Затем из списка **Add Filter** на вкладке **Filters** выберите команду **Presets ⇒ Save As** (Наборы параметров ⇒ Сохранить как), введите имя сохраненного набора параметров и нажмите **OK**. Теперь, чтобы применить этот набор параметров к любому выбранному объекту, достаточно выбрать его имя из меню **Add Filter ⇒ Presets**.

**C**

Созданными с помощью фильтров наборами параметров можно поделиться с другими пользователями Flash. Эффекты хранятся в XML-файлах, которые находятся в подпапке **Filters** папки **Configuration** уровня пользователя (о различных папках **Configuration** см. главу 1).

### Подводные камни, связанные с анимированными фильтрами

Фильтры можно анимировать, применяя интерполяцию движения. Определите движение с помощью символа-клипа, символа-кнопки или текстового поля; примените к анимируемому объекту фильтры, при этом в первом и последнем опорном кадре задайте разные параметры фильтров. На первый взгляд, все просто, но у Flash есть собственные понятия о том, насколько сильно могут отличаться начальные и конечные параметры. Проще и безопаснее всего интерполировать эффекты, создаваемые только одним фильтром. Например, путем анимирования тени объекта можно создать эффект перемещающегося источника цвета. Но если к интерполируемому объекту применено несколько фильтров, все становится сложнее, так как Flash требует согласованности между настройками фильтров у объекта в первом и последнем опорном кадре. Должны совпадать число, типы и порядок применения фильтров в начале и в конце последовательности.

Если при подготовке интерполяции вы начали с создания опорного кадра с одним объектом, к которому применены фильтры, затем выбрали какой-то последующий кадр и выполнили команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**, то Flash продублирует исходный кадр со всеми фильтрами и их настройками. В последнем кадре настройки фильтров можно модифицировать, чтобы создать анимированное изменение, и все будет работать нормально.

Если же вы создавали начальный и конечный опорные кадры независимо, применяя к объектам в каждом из них разные фильтры и разные настройки, а затем задали значение Motion свойства Tween, то возможно рассогласование фильтров. Рассогласование можно внести и в результате изменения интерполируемых объектов и их фильтров по ходу разработки.

Flash сравнивает фильтры и их настройки для начального и конечного опорного кадров. При обнаружении несогласованности Flash разрешает проблему путем добавления фильтров или фильтров-заглушек (тип заглушки совпадает с типом отсутствующего фильтра, а все параметры имеют значения по умолчанию) либо удаления каких-то фильтров. О том, что были внесены коррективы, никаких предупреждений не выдается. И понять, что Flash изменил документ по собственному усмотрению, может оказаться не просто. Ко всему прочему, автоматически внесенные изменения могут неожиданно отразиться на анимации. Если вы обнаружите, что эффекты работают не так, как задумано, проверьте состав и настройки фильтров в начальном и конечном опорных кадрах каждой интерполяции. Что именно придется изменить, зависит от характера проблемы.

# ДОБАВЛЕНИЕ ИНТЕРАКТИВНОСТИ (ПОМОЩЬЮ КНОПОК)

---

## 12

Овладев инструментами рисования и анимации, имеющимися в программе Flash CS3 Professional, вы можете создавать ролики, которые будут воспроизводиться с начала до конца без остановки. Но Flash умеет еще и организовывать интерактивную среду, превращающую зрителя в пользователя. Чтобы сделать ролик интерактивным, необходимо включить в него элементы интерфейса. Самым распространенным из таких элементов является кнопка. Кнопки обладают интерактивностью двоякого рода. Во-первых, они отвечают на действия пользователя, меняя свой внешний вид, например, перекрашиваясь в другой цвет при наведении курсора. Во-вторых, они инициируют выполнение заданий, скажем, переход к другой сцене при нажатии.

В состав дистрибутива Flash входит несколько предопределенных интерфейсных компонентов, в том числе символы-кнопки и компоненты-кнопки. В них уже встроено поведение, определяющее первый вид интерактивности, то есть они визуальным образом реагируют на движения мышь. Для реализации второго вида интерактивности необходимо написать сценарий. О некоторых простых способах программирования сценариев будет рассказано в главе 13.

А в этой главе вы научитесь реализовывать интерактивность первого уровня на примере символов-кнопок и компонентов-кнопок. Также будет показано, как настроить символ-клип, ведущий себя подобно визуальной реагирующей кнопке.

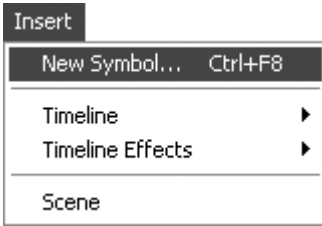
### О символах-кнопках

Во Flash для получения кнопки можно создать символ и назначить ему тип Button (Кнопка). По существу кнопка — это короткий интерактивный ролик. Если быть точным, он состоит ровно из четырех кадров. Когда выбирается тип символа Button, Flash подготавливает временную шкалу с четырьмя протокадрами. В первых трех вы создаете опорные кадры и графику для отображения кнопки в трех стандартных состояниях: отпущена, под курсором и нажата. В четвертом опорном кадре (который зритель никогда не видит) определяется активная область кнопки.

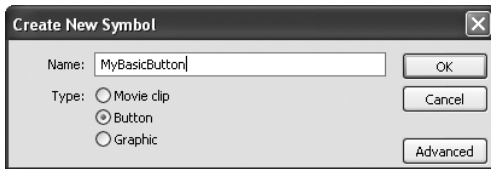
В состоянии «Отпущена» кнопка рисуется в статичном виде. Так кнопка выглядит, когда курсор находится вне ее активной области. В состоянии «Под курсором» кнопка оказывается, когда на нее наведен курсор мыши. В этом случае Flash автоматически рисует курсор в виде руки, однако зачастую вам нужны дополнительные визуальные ключи, которые сообщали бы пользователю о том, что курсор указывает на активную кнопку. В состоянии «Нажата» кнопка должна выглядеть так, будто на нее нажали. И в последнем опорном кадре создается графический объект, определяющий границы кнопки. Это может быть любая заливка; когда курсор оказывается внутри нее во время воспроизведения ролика, кнопка «оживает».

Любые изменения внешнего вида графических элементов в вышепомянутых опорных кадрах создают иллюзию движения. В прочих анимационных последовательностях изменения происходят во времени по мере того, как точка воспроизведения переходит от кадра к кадру. В случае же кнопок изменения обусловлены наведением курсора на определенную область экрана.

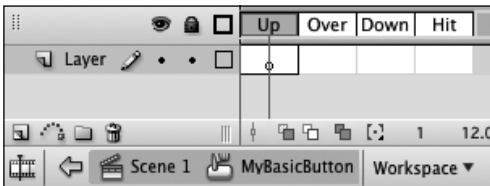
Можно включать в опорные кадры символы-клипы, тогда кнопки получатся анимированными. Можно также написать сценарий, который активирует кнопки, чтобы предоставить пользователям больший контроль над роликом.



**Рис. 12.1.** На первом шаге создания кнопки выполните команду Insert ⇒ New Symbol.



**Рис. 12.2.** Создайте новый символ и задайте для него тип Button в окне Create New Symbol. Здесь же присвойте кнопке имя.



**Рис. 12.3.** Временная шкала любой кнопки содержит четыре кадра: Up, Over, Down и Hit. Flash автоматически помещает в первую позицию (Up) опорный кадр.

## Создание простейшего символа-кнопки

Кнопка — это символ-клип, в опорных кадрах которого представлены различные состояния. У символа-кнопки есть три кадра, описывающие состояния «Отпущена», «Под курсором» и «Нажата», и еще один кадр, в котором определена активная область. Для создания простейшего символа-кнопки возьмите какую-нибудь простую форму и поместите ее во все кадры, описывающие состояния. Затем, чтобы отличить одно состояние от другого, измените в разных кадрах цвет формы или добавьте какие-нибудь внутренние элементы. Когда вы заполните все четыре опорных кадра, кнопка будет готова к работе. Вернитесь в режим редактирования документа и перетащите экземпляр символа-кнопки с библиотечной панели на арену.

### Чтобы создать символ-кнопку:

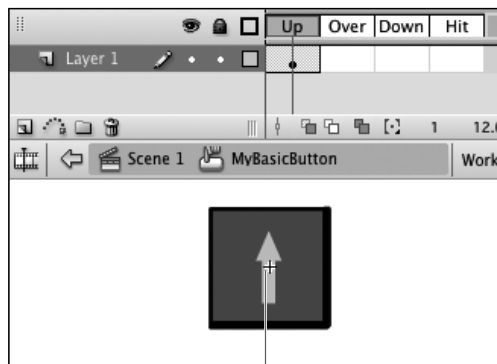
1. Откройте документ, в который будете добавлять кнопки.
2. Выполните команду **Insert ⇒ New Symbol** (Вставить ⇒ Новый символ) или нажмите комбинацию клавиш **⌘-F8** (Mac) либо **Ctrl-F8** (Windows) (рис. 12.1). Открывается окно Create New Symbol.
3. Введите имя символа (например, MyBasicButton), задайте тип Button и нажмите **OK** (рис. 12.2).

Flash создает на библиотечной панели новый символ и переходит в режим редактирования этого символа. Временная шкала символа-кнопки состоит из четырех кадров: «Отпущена», «Под курсором», «Нажата» и «Активна».

По умолчанию кадр «Нажата» является опорным (рис. 12.3). Остальные кадры вам предстоит сделать опорными и поместить в них графические элементы, описывающие кнопку. Чтобы пользователь мог опознать активную кнопку и понять, что она нажата, сделайте эти элементы отличающимися друг от друга.

### Чтобы создать состояние «отпущена»:

1. Используя файл из предыдущего упражнения, выберите на временной шкале опорный кадр Up.
2. Создайте на арене новое изображение или поместите на нее графический символ (рис. 12.4). Этот графический элемент станет кнопкой, то есть будет находиться на экране во время просмотра ролика, ожидая, что на него нажмут. Перекрестье в центре арены, которое видно в режиме редактирования символа, станет точкой регистрации символа.

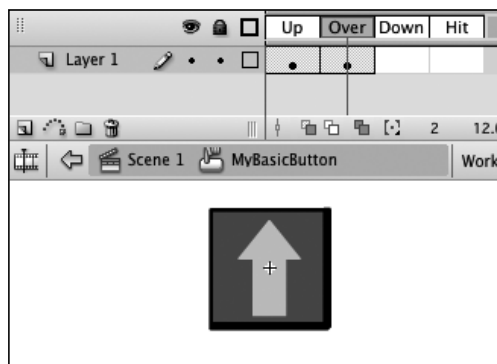


Перекрестье, обозначающее точку регистрации, находится в центре арены

### Чтобы создать состояние «под курсором»:

1. Используя файл из предыдущего упражнения, выберите на временной шкале опорный кадр Over.
2. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**.

Flash продублирует содержимое опорного кадра Up, после чего вы можете внести мелкие изменения для преобразования его к виду, отображаемому в кадре Over. Например, увеличьте элемент внутри кнопки (рис. 12.5). Дублирование содержимого предыдущего опорного кадра упрощает выравнивание элементов, так чтобы они не «скакали» в кадре при изменении состояния.



**Рис. 12.5.** Когда пользователь наводит курсор на кнопку (или задерживает его над кнопкой), Flash отображает состояние опорного кадра Over.



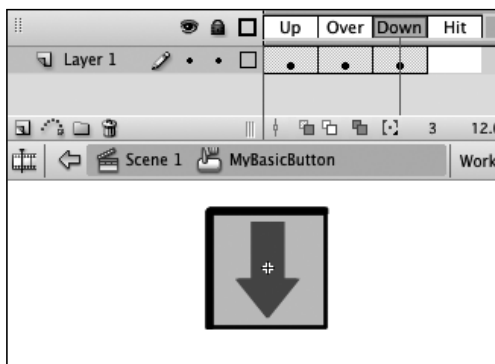


Рис. 12.6. Когда пользователь нажимает кнопку, Flash отображает содержимое опорного кадра Down.

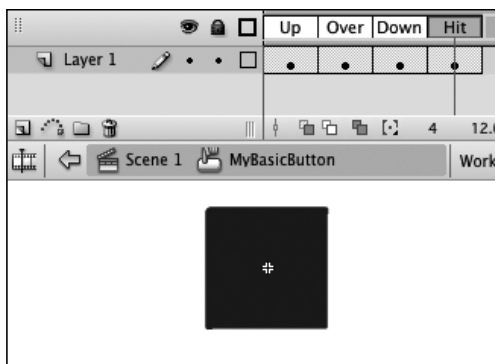


Рис. 12.7. В кадре Hit необязательно должно находиться детальное изображение кнопки в каком-либо состоянии, достаточно силуэта, описывающего ее форму. Это изображение используется только для определения активной области кнопки. В данном случае опорный кадр Hit содержит копию формы из опорного кадра Down, закрашенную темно-серым цветом.

### Чтобы создать состояние «нажата»:

1. Используя файл из предыдущего упражнения, выберите на временной шкале опорный кадр Down.
2. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**.

Flash продублирует содержимое опорного кадра Over, после чего вы сможете внести мелкие изменения для преобразования его к виду, отображаемому в кадре Down. Например, измените цвет и положение тени, так чтобы кнопка казалась нажатой (рис. 12.6). После того, как все три состояния кнопки созданы, необходимо определить активную область.

### Чтобы создать состояние «активна»:

1. Используя файл из предыдущего упражнения, выберите на временной шкале опорный кадр Hit.
2. Выполните команду **Insert** ⇒ **Timeline** ⇒ **Keyframe**.

Flash продублирует содержимое опорного кадра Down. Если во всех трех состояниях кнопки использовались изображения одинаковой формы и размера, то можно спокойно скопировать в кадр Hit содержимое любого из предшествующих опорных кадров.

3. Если хотите, можете воспользоваться инструментами «Ведро с краской» и «Чернильница», чтобы закрасить форму в кадре Hit одним цветом (рис. 12.7). Этот шаг не обязателен, но так будет легче запомнить, что это изображение никогда не показывается зрителю.
4. Выполните команду **Edit** ⇒ **Edit Document** или нажмите кнопку **Назад** на панели редактирования.

Flash возвращается в главную временную шкалу. Теперь символ-кнопку можно вставлять в ролик, как любой другой символ.

## С

Если в кадре Hit вы используете копию содержимого любого из кадров Up, Over или Down, попробуйте ее немного увеличить. В этом случае кнопка гарантированно активируется, когда курсор окажется рядом с ней. Чтобы быстро увеличить размер изображения, выберите его на арене и выполните команду **Modify** ⇒ **Shape** ⇒ **Expand Fill** (Изменить ⇒ Форма ⇒ Расширить заливку). В диалоговом окне Expand Fill введите небольшое значение (скажем, 2 пикселя). Отметьте переключатель **Expand** (Расширить) и нажмите **OK**.

### Чтобы поместить созданную кнопку в ролик:

- Продолжая работу с файлом из предыдущего упражнения, перетащите экземпляр на арену (рис. 12.8).

Можно изменить экземпляр, задав другой цвет, угол поворота или размер. (Подробнее о модификации экземпляров символа см. главу 7.)

**С**

Чтобы страницы сайта выглядели согласованно, вам, вероятно, захочется многократно использовать один и тот же набор кнопок. Можно даже использовать слегка различающиеся кнопки в разных проектах. Чтобы сэкономить время, заведите специальный документ исключительно для кнопок и создавайте символы-кнопки только в нем. Затем можно будет скопировать кнопку из этого файла в текущий документ и там провести изменения.

**С**

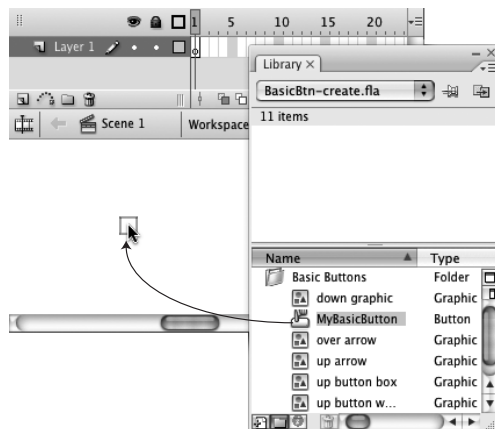
Всегда закрашивайте графику в опорном кадре Hit одним и тем же цветом, скажем, яркосиним или каким-то другим ярким цветом, который больше нигде в ролике не используется. Это станет еще одним визуальным напоминанием о том, что вы редактируете именно кадр Hit, входящий в состав кнопки.

**С**

Чтобы посмотреть, как выглядят состояния «отпущена», «под курсором» и «нажата», выберите кнопку на библиотечной панели и нажмите кнопку **Play** в области предварительного просмотра. Flash по очереди выведет каждый кадр.

### Чтобы просмотреть состояния кнопки на арене:

- Отметьте в меню флажок **Control** ⇒ **Enable Simple Buttons** (Управление ⇒ Разрешить использование простых кнопок) (рис. 12.9.). В ответ на движения мышью и нажатия Flash будет отображать кадры Up, Over и Down. Не забывайте, что когда режим простых кнопок активирован, нельзя ни выбрать кнопку, ни работать с ней. Чтобы отключить режим активации простых кнопок, сбросьте флажок **Control** ⇒ **Enable Simple Buttons**.



**Рис. 12.8.** Чтобы включить символ-кнопку в ролик, перетащите ее экземпляр с библиотечной панели на арену.



**Рис. 12.9.** По умолчанию в режиме редактирования документа на арене отображается только содержимое опорного кадра Up символа-кнопки. Чтобы увидеть, как меняется состояние при взаимодействии с мышью, отметьте флажок **Control** ⇒ **Enable Simple Buttons**.

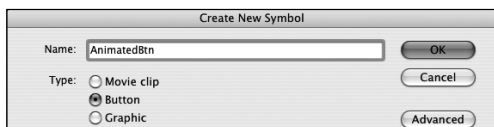


Рис. 12.10. В окне Create New Symbol назовите свой символ-кнопку и задайте для него тип Button.

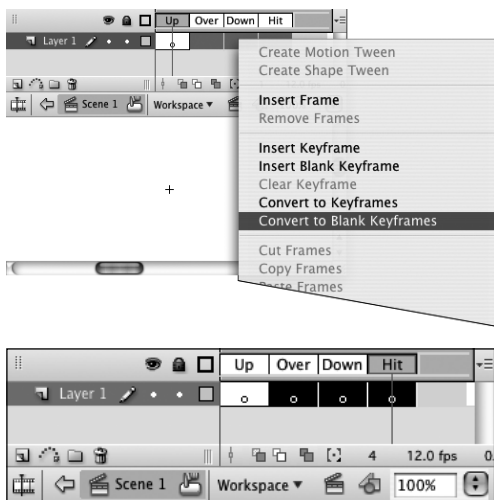


Рис. 12.11. Выбрав кадры Over, Down и Hit, откройте контекстное меню и выберите из него пункт Convert to Blank Keyframes. Flash добавит пустые опорные кадры во все выбранные позиции.

## Создание символов-кнопок с изменяющейся формой

Графические кнопки могут имитировать реальные переключатели. Например, в игре можно замаскировать кнопку под подмигивающий глаз персонажа. Отыскание на картинке активных областей или кнопок становится дополнительным развлечением. Однако, если в кадрах Up, Over и Down символа-кнопки находятся изображения разной формы и размера, то создание хорошего изображения в кадре Hit может оказаться непростой задачей. Это изображение должно включать в себя графику во всех остальных состояниях.

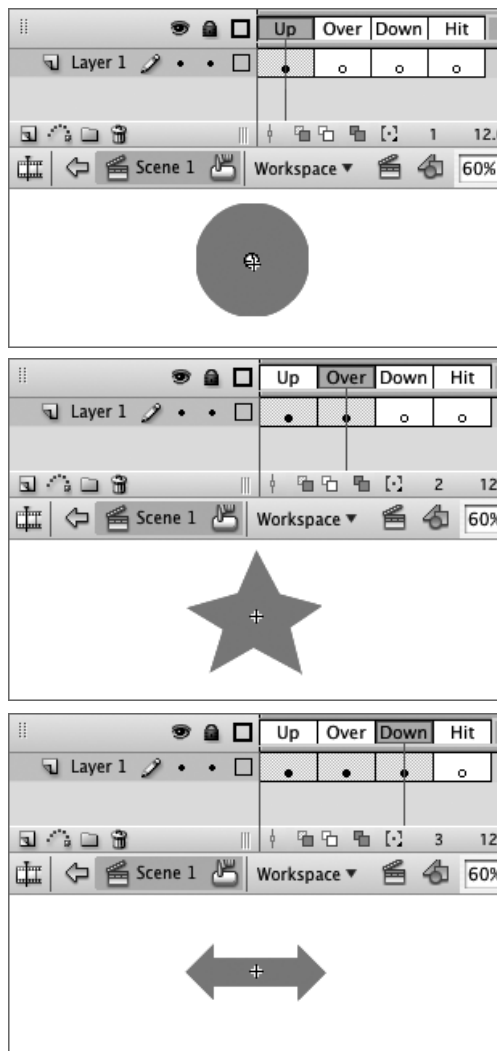
### Чтобы создать состояния «отпущена», «под курсором» и «нажата» с различной графикой:

1. Откройте документ, в который хотите добавить кнопки.
2. Выполните команду **Insert ⇒ New Symbol** (Вставить ⇒ Новый символ). Открывается диалоговое окно New Symbol.
3. Введите имя символа (например, AnimatedBtn), задайте тип Button и нажмите **OK** (рис. 12.10).  
Flash создаст новый символ на библиотечной панели и перейдет в режим редактирования символа на арене и временной шкале. Временная шкала символа-кнопки содержит четыре кадра: Up, Over, Down и Hit.
4. Выберите на временной шкале кадры Over, Down и Hit.
5. Щелкните мышью с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows) по любому из выбранных кадров и выберите из контекстного меню пункт **Convert to Blank Keyframes** (Преобразовать в пустые ключевые кадры) (рис. 12.11).

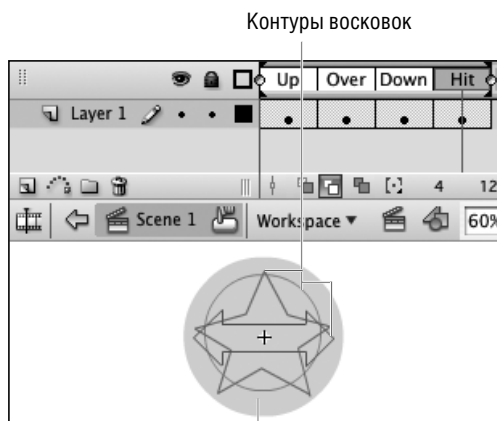
Теперь все кадры кнопки являются опорными и можно помещать в них графику.

6. Выбрав на временной шкале опорный кадр Up, создайте в нем какое-нибудь изображение на арене или поместите экземпляр графического символа, который станет состоянием Up кнопки.
7. Повторите шаг 6 для опорных кадров Over и Down.

В этом упражнении выберите различные формы для каждого кадра, скажем круг, звезду и двунаправленную стрелку (рис. 12.12).

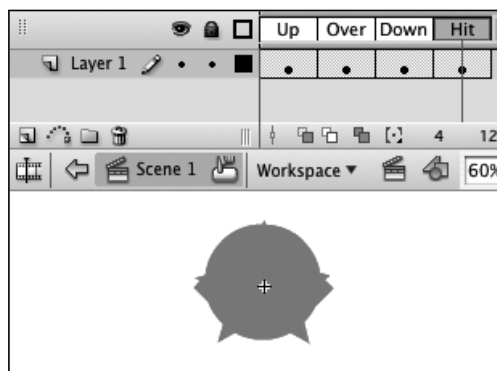


**Рис. 12.12.** Можно создавать замысловатые кнопки, меняющие форму при наведении курсора или нажатии. В данном случае в неактивном состоянии кнопка выглядит как круг (верхний рисунок). Когда курсор находится над ней, вместо круга отображается звезда (средний рисунок). А когда пользователь нажимает кнопку, она превращается в двунаправленную стрелку (верхний рисунок).



Прозрачная заливка для графики в кадре Hit

**Рис. 12.13.** Силуэт в кадре Hit должен охватывать все возможные состояния кнопки. Например, если вы включите в кадр Hit только круг, то лучи звезды останутся неохваченными. Когда пользователь наведет курсор на любой луч, кнопка вернется в состояние Over и активировать ее щелчком в этой области будет невозможно. Если же включить только звезду, то, проводя курсором по некоторым частям круга, пользователь так и не поймет, что это кнопка.



**Рис. 12.14.** Скопировав изображения из всех состояний кнопки и поместив их в кадр Hit командой Paste in Place, вы построите идеальный силуэт, охватывающий все возможные состояния.

**Чтобы создать состояние «активна», когда используются различные изображения:**

1. Продолжая работу с файлом из предыдущего упражнения, выберите на временной шкале опорный кадр Hit.
2. Чтобы создать изображение, описывающее активную область кнопки, выполните одно из следующих действий:
  - нарисуйте простую геометрическую форму, охватывающую все области кнопки. Включив показ шлейфа восковок, вы сможете лучше увидеть, что именно необходимо охватить (рис. 12.13).
  - С помощью команд **Edit** ⇒ **Copy** и **Edit** ⇒ **Paste in Place** скопируйте графические элементы из первых трех опорных кадров кнопки в кадр Hit. Изображения накладываются друг на друга и в совокупности как раз определяют область, достаточную для представления любого состояния кнопки (рис. 12.14).
3. Вернитесь в режим редактирования документа, например, с помощью команды **Edit** ⇒ **Edit Document**.

Теперь можно поместить экземпляр кнопки на арену и протестировать ее, выполнив команду **Control** ⇒ **Test Movie**.

**С**

Для графики в опорном кадре используйте прозрачный цвет (с коэффициентом альфа меньше 100 процентов). В режиме показа восковок изображения в кадрах Up, Down и Over будут просвечивать сквозь графику в кадре Hit, поэтому последнюю будет проще позиционировать, так чтобы она охватывала все состояния.

## Создание анимированных символов-кнопок

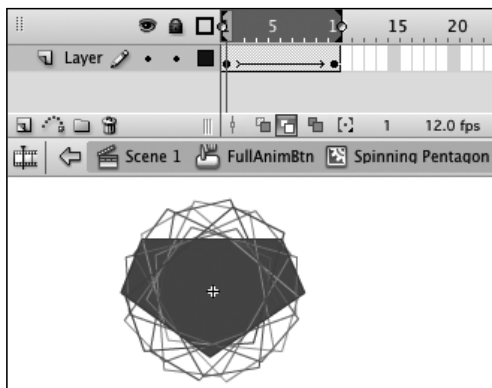
Символы-кнопки, созданные в предыдущих разделах, были анимированы лишь в том смысле, что изменялись при взаимодействии с пользователем. Но Flash позволяет создавать и полностью анимированные кнопки, например, мигающую лампочку или божью коровку, которая подпрыгивает и выкрикивает «Нажми меня!». Фокус в том, чтобы поместить в опорные кадры кнопки символы-клипы. Поскольку для воспроизведения символа-клипа используется отдельная временная шкала, то анимация такой кнопки не прекращается даже после приостановки основного ролика.

### Чтобы анимировать наведение курсора на кнопку:

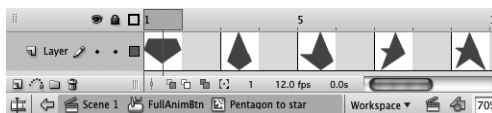
1. Откройте документ, в который будут добавлены кнопки.
2. Выполните команду **Insert** ⇒ **New Symbol**. Открывается диалоговое окно **New Symbol**.
3. Назовите свою кнопку (например, **FullAnimBtn**), присвойте ей тип **Button** и нажмите **OK**.
4. На временной шкале выберите кадры **Over**, **Down** и **Hit** и выполните команду **Modify** ⇒ **Timeline** ⇒ **Convert to Blank Keyframes**.

Flash создаст пустые опорные кадры в выбранных позициях.

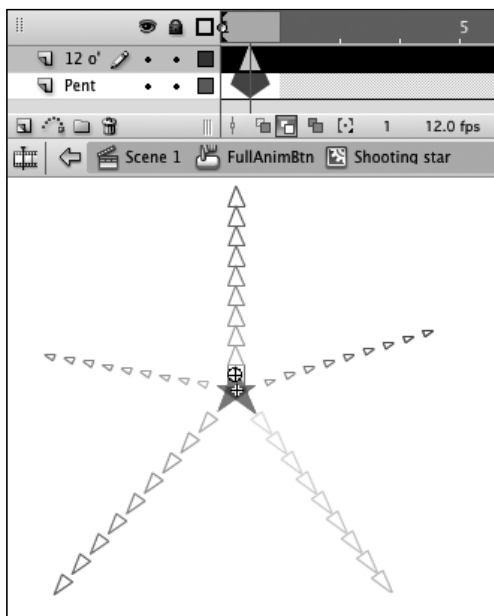
5. Выберите на временной шкале опорный кадр **Up** и вставьте в него экземпляр символа-клипа на арене. Для этого примера возьмите вращающийся пятиугольник (рис. 12.15).
6. Выберите на временной шкале опорный кадр **Over** и вставьте в него экземпляр символа-клипа на арене. В данном примере клип содержит пятиугольник, трансформирующийся в звезду (рис. 12.16).



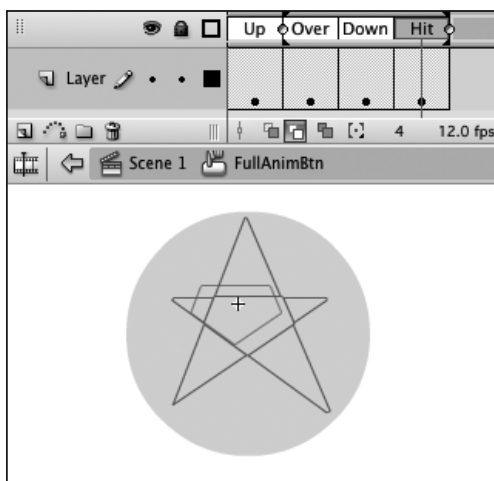
**Рис. 12.15.** Наличие анимированного символа в кадре **Up** символа-кнопки означает, что эта анимация будет воспроизводиться, когда кнопка отпущена. На восковках в режиме редактирования символа видны все кадры, описывающие анимацию вращающегося пятиугольника.



**Рис. 12.16.** Анимированный символ в кадре **Over** будет воспроизводиться, когда курсор проходит над кнопкой или задерживается над ней. В режиме предварительного просмотра временной шкалы видно, как пятиугольник трансформируется в звезду.



**Рис. 12.17.** Анимированный символ, находящийся в клипе, который помещен в опорный кадр Down, воспроизводится, когда пользователь щелкает мышью внутри активной области кнопки. На восковках в режиме редактирования символа видны все кадры разлетающейся звезды.



**Рис. 12.18.** Создавая изображение для опорного кадра Hit, включите показ восков, чтобы видеть начальные кадры роликов в каждом кадре кнопки. Здесь изображение в кадре Hit закрашено прозрачным цветом, это помогает расположить его так, чтобы была охвачена графика из трех остальных кадров.

7. Повторите шаг 6 для опорного кадра Down. В этом примере кадр Down содержит разлетающуюся звезду (рис. 12.17).
8. Выберите на временной шкале кадр Hit и создайте в нем изображение, охватывающее графику в остальных трех кадрах. Лучше всего подойдет большой овал (рис. 12.18). При таком выборе активная область оказывается больше вращающегося пятиугольника. Когда курсор окажется рядом с пятиугольником, кнопка переключится в состояние Over. В этом состоянии овал целиком охватывает звезду, а в состоянии Down, даже если рука пользователя немного дрогнет, курсор все равно не выйдет за пределы кнопки.
9. Вернитесь в режим редактирования документа, щелкнув по имени текущей сцены на панели редактирования.
10. Перетащите копию символа FullAnimBtn с библиотечной панели на арену.

**C**

Можно поместить символ-клип и в кадр Hit, но активную область будет определять только первый кадр клипа.

**C**

Если в режиме редактирования документа простые кнопки активированы, то Flash показывает кадры Up, Over и Down символа-кнопки, но не полную анимацию. В каждом состоянии виден только начальный кадр клипа. Чтобы увидеть анимацию целиком, необходимо экспортировать ролик и воспроизвести его в Flash-плеере (например, выполнив команду **Control** ⇒ **Test Movie**).

### О компонентах

Компонентом во Flash называется предопределенный элемент пользовательского интерфейса со встроенной графикой и поведением, который легко включить в собственные ролики. Компания Adobe спроектировала компоненты так, чтобы при их совместном использовании в сайте не создавалось впечатления разнobia. Поведение компонентов запрограммировано на языке ActionScript. Во Flash CS3 есть компоненты двух видов: отдельно для версии языка ActionScript 1.0 / 2.0 и отдельно для версии 3.0.

Компоненты размещаются на *панели компонентов*, которую можно считать библиотекой. Состав компонентов на этой панели зависит от того, какая версия ActionScript указана в параметрах публикации. Если вы работаете с документом типа Flash File (ActionScript 3.0), то панель компонентов настраивается на компоненты для версии 3.0, и никаких других на ней нет. (О параметрах публикации и переключении версии ActionScript см. главу 17.)

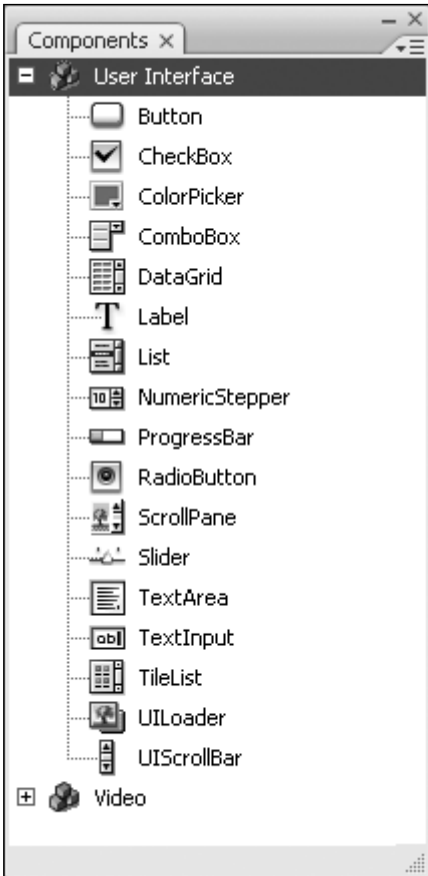
Компонент — это, по сути дела, сложный запрограммированный символ-клип. Для версий ActionScript 1.0 и 2.0 компонент представлен клипом особого вида — *откомпилированным клипом*. В ActionScript 3.0 видеокomпоненты — это тоже откомпилированные клипы, а компоненты пользовательского интерфейса — обычные клипы, которые можно легко модифицировать. См. врезку «Темы компонентов в ActionScript 3.0» ниже. (Внешний вид откомпилированного клипа тоже можно изменить, но эта процедура довольно сложна и выходит за рамки настоящей книги.)

Flash позволяет изменять некоторые свойства компонента (например, надпись на кнопке) путем настройки параметров на панели инспектора компонентов или на вкладке **Parameters** в инспекторе свойств.

Опытные пользователи Flash могут с помощью ActionScript организовать взаимодействие компонентов между собой и динамически изменять компоненты во время выполнения, но, даже не обладая обширными знаниями, можно с помощью компонентов добавить в ролик интерактивность.

Знающий программист может создать и собственные компоненты и поделиться ими с другими пользователями. Одним из источников новых компонентов является раздел Adobe Exchange на сайте компании Adobe. По мере появления компонентов, разработанных сторонними фирмами, вы можете добавлять их на панель компонентов.





**Рис. 12.19.** На панели компонентов перечислены все компоненты, поставляемые в составе Flash. Для ActionScript 3.0 доступно две категории компонентов: пользовательский интерфейс (User Interface) и видео (Video). Кнопки относятся к первой из них. Для сворачивания/разворачивания ветвей списка служит значок треугольника (Mac) или плюс (Windows) слева от названия категории компонентов.

## Использование компонентов-кнопок

Элементы, составляющие пользовательский интерфейс приложения, дают пользователю возможность взаимодействовать с содержанием. К их числу относятся меню, флажки, полосы прокрутки и т.д. Символ-кнопка — это частный случай интерфейсного элемента, в который уже встроено простое поведение (отображение predetermined состояний) и настраиваемый внешний вид. Однако во Flash есть много других интерфейсных элементов, которые представлены компонентами (см. врезку «О компонентах»). Поведение и внешний облик компонентов можно модифицировать. Поведение компонентов-кнопок управляется более развитым сценарием, а встроенная в них графика помогает сохранить единообразие ролика. Чтобы воспользоваться компонентом, нужно поместить его экземпляр в документ Flash.

### Чтобы поместить экземпляр компонента-кнопки:

1. Откройте документ, в который будет добавлен компонент-кнопка.
2. Откройте панель компонентов, инспектор компонентов, библиотечную панель и вкладку **Parameters** в инспекторе свойств.

Все эти панели можно открыть из меню **Window**. Так, чтобы попасть на вкладку **Parameters**, выберите из меню пункт **Window** ⇒ **Properties** ⇒ **Parameters** или, если инспектор свойств уже открыт, щелкните по язычку вкладки **Parameters**.

3. Если необходимо, раскройте список компонентов пользовательского интерфейса на панели компонентов (рис. 12.19).

Для сворачивания/разворачивания ветвей списка служит значок треугольника (Mac) или плюс (Windows) слева от названия категории компонентов.

4. Перетащите экземпляр компонента-кнопки на арену.

Flash добавит сам компонент-кнопку и папку Component Assets в библиотеку документа (рис. 12.20).

**С**

Компоненты аналогичны всем прочим библиотечным активам (подробнее о работе с библиотечными активами см. главу 7). У этой схожести есть и отрицательная сторона — вы можете переименовать компонент на библиотечной панели, дважды щелкнув по его имени (Button) и введя другое. Но делать это не рекомендуется. Так можно изменить имя главного компонента, но его экземпляр на арене по-прежнему будет называться Button. Кроме того, если вы включите в тот же документ еще одну копию оригинальной кнопки, то у обоих компонентов будет одинаковый идентификатор связи, что может создать проблемы при работе с ними в опубликованном ролике.



**Рис. 12.20.** Компонент — это частный случай символа-клипа. При перетаскивании компонента Button с панели компонентов на арену кнопка становится активом (типа Component) на библиотечной панели документа.

## Control

Play	Enter
Rewind	Ctrl+Alt+R
Go To End	
Step Forward One Frame	,
Step Backward One Frame	,
Test Movie	Ctrl+Enter
Test Scene	Ctrl+Alt+Enter
Test Project	Ctrl+Alt+P
Delete ASO Files	
Delete ASO Files and Test Movie	
Loop Playback	
Play All Scenes	
Enable Simple Frame Actions	Ctrl+Alt+F
Enable Simple Buttons	Ctrl+Alt+B
✓ Enable Live Preview	
Mute Sounds	Ctrl+Alt+M

**Рис. 12.21.** Чтобы во время редактирования документа получить представление о том, как выглядит компонент, отметьте флажок Control ⇒ Enable Live Preview.



**Рис. 12.22.** Если режим Enable Live Preview включен (по умолчанию), то во время разработки вы будете видеть на арене экземпляр компонента, определяемый текущими значениями параметров (верхний рисунок). Если этот режим выключен, то компоненты отображаются в виде прямоугольных контуров без всякого указания на то, какой именно элемент они представляют (нижний рисунок).

### Чтобы предварительно просмотреть экземпляр компонента:

- Чтобы во время разработки посмотреть, как выглядит компонент, отметьте в меню флажок **Control ⇒ Enable Live Preview** (Управление ⇒ Разрешить активный просмотр) (рис. 12.21). По умолчанию этот флажок уже отмечен. При работе с большим числом компонентов в этом режиме быстродействие может снизиться, поэтому лучше его на время отключить. Если флажок **Enable Simple Buttons** дает возможность увидеть состояния Up, Over и Down символа-кнопки, то **Enable Live Preview** показывает компонент-кнопку в состоянии Up, определяемом текущими значениями параметров (рис. 12.22). Чтобы увидеть работу компонента-кнопки в полном объеме необходимо просматривать ее в Flash-плеере. Если режим **Enable Live Preview** выключен, то Flash отображает экземпляры компонентов на арене просто как прямоугольные контуры.
- Чтобы увидеть все параметры и состояния компонента-кнопки, выполните команду **Control ⇒ Test Movie**.

## Модификация компонентов-кнопок

В инспекторе компонентов для компонента-кнопки есть семь изменяемых параметров. В упражнениях ниже мы будем работать только с двумя: изменять надпись на кнопке и задавать для кнопки режим переключателя. Для изменения размеров кнопки можно также воспользоваться вкладкой **Parameters** в инспекторе свойств.

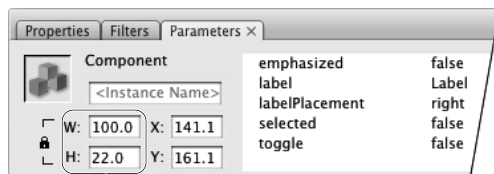
### Чтобы изменить размеры компонента-кнопки:

1. Продолжая работу с файлом из предыдущего упражнения, выберите на сцене экземпляр компонента Button.
2. На вкладке **Parameters** в инспекторе свойств введите новые значения ширины и высоты в поля W (Ш) и H (В) (рис. 12.23).

По умолчанию поля W и H заблокированы, то есть Flash сохраняет пропорции компонента (отношение ширины к высоте). Для целей этого упражнения щелкните по значку замка, чтобы отключить режим сохранения пропорций. Затем введите в поля W и H одинаковые значения, чтобы кнопка стала квадратной.

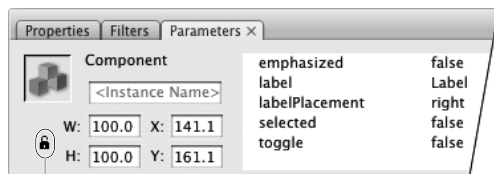
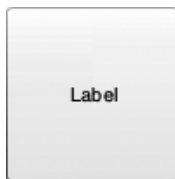
3. Нажмите **Enter** для фиксации новых значений.

Размеры кнопки изменятся (рис. 12.24). Активная область кнопки определяется охватывающим прямоугольником. При изменении размеров Flash автоматически модифицирует активную область компонента-кнопки.



Ввести ширину и высоту

**Рис. 12.23.** Чтобы изменить размеры компонента-кнопки, перейдите на вкладку Parameters в инспекторе свойств и введите ширину и высоту в поля W и H.



Сохранение пропорций включено



Сохранение пропорций выключено

**Рис. 12.24.** Прямоугольную кнопку можно превратить в квадратную, если ввести одинаковые значения ширины и высоты (здесь 100 пикселей). Но сначала необходимо отключить режим сохранения пропорций. Для этого щелкните по значку замка на вкладке Parameters в инспекторе свойств.

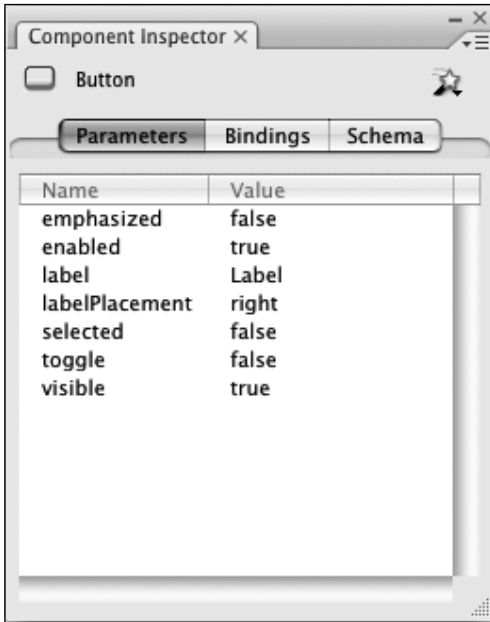


Рис. 12.25. В инспекторе компонентов представлены все параметры компонента-кнопки, которые можно изменять на этапе разработки.

**С**

Есть несколько способов изменить размер компонента-кнопки. Можно, выбрав экземпляр компонента, изменить его размер на панели преобразований или с помощью инструмента «Свободное преобразование». Или ввести ширину и высоту на панели информации или на вкладке **Properties** в инспекторе свойств.

**С**

Можно также повернуть или перекосить компонент-кнопку с помощью инструмента «Свободное преобразование» или панели преобразований. Однако следует помнить, что надпись на кнопке рисуется одним из шрифтов устройства, которые нельзя ни вращать, ни наклонять. Поэтому, если кнопка повернута или перекошена, то во время выполнения никакой надписи на ней не будет.

**С**

Чтобы точно позиционировать компонент-кнопку, введите значения координат  $x$  и  $y$  на панели информации или на вкладке **Properties** в инспекторе свойств.

### Чтобы изменить надпись на компоненте-кнопке:

1. Продолжая работу с файлом из предыдущего упражнения, выберите на арене экземпляр компонента **Button**.
2. Перейдите в инспектор компонентов. На этой панели есть три вкладки: **Parameters** (Параметры), **Bindings** (Привязки) и **Schema** (Схема).
3. Если активна не вкладка **Parameters**, перейдите на нее, нажав кнопку **Parameters** (Mac) или щелкнув по язычку вкладки (Windows). На этой вкладке отображается таблица с двумя колонками (рис. 12.25). В первой находятся имена параметров, во второй — значения.

4. Чтобы изменить надпись на кнопке, щелкните по слову **Label** во второй колонке. Это поле станет изменяемым, и вы сможете ввести новую надпись, например, **testComponent** (рис. 12.26).

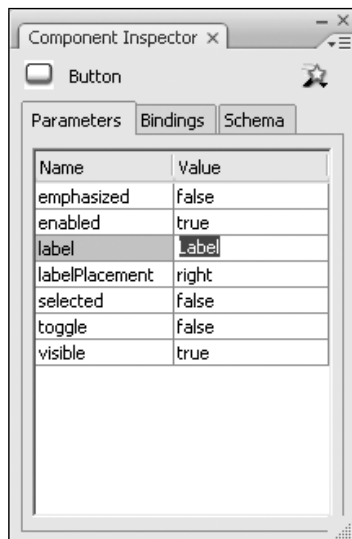
5. Нажмите **Enter** или щелкните мышью вне активного текстового поля.

При стандартных настройках Flash надпись появляется внутри компонента-кнопки (рис. 12.27).

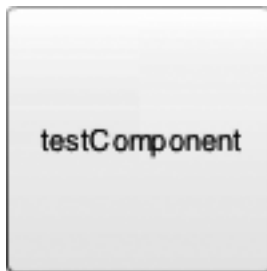
Если вы не видите новый текст, отметьте в меню флажок **Control ⇒ Enable Live Preview**. Но имейте в виду, что, если надпись окажется шире кнопки, то Flash обрежет текст по ширине видимой области.

**С**

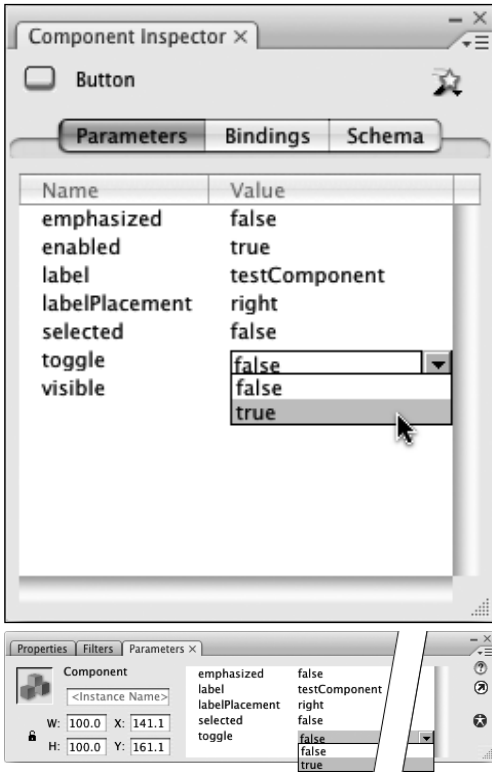
Большую часть свойств компонента-кнопки можно изменить также в инспекторе свойств. Выбрав кнопку на арене, перейдите на вкладку **Parameters** инспектора свойств, там вы увидите поля для задания параметров.



**Рис. 12.26.** В инспекторе компонентов щелкните по значению параметра в строке **Label**, чтобы активировать поле ввода. Введите новый текст надписи и нажмите **Enter**, чтобы подтвердить изменение.



**Рис. 12.27.** При включенном режиме **Live Preview Enabled** новая надпись появляется внутри компонента-кнопки на арене.



**Рис. 12.28.** При щелчке по значению в строке Toggle на вкладке **Parameters** в инспекторе компонентов (верхний рисунок) или в инспекторе свойств (нижний рисунок) раскрывается список, содержащий строки true и false.



**Рис. 12.29.** Если в строке Toggle для экземпляра компонента-кнопки выбрано значение true, то кнопка работает как переключатель. На этом рисунке показано, как выглядят на арене экземпляры компоненты-кнопки в режиме активного просмотра. Кнопка слева выключена, справа — включена (подсвечена).

### Чтобы перевести компонент-кнопку в режим переключателя:

1. Выберите экземпляр компонента-кнопки на арене.
2. В инспекторе компонентов или на вкладке **Parameters** в инспекторе свойств щелкните в любом месте строки Toggle, чтобы раскрыть список значений и выберите из него true (рис. 12.28). Теперь кнопка будет работать как переключатель, то есть при последовательных нажатиях ее внешний будет изменяться (рис. 12.29).

**С**

На вкладке **Parameters** в инспекторе свойств или на вкладке **Parameters** в инспекторе компонентов есть также поле selected, принимающее значение true или false. Оно позволяет задать начальное состояние кнопки-переключателя (если оно равно true, то кнопка в начальный момент подсвечена, если false — то нет).

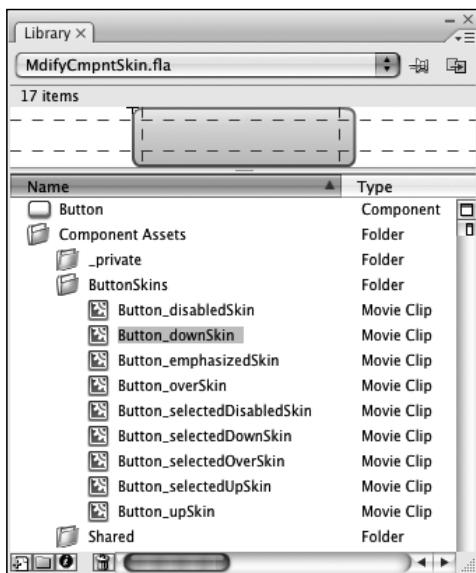
**С**

Чтобы быстро раскрыть список значений, дважды щелкните по значению в строке Toggle.

### Темы компонентов в ActionScript 3.0

Компоненты пользовательского интерфейса в ActionScript 3.0 можно редактировать так же, как любой другой символ-клип. Рассмотрим в качестве примера компонент Button.

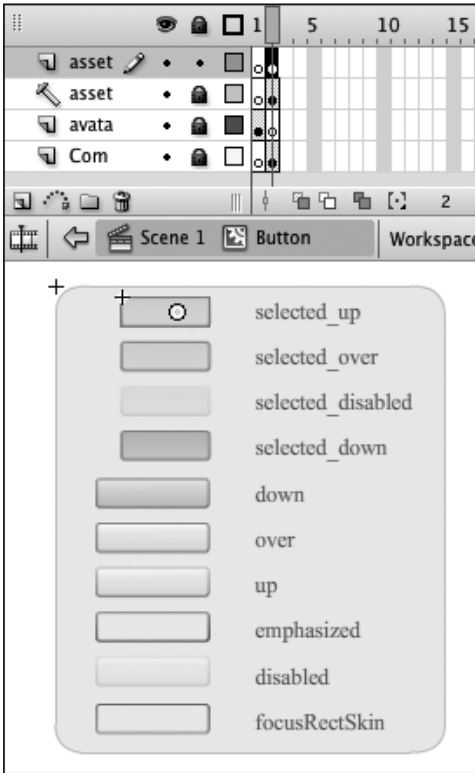
При перетаскивании экземпляра компонента Button на арену Flash добавляет главный компонент и папку Component Assets в библиотеку документа. Внутри папки с активами есть подпапка, содержащая темы (skins) компонента. Для компонента Button она называется ButtonSkins и содержит все символы, из которых составляются различные состояния кнопки (рис. 12.30). Как и любые символы, эти тоже можно модифицировать, изменяя тем самым внешний вид компонента. Чтобы изменить какой-то символ напрямую, дважды щелкните по нему в библиотеке; Flash откроет символ в режиме редактирования. Можете вместо этого дважды щелкнуть по экземпляру компонента на арене, чтобы получить доступ сразу ко всем темам в режиме редактирования символов. Верхний слой на временной шкале символа называется assets. Во втором опорном кадре в нем находится символ, который в свою очередь содержит все темы компонента, в данном случае — состояния кнопки (рис. 12.31). Дважды щелкните по любой теме, чтобы начать его редактирование (рис. 12.32).



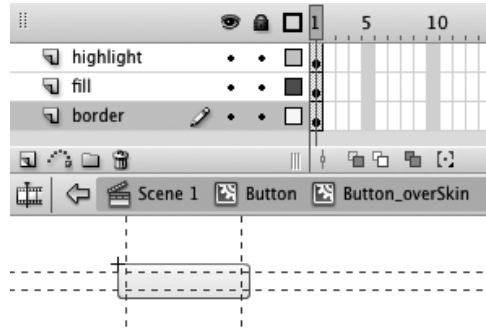
**Рис. 12.30.** Подпапка, в названии которой есть слово *skin*, содержит все символы, определяющие внешний облик компонента.



Чтобы после изменения кнопка осталась симпатичной и дружелюбной, вам, вероятно, придется модифицировать все относящиеся к ней символы. А, возможно, и другие компоненты, которые вы планируете использовать в том же приложении или сайте, что и кнопку. Изменения, внесенные в темы компонента, затрагивают все экземпляры этого компонента в ролике.



**Рис. 12.31.** Двойной щелчок по экземпляру компонента на арене открывает темы главного символа для редактирования. Символы-темы находятся в опорном кадре 1 в слое assets.



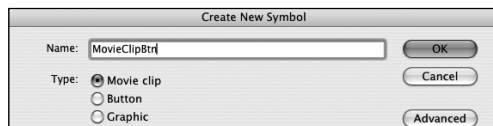
**Рис. 12.32.** Дважды щелкните по символу-теме, чтобы начать его редактирование.

## Создание кнопок-клипов

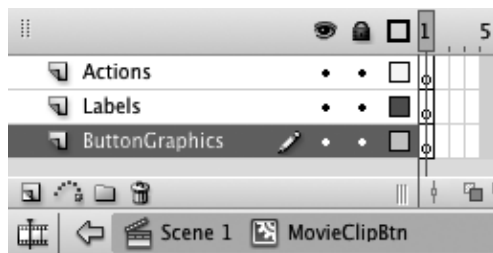
Для символов-кнопок и компонентов-кнопок во Flash определены встроенные правила, описывающие то, как кнопка выглядит во всех трех состояниях, сменяющих друг друга в ответ на действия пользователя. Можно взять управление этим на себя, а также создать кнопку, имеющую больше трех состояний. Для этого понадобится кнопка-клип. В следующих упражнениях вы научитесь создавать из клипа кнопку с четырьмя состояниями: `_up`, `_over`, `_down` и `Disabled`. Чтобы наделить кнопку-клип хотя бы первым уровнем интерактивности, то есть заставить ее реагировать на движения мыши, понадобится сценарий на языке ActionScript. Эта тема будет рассмотрена в главе 13.

### Чтобы создать состояния кнопки:

1. В документе, где вы собираетесь использовать кнопки-клипы, выполните команду **Insert** ⇒ **New Symbol**. Открывается диалоговое окно **New Symbol**.
2. Введите имя символа — `MovieClipBtn`, в качестве типа укажите **Movie Clip** (Фрагмент ролика) и нажмите **ОК** (рис. 12.33). Flash переходит в режим редактирования символа. На временной шкале имеется один слой с опорным кадром в первой позиции.
3. Добавьте еще два слоя. В каждом слое будет размещаться различная информация о кнопке (рис. 12.34). В верхнем слое мы поместим ActionScript-сценарий, управляющий кнопкой, поэтому назовите его **Actions**. Во втором слое будут храниться тексты, идентифицирующие все опорные кадры, которые представляют состояния кнопки; его назовем **Labels**. В последнем слое будут графические элементы, определяющие внешний вид кнопки в каждом состоянии; пусть он называется **ButtonGraphics**.



**Рис. 12.33.** Первый шаг конструирования кнопки-клипа — создание нового символа. Выполните команду **Insert** ⇒ **New Symbol** и в диалоговом окне присвойте символу имя и укажите тип **Movie Clip**.

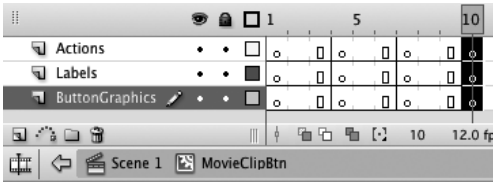


**Рис. 12.34.** При конструировании кнопки-клипа рекомендуется создать на временной шкале клипа отдельные слои для действий, текста и графики.

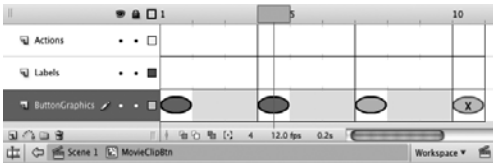
### Зачем нужны кнопки-клипы?

Символы-кнопки во Flash всем хороши, но у них всего три состояния: `Up`, `Over` и `Down`. А иногда нужно больше. В хороших пользовательских интерфейсах элементы используются последовательно и единообразно. Поэтому пользователь знает, что есть в его распоряжении и где искать необходимые для конкретной задачи элементы. Возьмем, к примеру, типичную форму для отправки электронного сообщения. В ней должна быть кнопка «Отправить». В идеале она присутствует всегда, но не активна (и выглядит соответствующим образом) до тех пор, пока не будут заполнены все обязательные поля, в частности, адрес получателя.

Создавая кнопку на основе клипа, вы можете сделать столько состояний, сколько вам требуется.



**Рис. 12.35.** Для каждого состояния кнопки-клипа необходим опорный кадр. Наличие длинных последовательностей промежуточных кадров позволяет визуально организовать слои более удобно и оставляет место для меток кадров.



**Рис. 12.36.** В режиме предварительного просмотра кадров на временной шкале отображаются все графические элементы, которые вы поместили в опорные кадры символа-клипа. Для состояния Disabled используйте оттенок серого цвета. (Поскольку книга печатается не в цвете, мы для ясности еще поместили букву X внутри овала, представляющего состояние Disabled.)

4. На временной шкале во всех трех слоях вставьте опорные кадры в позиции 4, 7 и 10 (рис. 12.35).

В добавленных слоях в первой позиции уже находятся опорные кадры. Но, чтобы реализовать все четыре состояния `_up`, `_over`, `_down` и `Disabled`, надо вставить дополнительные кадры. Оставив между ними промежутки, мы упростим себе работу и сможем видеть метки, которые создадим в следующем упражнении.

5. В слое `ButtonGraphics` выберите опорный кадр 1. Нарисуйте в нем овал с центром в точке регистрации на арене.

Эта форма будет представлять кнопку в состоянии `_up`. Залейте овал красным цветом и оставьте черную обводку. Пусть обводка будет пошире, чтобы форма больше напоминала кнопку.

6. Выберите овал и выполните команду **Edit** ⇒ **Copy**.
7. Выберите опорный кадр 4 в слое `ButtonGraphics` и выполните команду **Edit** ⇒ **Paste in Place**. Измените цвет заливки на зеленый; эта форма будет представлять кнопку в состоянии `_over`.
8. Повторите шаг 7 для кадров 7 и 10.

В кадре 7 измените цвет заливки на синий, это будет кнопка в состоянии `_down`. А в кадре 10 сделайте овал светло-серым, чтобы представить состояние `Disabled` (рис. 12.36).

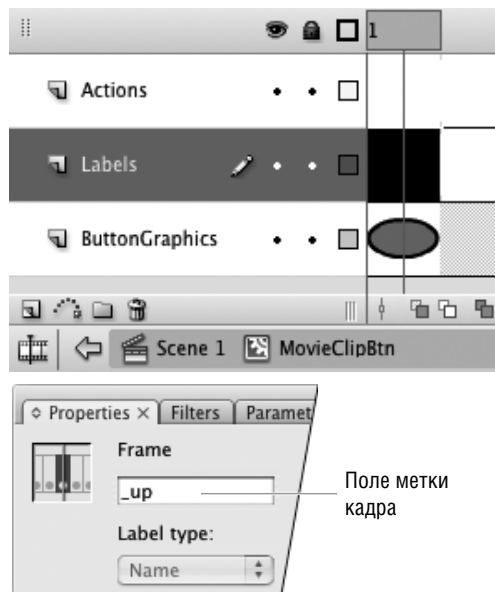
## С

В кнопке-клипе необязательно создавать состояние `Hit`. Когда вы добавите сценарий (см. главу 13), для определения активной области будут использоваться графические элементы, представляющие в клипе другие состояния.

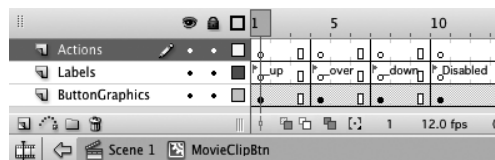
**Чтобы сопоставить метки кадрам, в которых представлены состояния кнопки:**

1. Продолжая работу с файлом из предыдущего упражнения, перейдите на вкладку **Properties** в инспекторе свойств. Если эта панель еще не открыта, выберите из меню пункт **Window** ⇒ **Properties** ⇒ **Properties**.
2. В слое Labels на временной шкале выберите опорный кадр 1.
3. Щелкните мышью в поле Frame Label (Метка кадра) на вкладке **Properties** в инспекторе свойств, введите имя состояния `_up` и нажмите **Enter** (рис. 12.37).

Как только вы введете первый символ имени, станет доступен раскрывающийся список Label Type (Тип метки). Оставьте значение по умолчанию — Name (Имя). По завершении ввода метки Flash поместит красный флажок во все опорные кадры, для которых задана метка типа Name. Если в последовательности опорного кадра достаточно места, то отображается также текст метки (рис. 12.38).



**Рис. 12.37.** Введите в поле Frame Label в инспекторе свойств метку выбранного опорного кадра, она будет представлять название состояния кнопки.



**Рис. 12.38.** Красный флажок в опорном кадре показывает, что с кадром связана метка. Если вслед за этим кадром идет достаточно промежуточных, то Flash отображает также саму метку (мы добавили в конец последовательности пустые кадры, так что метка Disabled видна). Помеченные опорные кадры в этом символе-клипе представляют различные состояния кнопки.

4. Повторите шаги 2 и 3 для кадров 4, 7 и 10, присваивая меткам значения `_over`, `_down` и `Disabled`.

Для сопоставления опорным кадрам меток есть две причины. Во-первых, метка сообщает о том, что находится в кадре. Во-вторых, — и это важнее — в сценарии можно будет сослаться на кадр по имени метки и сделать этот кадр текущим. Мы еще воспользуемся этим приемом, чтобы реализовать визуальные изменения кнопки в ответ на манипуляции мышью (см. главу 13).

5. Вернитесь в режим редактирования документа, например, нажав кнопку «Назад» на панели редактирования.
6. Перетащите экземпляр символа `MovieClipBtn` на арену.

Для этого клипа уже можно написать сценарий, который заставит его вести себя подобно кнопке. О том, как наделить кнопку интерактивностью, вы узнаете в главе 13. Чтобы посмотреть, как выглядят состояния, нажмите кнопку **Play** в области предварительного просмотра символа на библиотечной панели или выполните команду **Control ⇨ Test Movie** для воспроизведения клипа. Flash прокрутит опорные кадры и покажет по очереди все состояния кнопки.

7. Сохраните документ под именем `MyOwnBtn.fla`, он еще понадобится в главе 13

### Об именовании меток кадров

Flash очень трепетно относится к именам. Имена меток кадров становятся частью путей имен в сценариях на языке `ActionScript`. Поэтому некоторые символы, имеющие в сценариях специальное назначение, — косая черта, знак равенства, плюс и т.п. — не должны встречаться в именах. На всякий случай ограничьтесь только буквами, цифрами и символом подчеркивания; не используйте даже пробелы для разделения слов в именах. Вместо этого начинайте каждое слово с заглавной буквы или разделяйте слова подчеркиваниями.

C

ActionScript-сценарий воспринимает метки `_up`, `_down` и `_over` как состояния кнопки, поэтому для реализации функциональности кнопки предстоит сделать совсем немного (см. главу 13). Чтобы кнопка была более гибкой (например, по-разному реагировала на разные движения мышью), назовите метки по-другому. Можно использовать имена `MyUp`, `MyOver`, `MyDown`, `MyDragOut` и т.д.

C

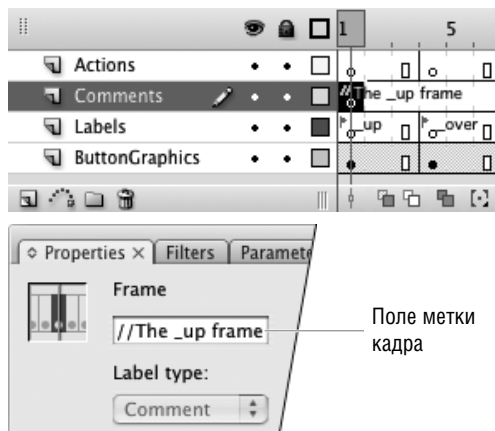
Еще один способ напомнить себе о том, для чего нужен каждый кадр, — оставить комментарий. Для этого выберите опорный кадр и в поле `Frame Label` в инспекторе свойств введите две косые черты (`//`) и текст комментария (рис. 12.39). Метки и комментарии взаимно исключают друг друга, у кадра может быть либо то, либо другое. Чтобы обойти это ограничение, заведите разные слои для меток и комментариев. В обоих слоях вставьте опорные кадры, а потом в одном слое введите метки, а в другом — комментарии.

C

Можно не вводить символы косой черты, а просто набрать текст комментария. Затем выберите из списка `Label Type` пункт `Comment` (Комментарий), и Flash добавит эти символы за вас.

C

Поскольку ActionScript может использовать метки кадров для реализации интерактивности, Flash экспортирует их наряду с другими данными ролика при публикации. Поэтому старайтесь делать метки короткими, чтобы не увеличивать размер файла. Длина комментария не ограничена, но не забывайте, что следующий опорный кадр обрезает комментарий к предыдущему. Поэтому комментарии должны уместиться в оставленном для них промежутке на временной шкале.



**Рис. 12.39.** Поле `Frame Label` на вкладке `Properties` в инспекторе свойств можно использовать и для ввода комментария к выбранному опорному кадру. Начните текст с двух символов `//` или выберите из списка `Label Type` пункт `Comment`. Комментарии отображаются на временной шкале в последовательности опорного кадра; длинный комментарий обрезается следующим опорным кадром.

# ОСНОВЫ ИНТЕРАКТИВНОСТИ

# 13

## Версии языка ActionScript

В состав Flash CS3 входят две версии языка ActionScript: 2.0 (она включает также версию 1.0) и 3.0. Версия ActionScript 3.0 существенно отличается от предыдущих, и в одном ролике смешивать их нельзя. Так какую же версию выбрать?

У ActionScript (AS) 2.0 есть некоторые достоинства с точки зрения пользователей, впервые приступающих к созданию интерактивного Flash-ролика. В этой версии интерактивность реализуется очень конкретно, сценарии можно присоединять к кнопкам и другим объектам напрямую. Но стоит выйти за пределы простейших манипуляций, как все становится очень сложно.

У версии ActionScript 3.0 свои преимущества. В ней процедуры написания сценариев стандартизованы, поэтому знания, приобретенные в ходе работы над одними аспектами, легко применить к другим. Поскольку именно AS 3.0 будет развиваться дальше, мы в этой книге сосредоточимся именно на ней. Для создания сценариев, в которых используется AS 3.0, необходимо в параметрах документа отметить, что он будет публиковаться в расчете на эту версию. Это можно сделать в момент создания документа или позже в диалоговом окне Publish Settings. (Дополнительную информацию о публикации см. в главе 17.)

Программа Adobe Flash CS3 Professional — это не просто инструмент для создания мультяшек или анимированных баннеров. С ее помощью можно строить сложные интерактивные Web-сайты для электронного обучения, торговли и других Интернет-приложений. В состав Flash входит полнофункциональный язык сценариев ActionScript, позволяющий разрабатывать интерактивные программы. Сценарии вообще и язык ActionScript в частности — достаточно обширные темы, и детальное их обсуждение выходит за рамки настоящей книги. Однако есть ряд типичных задач, для которых написать сценарий совсем нетрудно. В этой главе вы будете писать на языке ActionScript 3.0 сценарии, наделяющие Flash-ролик простыми интерактивными возможностями. Сценарий представляет собой структурированный текст. Некоторые символы, например открывающая и закрывающая фигурная скобка, служат для выделения блоков кода, без них сценарий не будет работать. Чтобы не запутаться, вы будете сразу же вводить такие символы парами, а потом между ними вставлять необходимый код. По ходу дела мы будем неоднократно дополнять и улучшать ранее написанные сценарии. На рисунках приводятся листинги, содержащие законченные фрагменты кода. Если листинг призван иллюстрировать какие-то конкретные шаги упражнения, то текст, напрямую относящийся к этим шагам, выделяется полужирным шрифтом, тогда как текст, рассмотренный выше, набран обычным шрифтом. Полные тексты сценариев имеются на сопроводительном сайте по адресу <http://www.peachpit.com/title/0321502914>.

## Знакомство с панелью действий

На панели действий выделяются три рабочие области: область сценария, инструментарий и навигатор по сценарию. Чтобы открыть эту панель, выберите из меню пункт **Window ⇨ Actions** (Окно ⇨ Действия) или нажмите комбинацию клавиш **Option-F9** (Mac) либо **F9** (Windows).

Область сценария (*Script pane*) — это окно, в котором собирается текст сценария. Код действий можно вводить вручную (как в текстовом редакторе) либо копировать из инструментария или с помощью всплывающего меню **Add** (Добавить). (Чтобы открыть это меню, щелкните по значку плюс на панели инструментов, расположенной прямой над областью сценария.) Сценарии или их фрагменты можно импортировать из внешнего файла, созданного в каком-нибудь автономном текстовом редакторе.

### О программировании на языке ActionScript

*Сценарий* представляет собой последовательность команд или предложений, которые Flash интерпретирует во время выполнения, то есть при воспроизведении опубликованного SWF-файла в Flash-плеере (см. главу 17).

Создание сценария сводится в основном к вводу текста. Обычно вы набираете текст в области сценария на панели действий, а Flash сохраняет его как часть FLA-файла. Можно сохранить сценарий и в отдельном файле с расширением *.as*, и сообщить Flash, что включать его нужно только в момент публикации. Так сохраненные файлы можно редактировать во внешней программе.

В режиме ассистирования (Script Assist) панель действий помогает вводить код. Слово *assist* (помощь) звучит очень привлекательно для начинающих, но на самом деле работа в этом режиме требует определенных знаний о языке. Суть его состоит в том, что вы создаете код, отмечая разные флажки, выбирая что-то из меню и вводя текст в специальные поля. При этом гарантируется, что синтаксис будет правильным (см. врезку «О синтаксисе языка ActionScript» ниже). Однако для работы в режиме ассистирования вы уже должны знать, какие предложения языка необходимы для создания интерактивности.

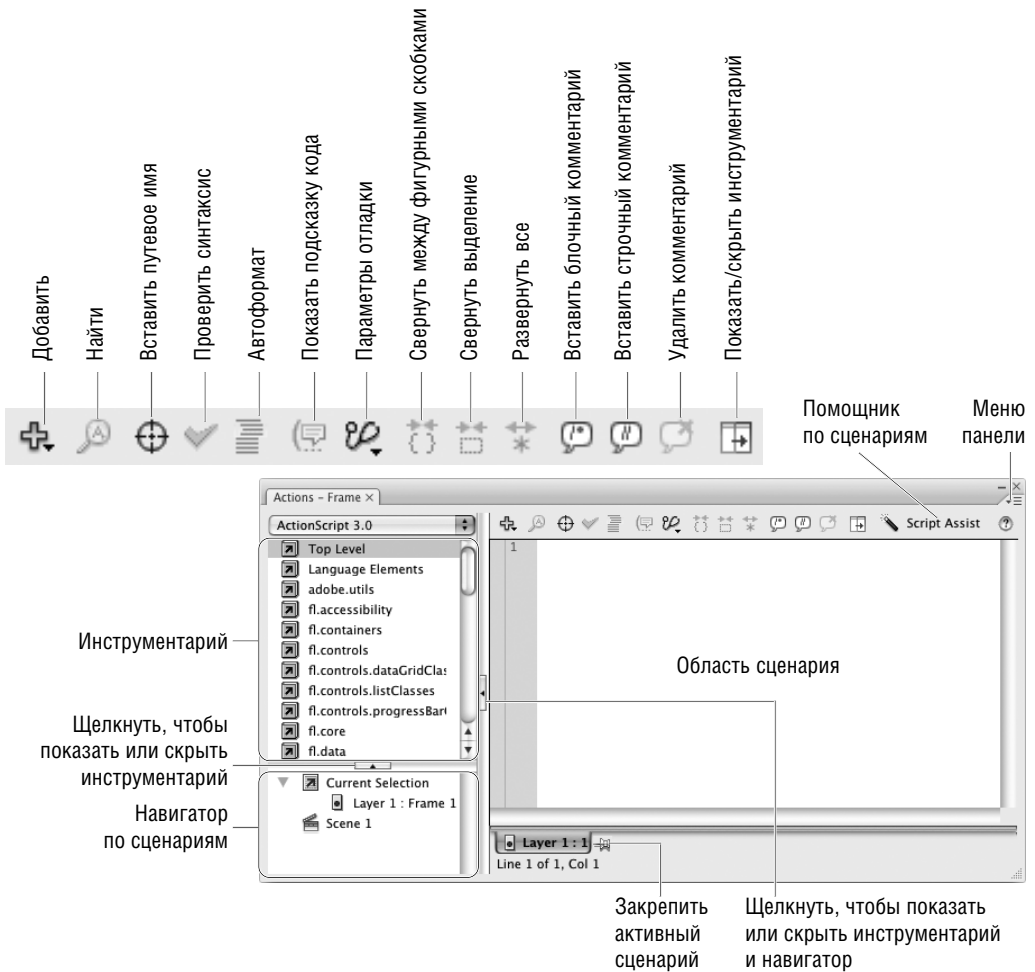
В предыдущих версиях Flash была попытка помочь в написании сценариев с помощью поведений (Behaviors), которые автоматически создавали код для решения конкретной задачи, например, создания ссылки на Web-сайт. Код, созданный поведением, не совместим с ActionScript 3.0. Поэтому, если в качестве языка документа выбран ActionScript 3.0, то панель поведений будет недоступна.

Самый лучший способ добавить интерактивность в ролик — написать необходимые сценарии самостоятельно. Хотя для тех, кто раньше никогда не программировал, подобное предложение звучит пугающе, на самом деле это проще, чем изучать инструменты, предоставляемые Flash для автоматизации процесса написания кода. В упражнениях ниже мы будем создавать сценарии для выполнения простых задач, например, настройки кнопок, которые запускают и останавливают воспроизведение ролика. Рассматриваемые приемы не годятся для разработки онлайн-магазина или сложного учебного курса. Но для проигрывания мультфильма, выбора любимой сцены или перехода на Web-страницу их вполне достаточно. А, быть может, это пробудит в вас желание изучить ActionScript более обстоятельно.



*Инструментарий* (Actions Toolbox) включает по большей части слова (названия действий), составляющие основу языка ActionScript. Действия организованы иерархически; щелкните по значку, напоминающему папку, чтобы просмотреть содержимое категории. Двойной щелчок по действию добавляет его код в область сценария. Можно также перетаскивать элементы из инструментария в область сценария. (Список действий доступен также с помощью меню **Add**.)

*Навигатор* (Script Navigator) позволяет переключаться с одного сценария на другой и производить поиск. Это особенно полезно, когда имеется много сценариев для различных опорных кадров на временной шкале. На рис. 13.1 показаны элементы панели действий.



**Рис. 13.1.** Панель действий состоит из трех основных частей: инструментарий, из которого можно выбирать действия; навигатор, где отображаются те элементы ролика, к которым присоединены сценарии, и область сценария, в которой собирается полный текст сценария.

## Настройка панели действий

Внешний вид панели действий можно настроить по своему вкусу. Например, изменить размер самой панели и ее частей или скрыть некоторые части. Можно настроить шрифт, которым отображается сценарий, цветовое выделение синтаксических элементов и число пробелов, добавляемых при нажатии клавиши табуляции. Можно включить или выключить режим выдачи подсказок к коду.

### Чтобы задать параметры для панели действий:

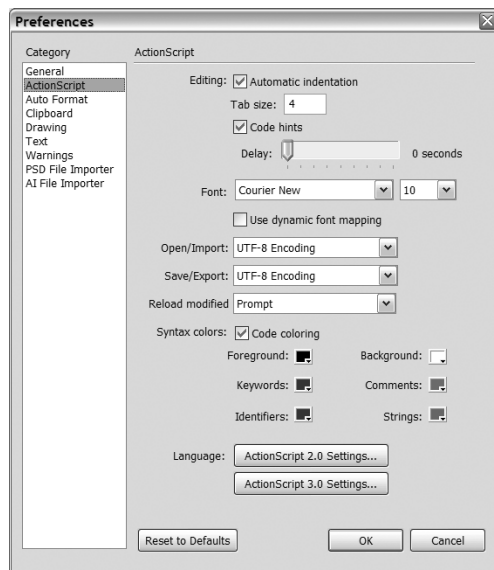
1. Выберите из меню пункт **Flash ⇨ Preferences** (Mac) или **Edit ⇨ Preferences** (Windows).

или

Из меню, расположенного в правом верхнем углу панели действий выберите пункт **Preferences** (Установки).

Открывается диалоговое окно Preferences.

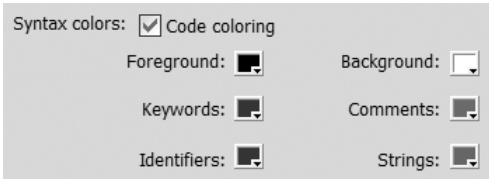
2. Выберите категорию **ActionScript**. В центральной области окна отображаются параметры работы с областью сценария на панели действий (рис. 13.2).
3. Чтобы получить подсказки к коду, отметить флажок **Code Hints** (Подсказки кода) (рис. 13.3). С помощью ползунка задайте, сколько секунд Flash должен подождать, прежде чем начать подсказывать, когда вы вводите текст непосредственно в области сценария.



**Рис. 13.2.** В окне Preferences выберите категорию ActionScript, в которой собраны параметры, относящиеся к работе со сценариями.



**Рис. 13.3.** Чтобы Flash предлагал подсказки к коду, отметьте флажок Code Hints. Чтобы подсказки появлялись быстрее, сдвиньте ползунок Delay (Задержка) влево.



**Рис. 13.4.** Чтобы включить подсветку синтаксиса отметьте флажок **Code Coloring** в секции **Syntax Coloring** из категории **ActionScript** в окне **Preferences**. Измените цвета, чтобы различные "части речи" языка **ActionScript** четко отличались друг от друга.

4. Для задания шрифта сценария выполните следующие действия:

- выберите шрифт из списка установленных шрифтов;
- выберите размер из списка размеров шрифта.

Сценарии могут отображаться шрифтом размера от 8 до 72 пунктов.

5. Для подсветки синтаксиса отметьте флажок **Code Coloring** (Цветовое оформление кода) (рис. 13.4).

Воспользуйтесь элементами управления цветом, чтобы открыть окно образцов, в котором задайте любой из следующих цветов:

**Foreground** (Основной цвет). Цвет, которым отображаются части сценария, не относящиеся к перечисленным ниже.

**Keywords** (Ключевые слова). Цвет слов, зарезервированных для самого языка **ActionScript**.

### Что такое интерактивность?

По умолчанию все сцены и кадры в опубликованном **Flash**-ролике воспроизводятся последовательно. Ролик начинается со сцены 1, проигрывает все принадлежащие ей кадры по порядку, переходит к сцене 2 и т.д. Иногда это приемлемо, иногда — нет.

Представьте себе учебный курс, в котором в начале каждого урока появляются одни и те же пять кадров с инструкциями общего характера. Эти кадры можно оформить в виде отдельной сцены и продублировать перед каждой сценой, содержащий материал урока. Но дублирование увеличивает размер файла, снижает производительность и затрудняет последующее редактирование. Более эффективно было бы потребовать, чтобы **Flash** повторял одну и ту же сцену перед каждым уроком. Кроме того, можно сделать так, чтобы обучаемый имел возможность совершать активные действия: нажимать на кнопки, перетаскивать элементы или вводить текст в ответ на контрольные вопросы. А сам курс мог бы оценивать правильность ответов. Это и есть примеры интерактивности во **Flash**.

Для реализации такой интерактивности необходим сценарий на языке **ActionScript**, который будет управлять воспроизведением ролика.

**Identifiers** (Идентификаторы). Имена объектов, переменных и функций, встроенных в ActionScript.

**Background** (Фон). Цвет фона, на котором отображается текст сценария.

**Comments** (Комментарии). Текст, содержащий пояснения о работе программы, который игнорируется при интерпретации сценария.

**Strings** (Строки). Последовательности символов (букв, цифр и знаков препинания), заключенные в кавычки.

## 6. Нажмите ОК.

Flash применяет новые значения параметров немедленно.

C

По умолчанию цвета ключевых слов и идентификаторов очень похожи (оттенки синего). Попробуйте сделать их более контрастными, скажем, розовым и оранжевым. Тогда вам будет проще распознавать различные элементы программы на языке ActionScript.

C

Подсказки к коду всплывают на панели действий, если режим ассистирования не активен. Если подсказки включены, то можно настроить панель действий так, что они будут отображаться либо в виде всплывающих окон, либо в виде выпадающего меню. Поместите каре (текстовый курсор) справа от символа точки или открывающей фигурной скобки в области сценария и нажмите кнопку **Show Code Hints** (Показать подсказку кода). Если режим показа подсказок включен, то они будут появляться автоматически после ввода точки, двоеточия или открывающей фигурной скобки в подходящем контексте.

### Два стиля написания сценариев

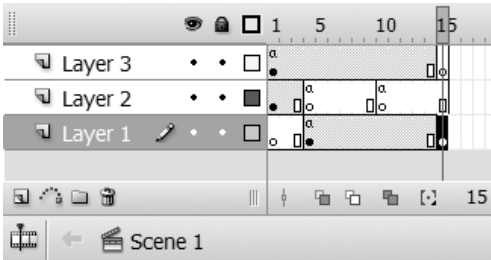
В ActionScript 3.0 есть два способа организации сценариев. Можно создавать сценарии, связанные с *действиями в кадре*, или основанные на *пользовательских классах*. Код в обоих случаях хранится по-разному. В упражнениях из этой книги создаются сценарии обработки действий в кадре, которые присоединяются к опорным кадрам и хранятся в FLA-файле. Сценарии, основанные на пользовательских классах, необходимо сохранять во внешних AS-файлах.

Для программирования действий в кадре вы выбираете опорный кадр, затем переходите на панель действий и вводите текст сценария. Flash выполняет содержащиеся в нем инструкции. С их помощью можно управлять временной шкалой напрямую (например, приостановить воспроизведение), настраивать интерактивность кнопок (например, сказать Flash, что нужно делать при нажатии конкретной кнопки) или изменять свойства объектов, находящихся в кадре (например, переместить экземпляр символа на арене).

Сценарии на основе пользовательских классов аналогичны библиотечным символам. Создание и редактирование классов выходит за рамки данной книги. Тем не менее отметим, что при выполнении упражнений из предыдущих глав вы на самом деле уже взаимодействовали с некоторыми из встроенных во Flash классов, например, Button и MovieClip. Знающие программисты могут создавать свои классы в редакторе ActionScript или во внешнем текстовом редакторе.



**Рис. 13.5.** Опорные кадры, содержащие действия, обозначаются буквой *a* на временной шкале.



**Рис. 13.6.** Когда действия распределены по многим слоям (верхний рисунок), их труднее отыскивать, и можно случайно назначить действия опорным кадрам, находящимся в одной позиции, но в разных слоях. Выделение отдельного слоя для действий (нижний рисунок) упрощает поиск и позволяет сразу увидеть, с какими кадрами ассоциированы действия.

## Организация действий в кадре

Маленькая буква *a* на временной шкале обозначает опорный кадр, к которому присоединены действия (рис. 13.5). Но невозможно же перелопачивать десятки, а то и сотни слоев, выискивая букву *a*, всякий раз, как вы хотите изменить действия в своем документе. Этого никакие глаза не выдержат. И к тому же как быть, если вы (или кто-то другой) впоследствии захотите изменить сценарий; ведь запомнить (или угадать), где притаились действия, очень трудно. Хорошо бы поместить все связанные с кадрами действия в отдельный слой.

Такое решение заодно предотвратит случайное присоединение действий к опорным кадрам, находящимся в одной и той же позиции, но в разных слоях; это могло бы стать причиной ошибок при изменении порядка слоев.

### Чтобы создать отдельный слой для действий в кадре:

1. Откройте документ Flash.
2. Добавьте новый слой на временную шкалу (см. главу 6).
3. Назовите слой *Actions*.
4. Перетащите слой наверх временной шкалы. Располагая слой с действиями в начале временной шкалы, вы упростите себе жизнь, когда настанет время добавлять или модифицировать действия (рис. 13.6).

**С**

Чтобы случайно не добавить графические элементы в слой *Actions*, заблокируйте его (щелкнув по маркеру в колонке с замком). Блокировка запрещает вносить изменения в элементы этого слоя на арене, но не мешает присоединению действий к опорным кадрам.

**С**

Полезно создавать отдельные слои также для меток и звуков. Если располагать их в начале временной шкалы, то будет проще искать неграфическое содержимое.

## Добавление действий в кадр

К числу простейших задач, требующих написания сценария, относится управление воспроизведением ролика: остановка и возобновление, а также переход из одного места в другое. По умолчанию ролик сразу начинает воспроизводиться; это можно изменить в параметрах публикации (см. главу 17) или с помощью действий.

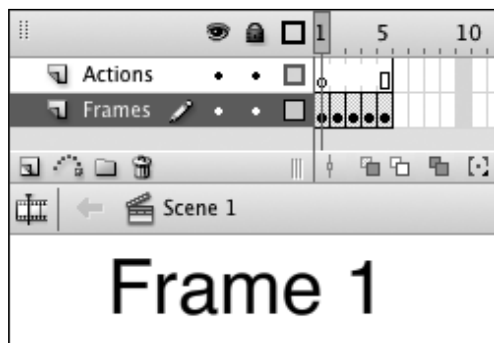
Для создания своего первого сценария во Flash подготовьте многокадровый документ, в каждом кадре которого находится какой-нибудь текст, позволяющий легко опознать кадр. Затем добавьте действие stop в опорный кадр 1, чтобы приостановить воспроизведение в самом первом кадре. Сохраните этот файл как шаблон для последующих упражнений.

### Чтобы подготовить документ для тестирования действий в кадре:

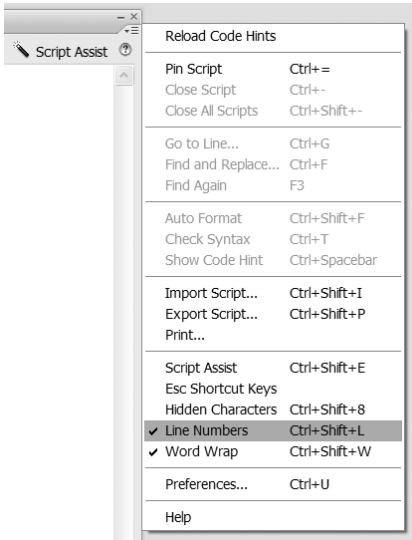
1. Создайте документ с двумя слоями: Actions и Frames.
2. В слой Actions добавьте промежуточные кадры в позициях 2–5; в первой позиции по умолчанию уже находится опорный кадр.
3. В слое Frames создайте опорные кадры в позициях 2–5; в первой позиции также находится опорный кадр
4. В каждый опорный кадр введите на арене текст, содержащий номер этого кадра (Frame 1, Frame 2 и т.д.). Документ должен выглядеть примерно так, как показано на рис. 13.7.

### Чтобы начать сценарий с добавления комментариев:

1. Продолжая работать с документом из предыдущего упражнения, выберите кадр 1 в слое Actions.
2. Перейдите на панель действий. (Если эта панель еще не открыта, выберите из меню пункт **Window** ⇒ **Actions**.) В полосе заголовка панели появится надпись Actions-Frame.



**Рис. 13.7.** Для тестирования действий в кадре (которые содержат указания Flash перейти к конкретному кадру или начать либо остановить воспроизведение) полезно иметь документ, в котором все кадры легко опознаются. Тогда вам будет проще наблюдать за работой сценария.



**Рис. 13.8.** Откройте меню панели действий и убедитесь, что нумерация строк и перенос по словам включены (если нет, включите). Показ номеров строк поможет вам ориентироваться в сценарии, а в режиме переноса по словам проще просматривать текст, поскольку он целиком виден в области сценария.

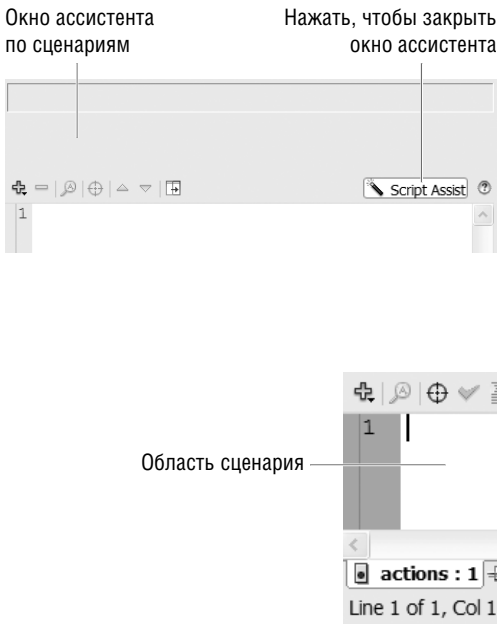
3. Открыв меню панели действий, убедитесь, что флажок **Line Numbers** (Номера строк) отмечен (рис. 13.8). Показ номеров строк поможет вам ориентироваться в своем сценарии.

4. Убедитесь, что флажок **Word Wrap** (Переносить по словам) в меню панели действий отмечен. Если это не так, отметьте его.

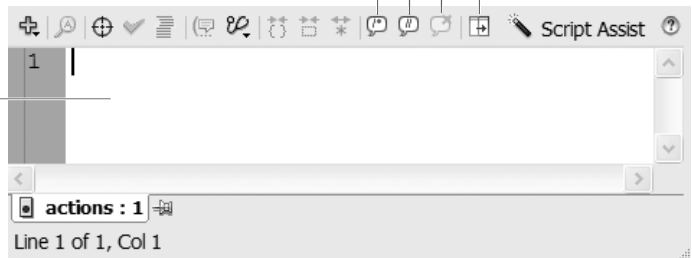
В этом режиме строка переносится, когда превышает ширину области сценария. Для интерпретации сценария это несущественно, так как смысл программы определяется не форматированием, а синтаксической структурой (см. врезку «О синтаксисе ActionScript» ниже).

5. Убедитесь, что режим ассистирования не активен (рис. 13.9). Если необходимо, нажмите кнопку **Script Assist** (Помощник по сценариям), чтобы закрыть окно ассистента.

6. Нажмите кнопку **Apply Line Comment** (Вставить строчный комментарий) на панели инструментов над областью сценария. Flash добавит две косые черты (//) в первую строку в области сценария. В конце строки появится мигающий текстовый курсор.



Удалить комментарий  
Вставить строчный комментарий  
Вставить блочный комментарий  
Показать/скрыть инструментарий



**Рис. 13.9.** На панели инструментов расположен ряд кнопок для работы со сценариями, в том числе Script Assist. Нажмите эту кнопку, чтобы закрыть окно ассистента и печатать непосредственно в области сценария.

- Введите в кадре 1 текст *Pause the movie* (рис. 13.10).

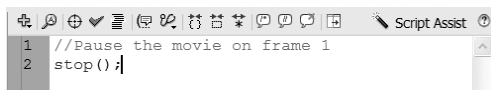
Введенный комментарий добавляется в область сценария. Если режим переноса по словам включен, то строка разбивается по достижении правой границы окна; но реально в тексте сценария комментарий занимает всего одну строку.

### Чтобы приостановить воспроизведение ролика:

- В документе из предыдущего упражнения укажите курсором на конец первой строки (комментария) и нажмите **Enter**. Flash создает в области сценария строку 2.
- Для добавления действия, которое приостановит воспроизведение, наберите `stop()`;  
В области сценария слово `stop`, появляющееся во второй строке, выделяется синим цветом (рис. 13.11).
- Сохраните документ как шаблон для будущего использования, назовите его *Frame-ActionsMaster*. (О сохранении документа в виде шаблона см. главу 1.)
- Закройте документ.



**Рис. 13.10.** Две косые черты обозначают начало комментария. Если нажать кнопку *Insert Line Comment*, Flash вставляет эти символы автоматически.



**Рис. 13.11.** Действие `stop()`; во второй строке приостанавливает воспроизведение ролика.

### О комментариях

Двойная косая черта называется *ограничителем комментария*; ограничитель устанавливает место начала однострочного комментария в сценарии. При выполнении сценария Flash игнорирует текст между двойной косой чертой и концом строки. В откомпилированном сценарии комментария нет, так что размер SWF-файла не увеличивается. Настоятельно рекомендуется оставлять в сценарии комментарии, чтобы напомнить себе о том, для чего он предназначен. Комментарии также окажут неоценимую услугу тому, кто будет модифицировать ваш сценарий впоследствии.

Длинные комментарии включаются иначе. Нажмите кнопку **Apply Block Comment** на панели инструментов, чтобы вставить блочный комментарий. Такой комментарий начинается с пары символов `/*` и заканчивается парой `*/`.

Когда вы тестируете длинные сценарии или пробуете различные способы решения задачи, бывает полезно временно удалить часть программы и посмотреть, что получится. Блочные комментарии позволяют сделать это, не удаляя код. Чтобы «закомментировать» большой участок текста, заключите его в ограничители `/*` и `*/`.



### О синтаксисе языка ActionScript

В языке ActionScript есть свои правила, аналогичные орфографическим и грамматическим правилам естественного языка. В совокупности они называются *синтаксисом* и описывают такие аспекты, как порядок слов, использование заглавных букв и знаков препинания. Ниже приведено краткое описание шести основных знаков препинания, которыми вам предстоит пользоваться в своих сценариях.

**Точка (.)**. В языке ActionScript точка связывает объекты с их *свойствами* (характеристиками) и *методами* (поведением). В предложении

```
myClip.nextFrame();
```

точка означает, что вызывается метод `nextFrame` объекта `myClip` (этот метод продвигает точку воспроизведения к следующему кадру на временной шкале). Точка также используется для разделения компонентов в путевых именах объектов (см. врезку «Путевые имена объектов» ниже).

**Точка с запятой (;)** обозначает конец предложения. Можно опускать точку с запятой — Flash правильно интерпретирует конец строки и без нее, — однако включать ее считается хорошим тоном. В некоторых предложениях употребление точки с запятой в качестве разделителя обязательно.

**Двоеточие (:)**. При первом употреблении переменной (контейнера с изменяющимся содержанием) двоеточие отделяет имя переменной от ее типа.

```
var myName:String;
```

Двоеточие используется также между именем и типом метода.

**Фигурные скобки ({}).** С помощью фигурных скобок определяются блоки логически неразделимых предложений. Например, предложения, следующие за объявлением функции `function greeting()`, должны быть заключены в фигурные скобки. Отметим, что внутри фигурных скобок могут встречаться предложения, для которых в свою очередь требуются фигурные скобки. Открывающие и закрывающие скобки должны быть сбалансированы. Обращайте внимание на то, чтобы каждая закрывающая скобка соответствовала «своей» открывающей.

**Круглые скобки (()).** В круглые скобки заключаются аргументы, относящиеся к конкретному предложению, например, имена сцены и кадра в действии `goto`. Кроме того, с помощью круглых скобок задается порядок выполнения операций в математических выражениях.

**Квадратные скобки ([]).** В сценариях часто используются *массивы*, то есть списки однотипных элементов. Список, помещенный между квадратными скобками, становится массивом.

## Программирование кнопок с помощью сценария кадра

В предыдущей главе вы узнали, что для символов-кнопок и компонентов-кнопок уже определены некоторые встроенные действия. По умолчанию, когда мышь оказывается внутри области кнопки, Flash переходит к кадру Over, а при нажатии кнопки — к кадру Down. Для того чтобы кнопка могла выполнить свою функцию или нестандартно отвечать на движения мышь, к опорному кадру нужно присоединить сценарий и в качестве объекта назначения указать символ-кнопку или компонент-кнопку.

### Чтобы добавить кнопку, управляемую сценарием кадра:

1. Откройте документ по шаблону Frame-ActionsMaster, который создали ранее (подробнее об открытии документов по шаблону см. главу 1).

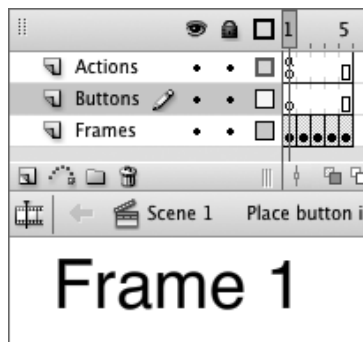
2. Добавьте на временную шкалу слой под слоем Actions и назовите его Buttons.

В позиции 1 нового слоя находится опорный кадр, а в позициях 2–5 — промежуточные кадры. Любые объекты, помещенные в этот слой, будут видны на протяжении всего пятикадрового ролика (рис. 13.12).

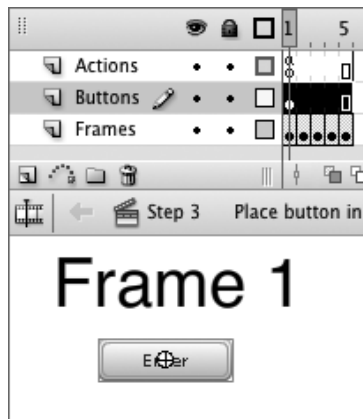
3. Выбрав слой Buttons, поместите на арену экземпляр символа-кнопки (рис. 13.13).

Для создания символа можно поступить так, как описано в главе 12, или взять какой-нибудь символ из разделяемой библиотеки кнопок (чтобы обратиться в этой библиотеке, выберите из меню пункт **Window** ⇒ **Common Libraries** ⇒ **Buttons** (Окно ⇒ Разделяемые библиотеки ⇒ Кнопки)). Перетащите экземпляр символа-кнопки из библиотеки на арену.

4. Чтобы назвать экземпляр кнопки, введите имя в поле Instance name (Имя экземпляра) на вкладке **Properties** инспектора свойств, например, enterBtn (рис. 13.14).

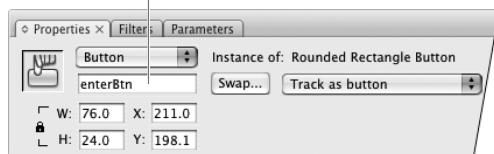


**Рис. 13.12.** Чтобы попрактиковаться с программированием кнопок, создайте ролик со слоями для кнопок и действий. Добавьте необходимое число промежуточных кадров, чтобы оба слоя имели одинаковую длину.



**Рис. 13.13.** За счет промежуточных кадров длина слоя Buttons уравнивается с длиной остальных слоев; символ-кнопка в опорном кадре 1 будет виден на протяжении всего ролика, хотя текст в кадрах из слоя Frames будет меняться.

Поле для ввода имени экземпляра



**Рис. 13.14.** Выберите символ-кнопку на арене и введите его имя на вкладке Properties инспектора свойств.

Возьмите за правило именовать все экземпляры символов в документе. Чтобы экземпляром символа можно было управлять из сценария, у него должно быть уникальное имя.

5. Сохраните файл (например, командой **File** ⇒ **Save**).

**С**

Программирование сценариев — сложный процесс, часто приходится идти путем проб и ошибок. Иногда приходится возвращаться к тому месту, где сценарий еще работал правильно. Команда **Save As** (Сохранить как) упрощает эту процедуру. Когда объекты на арене размещены, и все готово к добавлению кода, выполните команду **File** ⇒ **Save As**. Всякий раз после завершения очередного куска кода выполняйте ее снова. Включайте в имя файла порядковый номер или букву, чтобы потом было легко найти нужную версию сценария.

### Механизм работы подсказок

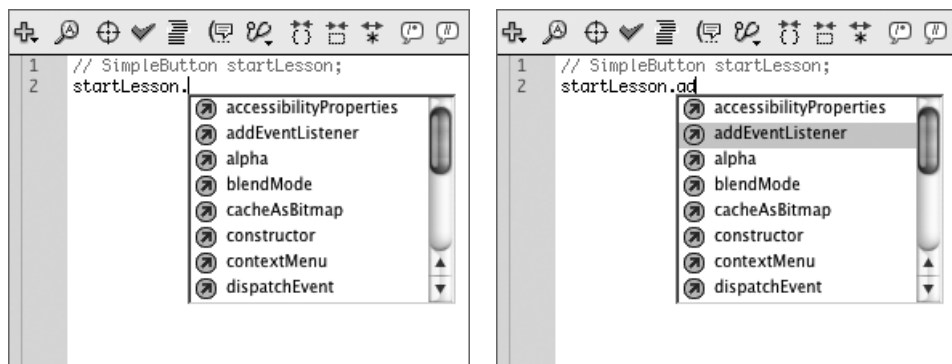
Как работают подсказки к коду? Когда вы вводите текст в области сценария, Flash пытается угадать, что вы собираетесь написать, и оказать вам помощь. Рассмотрим пример:

В строке 1 вы ввели комментарий:

```
// SimpleButton startLesson;
```

Когда в следующей строке вы введете имя экземпляра (`startLesson`) и точку, откроется всплывающее окно, в котором перечислены действия и свойства, применимые к символам-кнопкам (рис. 13.15).

По списку можно перемещаться с помощью полос прокрутки или клавиш со стрелками. Выберите нужный элемент и нажмите **Enter**, чтобы вставить его в текст. Или продолжайте печатать, тогда Flash будет пытаться автоматически завершить код, выделяя в списке первый элемент, начинающийся с тех символов, которые вы уже ввели. Когда нужное вам действие или свойство окажется выделенным, нажмите **Enter**. Можно, конечно, не обращать внимания на автозавершение и продолжать печатать, но этот механизм позволяет избежать опечаток и ускорить ввод текста.



**Рис. 13.15.** Когда набрано имя экземпляра и точка, открывается окно подсказки (левый рисунок). По мере того как вы продолжаете печатать, в это окне выделяется наиболее близкий к уже набранному тексту элемент (правый рисунок).

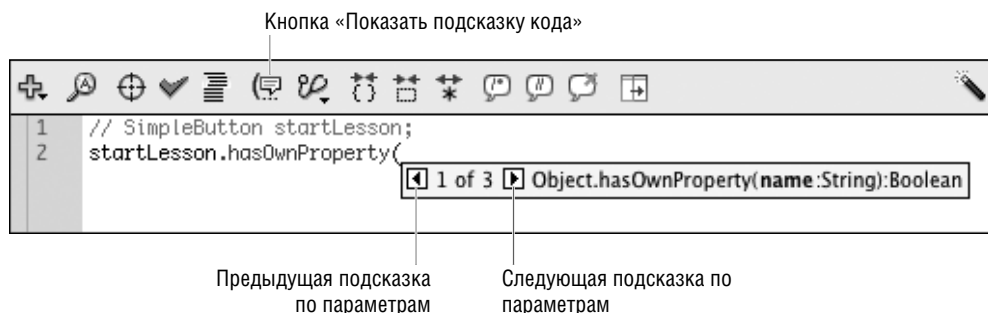
**Чтобы управлять кнопкой из сценария кадра:**

1. Продолжая работу с тем же документом, убедитесь, что панель действий открыта и режим подсказок включен (см. раздел «Настройка панели действий» выше).
2. Выберите опорный кадр 1 в слое Actions.  
В области сценария отображается ранее введенный код — комментарий в строке 1 и действие `stop()`; в строке 2.

**Механизм работы подсказок (продолжение)**

При использовании подсказок для автоматизации ввода действия с параметрами, Flash автоматически добавляет открывающую скобку и предлагает подсказку другого типа — по параметрам (рис. 13.16). (Если вы проигнорировали автозавершение, то подсказка по параметрам появляется после ввода открывающей скобки.) В этой подсказке перечислены все возможные параметры для только что введенного действия. Подсказки по параметрам не приводят к автозавершению, а просто напоминают о том, какие параметры необходимы. Иногда одно и то же действие применимо к нескольким объектам; например, и у `Button`, и у `MovieClip` есть действие `addEventListener`. В таком случае можно воспользоваться стрелочками, чтобы найти то, что вас интересует. Подсказки по параметрам также помогают понять, к какой категории инструментов относится текущий объект; в начале каждой подсказки указан путь к папке, содержащей введенное действие.

Если вы случайно закрыли окно подсказки или позже передумали, то можете открыть его снова. Для этого расположите курсор справа от точки или открывающей скобки и нажмите кнопку **Code Hint** на панели инструментов. Можно также нажать комбинацию клавиш **Option-пробел** (Mac) либо **Ctrl-пробел** (Windows).



**Рис. 13.16.** Подсказка по параметрам появляется после ввода открывающей скобки (или после автозавершения ввода действия или свойства, требующего параметров). Если существует несколько вариантов задания параметров, то для перебора можно воспользоваться кнопками со стрелочками.



**Рис. 13.17.** Комментарий для механизма подсказок должен располагаться в отдельной строке. Расположите курсор в самом конце строки 2, нажмите **Enter**, и Flash установит курсор в начале строки 3, где можно будет ввести текст комментария.

**Листинг. 13.1.** Задание комментария для механизма подсказок

```
1 //Pause the movie on frame 1
2 stop();
3 // code-hint comments
4 // SimpleButton enterBtn;
```

**Листинг. 13.2.** Определение обработчика события

```
3 // code-hint comments
4 // SimpleButton enterBtn;
5 // event handlers
6 function handleClick( pEvent:MouseEvent
7 );void
8 {
9 }
```

### Широкий сценарий — узкие колонки

В сценариях на языке **ActionScript** могут встречаться довольно длинные строки. Размер области сценария на панели действий можно увеличить, а вот ширина колонок в книге неизменна. Поэтому строки приходится переносить. В листингах (см., например, листинг 13.2) понять, где следует нажимать **Enter** для перехода на новую строку, можно по номерам строк. А в тексте упражнений для обозначения строк-продолжений мы будем ставить в начале строки стрелочку. Так, показанный ниже текст следует вводить как одну строку, нажимая клавишу **Enter** только после слова **void**:

```
function handleClick
⇒ ( pEvent:MouseEvent ):void
```

3. Чтобы добавить комментарий для механизма подсказок, выполните следующие действия:

- начните новую строку, для чего поставьте курсор в конце строки 2 — после **stop()**; — и нажмите **Enter**. Flash добавит строку 3 (рис. 13.17);
- в строках 3 и 4 введите текст  
`// Code-hint comments`  
`// SimpleButton enterBtn;`

Если вы назвали экземпляр кнопки иначе, введите свое имя вместо **enterBtn** (листинг 13.1).

4. Для создания функции-обработчика события выполните следующие действия:

- поставьте курсор в конце строки 4 и нажмите **Enter** для создания новой строки;
- в строках 5–8 введите текст:  
`// event handlers`  
`function handleClick(`  
`⇒ pEvent:MouseEvent ):void`  
`{`  
`}`

После ввода двоеточия открывается окно подсказки. Когда выделенным окажется текст, который вы собирались набрать, нажмите **Enter**, чтобы Flash закончил слово (или, если хотите, продолжайте набирать вручную). То, что находится внутри скобок, называется *параметром*; с его помощью передается информация о том, где произошло событие (**pEvent**) и какого типа это событие (**MouseEvent**). Эти два элемента разделяются двоеточием (листинг 13.2). (О синтаксисе функций см. врезку «Функции» ниже).

5. Чтобы убедиться в том, что событие действительно поступило от кнопки `enterBtn`:

- поставьте курсор в конце строки 7 (после открывающей скобки) и нажмите **Enter**. Flash сделает в новой строке 6 отступ, напоминая о том, что последующий код находится внутри скобок и является частью функции;
- в строках 8-10 введите текст  

```
if( pEvent.target == enterBtn )  
{  
}
```

Как только вы введете открывающую скобку, появится окно подсказки; продолжайте печатать. Сейчас вы ввели так называемое *условное предложение* (см. врезку «Предложение if» ниже). Код внутри скобок исполняется только в том случае, когда удовлетворяется заданное условие. Здесь мы проверяем, что источник события соответствует кнопке (`enterBtn`), которой мы хотим управлять из сценария (листинг 13.3).

6. Поставьте курсор в конце строки 9 (после открывающей скобки), нажмите **Enter** и введите в строках 10 и 11 такой текст:

```
// handle the event  
nextFrame();
```

В строках 10 и 11 Flash создаст удвоенный отступ, напоминая о том, что этот код находится внутри функции да еще и является частью предложения `if`.

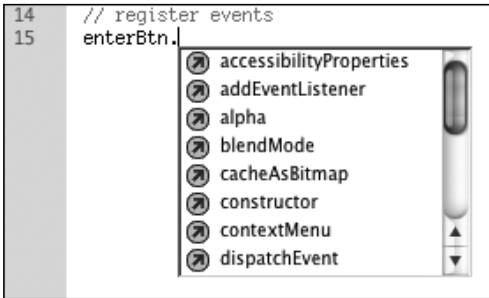
Действие `nextFrame()`; продвигает точку воспроизведения вперед на один кадр (листинг 13.4).

**Листинг 13.3.** Проверяем, что событие поступило от кнопки `enterBtn`.

```
6 function handleClick( pEvent:MouseEvent  
   ):void  
7 {  
8     if( pEvent.target == enterBtn )  
9     {  
10  
11     }
```

**Листинг 13.4.** Полный код обработчика события.

```
1 //Pause the movie on frame 1  
2 stop();  
3 // code-hint comments  
4 // SimpleButton enterBtn;  
5 // event handlers  
6 function handleClick( pEvent:MouseEvent  
   ):void  
7 {  
8     if( pEvent.target == enterBtn )  
9     {  
10  
11         // handle the event  
12         nextFrame();  
13     }
```



**Рис. 13.18.** По мере ввода текста в окне подсказки предлагаются возможные варианты продолжения. Когда подсветится нужный вам элемент, нажмите Enter, и он вставится в код сценария.

**Листинг 13.5.** Полный текст сценария для управления кнопкой (он станет частью шаблона ButtonActionsMaster).

```

1 //Pause the movie on frame 1
2 stop();
3 // code-hint comments
4 // SimpleButton enterBtn;
5 // event handlers
6 function handleClick( pEvent:MouseEvent
   ):void
7 {
8     if( pEvent.target == enterBtn )
9     {
10         // handle the event
11         nextFrame();
12     }
13 }
14 // register events
15 enterBtn.addEventListener(
   MouseEvent.CLICK, handleClick );
    
```

7. Чтобы зарегистрировать событие, выполните следующие действия.

Поставьте курсор в конце строки 13, нажмите **Enter** и в строках 14–15 введите текст:

```
// register events
enterBtn.addEventListener();
```

После ввода точки появится окно подсказки со списком возможных действий (рис. 13.18). После того как вы наберете `ad`, в этом списке подсветится строка `addEventListener`. В этот момент нажмите **Enter** или дважды щелкните мышью. Flash вставит в код строку `addEventListener` и откроет другое окно подсказки. Не обращая на него внимания, наберите `);` и на этом закончите предложение.

8. Чтобы завершить регистрацию события, выполните следующие действия.

- в строке 15 поставьте курсор между скобками;
- наберите `MouseEvent.CLICK, handleClick`

Код `MouseEvent.CLICK` сообщает Flash о том, на какое из нескольких возможных событий кнопки будет реагировать ваш обработчик (см. врезку «Анатомия события `MouseEvent`» ниже). Код `handleClick` указывает на созданную ранее функцию-обработчик, эта строка должна совпадать с именем, которое было введено после ключевого слова `function` в строке 6.

Вы только что запрограммировали свой первый обработчик события от кнопки. Полный текст сценария должен выглядеть, как показано в листинге 13.5. Пора проверить его в действии.

9. Выполните команду **Control ⇒ Test Movie** (Управление ⇒ Тестировать ролик).

Flash опубликует ролик и откроет SWF-файл в Flash-плеере. Вы увидите текст *Frame 1* и кнопку. После нажатия кнопки появится текст *Frame 2*. В ответ на ваше действие точка воспроизведения передвинулась на второй кадр, как и было запрограммировано в сценарии. Каждое нажатие кнопки перемещает точку воспроизведения на один кадр до тех пор, пока не будет достигнут кадр 5.

10. Сохраните документ в качестве шаблона для последующего использования и назовите его `ButtonActionsMaster`.
11. Закройте документ.

С

Если при тестировании ролика вы не получаете ожидаемый результат, откройте окно `Compiler Errors` (Ошибки компиляции) (для этого выберите из меню пункт **Window ⇒ Compiler Errors**) и посмотрите, нет ли каких-нибудь ошибок. Закройте окно ролика и убедитесь, что код набран точно так, как показано в листинге 13.5. Обращайте особое внимание на точки, двоеточия, точки с запятой, фигурные и круглые скобки. Дополнительную информацию о том, как тестировать сценарий, см. в разделе «Тестирование действий» ниже.

С

Окно подсказок не мешает вам печатать, а лишь помогает избежать опечаток при вводе имен действий. Когда окно открыто, вы можете продолжать вводить текст вручную.

С

Вы только что создали шаблон, включающий сценарий программирования кнопок. Этот шаблон можно использовать снова и снова, модифицируя его в соответствии с возникающими потребностями.

### Анатомия события `MouseEvent`

Пользователи могут взаимодействовать с кнопкой разными способами: пересекать курсором активную область кнопки, нажимать и отпускать кнопку мыши, когда курсор находится в активной области, буксировать мышью внутри активной области и т.д. Конструкция `MouseEvent.CLICK` описывает лишь одно из возможных событий мыши. Часть `MouseEvent` сообщает Flash о том, что интересующее вас событие генерируется устройством ввода (например, мышью или пером), а часть `CLICK` — что Flash должен уведомлять о том, что пользователь нажал и отпустил кнопку мыши в активной области.



## Обработчики событий

В языке ActionScript, как и в английском, слово *event*, означает, что произошло некое *событие*, например, пользователь нажал кнопку. *Обработчик события* — это *функция*, описывающая специальное действие, которое должно выполняться в ответ на конкретное событие.

События могут возникать в результате действий человека или наступления определенных условий в самом ролике. В этой главе нас интересует *взаимодействие с пользователем*, то есть события, порождаемые людьми в процессе просмотра ролика, например: нажатие кнопки, ввод текста или отметка флажка. Внутренние события тоже оказывают влияние на работу ролика, и можно написать сценарий, реагирующий на них. (К числу наиболее интересных внутренних событий относится конец загрузки ролика в компьютер пользователя или продвижение точки воспроизведения к следующему кадру.)

## Программирование интерактивности

Чтобы ролик реагировал на то или иное действие пользователя (событие), в сценарии необходимо сделать две вещи: *зарегистрироваться для получения события* и *обработать событие*. Обработка события сводится к написанию функции, которую Flash будет вызывать при наступлении события. Эта функция называется *обработчиком события*. Для регистрации следует написать код, который свяжет объект, возбудивший событие (например, кнопку), с его обработчиком. Код обработки события выглядит примерно так:

```
function handleEvent( event )
{
    if( event.target == eventSource )
    {
        // обработать событие...
    }
}
```

Функция-обработчик *handleEvent* ждет наступления события. Если событие поступило от объекта *eventSource*, то обработчик выполняет то, что от него требуется. Вместо *handleEvent* обычно задается имя, соответствующее конкретной ситуации. Источник *eventSource* может быть экземпляром кнопки, компонента-кнопки или символа-клипа (дополнительные технические детали см. на врезках «Функции» и «Предложение if» ниже).

Код регистрации событий имеет такой вид:

```
eventSource.addEventListener( eventName, handleEvent );
```

Здесь *eventSource* представляет объект-источник события, а *eventName* — само событие. Детали зависят от источника, например, кнопка может генерировать событие *click*. Наконец, *eventHandler* — это функция, созданная для обработки события.

### Имена экземпляров и подсказки

Имя *экземпляра* объекта на арене — это идентификатор, который может встречаться в сценарии. Во избежание ошибок не включайте в имена пробелы и символы, которые в языке ActionScript имеют специальный смысл, например, / или =. Лучше всего ограничиться буквами, цифрами и подчеркиком. В идеале имя экземпляра должно начинаться с маленькой буквы.

Присваивая имена экземплярам символов-кнопок, клипов и текстовых полей, вы можете сообщить Flash дополнительную информацию об именуемом объекте. Эта информация используется, чтобы помочь вам во время ввода текста сценария.

Существует два способа активировать режим подсказок для конкретных объектов. Первый состоит в том, чтобы включить в имя экземпляра предопределенный суффикс. Так, кнопки идентифицируются суффиксом `_btn`, клипы — суффиксом `_mc`, а текстовые поля — суффиксом `_txt`.

Другой способ заключается в том, чтобы добавить строку комментария, в которой создается связь между типом объекта и именем экземпляра, созданного для объекта этого типа на арене. Такая строка начинается с двух символов косой черты за которыми следует имя типа и имя экземпляра. Например, в следующих строках

```
// SimpleButton startLesson;
```

```
// MovieClip circle;
```

```
// TextField message;
```

идентифицируются экземпляры кнопки с именем `startLesson`, клипа `circle` и текстового поля `message`.

В упражнениях из этой главы мы пользуемся вторым способом, так как он считается предпочтительным в AS 3.0.

### Предложение if

Часто требуется, чтобы при одних условиях в ролике происходило одно, а при других — нечто другое. Выполнение сценария должно идти по разным путям в зависимости от сложившейся ситуации.

Один из способов решить эту задачу — воспользоваться *предложением if*, которое имеет следующий синтаксис:

```
if( condition )
```

```
{
    action();
}
```

Вот реальный пример:

```
if( numLives == 0 )
{
    gotoAndStop( «gameOver» );
}
```

Предложение if состоит из следующих частей: ключевое слово if, открывающая круглая скобка, условие, которое может быть истинным или ложным, закрывающая круглая скобка и список предложений, заключенный в фигурные скобки. Предложения выполняются только тогда, когда условие истинно. Условие может быть сравнением. Например, можно сравнить, равны ли два элемента, или выяснить, какой элемент больше. Для этого используется оператор сравнения. В языке ActionScript для проверки на равенство служит оператор ==, а одиночный знак равенства применяется для присваивания значений (например, чтобы задать ширину прямоугольника равной 1 дюйму).

Чаще всего употребляются следующие операторы сравнения:

```
> больше
>= больше или равно
< меньше
<= меньше или равно
!= не равно.
```

Иногда необходимо выполнить один блок предложений, когда некоторое условие истинно, и другой — когда оно ложно. Можно написать для этого два отдельных предложения if или воспользоваться предложением if-else.

Синтаксис предложения if-else следующий:

```
if( condition )
{
    // код
}
else
{
    // другой код
}
```

Можно также сцеплять несколько предложений в конструкцию if-else-if-else:

```
if( foodInFridge )    // есть еда в холодильнике
{
    // поесть дома
}
else if( haveMoney ) // есть деньги
{
    // сходить в ресторан
}
else
{
    // позвонить другу
}
```

### Функции

Функция позволяет сгруппировать несколько действий, присвоив им общее имя. Встречая в сценарии имя функции, Flash выполняет всю группу действий. Базовый синтаксис выглядит следующим образом:

```
function functionName():Type  
{  
    // код функции  
}
```

Блок кода начинается с ключевого слова `function`. За ним следует имя функции (в данном случае `functionName`). Имя придумываете вы сами, старайтесь выбирать осмысленное. Сразу вслед за именем идет пара круглых скобок.

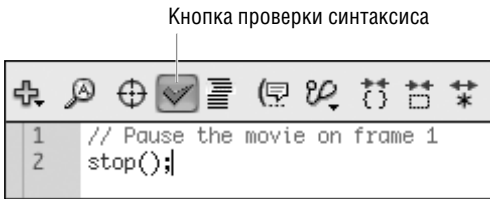
Далее следует двоеточие, отделяющее имя функции от ее типа. В данном примере тип называется `Type`. Он определяет, что возвращает функция. Чтобы понять, для чего нужен тип, рассмотрим калькулятор чаевых; в качестве результата он должен вернуть число (а не, скажем, букву). Поэтому для такой функции типом будет `Number`. Но чаще всего в качестве типа указывается ключевое слово `void`, означающее отсутствие возвращаемого значения. Так, функции, управляющие воспроизведением ролика, имеют тип `void`.

Собственно код функции заключен в фигурные скобки. Этот код выполняется при наступлении определенных условий; в нашем примере — после нажатия кнопки расчета чаевых.

Часто функции для работы необходима дополнительная информация, она передается в помощью параметров. Функция расчета чаевых должна знать стоимость обеда. При наличии параметров синтаксис функции несколько меняется:

```
function functionName( param:Type, nextParam:Type ):Type  
{  
    // код  
}
```

Если у функции несколько параметров, то их имена записываются внутри круглых скобок и разделяются запятыми. У каждого параметра имеется тип, как и у самой функции.



**Рис. 13.19.** Нажмите кнопку проверки синтаксиса на панели действий, чтобы Flash проверил код на предмет наличия синтаксических ошибок.

## Тестирование действий

Чтобы посмотреть, как работает ролик, необходимо воспроизвести его во Flash-плеере. Для этого можно опубликовать ролик (см. главу 17) или воспользоваться одним из тестовых режимов. Тестовый режим — это урезанный вариант публикации, без выхода из среды разработки. Чтобы протестировать сценарий, необходимо выполнить три шага: (1) сохранить документ; (2) проверить правильность синтаксиса и (3) запустить ролик в тестовом режиме.

### Чтобы проверить правильность синтаксиса:

1. Открыв документ, выберите опорный кадр, содержащий сценарий, который нужно проверить. Его текст появится в области сценария на панели действий.
2. Нажмите кнопку проверки синтаксиса (галочка) на панели инструментов над областью сценария или воспользуйтесь комбинацией клавиш **Option-T** (Mac) или **Ctrl-T** (Windows) (рис. 13.19).

Flash запускает компилятор (см. врезку «Что такое компилятор?» ниже) для проверки отсутствия синтаксических ошибок в выбранном коде (см. врезку «Синтаксис языка ActionScript» выше). Результаты проверки выводятся в диалоговое окно, имеющее кнопку **ОК**. Если были обнаружены ошибки, то открывается также панель ошибок компилятора.

3. Если код не содержит ошибок, нажмите **ОК** и закройте диалоговое окно.

или

Если код содержит ошибки, все равно закройте окно, нажав **ОК**, и выполните одно из следующих действий:

- просмотрите код в области сценария и исправьте те ошибки, которые сумеете найти;
- на панели ошибок компилятора выберите какую-нибудь ошибку и нажмите кнопку **Go to Source** (Перейти к источнику) (рис. 13.20). Flash открывает панель действий и подсвечивает строку, в которой обнаружена ошибка. Исправьте ее и повторите шаг 2.

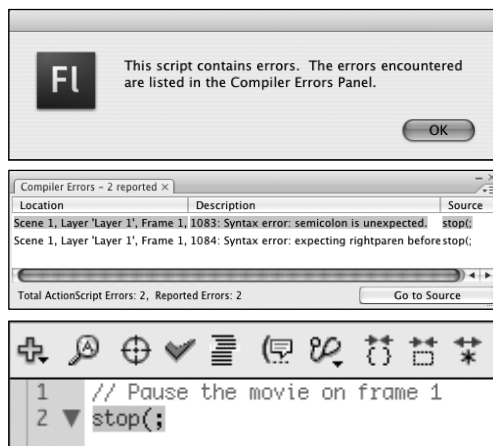
Обычно Flash отказывается выполнять сценарии с ошибками; прежде чем двигаться дальше, ошибки необходимо исправить.

**С**

Часто ошибки связаны с пропущенными, поставленными не в том месте или двойными запятыми, двоеточиями, скобками и точками с запятой. Парные элементы, например круглые и фигурные скобки, должны быть сбалансированы и правильно вложены (см. врезку «Синтаксис языка ActionScript» выше).

**С**

Если сценарий не содержит синтаксических ошибок, это еще не означает, что он будет делать то, что вам нужно. Компилятор не проверяет логику программы и не знает, чего вы хотели добиться.



**Рис. 13.20.** Если обнаружены синтаксические ошибки, об этом сообщается в диалоговом окне, а, кроме того, открывается панель ошибок компилятора, на которой приведены подробные сведения о каждой ошибке. Выбрав любую ошибку и нажав кнопку **Go to Source**, вы попадете на ту строку в области сценария, которая содержит эту ошибку.

### Что такое компилятор?

ActionScript — это язык программирования, на котором записываются инструкции для программы Flash Player. На его изучение требуется время, но все же он предназначен для чтения людьми. Процессор же работает только с двоичными данными и интерпретировать программу на языке ActionScript не в состоянии. Компилятор транслирует код, написанный на языке программирования, в команды, понятные процессору. В случае Flash сценарий транслируется в компактный байт-код, который понимает Flash Player.

Control	
Play	Enter
Rewind	Ctrl+Alt+R
Go To End	
Step Forward One Frame	.
Step Backward One Frame	,
Test Movie	Ctrl+Enter
Test Scene	Ctrl+Alt+Enter
Test Project	Ctrl+Alt+P

**Рис. 13.21.** Для проверки запрограммированной в ActionScript-сценарии анимации и интерактивности в полном объеме необходимо экспортировать ролик, например, выполнив команду Control ⇒ Test Movie

### Чтобы протестировать сценарии:

- Выполните команду **Control ⇒ Test Movie** (Управление ⇒ Тестировать ролик) или **Control ⇒ Test Scene** (Управление ⇒ Тестировать сцену) (рис. 13.21).

Flash экспортирует ролик или сцену в SWF-файл, применяя текущие параметры публикации. (Дополнительную информацию о параметрах публикации см. в главе 17.) Пока происходит экспорт, на экране присутствует диалоговое окно Exporting Flash Movie, содержащее индикатор выполнения и кнопку для отмены операции.

По завершении экспорта Flash открывает SWF-файла в Flash-плеере. Все кнопки и символы-клипы в тестовом окне работоспособны, и вы можете посмотреть, как они реагируют на манипуляции мышью. Все написанные вами сценарии будут выполняться.

Закончив тестирование, выйдите из плеера, нажав кнопку закрытия окна. Происходит возврат в среду редактирования документа.

**C**

Иногда для сценария, не содержащего синтаксических ошибок, возникают ошибки во время выполнения. Столкнувшись с такой ошибкой, Flash открывает панель ошибок компилятора. Ошибка времени выполнения может произойти, например, при обращении к несуществующему объекту (скажем, вместо enterBtn вы ввели enerBtn).

**C**

Если вы выполняли команду **Control ⇒ Test Scene**, то Flash включает в имя SWF-файла еще и имя сцены. В таком случае длина имени файла может превысить допустимый предел. Если команда **Test Movie** работает нормально, а **Test Scene** выдает предупреждение, попробуйте укоротить имя сцены.

**C**

Для тестирования интерактивности символа, например клипа, выполняйте команду **Control ⇒ Test Scene**, не выходя из режима редактирования символа. Тогда Flash опубликует только этот символ и вложенные в него элементы.

## Усовершенствование сценариев для работы с кнопками

При написании сценария трудны самые первые шаги, потом вы возводите новые этажи на уже готовом фундаменте. В предыдущих разделах был написан сценарий, управляющий поведением кнопки на арене. Сейчас мы на его основе реализуем другие виды интерактивности.

### Чтобы создать несколько действий в одном обработчике событий:

1. Откройте документ по шаблону Button-ActionsMaster, созданному в предыдущем разделе. В этом шаблоне пять кадров, в каждом из которых находится идентифицирующий текст, и один экземпляр кнопки. Также с опорным кадром 1 связано действие stop и сценарий, который активирует кнопку.

2. Выберите на временной шкале опорный кадр 1 в слое Actions.

Flash отображает текст сценария на панели действий. Если эта панель не видна, откройте ее, выполнив команду **Window** ⇨ **Actions** (Окно ⇨ Действия).

3. Чтобы заставить Flash напечатать сообщение в окне вывода, поставьте курсор в конце строки 11, нажмите **Enter** и в строке 12 введите такой текст:

```
trace( «click! new frame: « +
⇨ currentFrame );
```

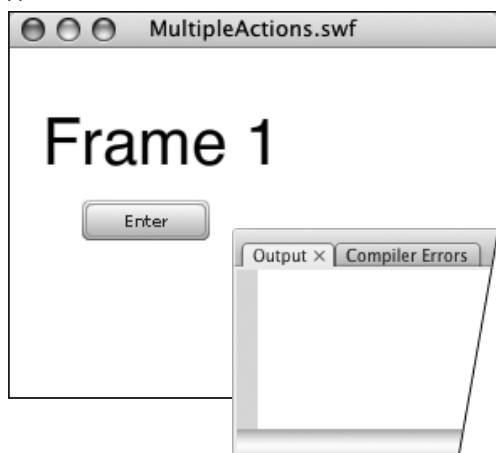
Этот код должен находиться перед закрывающей фигурной скобкой в строке 13. Строки 12 и 13 сейчас должны выглядеть, как показано в листинге 13.6.

**Листинг 13.6.** Добавление действия trace в обработчик события.

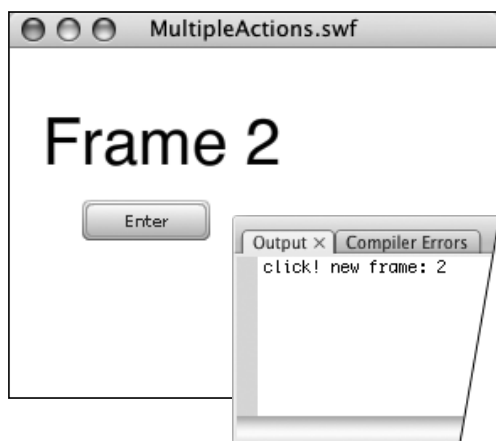
```
6 function handleClick( pEvent:MouseEvent
   ):void
7 {
8     if( pEvent.target == enterBtn )
9     {
10         // handle the event
11         nextFrame();
12         trace( «click! new frame: « +
               currentFrame );
13     }
14 }
```



До нажатия кнопки



После нажатия кнопки



**Рис. 13.22.** При тестировании ролика в окне вывода появляются трассировочные сообщения. В данном случае сообщение печатается при каждом нажатии кнопки.

Теперь обработчик события содержит два предложения. Предложение `nextFrame` перемещает точку воспроизведения к следующему кадру. А предложение `trace` заставляет Flash напечатать сообщение в окне вывода. (Текст сообщения состоит из слов, заключенных в кавычки, плюс номер кадра, который воспроизводится в данный момент.)

4. Сохраните документ и дайте ему осмысленное имя, например, `EventHandlerMultipleActions`.
5. Проверьте синтаксис сценария. Подробнее об этом см. раздел «Тестирование действий» выше.
6. Выполните команду **Control** ⇒ **Test Movie**.

Ролик открывается к окну Flash-плеера. При каждом нажатии на кнопку на панели вывода появляется фраза *Click! new frame*, за которой следует номер текущего кадра (рис. 13.22). Если панель вывода была закрыта, она откроется автоматически.

#### Чтобы установить точку воспроизведения на кадр с заданным номером:

1. Чтобы взять за основу сценарий, созданный в предыдущем упражнении, создайте копию документа `EventHandlerMultipleActions`, выполнив команду **File** ⇒ **Save As**. Назовите ее `GoToFrame`.
2. Выберите на временной шкале опорный кадр 1 в слое `Actions`.

На панели действий отображается текст ранее созданного сценария.

3. Выберите в области сценария следующий код в строке 11:

```
nextFrame();
```

и замените его таким:

```
gotoAndStop( 5 );
```

Сохраните отступы. Действие `gotoAndStop` перемещает точку воспроизведения не на один кадр, а сразу к кадру, номер которого указан в скобках.

4. Протестируйте действие.

Сохраните файл, проверьте синтаксис (строка 11 должна быть написана так, как в листинге 13.7), исправьте ошибки и запустите ролик в тестовом режиме (см. раздел «Тестирование действий»). Ролик открывается в окне Flash-плеера. При нажатии кнопки происходит переход к кадру 5, а в окне вывода печатается подтверждающее этот факт сообщение.

### Чтобы создать метку кадра:

1. Создайте копию документа из предыдущего упражнения (`GoToFrame`), выполнив команду **File** ⇒ **Save As**. Назовите ее `CreateFrameLabel`.
2. На панели временной шкалы создайте слой над слоем `Actions`, назовите его `Labels` и заблокируйте.  
Располагать слой `Labels` наверху удобно, так как вы можете использовать метки для обозначения различных участков ролика.
3. Вставьте в позицию 3 в слое `Labels` пустой опорный кадр, например, выполнив команду **Insert** ⇒ **Timeline** ⇒ **Blank Keyframe**.
4. Выбрав пустой кадр, перейдите в инспектор свойств и введите в поле `Frame Label` (Метка кадра) значение `myLabel`.

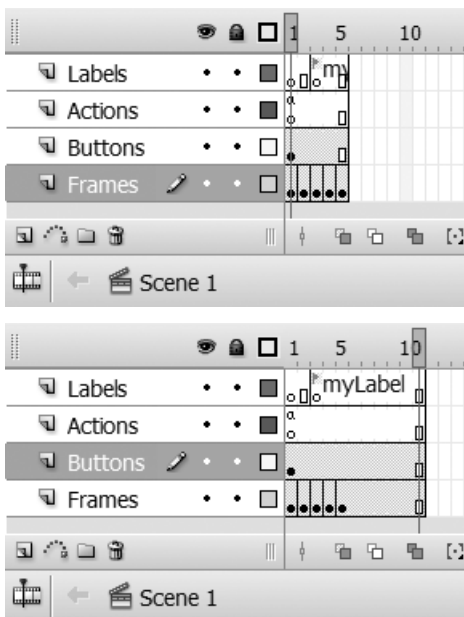
**Листинг 13.7.** Перемещение точки воспроизведения на кадр с указанным номером

```

6  function handleClick( pEvent:MouseEvent
   ):void
7  {
8      if( pEvent.target == enterBtn )
9      {
10         // handle the event
11         gotoAndStop( 5 );
12         trace( "click! new frame: " +
               currentFrame );
13     }
14 }
```

### Ссылка на кадр по имени

В качестве параметра действий `gotoAndStop` и `gotoAndPlay` можно задавать как номер, так и метку кадра. Лучше пользоваться метками, поскольку при добавлении или удалении кадров из кнопки-клипа (вы научились это делать в главе 12), номера кадров, описывающих состояние кнопки, могут измениться. В таком случае придется вносить изменения в сценарий. Если же вы ссылаетесь на кадр по метке, то модифицировать сценарий из-за изменения нумерации кадров не потребуется.



**Рис. 13.23.** Метка кадра отображается частично, если последовательность опорного кадра слишком коротка, чтобы вместить полный текст (верхний рисунок). Чтобы преимущества меток как средства визуальной организации шкалы проявились в полной мере, расширьте последовательность, добавив промежуточные кадры (нижний рисунок).

**Листинг 13.8.** Установка точки воспроизведения на помеченном кадре

```

6 function handleClick( pEvent:MouseEvent
7 );void
8 {
9     if( pEvent.target == enterBtn )
10    {
11        // handle the event
12        gotoAndStop( "myLabel" );
13        trace( "click! new frame: " +
14            currentFrame );
15    }
16 }

```

**С**

В предыдущем упражнении вы перешли к указанному кадру и остановили воспроизведение. Можно также продолжить воспроизведение ролика с указанного кадра. Для этого на шаге 2 вместо действия `gotoAndStop` укажите `gotoAndPlay`.

Чтобы зафиксировать имя метки, нажмите **Enter** или щелкните мышью вне поля. В опорном кадре 3 в слое **Labels** на временной шкале появится красный флажок. Flash отображает часть имени метки, которая помещается в последовательности помеченного опорного кадра (рис. 13.23).

5. Поместите точку воспроизведения в кадр 5, но не выбирайте его.
6. Вставьте столько промежуточных кадров (например, нажимая клавишу **F5**), сколько необходимо, чтобы имя метки было видно целиком.

Метки кадра помогают лучше организовать временную шкалу. Их применение особенно эффективно, если имена видны прямо на шкале и не надо каждый раз справляться с инспектором свойств.

### Чтобы установить точку воспроизведения на помеченный кадр:

1. Продолжая работу с файлом из предыдущего упражнения, выберите опорный кадр 1 в слое **Actions** и перейдите на панель действий. В области сценария появится текущий текст программы.
2. Замените текст в строке 11 `gotoAndStop( 5 );` таким:  
`gotoAndStop( «myLabel» );`  
 Сохраните отступы. Действие `gotoAndStop` перемещает точку воспроизведения на кадр с указанным номером или меткой.
3. Протестируйте действие.  
 Сохраните файл, проверьте синтаксис (строка 11 должна быть написана так, как в листинге 13.8), исправьте ошибки и запустите ролик в тестовом режиме (см. раздел «Тестирование действий»). Ролик открывается в окне Flash-плеера. При нажатии кнопки происходит переход к кадру с меткой `myLabel`.

## Выбор событий

Когда пользователь взаимодействует с кнопкой, движения мышью приводят к возникновению различных событий. Если отпустить кнопку мыши в активной области кнопки, то произойдет событие `click`. При входе в активную область и выходе из нее происходят другие события. Если вы хотите, чтобы кнопка реагировала на эти события, нужно зарегистрировать их в сценарии вместо `click`.

### Чтобы обрабатывать событие наведения курсора на кнопку:

1. Откройте новый документ по шаблону `ButtonActionsMaster`, созданному в этой главе ранее. Сохраните этот документ, назвав его `ButtonEvents.fla`.
2. Выберите кадр 1 в слое `Actions` и перейдите на панель действий. В области сценария появится текст программы.
3. В строке замените текст `MouseEvent.CLICK` на `MouseEvent.ROLL_OVER`.
4. Протестируйте действие.

Сохраните файл, проверьте синтаксис (строки 14–15 должны быть написаны так, как в листинге 13.9), исправьте ошибки и запустите ролик в тестовом режиме.

Ролик открывается в окне Flash-плеера. При наведении курсора на кнопку (когда курсор входит в активную область) точка воспроизведения сдвигается вперед на один кадр. При щелчке не происходит ничего, разве что кнопка перерисовывается в нажатом состоянии.

**Листинг 13.9.** Замена в обработке события `click` на событие `rollover`

```
14 // register events
15 enterBtn.addEventListener(
    MouseEvent.ROLL_OVER, handleClick );
```

**Листинг 13.10.** Регистрация события `rollout`

```
14 // register events
15 enterBtn.addEventListener(
    MouseEvent.ROLL_OUT, handleClick );
```

**C**

Заменить событие можно и быстрее. Сотрите текст `.CLICK` и введите точку, чтобы появилось окно подсказки. Или удалите только слово `CLICK`, поместите курсор справа от точки и нажмите кнопку **Code Hint** на панели инструментов над областью сценария.

**C**

В предыдущем упражнении обработчик событий по-прежнему называется `handleClick`. Было бы правильнее переименовать его в `handleRollOver`, только не забудьте сделать это в обеих строках 6 и 15.

**C**

Если вы хотите, чтобы сценарий реагировал на выход из кнопки, а не на вход, то в строке 15 замените текст `MouseEvent.ROLL_OVER` на `MouseEvent.ROLL_OUT` (листинг 13.10).

**Листинг 13.11.** Добавление второго обработчика события.

```
11 nextFrame();
12 }
13 }
14 function handleRollOver(
   pEvent:MouseEvent ):void
15 {
16     if( pEvent.target == enterBtn )
17     {
18         // handle the event
19         gotoAndStop( 2 );
20     }
21 }
22 // register events
```

### Чтобы получать извещения о нескольких событиях кнопки:

1. Откройте новый документ по шаблону ButtonActionsMaster, созданному в этой главе ранее. Сохраните этот документ, назвав его ButtonMultiEvents fla.
2. Выберите кадр 1 в слое Actions и перейдите на панель действий. В области сценария появится текст программы.
3. Для определения второго обработчика события выполните следующие действия:

- поместите курсор в конец строки 13 (после закрывающей скобки) и нажмите **Enter**, чтобы открыть новый блок (код, который до этого находился в строках 14 и 15, съезжает вниз);
- в строках 14–21 введите такой текст:

```
function handleRollOver(
    ⇨ pEvent:MouseEvent ):void
{
    if( pEvent.target == enterBtn )
    {
        // handle the event
        gotoAndStop( 2 );
    }
}
```

Строки 14–21 должны быть такими, как в листинге 13.11.

4. Чтобы зарегистрироваться для получения события `rollOver`, поставьте курсор в конец строки 23, нажмите **Enter** и в новой строке 24 введите такой текст:

```
enterBtn.addEventListener(  
    ⇨ MouseEvent.ROLL_OVER,  
    ⇨ handleRollOver );
```

Чтобы получать разные события от одной кнопки, необходимо на каждое событие регистрироваться отдельно. Сценарий должен получать извещения об обоих событиях, но реагировать на них по-разному. Для этого необходимо определить еще одну функцию-обработчик (листинг 13.12).

5. Протестируйте действие.

Сохраните файл, проверьте синтаксис (полный текст приведен в листинге 13.13), исправьте ошибки и запустите ролик в тестовом режиме.

Ролик открывается в окне Flash-плеера. При наведении курсора на кнопку точка воспроизведения переходит на кадр 2, а при нажатии — на кадр 3.

**Листинг 13.12.** Обратите внимание на второе событие (`rollover`)

```
22 // register events  
23 enterBtn.addEventListener(  
    MouseEvent.CLICK, handleClick );  
24 enterBtn.addEventListener(  
    MouseEvent.ROLL_OVER, handleRollOver );
```

**Листинг 13.13.** Окончательный сценарий для обработки нескольких событий.

```
1 //Pause the movie on frame 1  
2 stop();  
3 // code-hint comments  
4 // SimpleButton enterBtn;  
5 // event handlers  
6 function handleClick( pEvent:MouseEvent  
    ):void  
7 {  
8     if( pEvent.target == enterBtn )  
9     {  
10         // handle the event  
11         nextFrame();  
12     }  
13 }  
14 function handleRollOver( pEvent:MouseEvent  
    ):void  
15 {  
16     if( pEvent.target == enterBtn )  
17     {  
18         // handle the event  
19         gotoAndStop( 2 );  
20     }  
21 }  
22 // register events  
23 enterBtn.addEventListener(  
    MouseEvent.CLICK, handleClick );  
24 enterBtn.addEventListener(  
    MouseEvent.ROLL_OVER, handleRollOver );
```

### События мыши

Символы-кнопки и символы-клипы относятся к так называемым *интерактивным объектам* Flash. При взаимодействии с такими объектами посредством любого графического устройства (например, мыши или пера) происходят одни и те же события, они называются *событиями мыши*. (Компонент-кнопка, несмотря на наличие слова *кнопка* в названии, на самом деле является частным случаем клипа.)

Всего для интерактивных объектов определено 16 встроенных событий, из них 9 — это события мыши. Поскольку они встроены, их функциональность предопределена, и вы не можете изменить ни названия, ни поведение. Ниже приведены общепринятые имена событий (например, `click`) и в скобках константа `ActionScript`, описывающая это событие (например, `MouseEvent.CLICK`).

`click` (`MouseEvent.CLICK`) возникает, когда пользователь нажимает и отпускает кнопку мыши, находясь в активной области кнопки или клипа (интерактивного объекта).

`rollOver` (`MouseEvent.ROLL_OVER`) возникает, когда курсор входит в активную область интерактивного объекта. После того как событие `rollOver` произошло, оно не повторится раньше, чем произойдет событие `rollOut`.

`rollOut` (`MouseEvent.ROLL_OUT`) возникает, когда курсор покидает активную область интерактивного объекта.

`mouseMove` (`MouseEvent.MOUSE_MOVE`) возникает, когда курсор перемещается внутри активной области интерактивного объекта.

`doubleClick` (`MouseEvent.DOUBLE_CLICK`) возникает, когда пользователь дважды быстро щелкает кнопкой мыши в активной области интерактивного объекта. Это событие появилось только в версии `ActionScript 3.0`. Чтобы воспользоваться им в сценарии, необходимо установить в `true` свойство `doubleClickEnabled` кнопки или клипа (для этого нужно включить в сценарий код `myBtn.doubleClickEnabled = true;`).

Чаще всего употребляются события `click`, `rollOver` и `rollOut`, и в большинстве ваших сценариев ничего другого и не потребуется. О событиях `mouseMove` и `doubleClick` упомянуто лишь для того, чтобы вы знали, какие еще имеются возможности.

## Компоненты-кнопки

В составе Flash имеется ряд готовых элементов пользовательского интерфейса, которые находятся на панели компонентов, например, `RadioButton` и `ComboBox` (раскрывающийся список). Сценарии для описания взаимодействия с компонентами-кнопками и обычными кнопками очень похожи.

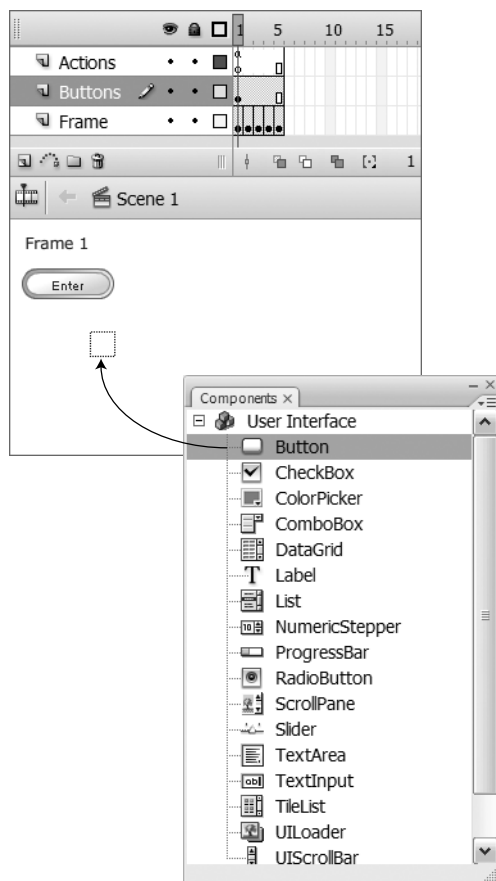
### Чтобы добавить в ролик компонент-кнопку:

1. Откройте новый документ по шаблону `ButtonActionsMaster`, созданному в этой главе ранее. Сохраните этот документ, назвав его `ButtonComponent.fla`.
2. Выберите опорный кадр 1 в слое `Buttons`.
3. Перейдите на панель компонентов и выберите компонент `Button` из категории `User Interface`.

Если эта панель еще не открыта, выполните команду **Window** ⇒ **Components** (Окно ⇒ Компоненты).

4. Перетащите экземпляр компонента-кнопки на арену (рис. 13.24).
5. Выберите экземпляр компонента-кнопки на арене и перейдите на вкладку **Properties** инспектора свойств.
6. Введите имя экземпляра `myComponent` в поле `Instance Name` (рис. 13.25).

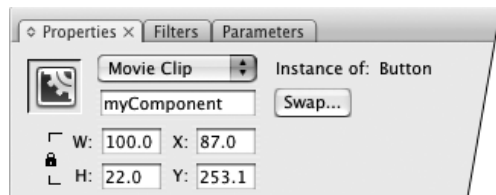
Теперь все готово к созданию сценария для программирования этого компонента.



**Рис. 13.24.** Перетащите экземпляр компонента-кнопки с панели компонентов на арену, чтобы включить его в ролик.

**C**

Внешний вид компонента можно изменить, модифицировав его тему (детали см. в главе 12).



**Рис. 13.25.** Чтобы на экземпляр компонента можно было сослаться из сценария, ему нужно присвоить имя. Обратите внимание, что в списке `Behavior` (Поведение экземпляра) в инспекторе свойств видно, что компонент `Button` на самом деле является фрагментом ролика (клипом).



**Листинг 13.14.** Добавление комментария, описывающего компонент-кнопку для механизма подсказки.

```
3 // code-hint comments
4 // SimpleButton enterBtn;
5 // Button myComponent;
6 // event handlers
```

**Листинг 13.15.** Обработчик события от компонента-кнопки.

```
12 nextFrame();
13 }
14 }
15 function handleComponent(
  pEvent:MouseEvent ):void
16 {
17   if( pEvent.target == myComponent )
18   {
19     // handle the event
20     enterBtn.visible = false;
21   }
22 }
```

### Чтобы запрограммировать интерактивность компонента-кнопки:

1. Продолжая работу с файлом из предыдущего упражнения, выберите опорный кадр 1 в слое Actions и перейдите на панель действий. В области сценария появится текущий текст программы.

2. Чтобы включить подсказку для экземпляра компонента, поставьте курсор в конец строки 4, нажмите **Enter** и введите в строке 5 текст:

```
// Button myComponent;
```

Компонент-кнопка сложнее обычной кнопки, поэтому включение подсказки упростит его программирование (листинг 13.14).

3. Чтобы создать обработчик события для компонента, выполните следующие действия:

- поставьте курсор в конец строки 12 (после закрывающей скобки) и нажмите **Enter**, чтобы открыть новый блок кода;

- в строках 15-22 введите такой текст:

```
function handleComponent(
  ⇨ pEvent:MouseEvent ):void
{
  if( pEvent.target ==
    ⇨ myComponent )
  {
    // handle the event
    enterBtn.visible = false;
  }
}
```

Код должен быть написан так, как показано в листинге 13.15. Этот обработчик не перемещает точку воспроизведения к другому кадру, а скрывает кнопку enterBtn.

4. Поставив курсор в конец строки 24 (после точки с запятой), нажмите **Enter** и в строке 25 введите такой текст:

```
myComponent.addEventListener(
```

```
    ⇨ MouseEvent.CLICK,
```

```
    ⇨ handleComponent );
```

Код должен выглядеть так, как показано в листинге 13.16.

5. Протестируйте действие.

Сохраните файл, проверьте синтаксис (сценарий должен быть написан так, как в листинге 13.17), исправьте ошибки и запустите ролик в тестовом режиме.

Ролик открывается в окне Flash-плеера. При нажатии на компонент-кнопку с именем Label символ-кнопка с именем Enter становится невидимым. Нажатие на символ-кнопку, как и раньше, сдвигает точку воспроизведения вперед на один кадр.

**Листинг 13.16.** Добавление обработчика события click от компонента-кнопки.

```
23 // register events
24 enterBtn.addEventListener(
    MouseEvent.CLICK, handleClick );
25 myComponent.addEventListener(
    MouseEvent.CLICK, handleComponent );
```

**Листинг 13.17.** Полный текст сценария для компонента-кнопки.

```
1 //Pause the movie on frame 1
2 stop();
3 // code-hint comments
4 // SimpleButton enterBtn;
5 // Button myComponent;
6 // event handlers
7 function handleClick( pEvent:MouseEvent
8 ):void
9 {
10     if( pEvent.target == enterBtn )
11     {
12         // handle the event
13         nextFrame();
14     }
15 }
16 function handleComponent(
17 pEvent:MouseEvent ):void
18 {
19     if( pEvent.target == myComponent )
20     {
21         // handle the event
22         enterBtn.visible = false;
23     }
24 }
25 // register events
26 enterBtn.addEventListener(
27     MouseEvent.CLICK, handleClick );
28 myComponent.addEventListener(
29     MouseEvent.CLICK, handleComponent );
```

## Использование одного обработчика для нескольких событий

Ранее в этой главе вы создавали отдельные обработчики для каждого события. Но можно обрабатывать несколько событий в одной функции с параметром. При таком подходе сценарий должен проверять передаваемый параметр `rEvent` и по-разному реагировать на события от разных кнопок.

### Чтобы добавить еще одну программируемую кнопку:

1. Откройте новый документ по шаблону `ButtonActionsMaster`, созданному в этой главе ранее. В этом документе на арене есть всего одна кнопка с именем `enterBtn`. Сохраните документ, назвав его `MultiEventsOneHandler.fla`.
2. Выберите экземпляр `enterBtn` и выполните команду **Edit** ⇒ **Duplicate** (Редактирование ⇒ Дублировать), чтобы создать его копию. Расположите копию так, чтобы она не перекрывала исходный экземпляр.
3. Перейдите на вкладку **Properties** в инспекторе свойств и введите в поле `Instance Name` имя `prevBtn`. Теперь на арене есть две кнопки, которые мы запрограммируем в одном обработчике событий.

### Чтобы запрограммировать вторую кнопку:

1. Продолжая работу с файлом из предыдущего упражнения, выберите опорный кадр 1 в слое `Actions` и перейдите на панель действий.

- Чтобы включить подсказку для второй кнопки (`prevBtn`), поставьте курсор в конец строки 4, нажмите **Enter** и введите в строке 5 текст:

```
// SimpleButton prevBtn;
```

(см. листинг 13.18).

- Поставьте курсор в конец строки 13 после закрывающей скобки и нажмите **Enter**. Проверьте, что курсор находится после нужной закрывающей скобки. Скобка в строке 13 закрывает предложение `if`, а скобка в строке 14 закрывает всю функцию.

- В строках 14–17 введите такой текст:

```
else if( pEvent.target == prevBtn )
{
    // handle the prevBtn event
    prevFrame();
}
```

Здесь мы продолжили предложение `if`, так чтобы при невыполнении первого условия проверялось второе. Нас интересует объект `prevBtn`. Если событие пришло от него, то точка воспроизведения сдвигается к предыдущему кадру (листинг 13.19).

- Чтобы зарегистрироваться для получения события `click` от кнопки `prevBtn`, поставьте курсор в конец строки 21, нажмите **Enter** и в строке 22 введите такой текст:

```
prevBtn.addEventListener(
    ⇨ MouseEvent.CLICK, handleClick );
```

- Протестируйте действие.

Сохраните файл, проверьте синтаксис (строки 3–22 должны быть написаны так, как в листинге 13.20), исправьте ошибки и запустите ролик в тестовом режиме. Ролик открывается в окне Flash-плеера. Нажатие на кнопку `enterBtn` сдвигает точку воспроизведения вперед на один кадр, а нажатие на кнопку `prevBtn` — назад на один кадр.

**Листинг 13.18.** Добавление комментария, описывающего вторую кнопку для механизма подсказки.

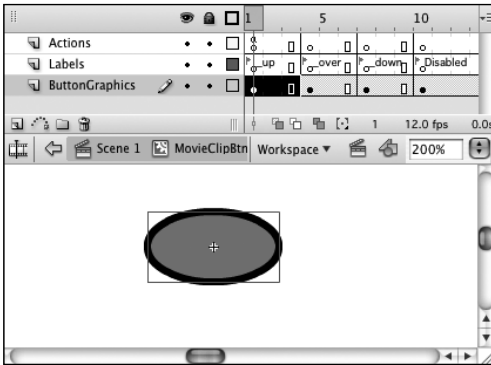
```
3 // code-hint comments
4 // SimpleButton enterBtn;
5 // SimpleButton prevBtn;
6 // event handlers
```

**Листинг 13.19.** Взаимодействие со второй кнопкой.

```
12         nextFrame();
13     }
14     else if( pEvent.target == prevBtn )
15     {
16         // handle the prevBtn event
17         prevFrame();
18     }
19 }
```

**Листинг 13.20.** Один обработчик событий от двух кнопок.

```
3 // code-hint comments
4 // SimpleButton enterBtn;
5 // SimpleButton prevBtn;
6 // event handlers
7 function handleClick( pEvent:MouseEvent
8 ):void
9 {
10     if( pEvent.target == enterBtn )
11     {
12         // handle the event
13         nextFrame();
14     }
15     else if( pEvent.target == prevBtn )
16     {
17         // handle the prevBtn event
18         prevFrame();
19     }
20     // register events
21     enterBtn.addEventListener(
22         MouseEvent.CLICK, handleClick );
23     prevBtn.addEventListener(
24         MouseEvent.CLICK, handleClick );
```



**Рис. 13.26.** У этого символа-клипа есть опорные кадры, описывающие четыре состояния кнопки: три стандартных (отпущена, под курсором и нажата) и дополнительное «не активна». Каждому кадру соответствует понятная метка.

## Программирование клипов как кнопку

Символы-кнопки во Flash поддерживают базовые требования, предъявляемые к кнопке: разный внешний вид в состояниях «отпущена», «под курсором» и «нажата». Но иногда бывают нужны и другие состояния. Например, в форме для отправки электронной почты кнопка «Отправить» должна быть деактивирована (и выглядеть соответственно) до тех пор, пока не будут заполнены все обязательные поля.

В главе 12 вы научились настраивать символ-клип для использования в качестве кнопки, имеющей состояние «не активна». В следующем упражнении мы напишем программу, которая заставит такой символ работать, как кнопка.

### Чтобы наделить кнопку-клип поведением обычной кнопки:

1. Откройте файл `MyOwnBtn.fla`, созданный в разделе «Создание кнопок-клипов» главы 12. Этот документ содержит символ-клип `MoveClipBtn`.
2. Дважды щелкните по экземпляру символа `MoveClipBtn` на арене, чтобы войти в режим редактирования символа.

На временной шкале для этого символа имеется три слоя: `Actions`, `Labels` и `ButtonGraphics`. Для него определены три опорных кадра с именами `_up`, `_over`, `_down` и графические изображения, соответствующие всем четырем состояниям (рис. 13.26).

3. Выберите на временной шкале символа опорный кадр 1 в слое `Actions`.

4. Чтобы остановить воспроизведение клипа, введите в строке 1 в области сценария на панели действий текст

```
stop();
```

По умолчанию клипы во время выполнения воспроизводятся. Для анимации это имеет смысл, но кнопка должна оставаться в состоянии «отпущена», пока пользователь не начнет с ней взаимодействовать.

5. Поставив курсор в конец строки 1, нажмите **Enter**, чтобы открыть новую строку.
6. Чтобы перевести клип в режим кнопки, наберите в строке 2 текст

```
buttonMode = true;
```

Если этого не сделать, то клип не будет знать о том, что должен вести себя, как кнопка, хотя вы присвоили его кадрам специальные имена.

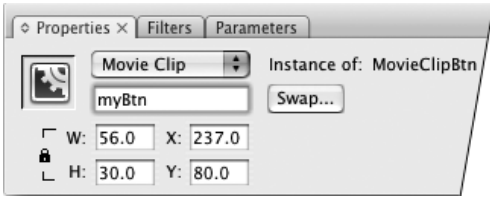
7. Сохраните документ, проверьте синтаксис (сценарий должен быть написан так, как в листинге 13.21) и выполните команду **Control ⇒ Test Movie**.

При наведении на кнопку-клип курсор принимает форму руки, а точка воспроизведения смещается к кадру `_over` клипа. При нажатии кнопки мыши точка воспроизведения смещается к кадру `_down` — в точности, как для обычной кнопки.

8. Закройте тестовое окно и вернитесь к главной временной шкале, например, выполнив команду **Edit ⇒ Edit Document**.

**Листинг 13.21.** Код, переводящий символ-клип в режим кнопки.

```
1 stop();  
2 buttonMode = true;
```



**Рис. 13.27.** Чтобы сослаться на кнопку-клип, необходимо задать ее имя в инспекторе свойств.

**Листинг 13.22.** Комментарий, содержащий описание сценария, остановка воспроизведения и комментарий для механизма подсказок.

```
1 // Scripting a movie-clip button
2 stop();
3 // code-hint comments
4 // MovieClip myBtn;
```

**Чтобы добавить обработчик события мыши, который отображает кнопку в неактивном состоянии:**

1. Продолжая работу с файлом из предыдущего упражнения, выберите экземпляр клипа `MovieClipBtn` на арене и перейдите на вкладку **Properties** в инспекторе свойств.
2. Введите имя экземпляра, например, `myBtn` (рис. 13.27).

3. На панели временной шкалы создайте новый слой над существующим и назовите его **Actions**.

Когда на временной шкале есть несколько слоев, рекомендуется присвоить имя каждому. Поэтому назовите слой, содержащий кнопку-клип, `MC Button`.

4. Выберите опорный кадр 1 в слое **Actions** и перейдите на панель действий.
5. В строке 1 в области сценария документируйте свой сценарий, для чего введите символы `//` и краткое описание (например, *Программирование кнопки-клипа*) а затем нажмите **Enter**. Комментарий напомнит вам (и другим людям), что вы хотели сделать в этом сценарии.
6. Поставьте курсор в строку 2 и наберите `stop();`, чтобы остановить воспроизведение, затем нажмите **Enter**.
7. В строках 3 и 4 оставьте комментарий для механизма подсказки:

```
// code-hint comments
```

```
// MovieClip myBtn;
```

Он сообщает Flash, что `myBtn` — экземпляр объекта типа `MovieClip` (листинг 13.22).

8. Оставив курсор в конце строки 4, нажмите **Enter**, чтобы открыть новый блок.

9. Чтобы определить обработчик события, введите в строках 5–13 такой текст:

```
// event handlers
function handleClick
⇒ ( pEvent:MouseEvent ):void
{
    if( pEvent.target == myBtn )
    {
        myBtn.enabled = false;
        myBtn.gotoAndStop( "Disabled" );
    }
}
```

В строке 6 объявлен обработчик события. В строке 8 проверяется, что событие поступило от интересующего нас экземпляра кнопки — `myBtn`. В строке 10 мы запрещаем кнопке реагировать на манипуляции мышью, устанавливая свойство `enabled` экземпляра клипа в `false`. В строке 11 точка воспроизведения перемещается на кадр `Disabled` (иначе точка воспроизведения вернулась бы на кадр `_up`, как только мышь покинула бы активную область клипа) (листинг 13.23).

10. Чтобы зарегистрироваться для получения события `click`, поставьте курсор в конец строки 13, нажмите **Enter** и в строках 14–15 введите такой текст:

```
// register events
myBtn.addEventListener(
⇒ MouseEvent.CLICK, handleClick );
```

11. Сохраните документ, проверьте синтаксис (сценарий должен быть написан так, как в листинге 13.24) и выполните команду **Control ⇒ Test Movie**.

Ролик открывается в окне Flash-плеера. Кнопка-клип отображается в состоянии `_over`. При нажатии на нее точка воспроизведения сдвигается к кадру `_down`. Как только вы отпускаете кнопку мыши, точка воспроизведения переходит на кадр `Disabled`, и кнопка-клип перестает реагировать на мышшь.

Листинг 13.23. Добавление обработчика события.

```
4 // MovieClip myBtn;
5 // event handlers
6 function handleClick( pEvent:MouseEvent
):void
7 {
8     if( pEvent.target == myBtn )
9     {
10         myBtn.enabled = false;
11         myBtn.gotoAndStop( "Disabled" );
12     }
13 }
```

Листинг 13.24. Полный текст сценария для кнопки-клипа.

```
1 // Scripting a movie-clip button
2 stop();
3 // code-hint comments
4 // MovieClip myBtn;
5 // event handlers
6 function handleClick( pEvent:MouseEvent
):void
7 {
8     if( pEvent.target == myBtn )
9     {
10         myBtn.enabled = false;
11         myBtn.gotoAndStop( "Disabled" );
12     }
13 }
14 // register events
15 myBtn.addEventListener(
MouseEvent.CLICK, handleClick );
```

## C

Процедуры программирования интерактивных кнопок и кнопок-клипов одинаковы. Если вы сначала воспользуетесь в своем проекте обычными кнопками, а потом решите, что необходима дополнительная функциональность, то начинать все с нуля не придется. Просто замените обычные кнопки кнопками-клипами с такими же именами, и программа будет работать как ни в чем ни бывало. Затем можно добавить в сценарий код для отображения дополнительных состояний клипа, что мы и сделали в предыдущем упражнении.



## Управление временной шкалой с помощью кнопок

Во многих Flash-роликах встречаются разнообразные интерфейсные объекты: кнопки-символы, кнопки-компоненты и клипы. Можно запрограммировать кнопку (или компонент) так, чтобы она начинала или останавливала воспроизведение либо выполняла переход к конкретному кадру. Самое главное тут — правильно задать путевое имя объекта.

### Путевые имена объектов

Для всех роликов Flash характерно наличие некоей структуры. Даже в простом ролике, созданном по шаблону ButtonActionsMaster, имеется иерархия вложенных объектов. (Временная шкала, которую вы видите, открывая документ Flash в среде разработки, — это на самом деле символ-клип, поэтому любые объекты на главной временной шкале оказываются вложенными.) Эта структура может стать весьма сложной, если одни интерактивные объекты вложены в другие, например: кнопка в клип, один клип в другой, кнопка в клип, который сам вложен в другой клип и т.д.

Вложенный объект называется *дочерним*, а содержащий его объект — *родительским*. Если у каждого родителя и потомка имеется имя, то сложность отношений родства не имеет значения. Из сценария можно манипулировать любыми вложенными объектами. Для идентификации места объекта в иерархии временной шкалы применяются *путевые имена*.

Путевые имена бывают *относительные* и *абсолютные*. Относительное путевое имя начинается с имени объекта и описывает положение конечной цели относительно других временных шкал в иерархии, например: «конечной целью являюсь я сам» или «конечная цель находится на один уровень выше меня» или «конечная цель находится внутри меня одним уровнем ниже». Абсолютное путевое имя начинается с временной шкалы верхнего уровня и перечисляет все шкалы на пути к конечному объекту вниз по иерархии.

Ситуация осложняется из-за того, что в языке ActionScript 3.0 появился новый тип интерактивных объектов — *спрайт*. Спрайт похож на ролик, но у него нет ни временной шкалы, ни кадров. Вся анимация и интерактивность спрайта реализуется исключительно сценарием. Спрайты тоже могут быть вложенными и образовывать иерархии. И для их идентификации тоже применяются путевые имена. При записи путевых имен нужно отличать спрайты от объектов на базе временной шкалы (клипов). Как это делается, показано в примерах ниже. (В настоящей книге мы будем иметь дело только с объектами на базе временной шкалы.)

### Чтобы заставить кнопку останавливать воспроизведение клипа:

1. Откройте новый документ типа ActionScript 3.0, добавьте два слоя (чтобы всего было три) и выполните следующие действия:
  - назовите верхний слой Actions;
  - назовите второй слой Buttons. В этот слой поместите экземпляр символа-кнопки, расположив его в левом верхнем углу арены;
  - назовите третий слой MovieClips. В этом слое поместите в центр арены экземпляр клипа, содержащего анимацию (например, простую интерполяцию геометрической формы с вращением и уменьшением);
  - сохраните файл и назовите его pauseClipControl fla.
2. Для подготовки экземпляров к использованию в сценарии выполните следующие действия:
  - перейдите в инспектор свойств;
  - выберите кнопку и назовите ее controlBtn;
  - выберите клип и назовите его animMc.
3. Выберите в слое Actions кадр 1 и перейдите на панель действий.
4. Для того чтобы описать свой сценарий и остановить воспроизведение в самом начале, введите в строках 1 и 2 следующий текст:

```
// Controlling a movie clip with a
⇒ button
stop();
```

### Путевые имена объектов (продолжение)

Чтобы начать относительное путевое имя, просто введите имя экземпляра того объекта, на который хотите сослаться, например, animMc.

Относительный путь может вести не только к объектам в текущей временной шкале. Можно подниматься и опускаться на несколько уровней по иерархии. Для этого как раз и оказываются полезны понятия родителя и потомка. Пусть есть кнопка с именем controlBtn на той же главной временной шкале, где находится и объект animMc. Тогда можно записать такой путь (parent as MovieClip).controlBtn.

Этот путь означает, что нужно подняться на один уровень вверх до родительской временной шкалы (в данном случае это главная шкала) и найти там объект с именем controlBtn. (При употреблении слова parent надо различать спрайты и клипы.)

Для ссылки на дочерние объекты применяется точка между именем родительского и дочернего объекта. Чтобы перейти на уровень вниз и обратиться к экземпляру клипа с именем starMc, который вложен в animMc, нужен такой относительный путь: animMc.starMc.

Хотя это и не обязательно, можете начинать относительный путь со слова this. Так, пути this.animMc.starMc и animMc.starMc эквивалентны.

Абсолютный путь всегда начинается с главной временной шкалы, которая называется root. Абсолютный путь к экземпляру starMc выглядит так: (root as MovieClip).animMc.starMc. (При употреблении слова root надо различать спрайты и клипы.)

**Листинг 13.25.** Прокомментированный сценарий, который останавливает временную шкалу и содержит комментарии для механизма подсказок.

```
1 // Controlling a movie clip with a button
2 stop();
3 // code-hint comments
4 // SimpleButton controlBtn;
5 // MovieClip animMc;
```

**Листинг 13.26.** Обработчик, который проверяет источник события.

```
5 // MovieClip animMc;
6 // event handler
7 function handleClick( pEvent:MouseEvent
8 ):void
9 {
10     if( pEvent.target == controlBtn )
11     {
12         // handle event
13     }
```

**Листинг 13.27.** Регистрация для получения события click от кнопки controlBtn.

```
14 // register events
15 controlBtn.addEventListener(
    MouseEvent.CLICK, handleClick );
```

5. Чтобы активировать подсказки для объектов на арене, введите в строках 5–7 следующие комментарии (листинг 13.25):

```
// code-hint comments
// SimpleButton controlBtn;
// MovieClip animMc;
```

6. Для определения обработчика, который проверяет источник события, введите в строках 6–13 такой код:

```
// event handler
function handleClick(
    ⇨ pEvent:MouseEvent ):void
{
    if( pEvent.target == controlBtn )
    {
        // handle event
    }
}
```

Этот обработчик проверяет, что событие поступило от кнопки controlBtn (листинг 13.26).

7. Чтобы зарегистрироваться для получения события click от кнопки controlBtn, введите в строках 14–15 такой код:

```
// register events
controlBtn.addEventListener(
    ⇨ MouseEvent.CLICK, handleClick );
```

Теперь главная временная шкала будет получать событие click от кнопки controlBtn, и вы сможете перехватить управление экземпляром клипа animMc (листинг 13.27).

8. Поставьте курсор в конец строки 11, нажмите **Enter** и введите текст:

```
animMc
```

Это относительное путевое имя; данный сценарий ссылается на экземпляр `animMc` из главной временной шкалы (в которой находится сам сценарий).

9. Расположив курсор в конце строки 12, сразу после путевого имени, введите:

```
.stop();
```

Это действие, присоединенное к объекту, заданному путевым именем, приостанавливает воспроизведение экземпляра клипа с именем `animMc`.

10. Протестируйте действие.

Сохраните документ, проверьте синтаксис (код должен быть написан так, как в листинге 13.28), исправьте ошибки и запустите ролик в тестовом режиме.

В окне Flash-плеера анимация продолжается без перерыва. При нажатии кнопки анимация приостанавливается.

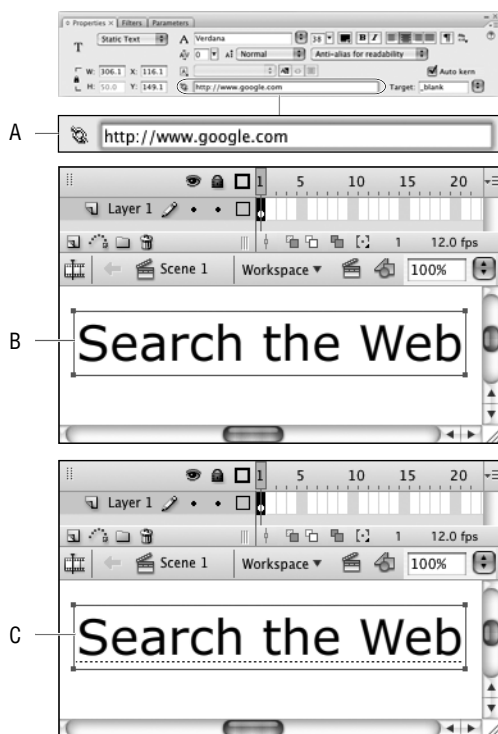
**Листинг 13.28.** Полный текст сценария для приостановки клипа при нажатии кнопки.

```
1 // Controlling a movie clip with a button
2 stop();
3 // code-hint comments
4 // SimpleButton controlBtn;
5 // MovieClip animMc;
6 // event handler
7 function handleClick( pEvent:MouseEvent
8 ):void
9 {
10     if( pEvent.target == controlBtn )
11     {
12         // handle event
13         animMc.stop();
14     }
15 }
16 // register events
17 controlBtn.addEventListener(
18     MouseEvent.CLICK, handleClick );
```

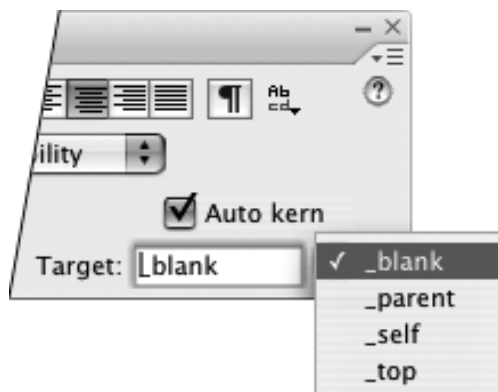
### Подводные камни, связанные с кнопкой **Insert Target Path**

На панели инструментов над областью сценария имеется кнопка **Insert Target Path** (Вставить путевое имя). При нажатии на нее открывается диалоговое окно **Insert Target Path**. По идее, оно могло бы стать ценным подспорьем для понимания иерархии временной шкалы и правильной записи путевых имен. К сожалению, эта функция не очень хорошо перенесла переход к версии **ActionScript 3.0**.

Не вдаваясь в детали, отметим, что в диалоговом окне **Insert Path** как правило создаются некорректные путевые имена. Использовать его для ознакомления с иерархией клипа при создании ролика можно, но кодировать путевые имена лучше вручную.



**Рис. 13.28.** После ввода адреса в поле URL Link на вкладке Properties в инспекторе свойств (A) выбранное текстовое поле (B) становится гиперссылкой в опубликованном ролике. В среде разработки ссылки подчеркиваются (C).



**Рис. 13.29.** В списке Target указывается, как следует открывать URL в окне браузера.

## Ссылки на другие Web-страницы

Flash предлагает два способа открывать новые файлы посредством ссылки на URL. Можно выбрать текст на арене и преобразовать его в гиперссылку, введя URL в поле Link на вкладке свойств инструмента «Текст» в инспекторе свойств. Можно также открыть URL из ActionScript-сценария. Оба способа позволяют открыть файл в новом окне браузера или в отдельном фрейме текущего окна.

### Чтобы создать текстовую ссылку на URL:

1. Выберите на арене текст, который должен стать ссылкой.  
Можно выбрать отдельные буквы или слова с помощью инструмента «Текст» или все текстовое поле инструментом «Стрелка».
2. Перейдите в инспектор свойств и введите URL в поле URL Link (URL-адрес) (справа от значка звена цепи) (рис. 13.28).
3. Выберите способ отображения страницы по указанному URL в окне браузера из списка Target (Назначение) (рис. 13.29):

**\_blank** открывает URL в новом окне браузера;

**\_parent** открывает URL в родительском окне текущего фрейма;

**\_self** открывает URL в том же фрейме окна браузера, где отображается текущий контент.

**\_top** открывает URL во фрейме верхнего уровня того окна браузера, где отображается текущий контент.

Если вы открываете опубликованный HTML-файл со своего жесткого диска и щелкаете по ссылке, ведущей на другой сайт, то можете получить сообщения об ошибках, связанных с безопасностью (см. врезку «Замечание о безопасности в Flash Player 9» в главе 17). Чтобы ссылки работали правильно, опубликованные файлы следует размещать на Web-сервере.

4. Выполните команду **Control ⇒ Test Movie**, чтобы протестировать работу ссылки.

При наведении курсора на текст он принимает форму указующего перста, а при щелчке по тексту Flash открывает URL в указанном окне или фрейме браузера.

### **Чтобы запрограммировать кнопку, которая открывает Web-страницу:**

1. Откройте новый документ по шаблону **ButtonActionsMaster**, созданному в этой главе ранее, и сохраните его, назвав его **LinkBtn.fla**.

На временной шкале есть три слоя: **Actions**, **Buttons** и **Frames**.

2. Чтобы создать кнопку, которая открывала бы новую Web-страницу, выполните следующие действия:

- выберите кадр 5 в слое **Buttons** и вставьте в него опорный кадр (например, с помощью команды **Insert ⇒ Timeline ⇒ Blank Keyframe**);
- выбрав кадр 4 в слое **Buttons**, перетащите на арену экземпляр символа-кнопки и перейдите на вкладку **Properties** в инспекторе свойств;
- введите имя экземпляра — **linkBtn**.

**Листинг 13.29.** Обработчик, проверяющий источник события.

```

1 // SimpleButton linkBtn;
2 function handleLink( pEvent:MouseEvent
3 ):void
4 {
5     if( pEvent.target == linkBtn )
6     {
7     }
```

**Листинг 13.30.** Регистрация для получения события click.

```

6 }
7 }
8 linkBtn.addEventListener(
  MouseEvent.CLICK, handleLink );
```

3. Выберите кадр 5 в слое Actions, вставьте в него пустой опорный кадр и перейдите на панель действий.

В предыдущих упражнениях вам уже доводилось создавать в опорном кадре 1 сценарии для работы с объектами, расположенными в кадре 1. Чтобы создать сценарий, который ссылается на объекты в одном из следующих кадров, можно добавить в соответствующую позицию пустой опорный кадр в слое Actions и присоединить к нему сценарий.

4. Введите в строке 1 комментарий для механизма подсказки:

```
// SimpleButton linkBtn;
```

5. Чтобы создать, обработчик, который проверяет источник события, выполните следующие действия:

- поставьте курсор в конце строки 1 и нажмите **Enter** для открытия нового блока;

- в строках 2–7 введите следующий код:

```

function handleLink(
    ⇨ pEvent:MouseEvent ):void
{
    if( pEvent.target == linkBtn )
    {
    }
}
```

Код должен выглядеть, как показано в листинге 13.29.

6. Чтобы зарегистрироваться для получения события click, поставьте курсор в конец строки 7 (после закрывающей скобки), нажмите **Enter** и введите такой текст:

```

linkBtn.addEventListener(
    ⇨ MouseEvent.CLICK, handleLink );
```

Код должен выглядеть, как показано в листинге 13.30.

7. Чтобы создать запрос к сети, выполните следующие действия:

- поставьте курсор в конец строки 5 (после открывающей скобки) и нажмите **Enter**;

- в строке 6 введите текст:

```
var request:URLRequest =
```

```
⇒ new URLRequest( «http://
```

```
⇒ www.google.com» );
```

Вместо ссылки на сайт Google можете ввести любую другую. В ActionScript 3.0 для установления любого соединения с внешним сервером необходимо создавать объект URLRequest. Здесь мы сохраняем его в переменной request.

8. Чтобы написать код для перехода по этой ссылке, поставьте курсор в конец строки 6 (после точки с запятой), нажмите **Enter** и наберите

```
navigateToURL( request, «_blank» );
```

Параметр `_blank` точно так же определяет способ открытия документа, как выбор из списка на шаге 3 предыдущего упражнения. Вместо него можно задавать значения `_self`, `_parent` или `_top`; однако не забывайте, что для этих способов может понадобиться настроить параметры безопасности (см. врезку «Замечание о безопасности в Flash Player 9» в главе 17).

9. Протестируйте действие.

Сохраните файл, проверьте синтаксис (код должен быть написан так, как в листинге 13.31), исправьте ошибки и запустите ролик в тестовом режиме.

При нажатии на кнопку в окне Flash-плеера в браузере открывается новое окно или вкладка, в которой отображается страница с указанным URL.

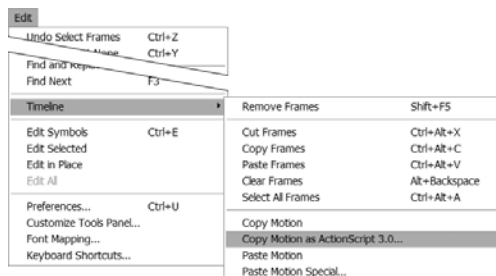
**Листинг 13.31.** Создание кнопки, ссылающейся на другую Web-страницу.

```
1 // SimpleButton linkBtn;
2 function handleLink( pEvent:MouseEvent
3 ):void
4 {
5     if( pEvent.target == linkBtn )
6     {
7         var request:URLRequest = new
8           URLRequest( "http://www.google.com" );
9         navigateToURL( request, "_blank" );
10    }
11 }
12 linkBtn.addEventListener( MouseEvent.CLICK,
13   handleLink );
```

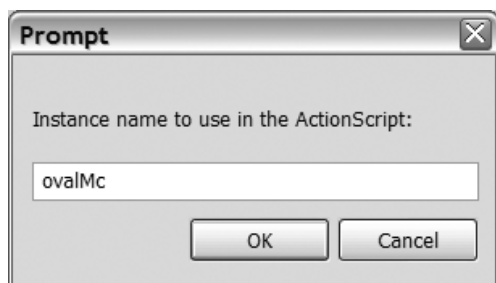
**C**

Необязательно создавать сценарии в пустом опорном кадре. Часто опорный кадр в слое Actions уже существует, и в нем находится сценарий для управления какой-то другой кнопкой. Тогда можно просто дописать новый сценарий в конец имеющегося. Попробуйте, например, в предыдущем упражнении добавить кнопку перехода на Web-страницу в опорный кадр 1 и допишите код в конец сценария, который уже имеется в шаблоне `ButtonActionsMaster`. Номера строк будут другими, но кнопка будет прекрасно работать.





**Рис. 13.30.** Чтобы преобразовать анимацию на временной шкале в код на языке ActionScript, выберите содержащие анимацию ключевые кадры и выполните команду **Edit** ⇒ **Timeline** ⇒ **Copy Motion As ActionScript 3.0**.



**Рис. 13.31.** В окне Prompt задайте имя экземпляра, к которому будет применен сохраненный код анимации.

## Преобразование анимации на временной шкале в код

С тех пор как в версии Flash 4 появились сценарии, программисты начали описывать движение с помощью кода. Но попытки запрограммировать повторно используемые анимированные эффекты на основе тщательно разработанной дизайнером интерполяции движения превращались в бесконечную череду проб и ошибок. Новая команда **Copy Motion As ActionScript 3.0** позволяет преобразовать в код тонкие нюансы анимации графики в опорных кадрах. Вам нужно лишь выбрать кадры на временной шкале, скопировать их в виде ActionScript-кода, а потом вставить код в свой сценарий.

### Чтобы скопировать движение на временной шкале:

1. Откройте новый документ Flash (типа ActionScript 3.0), создайте в нем символ-клип, содержащий звезду, и назовите его **Star**.
2. Поместите экземпляр символа **Star** на арту и анимируйте с помощью методов покадровой анимации и интерполяции движения, которые были рассмотрены в главах 8 и 9.
3. Выберите на временной шкале все слои и кадры, содержащие анимацию, и выполните команду **Edit** ⇒ **Timeline** ⇒ **Copy Motion As ActionScript 3.0** (Редактирование ⇒ Временная шкала ⇒ Копировать движение как код ActionScript 3.0) (рис. 13.30). Открывается диалоговое окно Prompt (Запрос).
4. В окне Prompt введите имя экземпляра символа-клипа (или другого графического объекта), который вы хотите анимировать, например, **ovalMc** (рис. 13.31).

Этот экземпляр не обязан существовать в данный момент, но именно он будет анимироваться из программы.

## 5. Нажмите ОК.

Flash преобразует покадровую и интерполяционную анимацию в код на языке ActionScript 3.0 и помещает этот код в буфер обмена, откуда его можно будет вставить в сценарий.

## C

Суть этой техники состоит в том, что анимация одного графического объекта на протяжении последовательности опорных кадров применяется к другому объекту. Если анимация содержит комбинированные формы или несколько графических объектов, то при выборе команды Copy Motion As ActionScript 3.0 Flash предупреждает об этом еще до открытия окна Prompt.

### Чтобы вставить скопированное движение в виде кода на языке ActionScript:

1. Откройте новый Flash-файл (типа ActionScript 3.0) и создайте символ-клип, содержащий овал. Назовите его Oval.
2. Поместите экземпляр символа Oval на арену и назовите его ovalMc.
3. Создайте новый слой с именем Actions.
4. Выбрав кадр 1 в слое Actions, перейдите на панель действий.
5. Чтобы приостановить ролик во время выполнения, введите в строке 1 код stop();.
6. Оставив курсор в конце строки 1 (после точки с запятой), нажмите **Enter**.
7. Теперь, когда курсор находится в строке 2, щелкните мышью с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows) и выберите из контекстного меню пункт **Paste** (Вставить).

Flash вставит скопированный ранее код, описывающий анимацию. Этот код включает части на языках XML и ActionScript (рис. 13.32).

## Начало XML

```
<Motion duration="15" xmlns="fl.motion.*"
```



```
</Motion>;
```

## Конец XML

**Рис. 13.32.** Вставленный код сложен, но пусть это вас не пугает. Вам не надо его редактировать, достаточно просто использовать. Если хотите, можете поэкспериментировать с изменением XML-кода и даже ActionScript-кода.

## 8. Протестируйте ролик.

В окне Flash-плеера экземпляр `ovalMc` совершает те же движения, что были созданы для символа `Star` в предыдущем упражнении, хотя в этом файле всего один опорный кадр.

**С**

Если после вставки сценария, скопированного командой `Copy Motion As ActionScript 3.0`, вы обнаружите, что ввели неверное имя в окне `Prompt`, то повторять всю процедуру копирования с самого начала необязательно. Нажмите кнопку **Найти** на панели инструментов (значок лупы). Откроется стандартный диалог поиска и замены. Теперь просто замените неправильно введенное имя экземпляра в тексте сценария.

**С**

Сценарий, генерируемый командой `Copy Motion As ActionScript 3.0`, для начинающего выглядит довольно сложно, но не пугайтесь. Если пока вам не хочется заниматься нетривиальным программированием, можете вообще ничего в нем не менять. Если понадобится изменить сгенерированный сценарий анимации, то всегда можно модифицировать оригинал на временной шкале и повторить процедуру копирования и вставки. Поэтому разумно подготовить систему для совместной работы графического дизайнера и программиста. Сохраняйте исходные документы Flash, содержащие анимацию на временной шкале до тех пор, пока проект не будет завершен.

**С**

Сценарий, сгенерированный командой `Copy Motion As ActionScript 3.0`, точнее его XML-часть — это детальное описание каждого опорного кадра. Если у вас есть авантюрная жилка, можете попробовать изменить XML-код и посмотреть, что получится.

## Использование кнопок для управления графическими объектами

Ранее в этой главе вы научились ссылаться на символ-клип и приостанавливать его воспроизведение. Но раз уж вы знаете, как обращаться к клипу, то почему бы не попробовать и другие манипуляции с ним, изменяя различные свойства? В этом разделе вы узнаете, как переместить клип, изменить прозрачность и даже анимировать его из сценария. (Такая же техника применима к кнопкам и текстовым полям.)

### Чтобы подготовить файл к размещению нескольких элементов управления:

1. Откройте файл `pauseClipControl.fla`, созданный в разделе «Использование кнопок для управления временной шкалой».
2. Сохраните его копию под именем `Control-GraphicObjectsStart.fla`.
3. В слое `Buttons` на временной шкале выберите опорный кадр 1 и добавьте на арену еще два экземпляра кнопок.
4. Назовите новые кнопки `moveBtn` и `transparentBtn`. Для этого поочередно выберите их на арене и введите имена в поле `Name` на вкладке **Properties** в инспекторе свойств.
5. В слое `Actions` выберите опорный кадр 1 и перейдите на панель действий. Появляется текст созданного ранее сценария.
6. Оставьте комментарий механизму подсказки, для чего поместите курсор в конец строки 5, нажмите **Enter** и в строках 6 и 7 введите текст:

```
// SimpleButton moveBtn;  
// SimpleButton transparentBtn;
```

Тем самым вы описали обе новых кнопки (листинг 13.32).

**Листинг 13.32.** Добавление комментария с описанием кнопок.

```
5 // MovieClipAnimMc;  
6 // SimpleButton moveBtn;  
7 // SimpleButton transparentBtn;  
8 // event handler
```

**Листинг 13.33.** Регистрация обработчика события click от кнопок.

```
17 // register events
18 controlBtn.addEventListener(
   MouseEvent.CLICK, handleClick );
19 moveBtn.addEventListener(
   MouseEvent.CLICK, handleClick );
20 transparentBtn.addEventListener(
   MouseEvent.CLICK, handleClick );
```

7. Чтобы зарегистрироваться для получения события click, поставьте курсор в конец строки 18, нажмите **Enter** и в строках 19 и 20 введите такой код:

```
moveBtn.addEventListener(
    ⇨ MouseEvent.CLICK, handleClick );
transparentBtn.addEventListener(
    ⇨ MouseEvent.CLICK, handleClick );
```

Теперь для обеих кнопок активирован механизм подсказок и зарегистрирован обработчик события click (листинг 13.33).

8. Протестируйте действие.

Сохраните файл, проверьте синтаксис, исправьте ошибки и запустите ролик в тестовом режиме (см. раздел «Тестирование действий»).

Вы еще не написали обработчик, реагирующий на нажатия новых кнопок; пока лишь проверяется, что все сделанное к настоящему моменту, работает правильно. В следующих упражнениях мы наделим кнопки интерактивностью.

9. Сохраните документ как шаблон, назвав его ClipControlMaster, и закройте файл. Этот сценарий можно будет в дальнейшем использовать как образец для других кнопок, управляющих воспроизведением клипов. О том, как сохранять документы в виде шаблонов, см. главу 1.

### Чтобы изменить координаты клипа:

1. Откройте новый документ по шаблону ClipControlMaster, созданному в предыдущем упражнении. Сохраните его, назвав MoveClip.fla.
2. В слое Actions выберите опорный кадр 1 и перейдите на панель действий. В области сценария появится текст сценария из шаблона.

3. Чтобы написать код взаимодействия с кнопкой `moveBtn`, выполните следующие действия:

- поставьте курсор в конец строки 15 (после закрывающей скобки) и нажмите **Enter**;
- введите в строках 16–18 такой текст:  

```
else if( pEvent.target ==
    ⇨ moveBtn )
{
}
```

В этой части обработчика код, заключенный в круглые скобки, выполняется только при нажатии экземпляра кнопки `moveBtn`. Строки 16–18 должны быть написаны так, как в листинге 13.34.

4. Чтобы поместить экземпляр ролика `animMc` в точку с заданными координатами, выполните следующие действия:

- поставьте курсор в конец строки 17 (после открывающей скобки) и нажмите **Enter**;
- введите в строках 18–20 такой текст:  

```
// move the clip
animMc.x = 20;
animMc.y = 50;
```

Здесь вы просите Flash изменить координаты *x* и *y* объекта `animMc` во время выполнения. Объект позиционируется относительно точки регистрации, как и при использовании инспектора свойств на этапе разработки.

5. Протестируйте действие.

Сохраните файл, проверьте синтаксис (код должен быть написан так, как в листинге 13.35), исправьте ошибки и запустите ролик в тестовом режиме (см. раздел «Тестирование действий»).

В окне Flash-плеера символ-клип `animMc` сначала располагается в центре арены. После нажатия кнопки `moveBtn` он перемещается в другую точку (рис. 13.33).

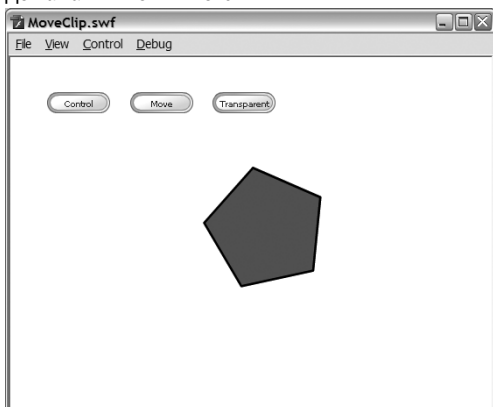
**Листинг 13.34.** Проверка того, что событие поступило от кнопки `moveBtn`.

```
14         this.animMc.stop();
15     }
16     else if( pEvent.target == moveBtn )
17     {
18     }
19 }
```

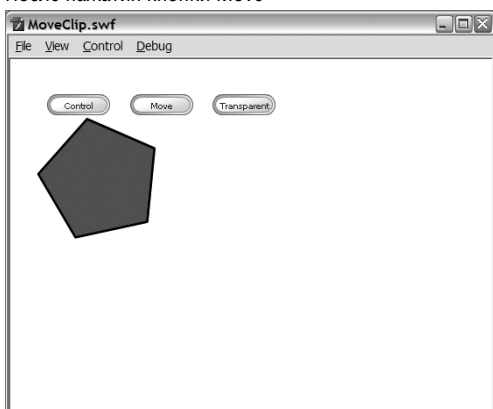
**Листинг 13.35.** Полный текст сценария перемещения символа-клипа в новое место.

```
9 function handleClick( pEvent:MouseEvent
):void
10 {
11     if( pEvent.target == controlBtn )
12     {
13         // handle event
14         animMc.stop();
15     }
16     else if( pEvent.target == moveBtn )
17     {
18         // move the clip
19         animMc.x = 20;
20         animMc.y = 50;
21     }
22 }
```

До нажатия кнопки Move



После нажатия кнопки Move



**Рис. 13.33.** Пятиугольник — это символ-клип с именем `animMc` (внутри него определена интерполяция движения, которая вращает и сжимает форму). Кнопка с надписью `Move` соответствует экземпляру объекта `moveBtn`. В начальный момент вращающаяся форма находится в центре арены. Сценарий смещает экземпляр `animMc` в левый верхний угол при нажатии кнопки `moveBtn`.

**Чтобы задать прозрачность клипа:**

1. Продолжая работу с файлом из предыдущего упражнения (`MoveClip.fla`), сохраните новую копию, назвав ее `TransparentClip.fla`.
2. В слое `Actions` выберите опорный кадр 1 и перейдите на панель действий.
3. Чтобы написать код взаимодействия с кнопкой `transparentBtn`, выполните следующие действия:

- поставьте курсор в конец строки 21 (после закрывающей скобки) и нажмите **Enter**;

- введите в строках 22–24 такой текст:

```
else if (pEvent.target ==
    transparentBtn )
{
}
```

Эта часть обработчика выполняется при нажатии кнопки `transparentBtn` (листинг 13.36).

4. Чтобы изменить прозрачность символа-клипа `animMc`, выполните следующие действия:

- поставьте курсор в конец строки 23 (после открывающей скобки) и нажмите **Enter**;

- введите в строках 24 и 25 такой текст:

```
// make the clip 50% transparent
animMc.alpha = 0.5;
```

Здесь для `animMc` устанавливается коэффициент прозрачности 50%, в результате чего содержимое клипа становится полупрозрачным.

**Листинг 13.36.** Проверка того, что событие поступило от кнопки `transparentBtn`.

```
20 animMc.y = 50;
21 }
22 else if (pEvent.target == transparentBtn
23 )
24 {
25 }
```

## 5. Протестируйте действие.

Сохраните файл, проверьте синтаксис (код должен быть написан так, как в листинге 13.37), исправьте ошибки и запустите ролик в тестовом режиме (см. раздел «Тестирование действий»).

В окне Flash-плеера при нажатии кнопки transparentBtn к символу animMc применяется 50-процентная прозрачность (рис. 13.34). Остальные кнопки по-прежнему выполняют ранее запрограммированные действия: кнопка moveBtn смещает клип animMc, а кнопка controlBtn останавливает его воспроизведение.

C

Если клип, который вы делаете прозрачным, является единственным объектом на арене, то изменение незаметно. Чтобы увидеть его эффект, поместите какой-нибудь текст или форму в слой, расположенный под слоем MovieClips, и еще раз протестируйте ролик.

### Чтобы перемещать клип по экрану путем обработки события enterFrame:

1. Продолжая работу с файлом из предыдущего упражнения (TransparentClip.fla), сохраните новую копию, назвав ее AnimateClip.fla.
2. В слое Actions выберите опорный кадр 1 и перейдите на панель действий.
3. Создайте переменную, для чего в области сценария выполните следующие действия:

- поставьте курсор в конец строки 31 (после точки с запятой) и нажмите **Enter**;
- введите в строках 32–34 такой текст:

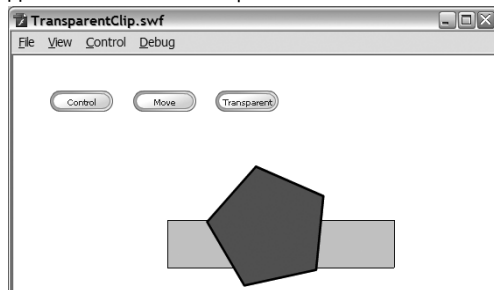
```
// animate the clip across the
// screen
// speed in pixels
var moveX:Number = 10;
```

**Листинг 13.37.** Сценарий, делающий клип animMc прозрачным.

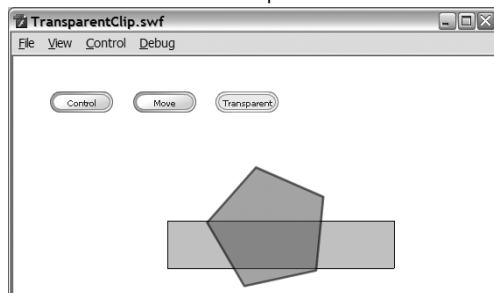
```

9  function handleClick( pEvent:MouseEvent
   ):void
10  {
11      if( pEvent.target == controlBtn )
12      {
13          // handle event
14          animMc.stop();
15      }
16      else if( pEvent.target == moveBtn )
17      {
18          // move the clip
19          animMc.x = 20;
20          animMc.y = 50;
21      }
22      else if( pEvent.target == transparentBtn
23      )
24      {
25          // make the clip 50% transparent
26          animMc.alpha = 0.5;
27      }
28  }
```

До нажатия кнопки Transparent



После нажатия кнопки Transparent



**Рис. 13.34.** Сценарий для кнопки transparentBtn устанавливает новый коэффициент прозрачности для символа-клипа animMc.



**Листинг 13.38.** Определение переменной `moveX` и регистрация для получения события `enterFrame`.

```
31 transparentBtn.addEventListener(  
    MouseEvent.CLICK, handleClick );  
32 // animate the clip across the screen  
33 // speed in pixels  
34 var moveX:Number = 10;
```

**Листинг 13.39.** Определение обработчика события `moveClip`.

```
33 // speed in pixels  
34 var moveX:Number = 10;  
35 // event handler  
36 function moveClip( pEvent:Event ):void  
37 {  
38 }
```

Этот сценарий должен выглядеть так, как показано в листинге 13.38. *Переменной* называется контейнер для хранения значений. Здесь число 10 означает 10 пикселей на арене. Это значение сохраняется в переменной `moveX` для последующего использования. В сценарии анимации, который будет написан далее, из переменной `moveX` Flash узнает, насколько сместить символ-клип `animMc`.

4. Чтобы определить обработчик события `moveClip`, выполните следующие действия:

- поставьте курсор в конец строки 34 (после точки с запятой) и нажмите **Enter**;
- введите в строках 35–38 такой текст:

```
// event handler  
function moveClip(  
    pEvent:Event ):void  
{  
}
```

Обработчик события `enterFrame` очень похож на созданные раньше. Разница в том, что событие теперь имеет типа `Event`, а не `MouseEvent` (листинг 13.39).

### Дополнительное упражнение на написание сценариев

Вы уже знаете, как создать кнопку, которая останавливает воспроизведение клипа. Теперь можно сделать кнопку, которая запустит воспроизведение снова. Знаний у вас достаточно, чтобы выполнить это упражнение самостоятельно.

Откройте файл `TransparentClip.fla` из предыдущего упражнения. Сохраните его копию, назвав ее `pausePlay.fla`. Добавьте еще одну кнопку на главную временную шкалу и как-то назовите ее. Зарегистрируйте все тот же обработчик `handleClick` для получения события `click` от новой кнопки. Добавьте в его текст еще одну проверку источника события и, если выясняется, что была нажата новая кнопка, вызовите метод `play()` объекта `animMc`.

5. Чтобы Flash перемещал клип `animMc` всякий раз, как точка воспроизведения переходит на новый кадр, выполните следующие действия:

- поставьте курсор в конец строки 37 (после открывающей скобки) и нажмите **Enter**;
- введите в строках 38 и 39 такой текст:  

```
// move the clip by the amount
// specified
animMc.x = animMc.x + moveX;
```

В этой строке изменяется координата `x` клипа `animMc` — к старому значению прибавляется значение переменной `moveX` (10). Поскольку Flash-плеер вызывает обработчик `moveClip` всякий раз, как точка воспроизведения переходит к новому кадру, то этот сценарий смещает клип `animMc` на 10 пикселей вправо в каждом кадре (листинг 13.40).

6. Чтобы зарегистрироваться для получения события `enterFrame`, выполните следующие действия:

- поставьте курсор в конец строки 40 (после закрывающей скобки) и нажмите **Enter**;
- введите в строках 41 и 42 такой текст:  

```
// register event
addEventListener(
    Event.ENTER_FRAME, moveClip );
```

При каждом продвижении точки воспроизведения любой объект, находящийся на арене, может получать событие `enterFrame`. Это событие не связано с действиями пользователя.

7. Протестируйте действие.

Сохраните файл, проверьте синтаксис (код должен быть написан так, как в листинге 13.41), исправьте ошибки и запустите ролик в тестовом режиме (см. раздел «Тестирование действий»).

Ролик открывается в окне Flash-плеера. Без всякого взаимодействия с пользователем клип `animMc` перемещается по экрану, не прекращая анимацию. В конце концов клип уходит за правый край окна. Чтобы вернуть его в видимую часть арены, можете нажать кнопку `Move`.

**Листинг 13.40.** Полный текст обработчика события для сдвига `animMc` на 10 пикселей.

```
36 function moveClip( pEvent:Event ):void
37 {
38     // move the clip by the amount specified
39     animMc.x = animMc.x + moveX;
40 }
```

**Листинг 13.41.** Полный текст сценария для перемещения `animMc` по экрану.

```
32 // animate the clip across the screen
33 // speed in pixels
34 var moveX:Number = 10;
35 // event handler
36 function moveClip( pEvent:Event ):void
37 {
38     // move the clip by the amount specified
39     animMc.x = animMc.x + moveX;
40 }
41 // register event
42 addEventListener( Event.ENTER_FRAME,
    moveClip );
```

**Листинг 13.42.** Проверка того, что клип animMc достиг края арены.

```
39 animMc.x = animMc.x + moveX;  
40 if( animMc.x >= stage.stageWidth )  
41 {  
42 }  
43 }
```

### Чтобы остановить перемещение клипа по достижении края окна:

1. Продолжая работу с файлом из предыдущего упражнения (AnimateClip.fla), выберите в слое Actions опорный кадр 1.
2. Чтобы сценарий проверял, достиг ли клип края арены, выполните следующие действия:
  - в области сценария на панели действий поставьте курсор в конец строки 39 (после точки с запятой) и нажмите **Enter**;
  - введите в строках 40–42 такой текст:  

```
if( animMc.x >= stage.stageWidth )  
{  
}
```

Получить доступ к арене из ActionScript можно с помощью свойства stage любого интерактивного объекта. Код в строке 40 проверяет, верно ли, что горизонтальная координата точки регистрации animMc (свойство x) больше либо равна ( $\geq$ ) ширине арены (stageWidth). Если это так, то клип animMc вышел за пределы арены (листинг 13.42).

3. Чтобы отказаться от получения события enterFrame, выполните следующие действия:
  - поставьте курсор в конец строки 41 (после открывающей скобки) и нажмите **Enter**;
  - введите в строке 42 такой текст:  

```
removeEventListener(  
    ⇨ Event.ENTER_FRAME, moveClip );
```

Если вы больше не хотите получать событие, на которое зарегистрировались, регистрацию можно отменить (см. врезку «О событии enterFrame» ниже).

## 4. Протестируйте действие.

Сохраните файл, проверьте синтаксис (код должен быть написан так, как в листинге 13.43), исправьте ошибки и запустите ролик в тестовом режиме (см. раздел «Тестирование действий»).

Клип `animMc` перемещается вправо. По достижении правого края окна Flash-плеера перемещение прекращается, но пятиугольник продолжает вращаться. В момент достижения правого края сценарий отменяет регистрацию события `enterFrame`, обработчик которого сдвигал `animMc` в каждом кадре. Если точка регистрации находится не на левом краю анимированного клипа, то часть его может все же выйти за пределы окна.

После отмены регистрации нажатие кнопки Move возвращает `animMc` в исходное положение, но больше он вправо не движется.

**Листинг 13.43.** Код, который прекращает перемещение клипа по достижении правого края арены.

```

36 function moveClip( pEvent:Event ):void
37 {
38     // move the clip by the amount specified
39     animMc.x = animMc.x + moveX;
40     if( animMc.x >= stage.stageWidth )
41     {
42         removeEventListener(
43             Event.ENTER_FRAME, moveClip );
44     }
45 }
```

### О событии `enterFrame`

В этой главе мы говорили в основном о событиях, инициируемых пользователем — щелчках, наведении курсора, нажатиях клавиш и т.п. Но в ActionScript 3.0 есть много других событий, которые напрямую не связаны с действиями пользователя, одно из них — `enterFrame`.

Событие `enterFrame` возникает всякий раз, как Flash перемещает точку воспроизведения по любой временной шкале, вне зависимости от того воспроизводится конкретный клип или остановлен. Этим можно воспользоваться, чтобы периодически выполнять некоторые действия (как, например, в разделе «Чтобы перемещать клип по экрану путем обработки события `enterFrame`»). Достаточно зарегистрировать обработчик, которые будет получать такие «бие-ния пульса» с помощью следующего кода:

```
addEventListener( Event.ENTER_FRAME, eventHandler );
```

Событие `enterFrame` очень полезно для программной анимации.

Однако имейте в виду, что событие продолжает генерироваться все время, пока ролик работает. Если необходимость в нем отпала, например завершилась анимация, то регистрацию следует отменить, вызвав метод `removeEventListener`. Код отмены регистрации выглядит примерно так:

```
removeEventListener( Event.ENTER_FRAME, eventHandler );
```

Точно так же можно отменить и реакцию на события мыши.

# ИСПОЛЬЗОВАНИЕ ГРАФИКИ, СОЗДАННОЙ В ДРУГИХ ПРОГРАММАХ

14

## Какие графические форматы умеет импортировать Flash?

Flash поддерживает широкое разнообразие векторных и растровых графических форматов, а именно: Adobe Illustrator версии 10 и младше, Adobe Photoshop, AutoCAD DXF, Bitmap (BMP), Enhanced Windows Metafile (EMF, только на платформе Windows), Adobe FreeHand версий 7–11, FutureSplash Player (SPL), GIF и анимированный GIF, JPEG, PNG, Flash Player 6/7 и Windows Metafile (WMF). Если на вашем компьютере установлена программа QuickTime версии 4 или старше, то можно импортировать также файлы в следующих форматах: MacPaint (PNTG), PICT (как растровое изображение на платформе Windows), QuickTime Image (QTIF), Silicon Graphics Image (SGI), TGA и TIFF.

Инструменты рисования, имеющиеся в Adobe Flash CS3 Professional, достаточно мощны и гибки, но вам, возможно, больше по душе инструменты из других программ создания векторной графики, например Adobe Illustrator или Adobe FreeHand. А, быть может, вы хотите включить во Flash-ролик картинки, созданные в Adobe Photoshop, отсканированные фотографии или другие растровые изображения. К счастью, Flash позволяет импортировать графику, а усовершенствования, внесенные во Flash CS3, делают этот процесс надежным и безболезненным.

Flash может импортировать векторную и растровую графику из буфера обмена или с помощью команды Import. Можно также перетаскивать изображения из программ Photoshop CS3, Illustrator CS3 и FreeHand версий 7–11 (MX) непосредственно во Flash.

Во Flash есть две команды импорта: **Import to Stage** (Импортировать на арену) и **Import to Library** (Импортировать в библиотеку). Обе работают примерно одинаково. В некоторых диалоговых окнах предлагаются слегка различающиеся параметры в зависимости от того, какой командой вы воспользовались. В упражнениях из этой главы мы будем работать только с командой **Import to Stage**.

## Импорт растровых изображений

Какой бы вид графики вы ни пожелали импортировать, начальные шаги всегда одинаковы. В случае простого растрового изображения команды импорта переносят его напрямую во Flash.

### Чтобы импортировать растровое изображение:

1. Открыв документ, выполните команду **File** ⇒ **Import** ⇒ **Import to Stage** (Файл ⇒ Импорт ⇒ Импорт на арену).

Открывается диалоговое окно Import (рис. 14.1). Команда **Import to Stage** импортирует файл в открытый документ, сохраняет главный актив-растр в библиотеке и помещает экземпляр растра на арену в текущий кадр в активном слое (рис. 14.2).

2. Из списка **Enable (Mac)** или **Files of Type** (Типы файлов) (Windows) выберите формат импортируемого файла.
3. Найдите нужный файл в своей системе.
4. Выберите файл.
5. Нажмите кнопку **Import** (Mac) или **Open** (Открыть) (Windows).

Растровое изображение теперь находится в библиотеке и на арене.

**C** На шаге 1 можно было бы выбрать команду **File** ⇒ **Import** ⇒ **Import to Library** (Файл ⇒ Импорт ⇒ Импорт в библиотеку). Тогда новый актив был бы помещен только в библиотеку, а на арене не появился бы.

**C** Импортированное изображение можно изменить в любом установленном графическом редакторе. Выберите его на библиотечной панели, щелкните мышью с нажатой клавишей **Control** (Mac) или правой кнопкой мыши (Windows) по значку и выберите команду **Edit With** (Редактировать с помощью) из контекстного меню. В открывающемся окне найдите подходящую программу и нажмите кнопку **Open**. Измените и сохраните изображение, Flash обновит его в библиотеке.

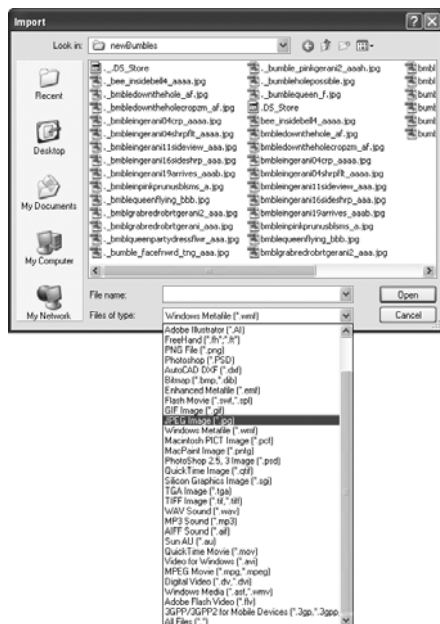


Рис. 14.1. Диалоговое окно Import предназначено для импорта в ваш документ Flash графики, созданной в других программах.

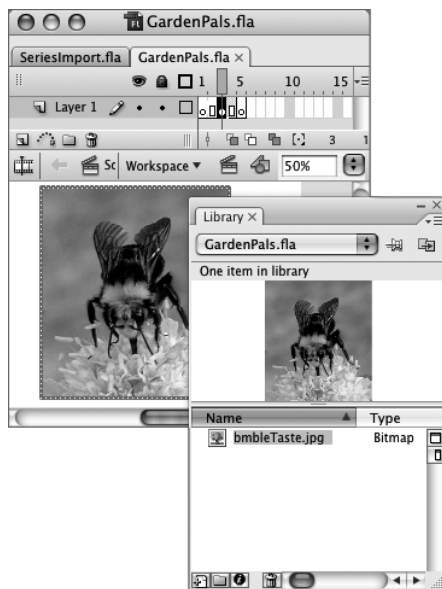
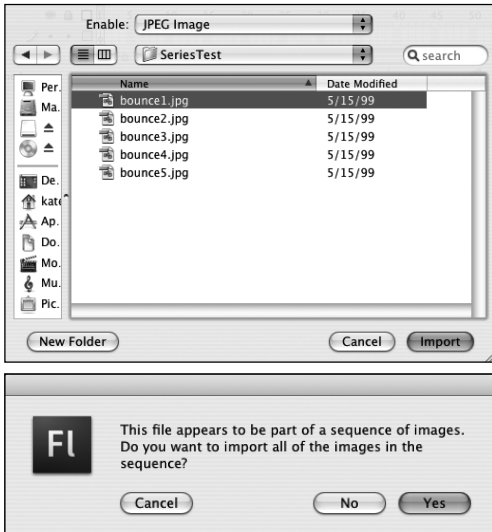


Рис. 14.2. Одновременно с импортом растрового изображения на арену Flash сохраняет главную копию в библиотеке.

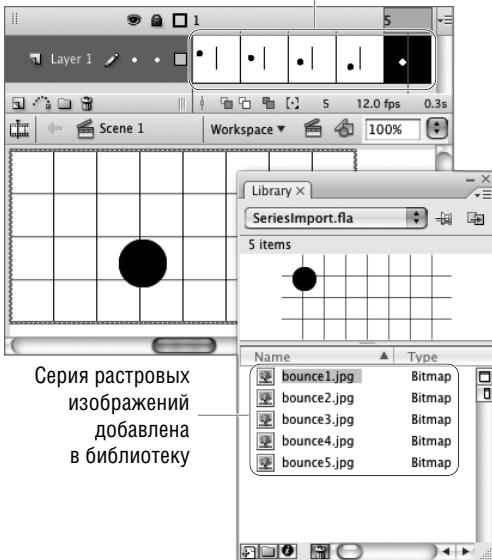


**Рис. 14.3.** При импорте первого из серии последовательно пронумерованных файлов (верхний рисунок) Flash спрашивает, хотите ли вы импортировать всю серию (нижний рисунок).

### Чтобы импортировать серию графических файлов:

6. Выполните шаги 1–5 из предыдущего упражнения. На шаге 2 выберите формат и найдите первый из подлежащих импорту файлов. Все файлы должны быть в одной папке, а их имена должны отличаться только последними цифрами, например: bounce1, bounce2, bounce3. Если эти условия выполнены, то появляется окно с вопросом, хотите ли вы импортировать серию последовательных изображений (рис. 14.3).
7. Нажмите **Yes**. Flash поместит каждое изображение в отдельный опорный кадр в активном слое (рис. 14.4).

В режиме предварительного просмотра видны изображения в опорных кадрах



Серия растровых изображений добавлена в библиотеку

**Рис. 14.4.** Каждый из серии импортированных файлов помещается в отдельный опорный кадр на временной шкале текущего документа.

## Импорт файлов, созданных в программе Photoshop

При импорте графики из Photoshop Flash автоматически принимает во внимание текущие параметры, установленные в окне Preferences. Команда импорта открывает диалоговое окно PSD Import, где можно просмотреть и изменить параметры для отдельных слоев.

### Чтобы задать параметры импорта для файлов Photoshop (PSD):

1. Из меню **Flash** (Mac) или **Edit** (Windows) выберите пункт **Preferences** (Установки). Открывается диалоговое окно Preferences.
2. Выберите категорию **PSD File Importer** (Импортер PSD-файлов). В правой части окна появляются параметры импорта PSD-файлов (рис. 14.5).
3. Для задания параметров импорта слоев изображения (рис. 14.6), выполните одно из следующих действий:
  - чтобы максимально сохранить прозрачность и режимы наложения в том объеме, который может обработать Flash, отметить переключатель **Bitmap images with editable layer styles** (Растровые изображения с изменяемыми стилями слоя);
  - чтобы наиболее достоверно передать изображение, отметьте переключатель **Flattened Bitmap Images** (Сведенные растровые изображения).

Если переключатель установлен в это положение, то для того чтобы Flash создал одновременно и само изображение, и содержащий его клип, отметьте флажок **Create Movie Clips** (Создать фрагменты ролика).

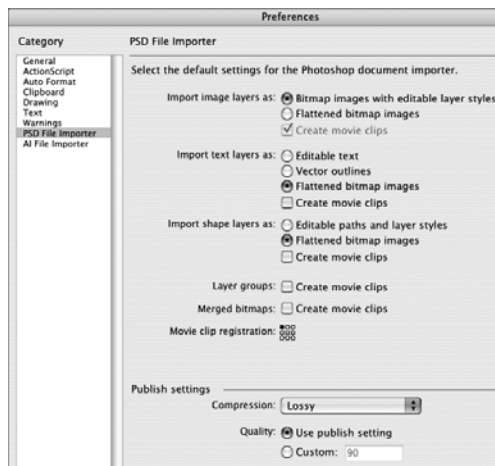


Рис. 14.5. Выбрав категорию PSD File Importer в окне Preferences, вы сможете изменять параметры импорта PSD-файлов.

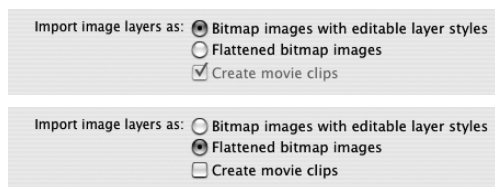
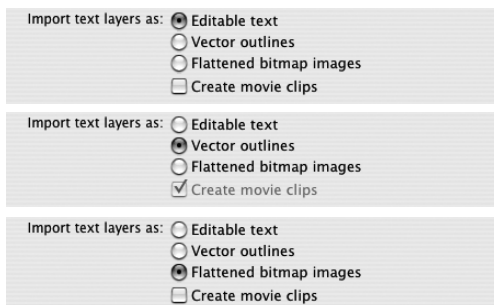
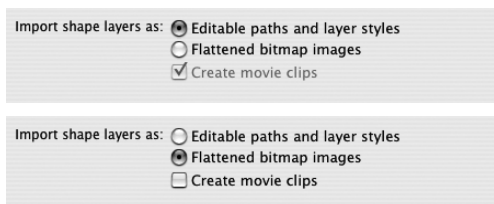


Рис. 14.6. Импортировать слои PSD-файлов можно двумя способами. Если выбрать вариант импорта с редактируемыми слоями (верхний рисунок), то Flash автоматически создает символы-клипы, содержащие импортированные слои и применяет подходящий режим наложения к экземпляру символа. Если же слои импортируются как растровые изображения (нижний рисунок), можно указать, следует ли создавать содержащие их клипы или ограничиться только импортом самих изображений.

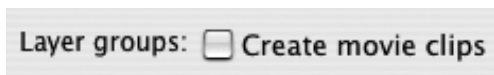




**Рис. 14.7.** Существует три варианта импорта текста из PSD-файла: в виде редактируемых текстовых полей (верхний рисунок), в виде редактируемых форм (средний рисунок) и в виде растровых изображений (нижний рисунок). Если выбран импорт текста в виде редактируемых форм, то Flash автоматически создает содержащие их символы-клипы.



**Рис. 14.8.** При импорте слоев, содержащих формы, в виде редактируемых путей (верхний рисунок) Flash автоматически помещает пути в символ-клип. При импорте же в виде растровых изображений (нижний рисунок) вы сами можете указать, следует ли помещать их в клип, отметив флажок **Create Movie Clips**.



**Рис. 14.9.** При импорте групп слоев можно указать, что они должны быть помещены в символы-клипы.



**Рис. 14.10.** Если вы хотите объединить слои в PSD-файле для импорта, то можете указать, что они должны быть помещены в символы-клипы.

4. Для задания параметров импорта текстовых слоев (рис. 14.7), выполните одно из следующих действий:

- для импорта текста в редактируемые статические текстовые поля выберите переключатель **Editable Text** (Редактируемый текст);
- для импорта текста в виде набора редактируемых форм выберите переключатель **Vector Outlines** (Векторные контуры);
- для импорта текста в виде растровых изображений выберите переключатель **Flattened Bitmap Images** (Сведенные растровые изображения);
- если хотите, чтобы Flash поместил редактируемый текст или растровое изображение в символ-клип, отметьте флажок **Create Movie Clips** (Создать фрагменты ролика).

5. Для задания параметров импорта слоев, содержащих формы (рис. 14.8), выполните одно из следующих действий:

- чтобы импортировать редактируемую форму с сохранением прозрачности и режима наложения в объеме, поддерживаемом Flash, выберите переключатель **Editable paths and layer styles** (Изменяемые пути и стили слоя);
- чтобы наиболее достоверно передать форму, выберите переключатель **Flattened Bitmap Images**. В этом случае можно указать, что должен быть создан содержащий изображение символ-клип; для этого отметьте флажок **Create Movie Clips**.

6. Чтобы преобразовать сгруппированные слои, отметьте флажок **Create Movie Clips** (Создать фрагменты ролика) в разделе **Layer Groups** (Группы слоев) (рис. 14.9).

7. Чтобы преобразовать объединенные слои в символы-клипы, отметьте флажок **Create Movie Clips** в разделе **Merged Bitmaps** (Объединенные растровые изображения) (рис. 14.10). (Об объединенных слоях см. следующее упоминание «Чтобы импортировать содержимое из PSD-файла».)

8. Чтобы установить положение точки регистрации для символов-клипов, создаваемых в процессе импорта, щелкните по одному из девяти квадратов в поле Movie Clip Registration (Расположение фрагмента ролика) (рис. 14.11).

По умолчанию Flash помещает точку регистрации в одно и то же место для всех клипов, создаваемых в процессе импорта. Для отдельного клипа умолчание можно переопределить в диалоговом окне PSD Import. Подробнее о точках регистрации и символах см. главу 7.

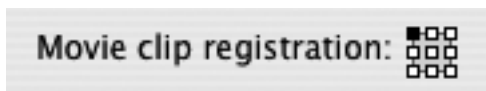
9. В разделе Publish Settings (Параметры публикации) (рис. 14.12) выберите режим сжатия из списка **Compression** (Сжатие):

- чтобы сохранить все данные изображения из импортированного PSD-файла при публикации Flash-ролика, выберите пункт Lossless (Без потерь). В этом случае на этапе создания SWF-файла применяется алгоритм сжатия без потери информации. В таком режиме сохраняется наивысшее качество изображения, но ценой увеличения размера файла. Дополнительную информацию о публикации см. в главе 17.

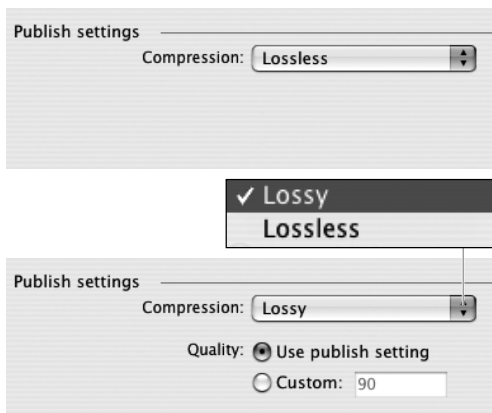
или

- чтобы применить алгоритм сжатия JPEG к импортированным данным (рис. 14.12), выберите пункт Lossy (С потерей качества). Отметьте переключатель Use Publish Setting (Использовать настройки публикации), если хотите применить уровень качества, определенный в параметрах публикации, или Custom (Заказное), если желаете задать его здесь. В последнем случае введите значение уровня от 0 до 100. Чем больше значение, тем выше качество сжатого изображения, но и тем больше размер результирующего файла.

10. Нажмите **ОК**, чтобы сохранить заданные параметры и закрыть окно.



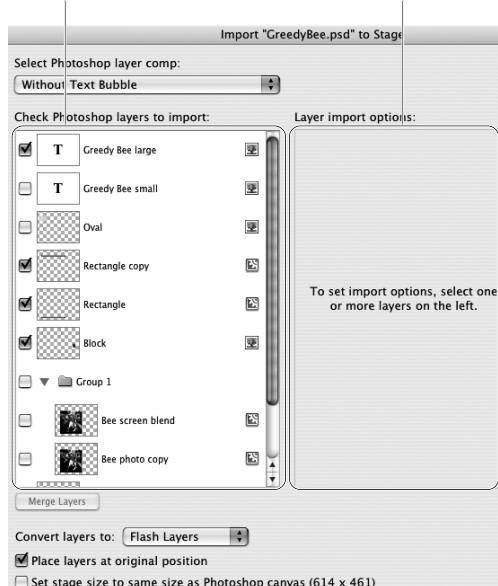
**Рис. 14.11.** В категории PSD Importer щелкните по какому-нибудь из девяти квадратов в поле Registration model, чтобы определить положение точки регистрации, принимаемое по умолчанию для символов-клипов, импортированных из PSD-файлов.



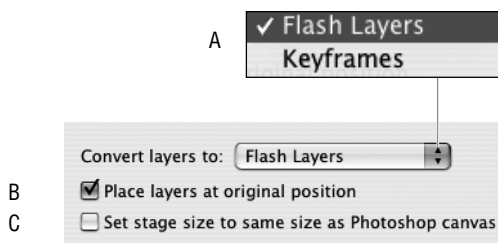
**Рис. 14.12.** Список Compression позволяет выбрать метод сжатия, применяемый на этапе публикации. Сжатие без потерь (верхний рисунок) сохраняет все данные изображения. В случае сжатия с потерей качества (нижний рисунок) можно еще определить уровень потерь.

Панель слоев

Панель параметров



**Рис. 14.13.** На панели слоев в окне PSD Import находится список, в котором представлены элементы, доступные для импорта. При выборе элемента из списка его характеристики отображаются на панели параметров справа. Параметры, относящиеся к импорту документа в целом, находятся в нижней части окна.



**Рис. 14.14.** В нижней части окна PSD Import задаются параметры, применяемые к импорту документа в целом. Можно определить, как Flash должен поступать со слоями исходного документа (A), как располагать импортированные объекты на арене (B) и следует ли изменять размер арены в соответствии с размером изображения в PSD-файле (C).

## Чтобы импортировать содержимое из PSD-файла:

1. Выполните шаги 1–5 из первого упражнения в предыдущем разделе «Импорт растровых изображений». На шаге 2 укажите формат Photoshop и выберите файл, который хотите импортировать. После нажатия кнопки **Import** (Mac) или **Open** (Windows) открывается диалоговое окно PSD Import (рис. 14.13).
2. Для задания параметров импорта, применяемых к документу в целом, выполните одно из следующих действий;
  - из списка **Convert Layers To** (Преобразовать слои в) выберите метод преобразования. Чтобы воспроизвести структуру слоев исходного PSD-файла в виде слоев на временной шкале, выберите пункт **Flash Layers** (Слои Flash). Чтобы преобразовать слои PSD-файла в опорные кадры, выберите пункт **Keyframes** (Ключевые кадры);
  - если хотите, чтобы Flash помещал все элементы на арену, не меняя координат, отметьте флажок **Place layers at original position** (Помещать слои в исходные позиции). В противном случае Flash будет центрировать импортированные слои на арене (позиции импортированных элементов относительно друг друга сохраняются) (рис. 14.14);
  - чтобы изменить размер арены Flash-документа в соответствии с размерами изображения в PSD-файле, отметьте флажок **Set Stage size to same size as Photoshop canvas** (Сделать размер арены таким же, как размер холста Photoshop).

## Замечание о корректирующих слоях в Photoshop

Во Flash нет понятия корректирующего слоя, присутствующего в Photoshop. Если у выбранного слоя есть корректирующий, и вы хотите сохранить визуальный эффект, то импортируйте выбранный слой как растровое изображение. В противном случае Flash проигнорирует корректирующий слой при импорте.

3. Чтобы пометить, какие слои вы хотели бы импортировать, отметьте флажки слева от названий слоев; если слой не нужен, сбросьте соответствующий ему флажок (рис. 14.15). Если PSD-файл содержит группы слоев, то в списке они будут представлены иерархически. Щелкните по треугольничку слева от имени группы, чтобы развернуть или свернуть ее содержимое. Чтобы отметить (или сбросить) флажки для всех слоев в группе, отметьте (или сбросьте) флажок против самой группы.
4. Чтобы объединить слои при импорте, щелкните по именам объединяемых слоев, удерживая клавишу **Shift**, а затем нажмите кнопку **Merge Layers** (Объединить слои) (рис. 14.16).

Flash создаст на панели слоев новый слой, который по умолчанию называется Merged Bitmap. Под ним будут перечислены отдельные объединяемые слои. Все элементы, принадлежащие любому из этих слоев, объединяются и импортируются как одно растровое изображение.

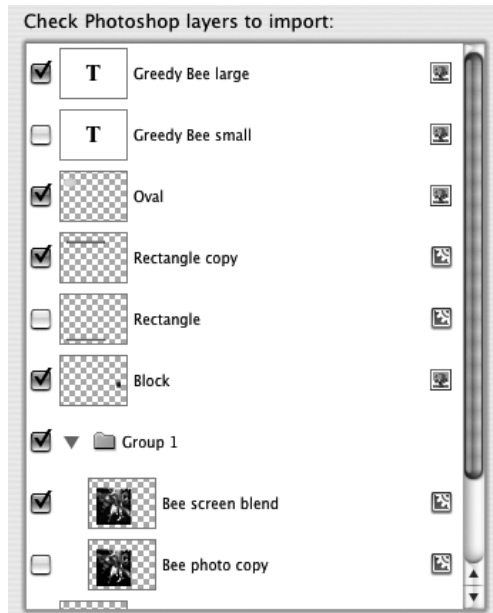


Рис. 14.15. Флажками отмечаются слои, предназначенные для импорта.

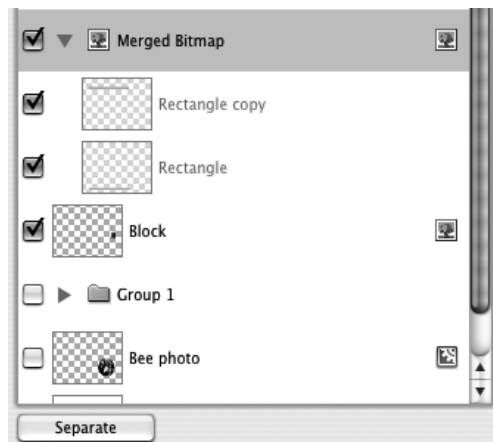
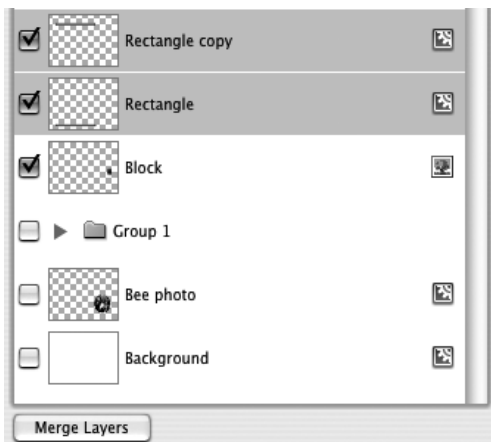
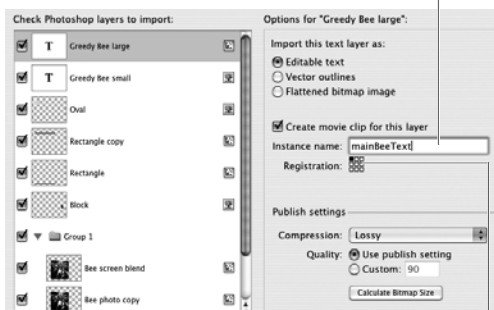


Рис. 14.16. Чтобы объединить слои при импорте, щелкните по именам объединяемых слоев с нажатой клавишей Shift (левый рисунок), а затем нажмите кнопку Merge Layers. Flash объединит слои. Исходные слои будут отображаться под объединенным с отступом (правый рисунок).

## Ввести имя экземпляра



Выбрать точку регистрации для этого клипа

**Рис. 14.17.** Выбор элемента на панели слоев открывает доступ к параметрам импорта этого слоя на панели Options. Если элемент можно импортировать в виде символа-клипа, то Flash отображает поля Instance Name (Имя экземпляра) и Registration (Регистрация). Задание нового положения точки регистрации отменяет то, что было задано в окне Preferences. Новая точка регистрации применяется только к ролику, созданному из выбранного слоя.



**Рис. 14.18.** В случае использования команды Import to Stage все выбранные слои импортируются и помещаются на арену (верхний рисунок). (Здесь параметры импорта заданы так, что PSD-слои воссоздаются на временной шкале Flash.) В библиотеку добавляется папка, в которой сохраняются активы, созданные при импорте PSD-файла (нижний рисунок).

5. Чтобы переопределить параметры, заданные в окне Preferences, при импорте отдельных слоев, щелкните по имени слоя и задайте другие параметры на правой панели (рис. 14.17).

6. После того как слои, подлежащие импорту, выбраны и параметры заданы, нажмите **ОК**. Flash импортирует все выбранные элементы, помещает их на арену, а соответствующие активы — в библиотеку (рис. 14.18).

**С**

Если исходный PSD-файл содержит композиции слоев (средство управления слоями для быстрого просмотра различных комбинаций видимых и скрытых слоев), то в верхней части окна PSD Import появляется список **Layer Comp** (Композиция слоев). После выбора из него варианта композиции Flash автоматически отмечает флажками видимые в этой композиции слои, а для всех остальных флажки сбрасывает.

**С**

Для выбора нескольких соседних PSD-слоев можно щелкнуть мышью с нажатой клавишей **Shift**; параметры, заданные в правой панели окна, будут применены ко всем выбранным элементам.

**С**

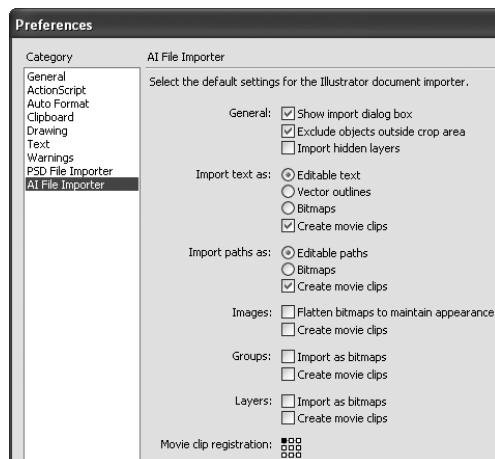
Во время импорта слои можно переименовать. Для этого дважды щелкните по имени слоя, после чего оно превратится в текстовое поле, где можно будет ввести новое имя. Для подтверждения нажмите **Enter**. Теперь при создании слоев и активов Flash будет использовать вновь введенное имя.

## Импорт файлов, созданных в программе Adobe Illustrator

При импорте графики из файла, созданного в программе Adobe Illustrator (AI), Flash автоматически использует параметры, заданные в окне Preferences. Один из них определяет, нужно ли открывать при импорте диалоговое окно AI Import. Если вы решите, что нужно, то сможете в нем просмотреть и изменить параметры для отдельных импортируемых элементов.

### Чтобы задать параметры импорта AI-файлов:

1. Из меню **Flash** (Mac) или **Edit** (Windows) выберите пункт **Preferences** (Установки). Открывается диалоговое окно Preferences.
2. Выберите категорию **AI File Importer** (Импортер AI-файлов). В правой части окна появляются параметры импорта (рис. 14.19).



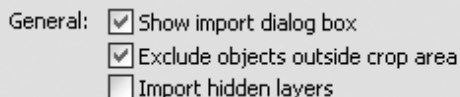
**Рис. 14.19.** Выберите категорию AI File Importer в диалоговом окне Preferences, чтобы задать параметры, принимаемые по умолчанию для импорта файлов Illustrator.

### Импорт на арену и импорт в библиотеку

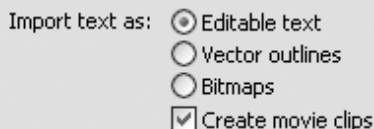
Файлы можно импортировать на арену или в библиотеку. В первом случае Flash размещает графическое содержимое на арене и при необходимости воссоздает иерархию слоев на главной временной шкале документа Flash. Кроме того, импортированное содержимое помещается в библиотеку.

При импорте в библиотеку импортированное содержимое помещается в библиотеку, но для файлов, созданных в программах Photoshop, Illustrator и Fireworks (в формате PNG), еще и создается включающий это содержимое символ-клип или графический символ. Этот символ помещается на арену, а иерархия слоев воссоздается в его временной шкале.

Отметим, что состав некоторых диалоговых окон слегка различается в зависимости от того, указан ли импорт на арену или в библиотеку. Например, если файлы, созданные в программах Illustrator или Photoshop, импортируются непосредственно в библиотеку, то не будет параметров, которые определяют, надо ли сохранять координаты объектов на арене и следует ли подстраивать размер арену под содержимое. Однако для импортированных объектов всегда сохраняется их взаимное расположение, а все импортированное изображение центрируется в символе-клипе.



**Рис. 14.20.** В разделе General диалогового окна AI Import задается, нужно ли открывать при импорте диалоговое окно. Можно указать, следует ли по умолчанию импортировать скрытые слои и объекты, лежащие вне мольберта AI.



**Рис. 14.21.** Параметры импорта текста определяют, нужно ли создавать из него редактируемые текстовые поля, редактируемые векторные формы или растровые изображения. Импортированный текст может быть помещен в символ-клип.

3. В разделе **General** (Общие) (рис. 14.20) выполните одно из следующих действий:

- чтобы Flash открывал перед началом импорта диалоговое окно AI Import (в котором можно изменить параметры импорта), отметьте флажок **Show Import dialog box** (Показать диалоговое окно «Импорт»). Если этот флажок сброшен, то файл импортируется с учетом параметров, заданных в окне Preferences без дополнительного уточнения;
- чтобы Flash не импортировал объекты, отсутствующие на мольберте (Artboard — эквивалент арены в Illustrator), отметьте флажок **Exclude objects outside crop area** (Исключить объекты вне области кадрирования);
- чтобы импортировать слои, которые в AI-файле скрыты, отметьте флажок **Import Hidden Layers** (Импортировать скрытые слои).

4. Для задания параметров импорта текста (рис. 14.21) выполните одно из следующих действий:

- для импорта текста в редактируемые текстовые поля Flash отметьте переключатель **Editable Text** (Редактируемый текст). (Если в Illustrator текст помечен как статический, вводимый или динамический, то такой же стиль будет применен и во Flash.)
- для импорта текста в виде набора редактируемых форм выберите переключатель **Vector Outlines** (Векторные контуры);
- для импорта текста в виде растровых изображений выберите переключатель **Bitmaps** (Растровые изображения);
- если хотите, чтобы Flash поместил импортированный текст в символ-клип, отметьте флажок **Create Movie Clips** (Создать фрагменты ролика).

5. Для задания параметров импорта путей (рис. 14.22) выполните одно из следующих действий:

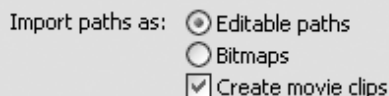
- для импорта редактируемых путей Безье выберите переключатель **Editable Paths** (Редактируемые пути);
- для импорта путей в виде растровых изображений выберите переключатель **Bitmaps** (Растровые изображения);
- если хотите, чтобы Flash поместил импортированный путь в символ-клип, отметьте флажок **Create Movie Clips** (Создать фрагменты ролика).

6. Для задания параметров импорта изображений, групп и слоев в соответствующих разделах (рис. 14.23) выполните одно из следующих действий:

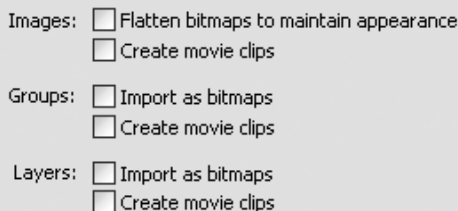
- чтобы сохранить внешний вид изображения, отметьте флажок **Flatten bitmaps to maintain appearance** (Свести в растровое изображение для сохранения внешнего вида) в разделе Images (Изображения);
- чтобы импортировать группы или слои в виде растровых изображений, отметьте флажок **Import As Bitmaps** (Импортировать как растровые изображения) в разделе Groups (Группы) или Layers (Слои);
- если хотите, чтобы Flash поместил каждое изображение, группу или символ в символ-клип, отметьте флажок **Create Movie Clips** (Создать фрагменты ролика) в соответствующем разделе.

7. Чтобы задать положение точки регистрации по умолчанию для символов-клипов, создаваемых в процессе импорта, щелкните по одному из девяти квадратиков в поле Movie clip registration (Расположение фрагмента ролика) (рис. 14.24).

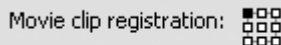
8. Нажмите **ОК**, чтобы сохранить заданные параметры и закрыть окно.



**Рис. 14.22.** Пути из файла, созданного Illustrator, можно импортировать как редактируемые кривые Безье или как растровые изображения.



**Рис. 14.23.** Изображения, группы и слои можно свести в растровое изображение или импортировать как символы-клипы с сохранением структуры слоев.

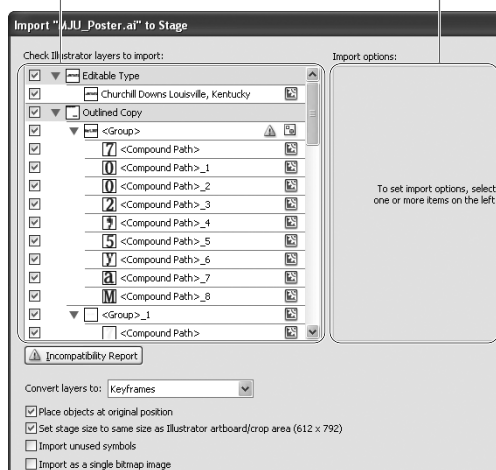


**Рис. 14.24.** Щелчок по одному из девяти квадратиков в окне AI Import Preferences устанавливает подразумеваемое положение точки регистрации для всех символов-клипов, создаваемых в процессе импорта AI-файла.

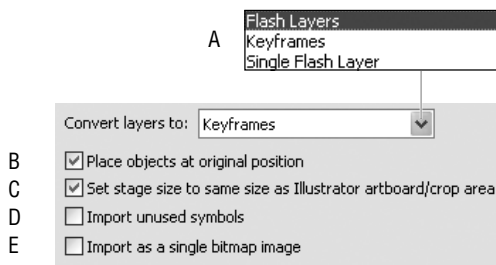


Панель слоев

Панель параметров



**Рис. 14.25.** На панели слоев в окне AI Import находится список, в котором представлены элементы, доступные для импорта. При выборе элемента из списка его характеристики отображаются на панели параметров справа. Параметры, относящиеся к импорту документа в целом, находятся в нижней части окна.



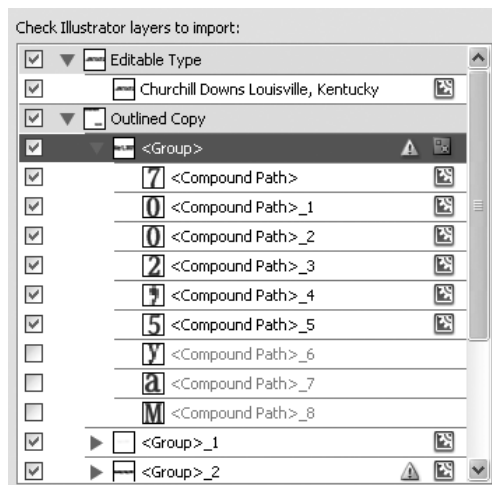
**Рис. 14.26.** В нижней части окна AI Import задаются параметры, применяемые к импорту документа в целом. Можно определить, как Flash должен поступать со слоями исходного документа (A), как располагать импортированные объекты на арене (B), следует ли изменять размер арены в соответствии с размером изображения в AI-файле (C) и нужно ли импортировать неиспользуемые символы (D). Имеется также флажок, позволяющий отменить все параметры для отдельных элементов и свести слои, отмеченные на панели Layers, в одно растровое изображение (E).

## Чтобы импортировать содержимое из AI-файла:

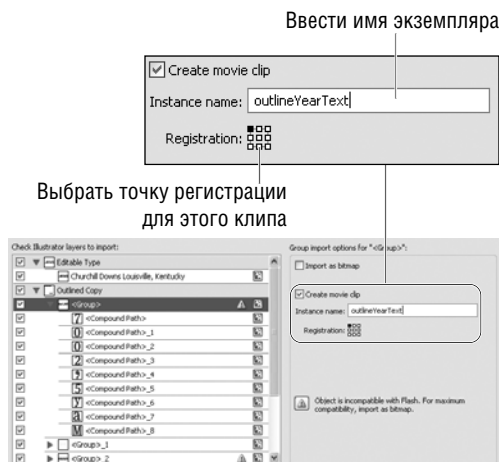
1. Выполните шаги 1–5 из первого упражнения «Импорт растровых изображений». На шаге 2 укажите формат Adobe Illustrator и выберите файл, который хотите импортировать. После нажатия кнопки **Import** (Mac) или **Open** (Windows) открывается диалоговое окно Importing External File (рис. 14.24). В нем находится индикатор выполнения и кнопка **Stop** (Mac) или **Cancel** (Windows), позволяющая прервать процедуру импорта. Если в окне Preferences вы указали, что нужно открывать диалоговое окно AI Import (см. предыдущее упражнение), то по завершении первого этапа импорта Flash откроет это окно (рис. 14.25). В противном случае содержимое AI-файла импортируется с применением параметров, заданных в окне Preferences.
2. Для задания параметров импорта, применяемых к документу в целом, выполните одно из следующих действий (рис. 14.26);

- из списка **Convert Layers To** (Преобразовать слои в) выберите метод преобразования. Чтобы воспроизвести структуру слоев исходного AI-файла в виде слоев на временной шкале, выберите пункт **Flash Layers** (Слои Flash). Чтобы преобразовать слои AI-файла в опорные кадры, выберите пункт **Keyframes** (Ключевые кадры). Если хотите, чтобы все импортированные элементы были помещены на временную шкалу, выберите пункт **Single Flash Layer** (Один слой Flash);
- если хотите, чтобы Flash помещал все элементы на арену, не меняя координат, отметьте флажок **Place objects at original position** (Помещать объекты в исходные позиции). В противном случае Flash будет центрировать импортированные слои на арене (позиции импортированных элементов относительно друг друга сохраняются);

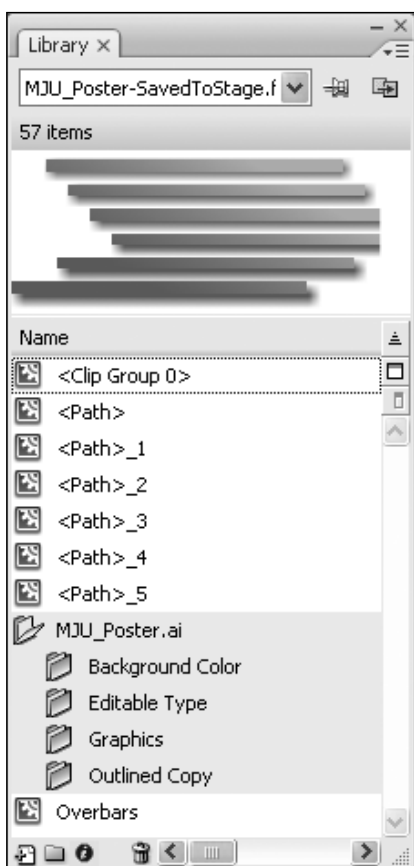
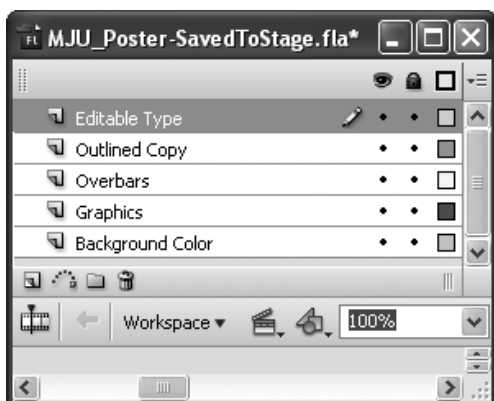
- чтобы изменить размер арены Flash-документа в соответствии с размерами изображения в AI-файле, отметьте флажок **Set Stage size to same size as Illustrator Artboard** (Сделать размер арены таким же, как размер мольберта Illustrator);
  - чтобы импортировать символы, которые есть в библиотеке AI-файла, но в данный момент не помещены на мольберт, отметьте флажок **Import Unused Symbols** (Импортировать неиспользуемые символы);
  - чтобы свести все выбранные слои в единое растровое изображение, отметьте флажок **Import as a single bitmap image** (Импортировать как единое растровое изображение).
3. Чтобы пометить, какие слои вы хотели бы импортировать, отметьте флажки слева от названий слоев; если слой не нужен, сбросьте соответствующий ему флажок (рис. 14.27). Слои, имеющиеся в AI-файле (и их подслои), представлены на панели Layers иерархически. Щелкните по треугольничку слева от имени слоя, чтобы развернуть или свернуть содержимое соответствующей ветки.
4. Чтобы переопределить параметры, заданные в окне Preferences, при импорте отдельных слоев, щелкните по имени слоя и задайте другие параметры на правой панели (рис. 14.28).



**Рис. 14.27.** На панели Layers определяется, что именно Flash должен импортировать. Если флажок слева от имени слоя отмечен, слой импортируется, иначе — нет.



**Рис. 14.28.** Выбор элемента на панели слоев открывает доступ к параметрам импорта этого слоя на панели Options. Если элемент можно импортировать в виде символа-клипа, то Flash отображает поля Instance Name (Имя экземпляра) и Registration (Регистрация). Задание нового положения точки регистрации отменяет то, что было задано в окне Preferences. Новая точка регистрации применяется только к ролику, созданному из выбранного слоя.



**Рис. 14.29.** Flash импортирует все элементы, выбранные на панели Layers, и помещает их на арену (здесь параметры импорта заданы так, что AI-слои воссоздаются на временной шкале Flash.) В библиотеку добавляется папка, в которой сохраняются активы, созданные при импорте AI-файла.

5. После того как слои, подлежащие импорту, выбраны и параметры заданы, нажмите **ОК**.

Flash импортирует все выбранные элементы, помещает их на арену, а соответствующие активы — в библиотеку (рис. 14.29).

**С**

Для выбора нескольких соседних AI-слоев можно щелкнуть мышью с нажатой клавишей **Shift**; параметры, заданные в правой панели окна, будут применены ко всем выбранным элементам.

**С**

Во время импорта слои можно переименовать. Для этого дважды щелкните по имени слоя, после чего оно превратится в текстовое поле, где можно будет ввести новое имя. Для подтверждения нажмите **Enter**. Теперь при создании слоев и активов Flash будет использовать вновь введенное имя.

**С**

Чтобы отметить или сбросить флажки для всех элементов в одном слое, сделайте это с флажком, относящимся к самому слою. Тот же прием работает для групповых слоев.

### Как быть с несовместимостью?

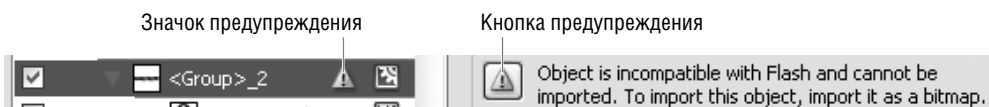
Хотя Adobe Illustrator CS3 и Flash CS3 стали более совместимыми, чем раньше, некоторые элементы все же не транслируются или транслируются плохо. Самая существенная несовместимость — пространство цветов — проистекает из различия в подходах к традиционному графическому дизайну и Web-дизайну. Художники, работающие в Illustrator, обычно пользуются пространством цветов CMYK, применяемом при подготовке печатных материалов. А Web-дизайнеры привыкли к пространству цветов RGB. При импорте AI-файла Flash может перевести CMYK-цвета в RGB, но результат иногда оказывается далеким от идеала. Несовместимости могут возникать также в случае, когда в исходном материале используются фильтры и наложения, не поддерживаемые Flash. В диалоговом окне импорта существует система предупреждений, которая извещает обо всех возникших несовместимостях и предлагает оптимальные параметры.

Чтобы ознакомиться с отчетом о несовместимостях для файла в целом, нажмите кнопку **Incompatibility Report** (Отчет о несовместимости). Flash откроет диалоговое окно, в котором перечислены потенциальные проблемы (рис. 14.30). Для изменения параметров импорта, так чтобы импортированный элемент выглядел максимально похоже на оригинал, отметьте флажок **Apply recommended import settings** (Применить рекомендованные параметры импорта) и нажмите кнопку **Close** (Закрыть). (Если проблема заключается в пространстве цветов, то в этом окне дается хороший совет: чтобы обеспечить наилучший перевод CMYK в RGB, преобразуйте файл в RGB в Illustrator.)

На панели Layers против каждого элемента, для которого обнаружена проблема, отображается значок предупреждения (желтый треугольник с восклицательным знаком). При выборе этого элемента такой же значок появляется на панели параметров (рис. 14.31). Щелкните по нему, чтобы открыть окно с описанием проблемы. Еще один вид несовместимости возникает, когда шрифты, используемые в AI-файле, недоступны для Flash. В отчете о несовместимости проблемы со шрифтами не отражены, но во время импорта Flash выводит предупреждение, предлагая вместо отсутствующего шрифта подставить шрифт по умолчанию или выбрать замену самостоятельно.



**Рис. 14.30.** Чтобы вывести перечень несовместимостей между AI-файлом и Flash, нажмите кнопку Incompatibility Report button (A). Flash открывает диалоговое окно, в котором описаны возникшие проблемы (B). Если вы хотите, чтобы Flash импортировал проблемные объекты в виде растровых изображений для сохранения исходного внешнего вида, то отметьте флажок Apply recommended import settings (C).



**Рис. 14.31.** Для тех элементов, для которых при трансляции из Illustrator во Flash возможны проблемы, на панели Layers отображается значок предупреждения. Выбрав такой элемент, вы получите общее описание проблемы на панели Options. Чтобы ознакомиться с деталями, нажмите кнопку Alert.



Рис. 14.32. В диалоговом окне Import Fireworks Document задаются параметры импорта файлов, созданных в программе Fireworks.



Рис. 14.33. Если флажок Import as a single flattened bitmap отмечен, то перед импортом во Flash все содержимое файла Fireworks сводится в единое растровое изображение.

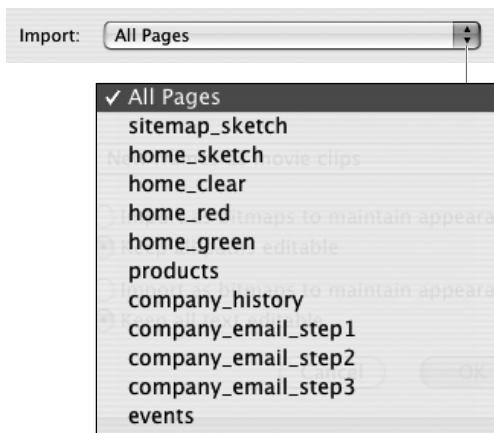


Рис. 14.34. В списке Import можно выбрать для импорта одну или все страницы из файла Fireworks.

## Импорт файлов, созданных в программе Fireworks

Во Flash CS3 возможностей импорта содержимого из файлов, созданных в Adobe Fireworks, больше, чем в предыдущих версиях.

### Чтобы импортировать содержимое из файла Fireworks в виде единого растрового изображения:

1. Выполните шаги 1–5 из первого упражнения «Импорт растровых изображений». На шаге 2 укажите формат PNG и выберите файл, который хотите импортировать. После нажатия кнопки **Import** (Mac) или **Open** (Windows) открывается диалоговое окно Import Fireworks Document (рис. 14.32).
2. Отметьте флажок **Import as a single flattened bitmap** (Импортировать как единое сведенное растровое изображение). В этом случае Flash деактивирует все остальные элементы управления импортом (рис. 14.33).
3. Нажмите **OK**.

Flash сводит все содержимое Fireworks в одно растровое изображение и помещает его в текущий опорный кадр на главной временной шкале и в библиотеку Flash-документа.

### Чтобы импортировать содержимое из файла Fireworks в виде редактируемого контента:

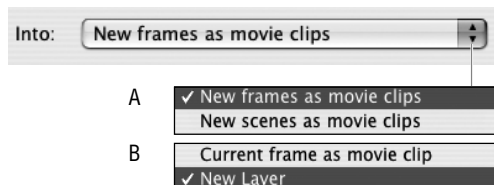
1. Выполните шаг 1 из предыдущего упражнения. Флажок **Import as a single flattened bitmap** должен быть сброшен.
2. Для выбора подлежащих импорту страниц выполните одно из следующих действий:
  - чтобы импортировать содержимое одной страницы, выберите эту страницу из списка Import;
  - чтобы импортировать все содержимое документа, выберите из списка Import пункт All Pages (Все страницы) (рис. 14.34).

3. Чтобы задать, как Flash должен размещать импортированное содержимое на временной шкале, выберите из списка **Into** (В) один из следующих пунктов (рис. 14.35):

- **New Frames As Movie Clips** (Новые кадры как фрагменты ролика). При импорте нескольких страниц Flash создает символы-клипы для каждой страницы Fireworks и помещает их экземпляры в отдельные опорные кадры на главной временной шкале документа.
- **New Scenes As Movie Clips** (Новые сцены как фрагменты ролика). При импорте нескольких страниц Flash создает символы-клипы для каждой страницы Fireworks и помещает их экземпляры в начальные опорные кадры отдельных сцен на главной временной шкале.
- **Current Frame As Movie Clip** (Текущий кадр как фрагмент ролика). При импорте одной страницы Flash создает символ-клип и помещает его экземпляр в текущий опорный кадр, выбранный на временной шкале.
- **New Layer** (Новый слой). При импорте одной страницы Flash добавляет новый слой и помещает в него импортированное содержимое в виде сведенного растрового изображения.

4. Чтобы указать, можно ли редактировать импортированные элементы, в разделах **Objects** (Объекты) и **Text** (Текст) выполните одно из следующих действий:

- для импорта объектов или текста в виде растровых изображений выберите переключатель **Import as bitmaps...** (Импортировать как растровые изображения) в соответствующем разделе;
- для импорта редактируемых объектов или текста выберите переключатель **Keep all paths/text editable** (Оставить все пути/текст редактируемыми) в соответствующем разделе. В этом режиме Flash создает объекты и текст, которые можно редактировать.

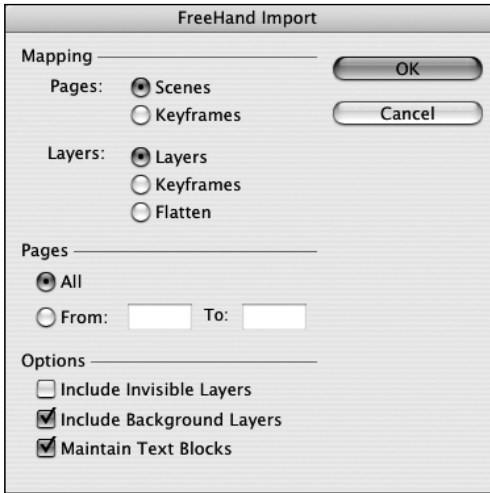


**Рис. 14.35.** Чтобы определить, как следует размещать импортированное содержимое на временной шкале, выберите параметр из списка **Into** в диалоговом окне **Import Fireworks Document**. Состав этого списка зависит от того, импортируется ли из Fireworks несколько страниц (A) или только одна (B).

5. Нажмите **ОК**.

Flash импортирует содержимое файла Fireworks в соответствии с заданными параметрами, помещая растровые изображения или экземпляры символов-клипов на арену. Кроме того, в библиотеке создается папка **Fireworks Objects**, в которой хранятся импортированные активы.

**С** Если вы импортируете объект Fireworks как сведенное растровое изображение, то он сохраняет связь с исходным файлом. Укажите растровое изображение в библиотеке и из меню панели выберите команду **Edit in Fireworks** (Редактировать в Fireworks). Flash откроет исходный файл в программе Fireworks. По завершении редактирования обновленный файл будет импортирован повторно.



**Рис. 14.36.** При импорте файла, созданного в программе FreeHand версии от 7 до 11 (MX), можно более точно управлять тем, как импортированные элементы будут выглядеть в документе Flash.

## Импорт из программы FreeHand

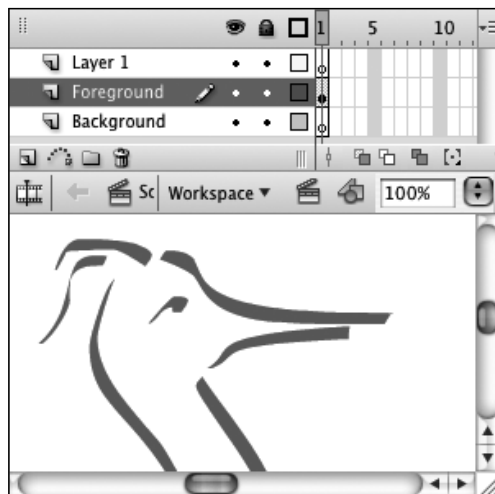
Чтобы импортировать содержимое файла, созданного в программе Adobe FreeHand, можно воспользоваться операциями копирования и вставки или перетащить содержимое из открытого файла FreeHand прямо на арену Flash. Однако для контента, созданного в FreeHand версии 7 или старше, команда **File** ⇒ **Import** ⇒ **Import to Stage** (или **Import to Library**) позволяет более точно контролировать процесс импорта. При ее выполнении открывается диалоговое окно FreeHand Import, в котором можно задать параметры.

### Чтобы импортировать файл FreeHand на арену:

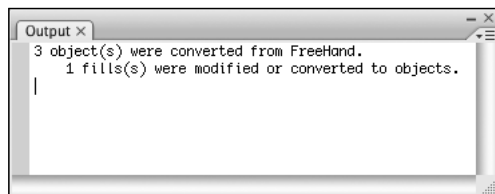
1. Откройте документ Flash.
2. Выполните шаги 1–5 из первого упражнения «Импорт растровых изображений». На шаге 2 укажите формат FreeHand и выберите файл, который хотите импортировать. После нажатия кнопки **Import** (Mac) или **Open** (Windows) открывается диалоговое окно FreeHand Import (рис. 14.36).
3. В разделе Mapping (Отображение) выполните любое из следующих действий:
  - для создания новой сцены из каждой страницы FreeHand установите переключатель **Pages** (Страницы) в положение Scenes (Сцены);
  - для создания нового опорного кадра из каждой страницы FreeHand установите переключатель **Pages** в положение Keyframes (Ключевые кадры);
  - для создания нового слоя из каждого слоя FreeHand установите переключатель **Layers** (Слои) в положение Layers;
  - для создания опорного кадра из каждого слоя FreeHand установите переключатель **Layers** в положение Keyframes;
  - для объединения нескольких слоев FreeHand в один слой Flash установите переключатель **Layers** в положение Flatten (Свести).

4. Чтобы выбрать подлежащие импорту страницы, в разделе Pages (Страницы) выполните одно из следующих действий:
  - для импорта всего файла FreeHand отметьте переключатель **All** (Все);
  - для импорта диапазона страниц отметьте переключатель **From/To** (С/по) и введите номера первой и последней страниц.
5. В разделе Options (Дополнительно) выполните любое из следующих действий:
  - чтобы импортировать все скрытые слои из файла FreeHand, отметьте флажок **Include Invisible Layers** (Включить невидимые слои);
  - чтобы импортировать фоновый слой из файла FreeHand, отметьте флажок **Include Background Layers** (Включить фоновые слои);
  - чтобы Flash преобразовывал редактируемые текстовые блоки FreeHand в редактируемые же блоки Flash, отметьте флажок **Maintain Text Blocks** (Сохранять текстовые блоки). В противном случае символы тексты импортируются как сгруппированные формы.
6. Нажмите **ОК**.

Открывается диалоговое окно Importing External File, в котором есть кнопка **Stop** (Mac) или **Cancel** (Windows) для отмены операции. Flash импортирует графическое содержимое из файла FreeHand и помещает его на арену, создавая на главной временной шкале слои и опорные кадры в соответствии с заданными параметрами (рис. 14.37). На панели вывода отображаются сообщения о том, сколько было импортировано или создано объектов (рис. 14.38). Для просмотра этой информации откройте панель вывода командой **Window ⇒ Output** (Окно ⇒ Вывод).



**Рис. 14.37.** Flash импортирует файлы FreeHand в соответствии с параметрами, заданными в диалоговом окне FreeHand Import. В данном случае было сказано включать фоновый слой.



**Рис. 14.38.** При импорте файлов FreeHand Flash сообщает о том, сколько объектов содержал файл FreeHand и сколько объектов Flash было создано в процессе импорта. Эта информация находится на панели вывода.

**C**

В версиях Flash младше 8, если импортированный из FreeHand файл содержал перекрывающиеся формы в одном слое, то во Flash они сегментировали друг друга. Flash CS3 импортирует формы FreeHand как объекты-рисунки, поэтому нежелательного сегментирования не возникает.

**C**

Если импортируется файл FreeHand, содержащий объекты с заливкой, к которой применен фильтр Transparent Lens, то Flash устанавливает для импортированных объектов прозрачность, чтобы воспроизвести нужный эффект.



# ДОБАВЛЕНИЕ ЗВУКОВОГО СОПРОВОЖДЕНИЯ

---

15

Поразительно, как много может передать классический немой фильм только с помощью изображения и текста. Увы, их время уже ушло. Звуковое сопровождение стало неотъемлемой частью современных Web-сайтов. Программа Adobe Flash CS3 Professional позволяет включить в проект звук либо в виде непрерывного фонового сопровождения, либо как синхронизированный элемент, сопоставленный конкретной части действия, скажем, аплодисменты.

Один из способов добавить звук во Flash-ролик состоит в том, чтобы импортировать аудиоклип в библиотеку и впоследствии присоединить его к опорному кадру. Обратиться к аудиоклипам и управлять синхронизацией позволяет вкладка свойств кадра в инспекторе свойств. Добавить звук можно также из ActionScript, но эта тема выходит за рамки настоящей книги.

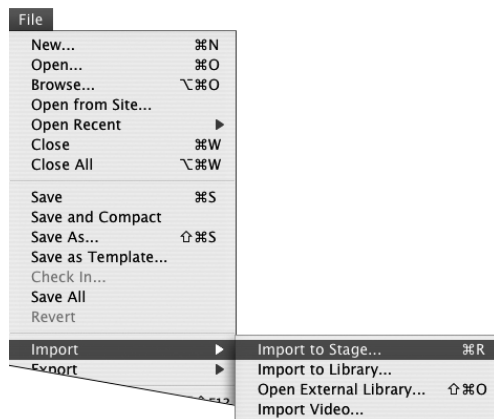
Во Flash есть ограниченные средства редактирования звука. Можно обрезать концы клипа или отрегулировать громкость, но прочие операции следует выполнять во внешних программах. При публикации ролика обращайтесь внимание на частоту дискретизации и уровень сжатия аудио, чтобы соблюсти баланс между качеством звукового сопровождения и размером файла. Подробнее на эту тему мы будем говорить в главе 17.

## Импорт звука

Процедура импорта звука напоминает импорт растровых или иных изображений. Можно воспользоваться командами **File ⇒ Import ⇒ Import to Stage** (Файл ⇒ Импорт ⇒ Импортировать на арену) или **File ⇒ Import ⇒ Import to Library** (Файл ⇒ Импорт ⇒ Импортировать в библиотеку). В любом случае звуковой файл включается в библиотеку текущего документа. Но ни одна из этих команд не помещает звук в кадр. Для этого вы должны явно перетащить копию звукового файла с библиотечной панели в нужный кадр.

### Чтобы импортировать звуковой файл:

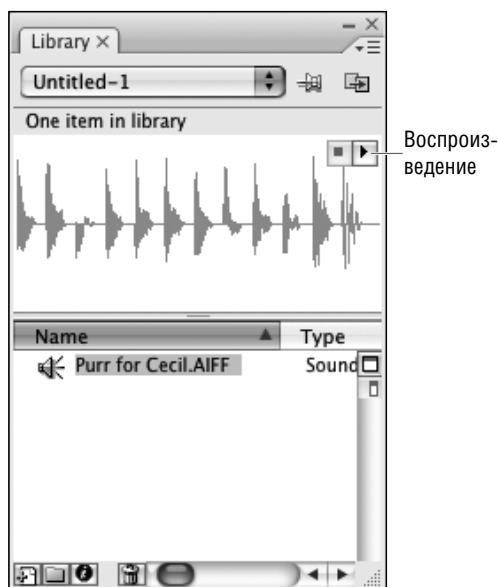
1. Откройте файл, в который хотите добавить звуковое сопровождение.
2. Выполните команду **File ⇒ Import ⇒ Import to Stage** или нажмите комбинацию клавиш **⌘-R** (Mac) либо **Ctrl-R** (Windows) (рис. 15.1). Годится также команда **File ⇒ Import ⇒ Import to Library**. Появляется стандартный диалог импорта файла (рис. 15.2).
3. Из списка **Enable** (Mac) или **Files of Type** (Windows) выберите формат звукового файла. Чтобы видеть файлы в любом формате, выберите пункт **All Sound Formats** (Все звуковые форматы).
4. Найдите нужный файл в своем компьютере.
5. Выберите этот файл.



**Рис. 15.1.** Для включения в документ звукового сопровождения выполните команду **File ⇒ Import ⇒ Import to Stage** (или **Import to Library**).



**Рис. 15.2.** Диалоговое окно **Import** позволяет импортировать звуковые файлы в документ Flash. Выберите из списка формат файла, подходящий для вашей платформы. При желании можете просмотреть звуковые файлы, записанные в любом формате.



**Рис. 15.3.** Flash хранит звуковые файлы в библиотеке. В области предварительного просмотра отображаются волновые формы. Чтобы услышать звук, нажмите кнопку Play.

- Нажмите кнопку **Import** (Mac) или **Open** (Windows). Flash импортирует выбранный звуковой файл, помещая его в библиотеку. В области предварительного просмотра отображаются хранящиеся в нем волновые формы (рис. 15.3). Звуковые файлы бывают весьма объемными, поэтому в процессе импорта появляется диалоговое окно Working с индикатором выполнения. Прервать импорт можно, нажав кнопку **Stop** (Mac) или **Cancel** (Windows).

**C** В случае импорта графики команда **File** ⇒ **Import** ⇒ **Import to Stage** помещает экземпляр на арену в выбранный опорный кадр. Для звуковых файлов это не так. Импортированный файл копируется только в библиотеку, а поместить его в кадр должны вы сами.

**C** Звуковой файл можно прослушать, и не помещая его в ролик. Выберите аудиоклип на библиотечной панели. Flash отображает в области предварительного просмотра волновые формы. Чтобы услышать звук, нажмите кнопку **Play** (Воспроизведение).

### Какие звуковые форматы поддерживает Flash?

Flash умеет работать только с дискретизированным звуком, который либо был записан цифровым способом, либо конвертирован в цифровой формат. Поддерживается формат AIFF для Mac OS, формат WAV для Windows и формат MP3 для обеих платформ. Кроме того, при наличии на том же компьютере программы QuickTime версии 4 или старше Flash CS3 может импортировать ролики QuickTime, содержащие только звук и AU-файлы, «родные» для платформы Sun. В этом случае пользователи Mac могут также импортировать файлы в форматах WAV, Sound Designer II и System 7, а пользователи Windows — файлы в формате AIFF. Любые звуковые файлы, импортированные или скопированные в документ Flash, помещаются в библиотеку.

## Добавление звука в кадры

Поместить звук в опорный кадр можно точно так же, как символ или растровое изображение. Достаточно выбрать кадр и перетащить на арену копию аудиоролика с открытой библиотечной панели (этого или другого документа). Можно также выбрать аудиоролик из списка **Sound** (Звук) на вкладке свойств кадра в инспекторе свойств. В этом списке представлены все аудиоролики из библиотеки текущего документа.

### Чтобы добавить звук в опорный кадр:

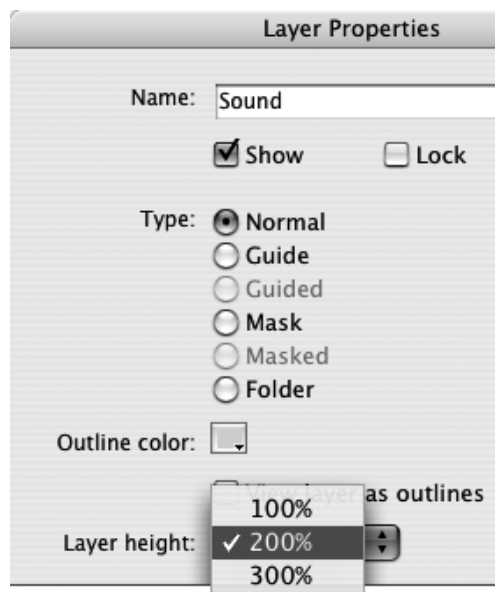
1. Откройте документ, в который собираетесь добавить звук.

Анимация игры в пинг-понг, созданная в главе 11, вполне подойдет. Этот документ содержит четыре опорных кадра, в которых мячик соприкасается с ракеткой. Наличие звукового сопровождения сделало бы контакт более реалистичным. Звук может быть настоящим (*чпоньк*) или неожиданным и потому смешным (скажем, удар колокола).

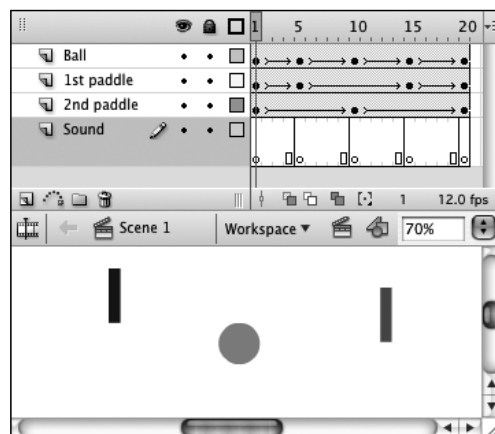
2. Добавьте в документ новый слой специально для звуков, назовите его **Sound** и задайте высоту 200% (рис. 15.4). Подробнее о подготовке звукового слоя см. врезку «Организация звукового сопровождения».
3. Выберите на временной шкале слой **Sound** и добавьте опорные кадры в позиции 5, 10, 15 и 20.

Эти четыре кадра соответствуют тем кадрам анимации, в которых мячик соприкасается с ракеткой (рис. 15.5).

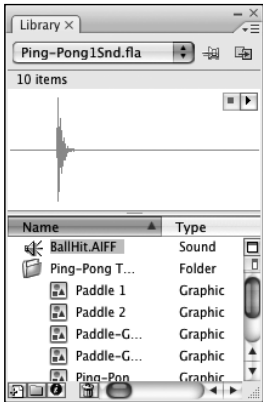
4. Импортируйте файл, содержащий звук, который вы хотите услышать при ударе ракеткой по мячу.



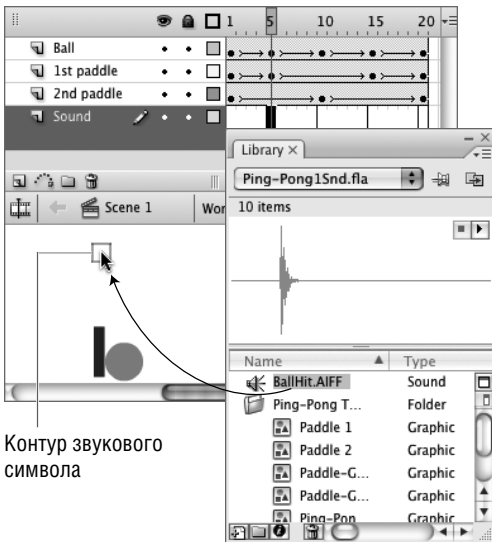
**Рис. 15.4.** Звуковые слои лучше делать побольше, чтобы лучше видеть волновые формы. Откройте окно свойств слоя, дважды щелкнув по значку слоя на временной шкале, и выберите из списка **Layer Height** значение 200% или 300%.



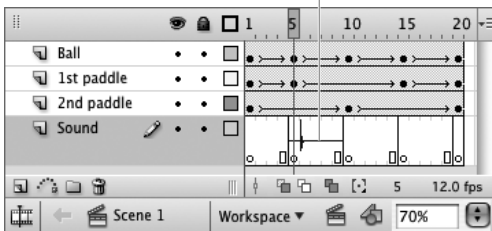
**Рис. 15.5.** Создайте отдельный слой для звуков. В этом слое добавьте опорные кадры в те позиции, в которых должен быть слышен звук. В данном случае опорные кадры в слое **Sound** соответствуют тем опорным кадрам в других слоях, в которых мячик ударяется о ракетку.



**Рис. 15.6.** Импортированный звуковой файл появляется на библиотечной панели. Выберите его и перетащите в опорный кадр.



Волновая форма звука, ассоциированного с опорным кадром 5



**Рис. 15.7.** В процессе перетаскивания звукового файла с библиотечной панели на арену отображается прямоугольный контур (верхний рисунок). Никаких визуальных признаков присутствия звука на арене нет, но на временной шкале отображаются волновые формы (нижний рисунок).

5. Выберите на временной шкале опорный кадр 5 в слое Sound. Это первый кадр, в котором мячик ударяется о ракетку.
6. Перейдите на библиотечную панель и выберите свой звуковой файл (рис. 15.6). В области предварительного просмотра отображаются волновые формы.
7. Перетащите копию звукового файла с библиотечной панели на арену.  
Хотя никаких визуальных признаков присутствия звука на арене нет, перетаскивать на нее звуковой файл все равно нужно. В процессе перетаскивания отображается прямоугольный контур. Когда вы отпустите кнопку мыши, Flash добавит звук в выбранный опорный кадр и отобразит волновые формы в этом кадре и всех ассоциированных с ним промежуточных кадрах (рис. 15.7).
8. Выберите на временной шкале опорный кадр 10 в слое Sound. Это второй кадр, в котором мячик ударяется о ракетку.
9. Перейдите на вкладку свойств кадра в инспекторе свойств.

10. Выберите свой звуковой файл из списка **Sound** (Звук).

В списке **Sound** на вкладке свойств кадра в инспекторе свойств перечислены все звуковые файлы, хранящиеся в библиотеке документа (рис. 15.8). Поэтому необязательно перетаскивать копию файла всякий раз, как нужно добавить звук в опорный кадр.

Больше никаких параметров на вкладке свойств кадра пока изменять не надо. Мы еще поговорим о них в последующих упражнениях.

11. Повторите шаги 6 и 7 (или 8, 9 и 10) для опорных кадров 15 и 20.

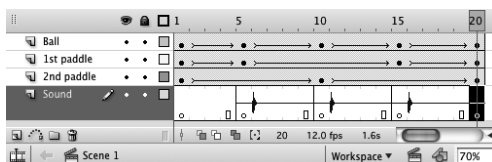
После добавления звука во все четыре опорных кадра ролик можно протестировать (рис. 15.9). При каждом ударе мячика о ракетку Flash воспроизводит назначенный звуковой файл, что придает игре дополнительный реализм.

## С

Если вы хотите включить в свой ролик звуковое сопровождение, но не располагаете звукозаписывающей аппаратурой, к вашим услугам множество бесплатных звуковых файлов. Можете приобрести компакт-диск, на котором они записаны, или загрузить с какого-нибудь сайта. Убедитесь только, что, включая звук в свой проект, вы не нарушаете ничьих авторских прав.



**Рис. 15.8.** Список Sound на вкладке свойств кадра в инспекторе свойств содержит все звуковые файлы, хранящиеся в библиотеке открытого документа. Выбранный из него звук будет добавлен в текущий опорный кадр.

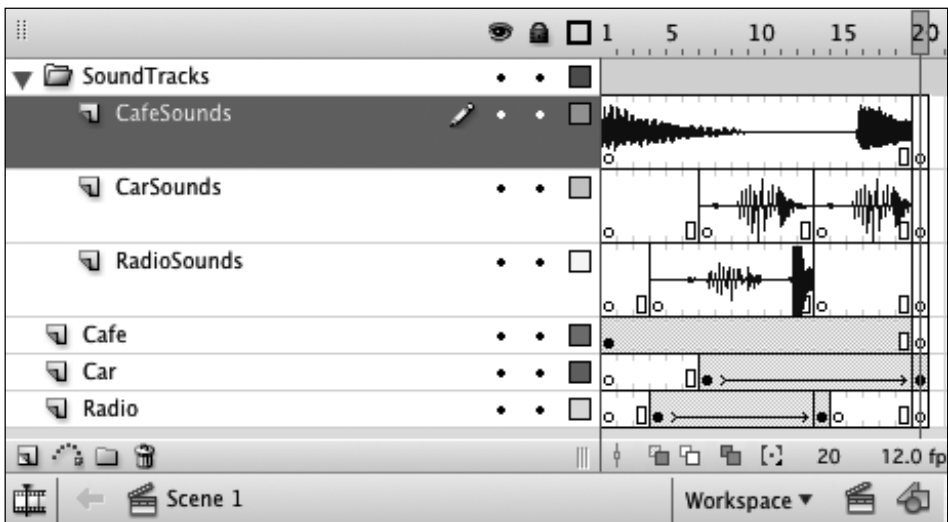


**Рис. 15.9.** Если в каком-то месте ролика должен быть слышен звук, добавьте его в соответствующий опорный кадр в слое Sound. В последовательности этого кадра будут отображаться волновые формы. В одном кадре вы много не увидите; так в кадре 20 видна лишь начальная линия.

### Организация звукового сопровождения

Ничто не мешает помещать звуковые файлы в слои, где уже есть другое содержимое, но работать с документом (в частности, находить звуковые файлы для обновления и редактирования) будет проще, если помещать их в отдельные слои, предназначенные специально для звуковой дорожки. Ниже приведены некоторые рекомендации по организации слоев для звукового сопровождения:

- Давайте таким слоям имена, напоминающие об их содержимом. Подробную информацию о работе со слоями см. в главе 6.
- Помещайте звуковые слои в начало или в конец временной шкалы, чтобы их проще было найти. Место слоя в стопке не влияет на воспроизведение звука в ролике.
- Создайте для звуковых слоев отдельную папку. Flash Player версий 8 и 9 поддерживает до 32 одновременно воспроизводимых звуковых файлов; в предыдущих версиях их могло быть не более 8. (О публикации ролика для Flash-плеера см. главу 17.) Если помещать каждый звуковой файл в отдельный слой, то придется обо всех помнить. Если таких слоев оказывается много, то лучше создать папку SoundTracks; тогда управляться с ними будет гораздо легче (рис. 15.10).
- Увеличьте высоту звуковых слоев, чтобы были лучше видны волновые формы (графическое представление звука). Чтобы открыть окно свойств слоя, выполните команду **Modify** ⇒ **Timeline** ⇒ **Layer Properties** или дважды щелкните по значку выбранного слоя. Чтобы увеличить высоту слоя, выберите из списка Layer Height значение 200% или 300%. Нажмите **OK**.
- После того как звуковые файлы помещены в слой, заблокируйте его, чтобы случайно не добавить туда же графику. Для этого щелкните по маркеру в колонке с изображением замка в заголовке.



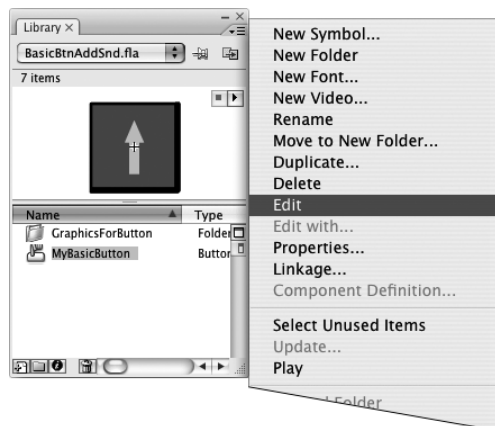
**Рис. 15.10.** Хранить звуковые файлы лучше в специальных слоях, отдельно от графики и действий. Для более эффективной организации звуковых слоев поместите их в отдельную папку. Увеличьте высоту звукового слоя, чтобы оставить больше места для отображения волновых форм.

## Добавление звука к кнопкам

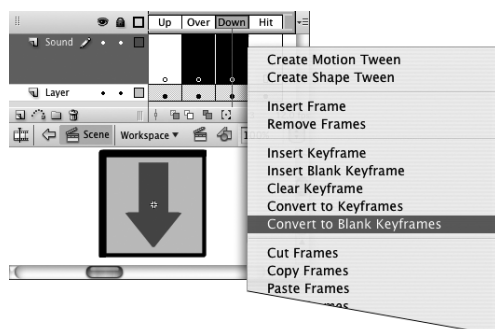
Акустическая обратная связь помогает зрителям работать с кнопками в вашем ролике. Если кнопка соответствует какому-то реальному объекту, то для усиления эффекта присутствия можно добавить звук щелчка в ее кадр Down. Для более причудливых кнопок, а также кнопок, замаскированных под часть сцены, можно включить звук в кадр Over, чтобы пользователь понимал, что наткнулся на активную область.

### Чтобы наделить кнопку акустической обратной связью:

1. Откройте документ, содержащий символ-кнопку, к которому собираетесь добавить звук. (О работе с символами-кнопками см. главу 12.)
2. Откройте библиотечную панель (команда **Window ⇨ Library**) и выберите нужный символ-кнопку.
3. В меню библиотечной панели выберите команду **Edit** (рис. 15.11). Flash открывает кнопку в режиме редактирования символа.
4. Добавьте на временную шкалу символа-кнопки новый слой (нажмите кнопку **Add Layer**) и назовите его **Sound**.
5. Выберите в слое **Sound** кадры **Over** и **Down**, откройте контекстное меню и выберите из него команду **Convert to Blank Keyframes** (рис. 15.12).



**Рис. 15.11.** Чтобы отредактировать символ, выберите его на библиотечной панели и выполните команду **Edit** из меню панели.



**Рис. 15.12.** Добавьте новый слой для звуков, ассоциированных с символом-кнопкой. В этом слое создайте опорные кадры, соответствующие тем состояниям кнопки, с которыми планируете связать звук.





## Использование звуков, синхронизированных с событиями

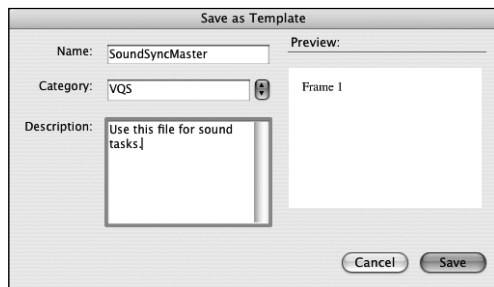
На вкладке свойств кадра в инспекторе свойств имеется раскрывающийся список Sync (Синхр.). Это свойство определяет, как Flash синхронизирует звуки в ролике. Оно может принимать любое из четырех значений: Event (Событие), Start (Начать), Stop (Остановить) и Stream (Поток). По умолчанию принимается значение Event. Звуки, синхронизированные с событиями, могут воспроизводиться независимо от главной временной шкалы. Flash начинает проигрывать такой звук, когда доходит до содержащего его кадра. Воспроизведение заканчивается по достижении конца аудиоклипа или при получении команды прекратить воспроизведение этого или всех вообще звуков. Продолжительный аудиоклип может звучать и после того, как был показан последний кадр ролика. Если ролик зациклен, то при каждом показе кадра, в котором есть синхронизированный с событиями звук, он начинает проигрываться заново. Чтобы понять, как работает синхронизация, полезно поработать с файлом, опорные кадры которого содержат идентифицирующий текст.

### Чтобы подготовить файл для тестирования звука:

1. Создайте документ, содержащий три слоя и 20 кадров.
2. Назовите слои Objects, Sound 1 и Sound 2.
3. В каждом слое вставьте опорные кадры в позиции 1, 5, 10, 15 и 20.
4. В каждый опорный кадр в слое Objects поместите какой-нибудь идентифицирующий текст.
5. Импортируйте в библиотеку несколько звуковых файлов разной длины.

Для этого примера мы взяли музыкальную гамму продолжительностью 15,8 с, звук падающей капли, мелодический пассаж и несколько ритмичных звуков.

6. Сохраните документ в качестве шаблона для последующего использования в этой главе и назовите его SoundSyncMaster (рис. 15.14).



**Рис. 15.14.** Подготовив тестовый файл, выполните команду File ⇒ Save As Template и в диалоговом окне Save As Template сохраните файл в виде шаблона с именем SoundSyncMaster. После создания документа-шаблона закройте его. Подробные инструкции по сохранению документов см. в главе 1.

### Независимые и синхронизированные звуки

Несинхронизированный аудиоклип воспроизводится независимо от кадров ролика и может продолжать звучать даже после того, как точка воспроизведения достигла конца ролика. Flash начинает воспроизведение такого клипа, когда входит в конкретный кадр, и, начиная с этого момент, проигрывает его независимо. На компьютере одного зрителя клип может звучать на протяжении десяти кадров, а на более медленной машине завершится, когда будет показано всего пять кадров.

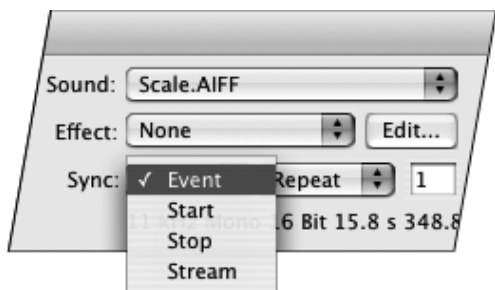
Flash может также синхронизировать аудиоклип с конкретными кадрами. В этом случае звуковой поток разбивается на мелкие куски, каждый из которых присоединяется к некоторому кадру. Для такого потокового аудио Flash принудительно синхронизирует анимацию со звуком. Если компьютер медленный, то частота кадров искусственно снижается, чтобы действие не отставало от звукового сопровождения.



**Рис. 15.15.** Выберите опорный кадр, к которому хотите присоединить звук. Параметры на вкладке свойств кадра применяются именно к этому кадру.



**Рис. 15.16.** На вкладке свойств кадра в инспекторе свойств выберите звуковой файл из списка Sound.



**Рис. 15.17.** В списке Sync выберите значение Event, чтобы звук начинался в выбранном кадре и продолжался до естественного завершения без синхронизации с последующими кадрами ролика.

### Чтобы синхронизировать звук с событием:

1. Откройте новый документ по шаблону SoundSyncMaster, созданному в предыдущем упражнении.
2. Выберите на временной шкале опорный кадр 5 в слое Sound 1 (рис. 15.15).
3. На вкладке свойств кадра выберите из списка **Sound** (Звук) протяженный аудиоклип, например, Scale.AIFF, который длится 15.8 с (рис. 15.16).
4. В списке **Sync** выберите значение Event (Событие) (рис. 15.17).

Звуковой файл Scale.AIFF присоединяется к опорному кадру 5 в слое Sound 1.

5. Поместите точку воспроизведения в опорный кадр 1 и воспроизведите ролик (команда **Control** ⇒ **Play**).

При стандартной частоте 12 кадров/с аудиоклип продолжительностью 15.8 с продолжает звучать после достижения конца ролика.

### С

Чтобы лучше понять, как Flash обрабатывает синхронизированные с событием звуки, выполните команду **Control** ⇒ **Loop Playback** (Управление ⇒ Цикл воспроизведения). Снова запустите ролик и дайте ему пару раз прокрутиться. При каждом прохождении через пятый кадр запускается новый экземпляр аудиоклипа, так что слышна не одна восходящая гамма, а дурная какофония. Если остановить воспроизведение, то каждый аудиоклип доигрывается до конца.

### Чтобы воспроизвести перекрывающиеся экземпляры одного и того же звукового файла:

1. Продолжаем работу с файлом из предыдущего упражнения. Чтобы начать воспроизведение того же аудиоклипа в другом месте ролика, выполните одно из следующих действий:

- выберите опорный кадр 15 в слое Sound 1;
- выберите опорный кадр 15 в слое Sound 2;

Поскольку Flash запускает воспроизведение звука, синхронизированного с событием, даже если предыдущий экземпляр еще продолжает звучать, то у вас есть выбор: включить второй экземпляр в тот же самый или в другой слой.

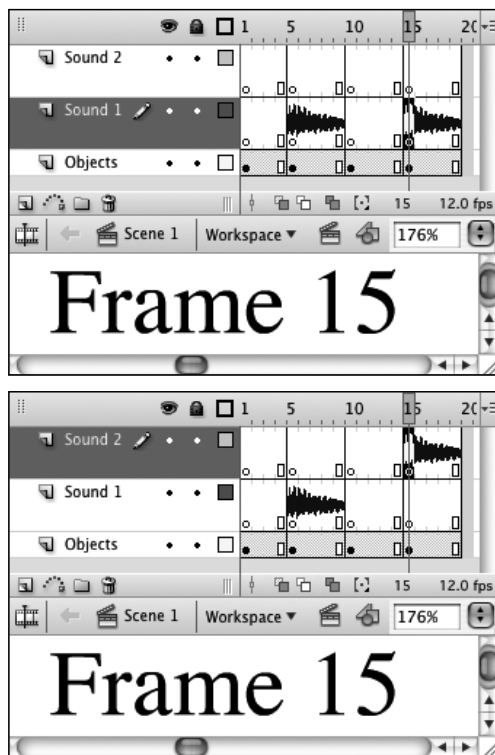
2. На вкладке свойств кадра выберите из списка **Sound** тот же самый файл (Scale.AIFF).
3. В списке **Sync** выберите значение Event.

Звуковой файл Scale.AIFF присоединяется к кадру 15 в выбранном вами слое (рис. 15.18).

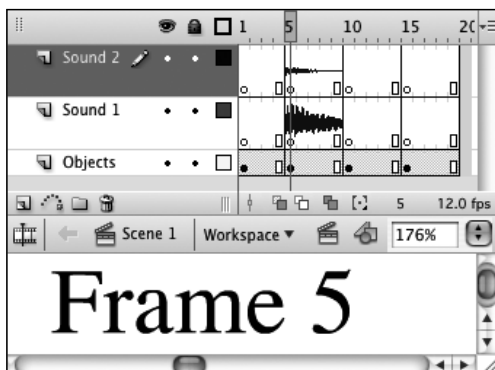
4. Поместите точку воспроизведения в кадр 1 и воспроизведите ролик один раз.

По достижении опорного кадра, начинает звучать файл Scale.AIFF. По достижении кадра 15 запускается второй экземпляр Scale.AIFF, и оба продолжают звучать вместе (слышится два голоса). Когда первый экземпляр завершится, останется только один голос. В пределах одного слоя кадр может содержать не более одного звукового файла. Если вы хотите, чтобы Flash начал воспроизведение разных звуков в одной и той же точке ролика, то файлы придется поместить в разные слои.

5. Сохраните этот файл для следующего упражнения, назовите его `OverlapSnds fla`.



**Рис. 15.18.** Можно добавить второй экземпляр звука и воспроизвести его поверх первого. Звуки, синхронизированные с событиями, воспроизводятся независимо от временной шкалы, поэтому второй экземпляр может находиться в том же слое, что и первый (верхний рисунок). Но можно поместить его и в другой слой (верхний рисунок).



**Рис. 15.19.** Чтобы два аудиоклипа звучали одновременно, необходимо поместить их в один и тот же опорный кадр (например, кадр 5), но в разные слои.

### Чтобы одновременно запустить разные звуковые файлы:

1. Откройте новый документ по ранее созданному шаблону SoundSyncMaster .
2. Выберите на временной шкале опорный кадр 5 в слое Sound 1.
3. На вкладке свойств кадра выберите из списка **Sound** файл Scale.AIFF.
4. В списке **Sync** выберите значение Event.
5. На временной шкале выберите опорный кадр 5 в слое Sound 2.
6. На вкладке свойств кадра выберите из списка **Sound** другой звуковой файл (Melody.AIFF).  
Можно также импортировать еще какой-нибудь звуковой файл в библиотеку или открыть библиотечную панель другого ролика, найти там подходящий аудиоклип и перетащить его на арену.
- Flash отображает волновые формы из второго звукового файла в опорном кадре 2 из слоя Sound 2 (рис. 15.19).
7. На вкладке свойств кадра выберите из списка **Sync** значение Event,
8. Поместите точку воспроизведения в кадр 1 и воспроизведите ролик один раз.

По достижении опорного кадра 5 Flash начинает проигрывать файлы Scale.AIFF и Melody.AIFF одновременно.

## С

Вся информация, необходимая для воспроизведения синхронизированного с событием звукового файла, хранится в том кадре, с которым этот файл связан. При воспроизведении ролик приостанавливается в этом кадре, пока не будет загружена вся информация. Поэтому такой способ синхронизации лучше оставить для коротких клипов, в противном случае задержки могут быть заметны.

## Использование звуков, синхронизированных с началом

Звуки, синхронизированные с началом, ведут себя подобно синхронизированным с событием, но с одним существенным отличием: Flash не запускает новый экземпляр, если такой же уже проигрывается.

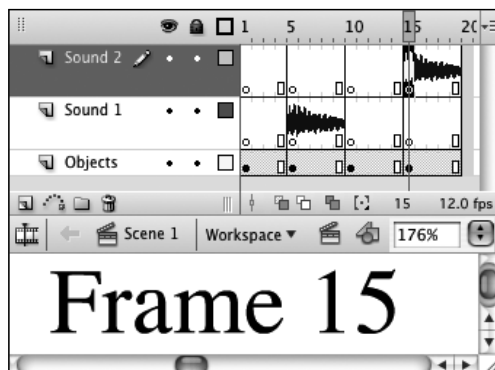
### Чтобы синхронизировать звук с началом:

1. Откройте файл `OverlapSnds.fla`, созданный ранее.  
У вас должен быть один экземпляр файла `Scale.AIFF` в опорном кадре 5, а другой — в кадре 15. Второй экземпляр может находиться в слое `Sound 1` или `Sound 2`.
2. Выберите на временной шкале опорный кадр 15, к которому присоединен файл `Scale.AIFF` (рис. 15.20).
3. На вкладке свойств кадра выберите из списка **Sync** значение **Start** (рис. 15.21).
4. Поместите точку воспроизведения в кадр 1 и воспроизведите ролик один раз.

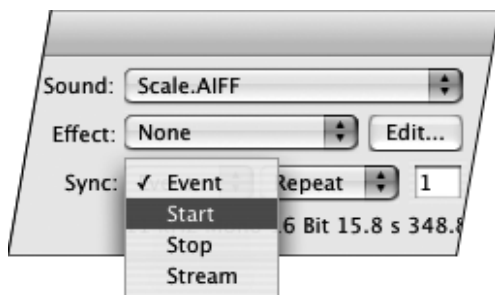
По достижении опорного кадра 5 Flash начинает проигрывать файл `Scale.AIFF`. По достижении кадра 15 ничего нового не происходит, по-прежнему слышится только один голос, так как файл `Scale.AIFF` еще не закончился. Если некоторый звуковой файл воспроизводится, а Flash встречает еще один экземпляр того же файла, то в случае синхронизации с началом новый экземпляр не запускается.

**С**

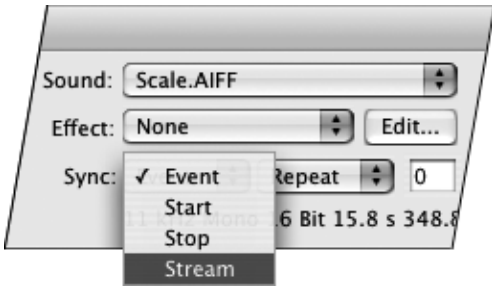
Чтобы избежать воспроизведения нескольких экземпляров одного и того же звукового файла при зацикливании ролика, устанавливайте свойство **Sync** в **Start**. Если файл еще проигрывается в момент возврата в начало ролика, то Flash дает ему доиграть, не запуская новый экземпляр. Если же к этому моменту файл уже завершился, то он запустится снова.



**Рис. 15.20.** Чтобы изменить свойство **Sync**, сначала выберите опорный кадр, к которому присоединен звук.



**Рис. 15.21.** Чтобы предотвратить запуск второго экземпляра звукового файла, если первый еще проигрывается, выберите значение **Start** из списка **Sync** на вкладке свойств кадра в инспекторе свойств.



**Рис. 15.22.** Чтобы Flash синхронизировал звук с конкретным кадром ролика, выберите значение Stream из списка Sync на вкладке свойств кадра.

### О потоковом аудио

Если в списке **Sync** выбрано значение **Stream**, то Flash разбивает аудиоклип на более мелкие подклипы и вставляет каждый из них в отдельный кадр. Размер подклипа определяется частотой кадров. Например, если задана частота 10 кадров/с, то каждый подклип будет звучать 1/10 секунды. Таким образом, в каждых 10 кадрах будет воспроизведена 1 секунда звучания.

Flash синхронизирует начало каждого подклипа с определенным кадром. Если звук воспроизводится быстрее, чем компьютер успевает рисовать кадры, то Flash готов пожертвовать частью визуального ряда (пропуская рисование некоторых кадров анимации), лишь бы звук и изображение точнее соответствовали друг другу. Например, если свойство **Sync** равно **Stream**, то вы услышите, как хлопнула дверь, одновременно с ее закрытием, а не спустя несколько секунд. Если несоответствие между скоростью воспроизведения звука и скоростью рисования кадров слишком велико, то из-за пропущенных кадров ролик может казаться дергающимся, как если бы вы с самого начала установили излишне низкую частоту кадров.

## Использование потокового аудио

Потоковые звуковые файлы особенно хорошо приспособлены к воспроизведению через Web. Если свойство **Sync** равно **Stream**, то Flash разбивает звуковой файл на небольшие кусочки. Эти миниклипы синхронизируются с таким количеством кадров ролика, какое необходимо для воспроизведения звука. Flash перестает воспроизводить поток по достижении следующего за ними кадра или при получении команды остановить воспроизведение этого или вообще всех звуков.

В отличие от звуковых файлов, синхронизированных с событиями, которые должны быть загружены целиком еще до начала воспроизведения, потоковое аудио можно начинать воспроизводить уже после загрузки небольшого числа кадров. Следовательно, этот вариант оптимален для продолжительных звуковых файлов, особенно если ролик доставляется через Web.

### Чтобы сделать звуковой файл потоковым:

1. Откройте новый документ по созданному ранее шаблону **SoundSyncMaster**.
2. На временной шкале удалите опорный кадр из позиции 10 в слое **Sound 1** (выберите его и выполните команду **Modify** ⇒ **Timeline** ⇒ **Clear Keyframe**).
3. На временной шкале выберите опорный кадр 5 в слое **Sound 1**.
4. На вкладке свойств кадра выберите продолжительный звуковой файл из списка **Sound** (в данном случае **Scale.AIFF**).
5. В списке **Sync** выберите значение **Stream** (рис. 15.22).

6. Чтобы посмотреть, как теперь распределяется звук в промежутке времени, отведенном для ролика, нажмите кнопку **Edit** (Редактировать) в разделе Sound на вкладке свойств кадра.

Открывается диалоговое окно Edit Envelope (Редактировать огибающую). Звуковой файл Scale.AIFF продолжительностью 15.8 с не успевает целиком воспроизвестись в кадрах между 5 и 15. Если свойство Sync равно Stream, то Flash воспроизводит лишь ту часть звука, которая помещается в отведенные для него кадры, в данном случае чуть меньше одной секунды. В окне Edit Envelope вертикальная линия показывает, в каком месте звук будет обрезан (рис. 15.23).

7. Чтобы закрыть окно Edit Envelope, нажмите кнопку **OK** или **Cancel** (Отмена).

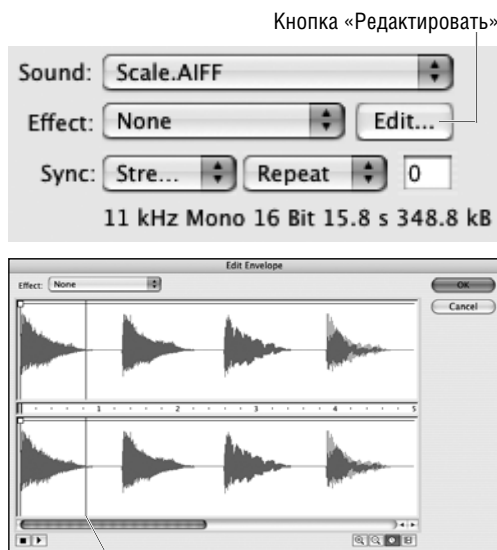
Обрезанная волновая форма показывается в кадрах 5–15 (рис. 15.24).

8. Поместите точку воспроизведения в кадр 1 и воспроизведите ролик, чтобы послушать, как он звучит.

По достижении кадра 5 начинает проигрываться файл Scale.AIFF. Когда точка воспроизведения доходит до кадра 15, последовательность опорного кадра заканчивается, и Flash прекращает проигрывать звуковой файл.

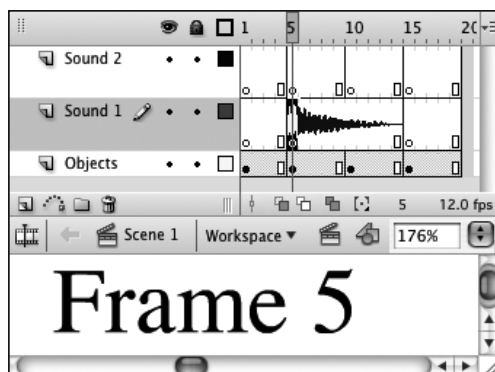
9. Выполните команду **Control ⇨ Loop Playback** и снова воспроизведите ролик.

Flash повторяет один и тот же фрагмент звукового файла, всякий раз прекращая воспроизведение по достижении кадра 15.



Здесь воспроизведение звука будет остановлено

**Рис. 15.23.** Если свойство Sync равно Stream, то можно посмотреть, какая часть звукового файла будет воспроизводиться при данном числе промежуточных кадров, отведенных под этот файл. На вкладке свойств кадра нажмите кнопку Edit (верхний рисунок), чтобы открыть окно Edit Envelope (нижний рисунок). В окне редактирования звукового файла отображается полная волновая форма в контексте времени или числа кадров.



**Рис. 15.24.** В десяти кадрах в позициях 5–15 хватает времени только для воспроизведения первой ноты гаммы. Именно такая часть полной волновой формы отображается на временной шкале.

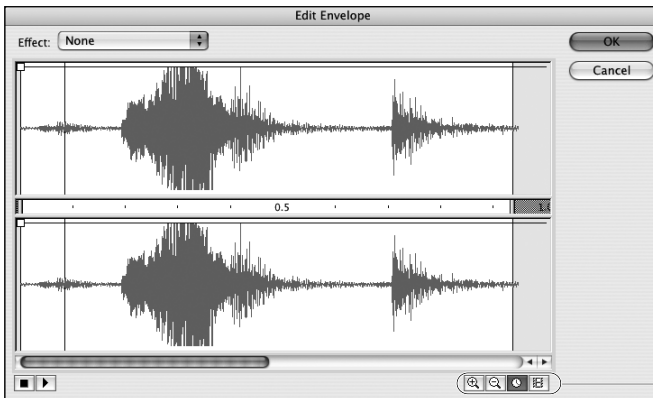


С

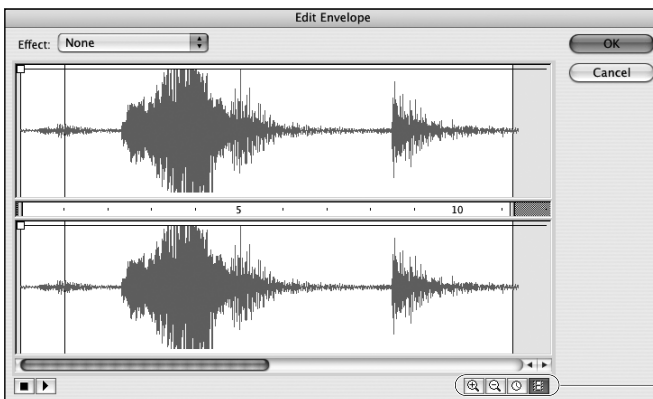
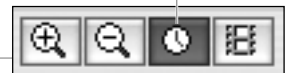
Услышать, как воспроизводится потоковое аудио, можно, протаскив точку воспроизведения по всей шкале (этот прием называется *ручным воспроизведением*). По мере перемещения вы сможете оценить, насколько хорошо совмещаются изображение и звук. По результатам можно будет добавить или удалить кадры для улучшения синхронизации.

С

Если потоковое аудио преждевременно обрывается, измените единицу измерения в окне Edit Envelope, чтобы понять, сколько кадров необходимо добавить для полного воспроизведения звукового файла (рис. 15.25).



Единица измерения «секунды»



Единица измерения «кадры»



**Рис. 15.25.** На графике волновой формы в окне Edit Envelope можно в качестве единицы измерения выбрать секунды (верхний рисунок) или кадры (нижний рисунок). Выбрав кадры, вы точно узнаете, сколько кадров необходимо, чтобы наиболее важная часть звукового файла была воспроизведена до конца. (В данном случае требуется 11 кадров.) Обычно желательно оставить место для частей волновой формы с наибольшей амплитудой.

## Прекращение звучания

Обычно синхронизированные с событиями звуковые файлы проигрываются до конца, но можно принудительно прекратить звучание в указанном кадре. Для этого следует свойству Sync присвоить значение Stop.

### Чтобы остановить воспроизведение звука:

1. Создайте однослойный документ с 15 кадрами и двумя относительно продолжительными звуковыми файлами (по меньшей мере 2–3 секунды каждый); один файл поместите в опорный кадр 1, другой — в опорный кадр 5.

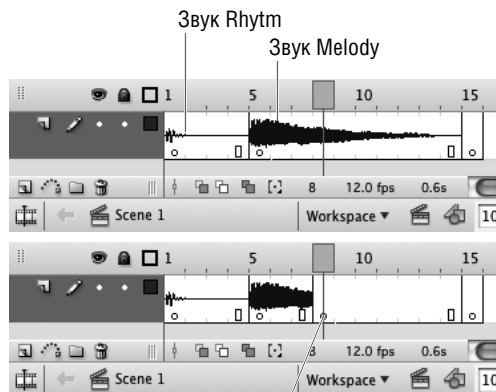
В данном примере кадр 1 содержит файл Rhythm.AIFF, а кадр 5 — файл Melody.AIFF. Для обоих файлов задайте свойство Sync равным Event.

2. Вставьте в позицию 8 на временной шкале пустой опорный кадр (рис. 15.26).

Пустой опорный кадр обрезает волновую форму на временной шкале, но при воспроизведении ролика оба файла продолжат звучать и после достижения кадра 8.

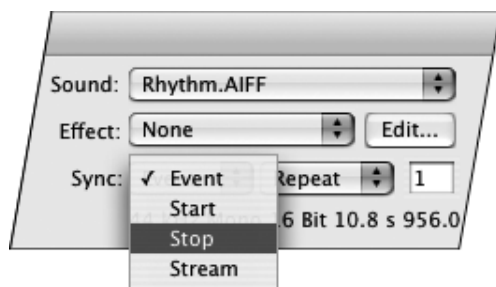
3. Выберите опорный кадр 8.
4. На вкладке свойств кадра в инспекторе свойств выберите из списка **Sound** файл Rhythm.AIFF.
5. В списке **Sync** выберите значение Stop (рис. 15.27).

Flash воспринимает это как команду прекратить воспроизведение файла Rhythm.AIFF по достижении кадра 8.



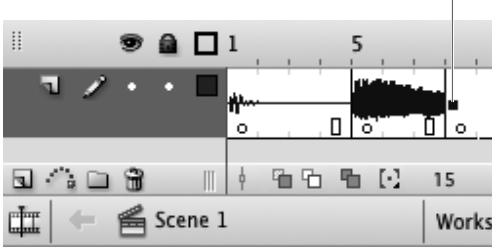
После добавления пустого опорного кадра

**Рис. 15.26.** В результате вставки нового опорного кадра волновая форма, соответствующая звуковому файлу в предшествующем кадре, обрезается на временной шкале. Однако, если файл был синхронизирован с событием, то он продолжает звучать и после того, как точка воспроизведения пройдет этот кадр.



**Рис. 15.27.** Чтобы прекратить воспроизведение звука в определенной точке ролика, создайте и выберите опорный кадр в этой точке. На вкладке свойств кадра выберите в списке Sound тот звуковой файл, который должен быть остановлен. В списке Sync выберите значение Stop. В данном случае команда Stop относится к звуку Rhythm.

В этом опорном кадре свойство Sync равно Stop



**Рис. 15.28.** Квадратик в центре опорного кадра на временной шкале говорит о наличии в нем команды прекращения звучания.

Чтобы показать, что кадр 8 на временной шкале содержит команду прекращения звучания, в его центре находится маленький квадратик (рис. 15.28).

6. Поместите точку воспроизведения в кадр 1 и воспроизведите ролик, чтобы послушать, как он звучит.

Воспроизведение файла *Rhythm.AIFF* начинается немедленно, а файла *Melody.AIFF* — с пятого кадра. По достижении кадра 8 звук *Rhythm.AIFF* обрывается, а *Melody.AIFF* продолжает звучать и после достижения конца ролика.

**С**

Команда Stop и останавливаемый ей звук могут находиться в разных слоях. Эта команда прекращает воспроизведение всех экземпляров заданного звукового файла, в каком бы слое они ни находились.

**С**

Чтобы остановить воспроизведение только одного экземпляра звукового файла, задайте для него свойство Sync равным Stream, и в слое, который содержит этот экземпляр, вставьте пустой опорный кадр в то место, где звучание должно прекратиться.

## Повтор звучания

Во Flash есть параметр, который позволяет повторить воспроизведение звука несколько раз подряд без добавления новых экземпляров. Если на вкладке свойств кадра ввести в поле Repeat (Повтор) (оно находится справа) некоторое значение, то находящийся в этом кадре звук будет воспроизведен указанное число раз. Повторять можно как звуки, синхронизированные с событиями, так и потоковое аудио. Свойство Sync применяется ко всей серии повторяющихся звуков. Можно также установить режим повтора до получения иной команды.

### Чтобы задать значение Repeat:

1. Создайте документ с пятью кадрами и коротким звуковым файлом в кадре 1. Для этого упражнения мы воспользуемся файлом Drip.AIFF.
2. Выберите на временной шкале опорный кадр 1.
3. На вкладке свойств кадра в инспекторе свойств выберите из списка **Repeat** в правой части значение по умолчанию Repeat (рис. 15.29).
4. Введите значение 3 в поле справа от списка **Repeat** (рис. 15.30).

Flash продолжит волновую форму звукового сигнала, добавив к ней еще две идентичных копии. На временной шкале отображается часть полной волновой формы, уместящаяся в отведенных для нее кадрах (рис. 15.31).

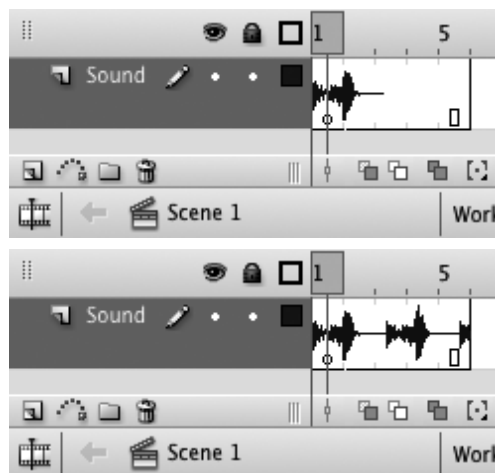
5. Сохраните документ, назвав его RepeatSnds fla.



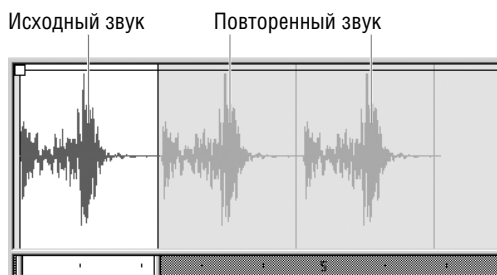
**Рис. 15.29.** В списке Repeat на вкладке свойств кадра можно задать два режима воспроизведения звукового файла: заданное число раз (Repeat — Повторить) или циклически до получения команды останова (Loop — Цикл).



**Рис. 15.30.** Значение в поле Repeat говорит, сколько раз повторять выбранный звук.



**Рис. 15.31.** Если свойство Repeat равно 0 или 1, Flash отображает на временной шкале только исходную волновую форму (верхний рисунок). Если же Repeat больше 1, то отображается повторенная волновая форма, но не больше, чем хватает места (нижний рисунок).



**Рис. 15.32.** Полную волновую форму повторенного звука в контексте секунд или кадров можно посмотреть в диалоговом окне Edit Envelope. Закрашенные серым цветом формы относятся к повторам.

**С**

Чтобы посмотреть, сколько кадров или секунд занимает повторенная волновая форма, нажмите кнопку **Edit** на вкладке свойств кадра в инспекторе свойств. В диалоговом окне Edit Envelope отображается полная волновая форма (рис. 15.32).

**С**

Задавать повтор для потокового аудио разрешается, но при этом увеличивается размер опубликованного файла.

### Управление звуком с помощью кнопок

В этой главе вы научились двум способам работы со звуком. Путем включения звука в опорные кадры символа-кнопки вы создали дружелюбные кнопки с акустической обратной связью. А кроме того, вы узнали, как добавить в ролик звуковую дорожку или звуковые эффекты путем помещения звука непосредственно на временную шкалу. Но есть и третий способ — воспользоваться языком ActionScript (AS) для управления звуком во время воспроизведения. Так можно загружать, проигрывать и останавливать звук на этапе выполнения. В версии AS 2.0 существует набор поведений, реализующих типичные задачи управления звуком. Поведение присоединяет сценарий прямо к объектам управления звуком (символам-кнопкам, компонентам-кнопкам или символам-клипам). С их помощью можно настроить кнопку так, чтобы она загружала и воспроизводила определенный звуковой файл, повторяла воспроизведение уже загруженного файла или останавливала все синхронизированные с событиями звуки вне зависимости от того, были они загружены из сценария или определены на этапе разработки. В версии AS 3.0 программное управление звуками возлагается на вас. Можно присоединить к объекту, сопровождаемому звуком, сценарий, который будет выполнять те же функции, что поведения в AS 2.0. В любом случае для настройки кнопки, управляющей звуком, необходимо задать для звукового файла свойство Linkage (Связывание). Программное управление звуком — это достаточно сложная задача, которую мы в этой книге рассматривать не будем. О создании простых сценариев на языке ActionScript 3.0 см. главу 13.

## Редактирование звука

Flash позволяет вносить некоторые изменения в экземпляры звуковых файлов, пользуясь диалоговым окном Edit Envelope. Можно изменить начальную и конечную точку звучания (то есть обрезать волновую форму по краям) и отрегулировать громкость.

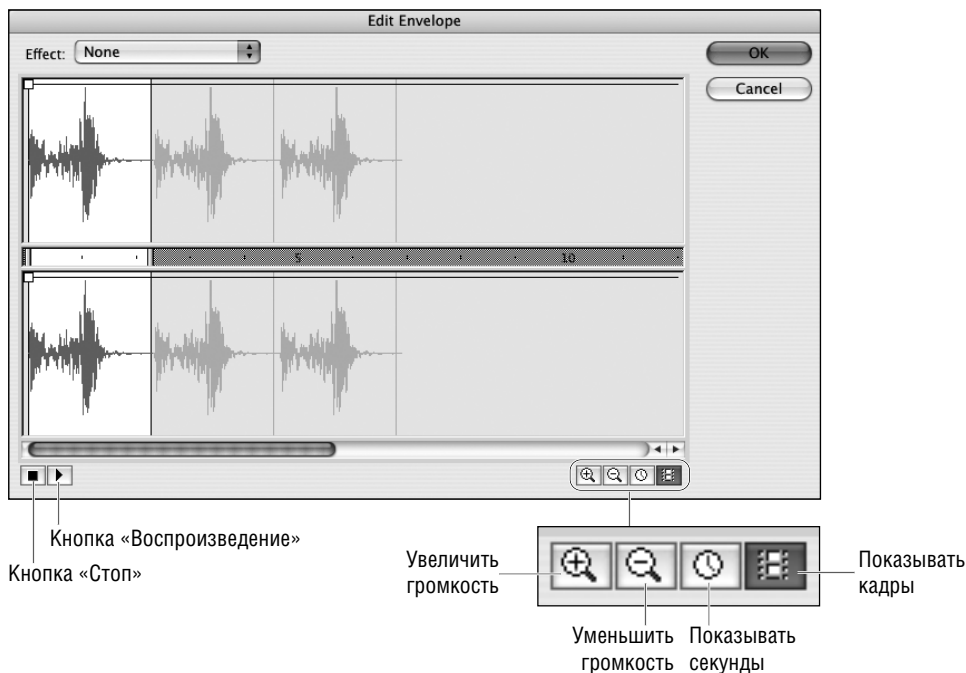
Во Flash определено шесть предустановленных шаблонов звука: Left Channel (Левый канал), Right Channel (Правый канал), Fade Left to Right (Перемещение слева направо), Fade Right to Left (Перемещение справа налево), Fade In (Сильнее) и Fade Out (Слабее). С помощью этих шаблонов создаются стандартные звуковые эффекты: постепенное нарастание силы звука (Fade In), уменьшение силы звука (Fade Out), а в случае стереозвучания — перетекание звука из одного динамика в другой.

Помимо изменения громкости, можно сократить время звучания, удалив часть сигнала в начале, в конце или с обеих сторон.

## Чтобы назначить готовый звуковой эффект:

1. Откройте документ, созданный в предыдущем упражнении (RepeatSnds.fla). Этот ролик содержит пять кадров и синхронизированный с событием звук, который повторяется в них три раза.
2. Выберите на временной шкале опорный кадр 1. В нем находится звуковой файл Drip.AIFF.
3. На вкладке свойств кадра нажмите кнопку **Edit**.

Открывается диалоговое окно Edit Envelope с областью редактирования звука, в которой показана волновая форма звукового файла из кадра 1 (рис. 15.33).



**Рис. 15.33.** В окне Edit Envelope можно выполнять простые операции редактирования звука: изменять длительность звучания и регулировать громкость.



**Рис. 15.34.** В списке Effect в диалоговом окне Edit Envelope предлагается шесть стандартных звуковых эффектов. Эффект Fade In приводит к постепенному нарастанию силы звука.



**Рис. 15.35.** В случае эффекта Fade In огибающая сигнала начинается с 0 процентов (нижний край области редактирования звука) и быстро поднимается до 100 процентов (верхний край).

4. В диалоговом окне Edit Envelope нажмите кнопку **Display Frames** (Кадры).

Измерять продолжительность звучания можно в секундах или в кадрах. Щелчок по значку с изображением пленки в правом нижнем углу окна устанавливает единицу измерения «кадры», а по значку с изображением часов — секунды.

5. Из списка **Effect** (Эффект) выберите значение Fade In (Сильнее) (рис. 15.34).

Flash изменяет огибающую сигнала (рис. 15.35). Если огибающая расположена у верхнего края области редактирования, то Flash воспроизводит звук со 100%-ной громкостью, а если у нижнего — то вообще не воспроизводит.

6. Нажмите кнопку **Play** (Воспроизведение), чтобы услышать эффект нарастания громкости.

На первой итерации звучание вначале тихое и постепенно становится громче. При повторении звук воспроизводится в полную силу.

7. Нажмите **ОК**.

Flash возвращается в режим редактирования документа.

**С**

Если волновая форма звукового сигнала вас не интересует, можно обойтись и без открытия окна Edit Envelope. Просто выберите эффект из списка **Effect** на вкладке свойств кадра в инспекторе свойств.

### Чтобы самостоятельно создать звуковой эффект:

1. Выполните шаги 1–3 из предыдущего упражнения.
2. Из списка **Effect** в окне Edit Envelope выберите пункт Custom (Заказной) (рис. 15.36).
3. В области редактирования на огибающей имеются квадратные манипуляторы, которые первоначально для обоих каналов находятся на уровне 0%.

4. В правом канале (верхний график) щелкните в трех точках волновой формы, чтобы задать разные уровни для каждого повтора. В данном примере поместите манипуляторы рядом с отметками кадров 2, 4 и 5.

Flash добавит на огибающую манипуляторы, так что теперь в обоих каналах имеется по четыре манипулятора.

5. В правом канале отбуксируйте второй манипулятор вверх примерно до 50%-ого уровня громкости (рис. 15.37).
6. Повторите шаг 5 для левого канала.
7. В обоих каналах отбуксируйте третий манипулятор до 50%-ого уровня, а четвертый — до 100%-ного.

В одном звуковом файле можно использовать до восьми манипуляторов.

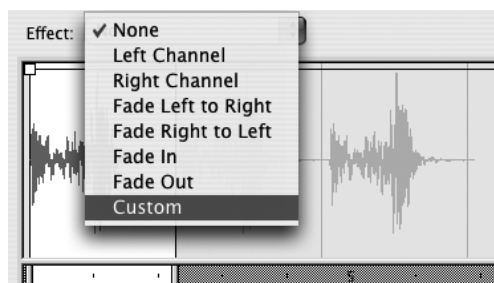
8. Нажмите кнопку **Play**, чтобы услышать эффект нарастания громкости.

Flash увеличивает силу звука на первой итерации, вторую воспроизводит с половинной громкостью, а третью — со стопроцентной.

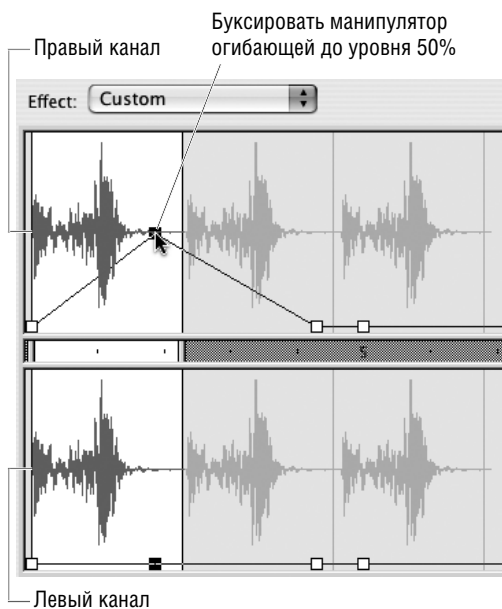
9. Нажмите **ОК**.

**С**

Чтобы удалить ненужные манипуляторы огибающей, отбуксируйте их за пределы области редактирования звука.

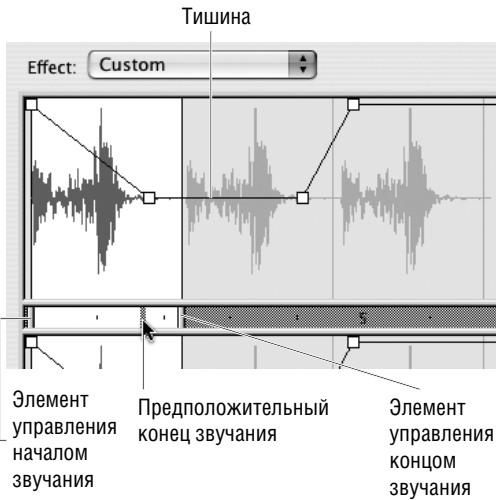


**Рис. 15.36.** Чтобы самостоятельно изменить уровень громкости, выберите из списка Effect в окне Edit Envelope пункт Custom.

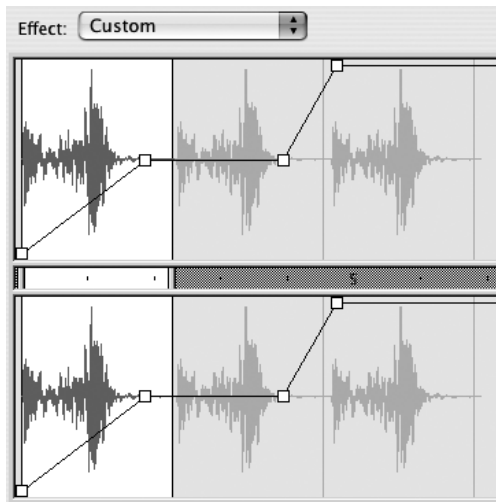


**Рис. 15.37.** Чтобы добавить манипулятор, щелкните по волновой форме в области редактирования звука. Для изменения формы огибающей буксируйте манипулятор. Можно сделать огибающие для двух каналов одинаковыми или различными. У монофонического звука обе волновые формы одинаковы.





**Рис. 15.38.** Flash позволяет обрезать начало и конец звукового файла с помощью инструментов в окне Edit Envelope. В данном случае буксировка элемента управления концом звучания обрезает часть звука, в которой амплитуда сигнала мала. (Сигнал малой амплитуды звучит очень тихо, почти неслышно.)

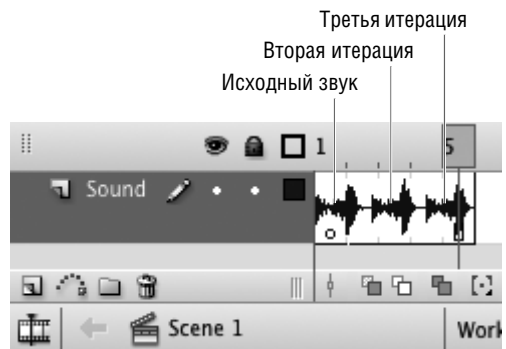


**Рис. 15.39.** После того как элемент управления концом звучания смещен влево, протяженность волновой формы в области редактирования звука уменьшилась.

### Чтобы изменить продолжительность звучания:

1. Выберите опорный кадр 1 в ролике, который был создан в предыдущем упражнении.
2. Откройте окно Edit Envelope, нажав кнопку **Edit** на вкладке свойств кадра.
3. В области редактирования звука отбуксируйте элемент управления концом звучания (вторая вертикальная планка в полосе, разделяющей каналы) туда, где волновая форма становится плоской (рис. 15.38).  
Flash сокращает продолжительность звучания в обоих каналах (рис. 15.39).
4. Нажмите **ОК**.

Flash возвращается в режим редактирования документа. Теперь все три итерации уместятся на временной шкале (рис. 15.40).



**Рис. 15.40.** После уменьшения продолжительности звучания в первых пяти кадрах ролика уместились все три итерации.

**С**

Чтобы удалить неслышимый кусок в начале звукового файла, буксируйте элемент управления началом звучания. Часть волновой формы до точки начала звучания будет закрашена серым цветом.

**С**

Хотя Flash и позволяет изменять начальную и конечную точку звучания, в SWF-файле по-прежнему хранится звуковой файл целиком. Если вам приходится часто обрезать звуки, то подумайте о приобретении специализированного редактора звука, который позволит выбросить обрезки, а не таскать их с собой.

**С**

Если вы задали повторение звука, то Flash соединяет и отображает в окне Edit Envelope несколько копий сигнала, где их можно отредактировать. Например, можно увеличивать громкость от итерации к итерации. (См. раздел «Повтор звучания» выше.)

От Web-камер в природных заповедниках до роликов на сайте YouTube — видео присутствует в Web повсюду. Программа Adobe Flash CS3 Professional позволяет без труда включать в ролики видео, чтобы зрителям было интереснее. Имеются следующие варианты: встроить видеоданные на временную шкалу, добавить в документ специальный компонент для отображения внешних видеофайлов, конвертированных в формат Adobe Flash Video (FLV) или включить в ролик ссылку на внешний файл в формате QuickTime. Первые два позволяют конечному пользователю просматривать видео во Flash-плеере, а последний предполагает публикацию Flash-контента в виде ролика QuickTime. Опытный программист может также написать сценарий для динамического отображения FLV-файла как потокового видео.

Для преобразования видеофайлов в формат, понятный Flash, можно пользоваться как встроенным во Flash мастером импорта видео, так и автономной утилитой Adobe Flash CS3 Video Encoder (она автоматически ставится, если выполнялась стандартная установка Flash). В этой главе вы научитесь работать с мастером импорта видео. Мы рассмотрим как внедренные видеоклипы, так и использование компонента FLVPlayback для отображения в Flash-плеере видео из внешних FLV-файлов.

### О мастере импорта видео

Мастер импорта видео в Flash CS3 проводит вас по всей процедуре импорта и кодирования видеоклипов. Ко всему прочему мастер предлагает ограниченные средства редактирования видео, позволяющие изменить область просмотра, размер кадра и длину.

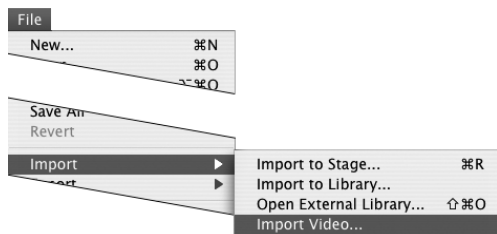
Самое важное решение, которое вам предстоит принять при работе с мастером, касается того, как Flash будет доставлять зрителям видеоданные. Видеоданные можно встроить в сам документ Flash или оставить их во внешнем файле и отображать видео с помощью специального компонента — FLVPlayback.

При внедрении видео увеличивается размер SWF-файла, и пользователь ничего не увидит, пока видеоданные не загрузятся полностью. Видео, отображаемое с помощью компонента FLVPlayback, не увеличивает размер ролика. Как долго пользователю придется ждать, зависит от того, используется ли прогрессивное или потоковое видео и от конфигурации его собственного компьютера.

Видео в формате FLV можно загружать с Web-сервера прогрессивно. В этом случае видеоданные разбиваются на небольшие сегменты. Воспроизведение начинается уже после загрузки нескольких сегментов, следовательно пользователю придется ждать не так долго.

Flash поддерживает также потоковую загрузку видео с помощью службы Flash Video Streaming Service или с вашего собственного сервера, на котором установлена программа Flash Media Server. При таком способе воспроизведение видео может начаться, как только получены данные для начальных кадров; пока пользователь просматривает их, сервер продолжает посылать данные. Для использования потокового видео вы должны иметь доступ к Flash Media Server, либо завести учетную запись Flash Video Streaming Service у вашего Интернет-провайдера, либо воспользоваться специализированной потоковой службой, например LimeLight Networks.

Дополнительно Flash может подготовить видео для мобильных устройств и организовать ссылку на внешние видеофайлы в формате QuickTime. Однако вариант с QuickTime работает только для Flash Player версий 3–5 и требует, чтобы окончательный контент был опубликован в формате MOV-файла. Подробнее о публикации см. главу 17.



**Рис. 16.1.** Чтобы запустить мастер импорта видео, выполните команду **File** ⇒ **Import** ⇒ **Import Video**.



**Рис. 16.2.** Мастер импорта видео проводит вас по всей процедуре подготовки видеоклипа к использованию во Flash-ролике. Первый шаг — выбор видеофайла.



**Рис. 16.3.** Чтобы импортировать файл со своего компьютера, отметьте переключатель **On Your Computer** и либо введите путь к файлу, либо найдите его с помощью инструментов навигации.



**Рис. 16.4.** Чтобы импортировать файл с внешнего Web-сервера, отметьте переключатель **Already deployed to a Web server** и введите URL.

## Импорт видео для прогрессивной или потоковой загрузки

Импорт видео начинается с запуска мастера, отыскания исходного видеофайла и выбора режима доставки (развертывания). Последующие шаги зависят от выбранного метода. Рассмотрим сначала развертывание видео для прогрессивной или потоковой загрузки через Web.

### Чтобы подготовить видео для прогрессивной или потоковой загрузки в Flash-плеер:

1. Открыв документ Flash, выберите из меню пункт **File** ⇒ **Import** ⇒ **Import Video** (Файл ⇒ Импорт ⇒ Импорт видео) (рис. 16.1)

Открывается страница **Select Video** (Выберите видео) мастера импорта (рис. 16.2).

2. Чтобы взять файл, хранящийся на вашем компьютере, отметьте переключатель **On Your Computer** (На вашем компьютере) (рис. 16.3) и выполните одно из следующих действий:

- введите путь к файлу в поле **File Path**;
- нажмите кнопку **Choose** (Выбор) (Mac) или **Browse** (Обзор) (Windows). Когда откроется диалоговое окно, найдите нужный файл, выберите его и нажмите кнопку **Open** (Открыть).

Мастер возвращается на страницу **Select Video** и копирует введенный путь в поле **File Path**.

или

Чтобы взять файл с Web-сервера, отметьте переключатель **Already deployed to a Web server** (Уже развернуто на Web-сервере) (рис. 16.4) и введите адрес файла в поле **URL**.

3. Нажмите кнопку **Continue** (Продолжить) (Mac) или **Next** (Далее) (Windows).

Появляется страница Deployment (Развертывание), на которой задаются параметры доставки видеоданных аудитории (рис. 16.5).

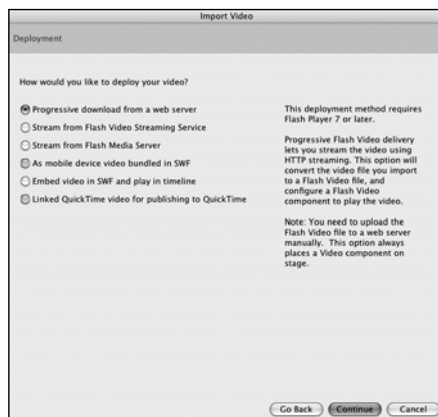
4. Для выбора метода доставки отметьте один из переключателей (рис. 16.6):

- для загрузки видеоданных фрагментами по протоколу HTTP отметьте переключатель **Progressive download from a Web server** (Последовательная загрузка с Web-сервер);
- для загрузки потокового видео с сервера с помощью службы Flash Video Streaming Service отметьте переключатель **Stream from Flash Video Streaming Service** (Потоковая передача с Flash Video Streaming Service) (этот способ предполагает, что вы имеете учетную запись для Flash Media Server у своего Интернет-провайдера или пользуетесь какой-то иной службой потоковой доставки);
- для загрузки видео со своего собственного сервера отметьте переключатель **Stream from Flash Media Server** (Потоковая передача с сервера Flash Media Server).

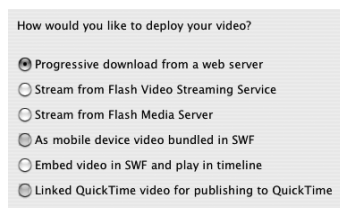
5. Нажмите кнопку **Continue** (Mac) или **Next** (Windows).

Появляется страница Encoding (Кодировка) (рис. 16.7).

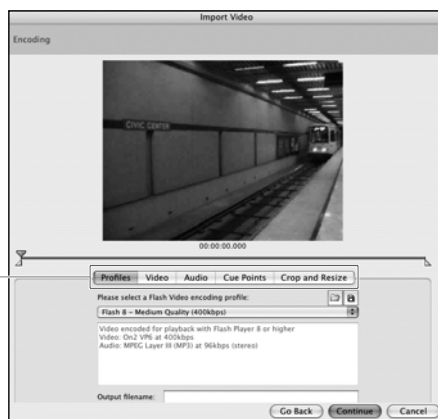
A	B	C	D	E
Profiles	Video	Audio	Cue Points	Crop and Resize



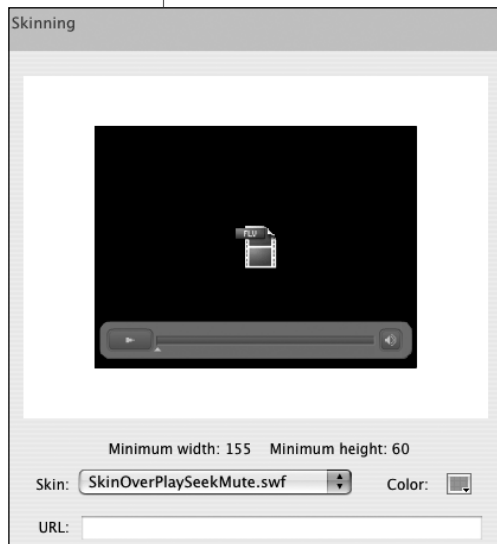
**Рис. 16.5.** На странице мастера Deployment определяется способ доставки видеоданных аудитории.



**Рис. 16.6.** Мастер импорта видео предлагает шесть способов доставки видеоданных.



**Рис. 16.7.** На странице Encoding есть пять вкладок: предопределенные параметры сжатия видео и аудио (A), заказные параметры сжатия видео и аудио (B), заказные параметры сжатия аудио (C), инструменты для создания ключевых точек, или маркеров для использования в ActionScript (D) и инструменты для модификации размеров и длины клипа (E).



**Рис. 16.8.** На странице мастера Skinning можно выбрать тему для оформления элементов управления прогрессивной и потоковой загрузкой видео в Flash-ролике.

6. Выберите одну из следующих вкладок:

- чтобы использовать встроенное кодирование видео и аудиоданных, нажмите кнопку **Profiles** (Профили) (Mac) или перейдите на вкладку **Encoding Profiles** (Профили кодирования) (Windows), а затем выберите из списка тот или иной предопределенный профиль;
- чтобы самостоятельно задать параметры видео, нажмите кнопку **Video** (Mac) или перейдите на вкладку **Video** (Видео) (Windows);
- чтобы самостоятельно задать параметры аудио, нажмите кнопку **Audio** (Mac) или перейдите на вкладку **Audio** (Звукозапись) (Windows);
- чтобы пометить некоторые кадры клипа, так чтобы впоследствии на них можно было сослаться из ActionScript-сценария, нажмите кнопку **Cue Points** (Mac) или перейдите на вкладку **Cue Points** (Ключевые точки) (Windows) и с помощью кнопок **Add Cue Point** (Добавить ключевую точку) и **Add Parameters** (Добавить параметры) настройте ключевые точки и их параметры;
- чтобы кадрировать видеоизображение, изменить размер окна для показа видео или изменить длину видеоролика, нажмите кнопку **Crop and Resize** (Mac) или перейдите на вкладку **Crop and Resize** (Кадрировать и изменить размер) (Windows) и задайте желаемые параметры. О назначении параметров на каждой вкладке будет рассказано в следующих упражнениях.

7. Нажмите кнопку **Continue** (Mac) или **Next** (Windows).

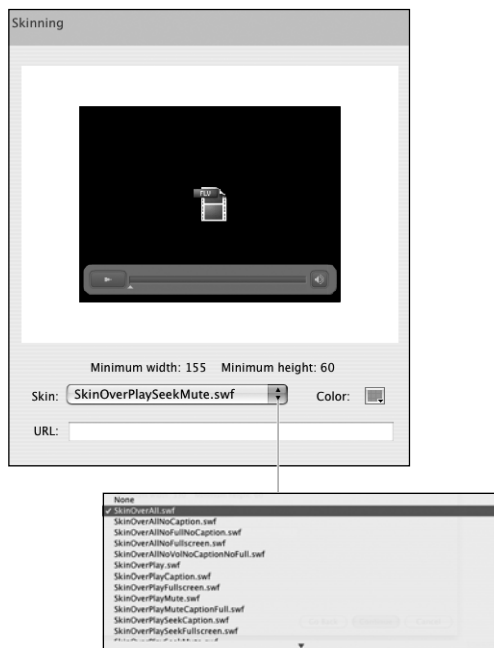
Появляется страница Skinning (Загрузка темы) (рис. 16.8).

8. Для создания элементов управления воспроизведением, которые будут видеть конечные пользователи, выберите тему из списка **Skin** (Тема):

- чтобы создать панель управления, которая будет плавать поверх изображения, выберите тему, название которой начинается со слов *SkinOver* (рис. 16.9);
- чтобы создать панель управления, которая будет расположена под изображением, выберите тему, название которой начинается со слов *SkinUnder*.

9. Нажмите кнопку **Continue** (Mac) или **Next** (Windows).

Появляется страница **Finish Video Import** (Завершить импорт видео) (рис. 16.10), на которой сведена вся информация, собранная мастером, и приводится краткое описание того, каков будет результат (см. врезку «Завершающие шаги для компонента FLVPlayback» ниже).



**Рис. 16.9.** В каждой из перечисленных тем для создания панели управления используются различные элементы. Темы, в названии которых есть слово *Over*, создают панель, плавающую поверх изображения, а темы, в названии которых есть слово *Under*, — панель, расположенную под изображением.

☐ After importing video, view video topics in Flash Help



**Рис. 16.10.** На странице **Finish Video Import** приведены сведения о том, как работать с видеофайлами, предназначенными для публикации. Чтобы получить более подробную информацию, отметьте флажок **After importing video, view video topics in Flash Help** (После импорта видео ознакомьтесь с соответствующими разделами справки Flash).



**10. Нажмите кнопку **Finish** (Готово).**

Если вы еще не сохранили файл, с которым работаете, то Flash откроет диалоговое окно **Save As (Mac)** или **Save In (Windows)**. Перейдите в папку, где хотите сохранить файл, введите имя файла и нажмите кнопку **Save** (Сохранить). Появляется диалоговое окно **Flash Video Encoding Progress** (Ход кодирования видео Flash). Чтобы прервать кодирование, нажмите кнопку **Cancel**. Затем появляется окно **Loading FLV Dimensions** (Загрузка размеров FLV-файла) с индикатором выполнения. По завершении загрузки Flash помещает на арену экземпляр компонента **FLVPlayback**. Его свойства уже настроены так, что он указывает на исходный видеофайл и использует тему, выбранную на шаге 8.

**C**

Чтобы создать видеоклип без панели управления, на шаге 8 выберите из списка **Skin** пункт **None** (Нет).

**C**

Чтобы задать цвет панели управления, на странице мастера **Skinning** щелкните по элементу управления цветом и выберите цвет в окне образцов.

**За пределами мастера**

Если нужно закодировать много видеофайлов с одними и теми же параметрами, то загружать их во Flash поодиночке с помощью мастера импорта видео довольно утомительно. На этот случай имеется автономная утилита кодирования видео — **Adobe Flash CS3 Video Encoder**.

С ее помощью можно создать список файлов, а затем установить параметры кодирования в диалоговых окнах. Окна для задания сжатия видео и аудио, создания ключевых точек, кадрирования и изменения размера и выбора темы почти не отличаются от страниц мастера.

Видеофайлы, пригодные для Flash, можно создавать также с помощью внешних приложений, которые умеют выводить файлы в формате FLV, например, **Premier Pro** или **After Effects** компании **Adobe**.

## Импорт внедряемого видео

При выборе в качестве способа развертывания видеоклипа прогрессивной или потоковой загрузки закодированные видеоданные остаются во внешнем файле. Если же видео внедряется, то эти данные становятся частью FLA и SWF-файлов. Процедура импорта внедряемого видео несколько отличается от импорта прогрессивного или потокового видео. Мастер импорта выводит дополнительную страницу, на которой можно задать параметры внедрения и выполнить некоторые операции редактирования. Но создавать ключевые точки и выбирать тему для внедренных клипов не разрешается.

### Чтобы импортировать и внедрить видеоклип:

1. Выполните шаги 1–3 из предыдущего упражнения.

Если вы хотите, чтобы Flash поместил видео на временную шкалу, то до начала процедуры импорта выберите опорный кадр, в котором должно начаться воспроизведение клипа.

2. На странице Deployment мастера отметьте переключатель **Embed video in SWF and play in Timeline** (Внедрить видео в SWF и воспроизвести во временной шкале) (рис. 16.11).
3. Нажмите кнопку **Continue** (Mac) или **Next** (Windows).

Появляется страница Embedding (Внедрение) с параметрами, которые можно задать для встроенного видеоклипа (рис. 16.12).

How would you like to deploy your video?

- ☐ Progressive download from a web server
- ☐ Stream from Flash Video Streaming Service
- ☐ Stream from Flash Media Server
- ☐ As mobile device video bundled in SWF
- ☒ Embed video in SWF and play in timeline
- ☐ Linked QuickTime video for publishing to QuickTime

**WARNING: Embedded deployment is likely to cause audio synchronization issues. This option is ONLY recommended for short video clips with no audio track.**

Embedding video in a SWF file lets you sync video with other visual elements on the stage. For example you can add interactive elements to a video frame to create hotspots that link to other content.

**Note: Embedding substantially increases SWF file size.**

**Рис. 16.11.** При выборе внедрения как метода развертывания (верхний рисунок) Flash помещает все видеоданные в SWF-файл. У этого способа есть свои плюсы и минусы, о чем вас предупреждают в сообщении, которое появляется в правой части диалогового окна (нижний рисунок).

### Embedding

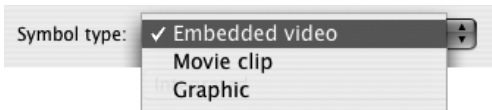
How would you like to embed the video?

Symbol type:

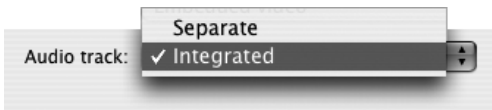
Audio track:

- ☒ Place instance on stage
- ☒ Expand timeline if needed
- ☒ Embed the entire video
- ☐ Edit the video first

**Рис. 16.12.** Перед тем как задавать параметры кодирования внедренного видеоклипа, необходимо определить, как вы собираетесь с ним работать.



**Рис. 16.13.** Список Symbol Type на странице Embedding позволяет выбрать способ обработки видеоклипа: преобразовать в анимированный графический символ или в символ-клип, либо разместить кадры видео прямо на главной временной шкале.



**Рис. 16.14.** В случае внедренного видеоклипа звуковую дорожку можно встроить вместе с видео или выделить ее в отдельный звуковой файл, который ничем не отличается от любого другого звукового актива.

4. Из списка **Symbol Type** (Тип символа) (рис. 16.13) выберите одно из следующих значений:

- **Embedded Video** (Встроенное видео). Помещает кадры видеоклипа непосредственно на временную шкалу самого ролика;
- **Movie Clip** (Фрагмент ролика). Помещает кадры видеоклипа на временную шкалу символа-клипа;
- **Graphic** (Графика). Помещает кадры видеоклипа на временную шкалу анимированного графического символа.

5. Из списка **Audio Track** (Звуковая дорожка) (рис. 16.14) выберите одно из следующих значений:

- **Separate** (Отдельно). Создает в библиотеке два актива, один для звуковой дорожки видеоклипа, другой — для видеоданных. На временную шкалу помещается лишь актив-видео.
- **Integrated** (Интегрировать). Объединяет звуковую дорожку и видео в единый актив.

6. Чтобы Flash автоматически поместил кадры видео на временную шкалу (если на шаге 4 было выбрано значение Embedded Video) или поместил экземпляр символа на арену (если на шаге 4 было выбрано значение Movie Clip или Graphic), отметьте флажок **Place instance on Stage** (Разместить экземпляр в рабочей области).

По завершении импорта Flash поместит видео или экземпляр символа в текущий выбранный опорный кадр.

7. Если на шаге 4 было выбрано значение Embedded или Graphic, то, чтобы Flash добавил на главную временную шкалу столько кадров, сколько необходимо для размещения всех кадров видео, отметьте флажок **Expand Timeline if needed** (При необходимости расширить временную шкалу).

- Чтобы импортировать исходный видео-файл целиком, отметьте переключатель **Embed the entire video** (Внедрить видео целиком) (рис. 16.15).

При желании можно обрезать видео или создать несколько коротких клипов до начала импорта (см. следующий раздел «Чтобы отредактировать внедренный видеоклип»).

- Нажмите кнопку **Continue** (Mac) или **Next** (Windows).

Появляется страница Encoding.

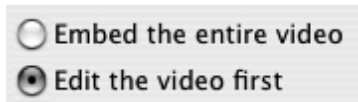
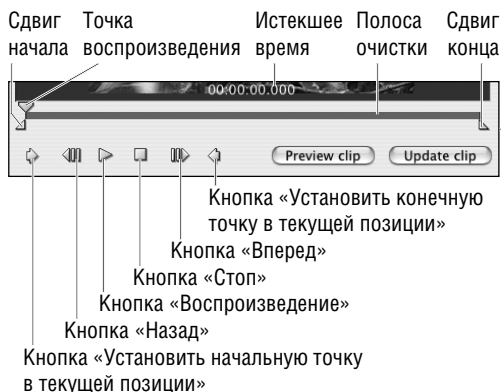
- Задайте параметры на вкладках Encoding Profiles, Video, Audio и Crop and Resize (детали см. в разделе «Задание параметров кодирования»).

- Выполните шаги 9 и 10 из предыдущего упражнения.

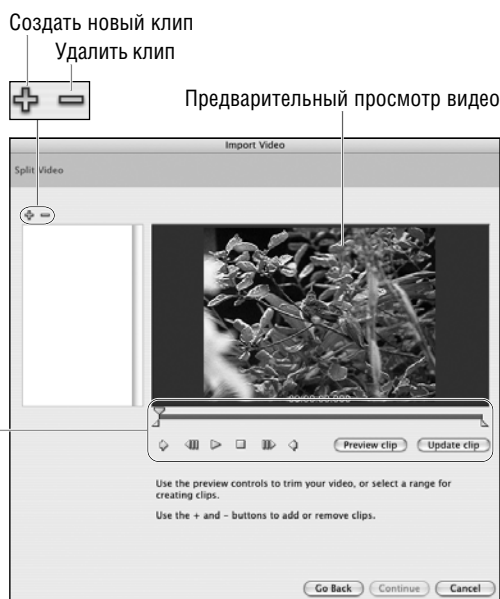
### Чтобы отредактировать внедренный видеоклип:

- Выполните шаги 1–7 из предыдущего упражнения.
- На странице мастера Embedding отметьте переключатель **Edit the video first** (Сначала редактировать видео).
- Нажмите кнопку **Continue** (Mac) или **Next** (Windows).

Появляется страница Split Video (Разделить видео) (рис. 16.16).

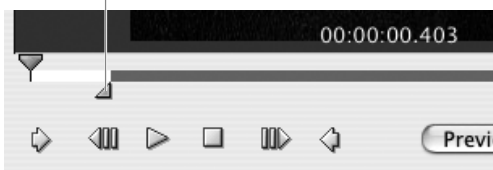


**Рис. 16.15.** Для импорта видео отметьте переключатель Embed the entire video на странице Embedding. Чтобы укоротить видеоклип перед внедрением или разрезать его на несколько частей, отметьте переключатель Edit the video first.



**Рис. 16.16.** Если вы решили отредактировать видео до начала импорта, то мастер предложит страницу Split Video, на которой есть инструменты для обрезания видеофайла и нарезания его на несколько частей.

Перетащить ползунок «Сдвиг начала»



Flash переназначает начальный кадр



Позиционировать точку воспроизведения



и нажать кнопку «Установить начальную точку в текущей позиции»

Flash переназначает начальный кадр



**Рис. 16.17.** Чтобы сделать начальным другой кадр внедряемого видео, можно перетащить ползунок «Сдвиг начала» (верхний рисунок) или точку воспроизведения, а затем нажать кнопку «Установить начальную точку в текущей позиции» (верхний рисунок).

**С**

Если вы точно не знаете, где должен начинаться или заканчиваться клип, нажмите кнопку **Воспроизведение**. Дойдя до места, где предполагается начать либо закончить клип, нажмите кнопку **Стоп**. Затем с помощью кнопок **Вперед** и **Назад** (либо буксируя точку воспроизведения) точно позиционируйтесь на нужный кадр. И напоследок нажмите кнопку **Установить начальную точку в текущей позиции** или **Установить конечную точку в текущей позиции**, чтобы сделать текущий кадр начальным или конечным.

4. Чтобы переназначить первый кадр во внедряемом видеоклипе, выполните любое из следующих действий (рис. 16.17):

- перетащите ползунок «Сдвиг начала» (треугольник слева от линейки) в то место, где должен начинаться клип;
- нажмите кнопку **Вперед**, чтобы передвинуть и точку воспроизведения, и ползунок «Сдвиг начала» вперед на один кадр;
- нажмите кнопку **Назад**, чтобы передвинуть и точку воспроизведения, и ползунок «Сдвиг начала» назад на один кадр;
- перетащите точку воспроизведения в новое место на линейке, а затем нажмите кнопку **Установить начальную точку в текущей позиции** (стрелка, повернутая вправо), чтобы переместить в то же место ползунок «Сдвиг начала» и укоротить линейку.

5. Чтобы переназначить последний кадр во внедряемом видеоклипе, проделайте те же операции, что в шаге 4, только с применением ползунка «Сдвиг конца» (треугольник справа от линейки) и кнопки **Установить конечную точку в текущей позиции** (стрелка, повернутая влево)

6. Чтобы применить новые параметры, нажмите кнопку **Создать клип** (знак плюс). Flash создаст клип с учетом переназначения начального и конечного кадров, присвоит ему имя по умолчанию и добавит в список клипов, предназначенных для импорта.

7. Нажмите кнопку **Continue** (Mac) или **Next** (Windows).

Появляется страница Encoding. Задайте параметры кодировки (см. следующий раздел).

8. Нажмите кнопку **Continue** (Mac) или **Next** (Windows).

Появляется страница Finish Video Import.

9. Нажмите кнопку **Finish**.

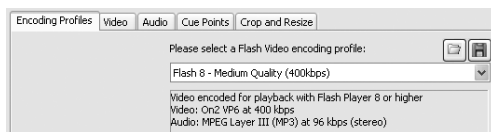
## Задание параметров кодирования

На странице мастера Encoding есть пять вкладок: **Profiles** (Профили) (Mac) или **Encoding Profiles** (Профили кодирования) (Windows), **Video** (Видео), **Audio** (Звукозапись), **Cue Points** (Ключевые точки), **Crop and Resize** (Кадрирование и изменение размера). Они одинаковы вне зависимости от того, разворачивается клип для прогрессивной или потоковой загрузки. Для внедренных клипов ключевые точки не задаются, а для клипов, которые редактировались до начала импорта, нет средств для обрезания. В остальном параметры кодирования для внедренных клипов такие же, как для клипов, загружаемых прогрессивно или потоком.

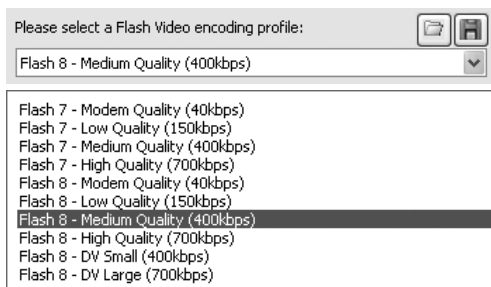
### Чтобы воспользоваться предопределенным набором параметров:

1. На странице Encoding нажмите кнопку **Profiles** (Mac) или перейдите на вкладку **Encoding Profiles** (Windows). Flash отображает список профилей (рис. 16.18).
2. Выберите из списка предопределенный профиль (рис. 16.19).

Flash помещает параметры из выбранного профиля на вкладки **Video** и **Audio**. Описание выбранных параметров приводится в сообщении под списком.



**Рис. 16.18.** На вкладке Profiles (Mac) или Encoding Profiles (Windows) предложен ряд предустановленных наборов параметров сжатия аудио и видео.



**Рис. 16.19.** В списке Flash Video Encoding Profile имеется десять предустановленных наборов параметров кодирования для оптимизации воспроизведения видео в различных версиях Flash Player при разной частоте кадров.

## О кодировании

При выборе параметров кодирования ваша цель (постоянная головная боль разработчиков для Web) — соблюсти баланс между качеством и размером файла. В случае видео это особенно важно, так как объем передаваемых данных весьма велик. Для представления каждого пикселя в видеоизображении требуется 24 бита. Если клип воспроизводится в квадрате 100x100 пикселей, то это уже получается 240 000 бит на каждый кадр, а для воспроизведения через Web необходима частота не менее 10 кадров/с. Таким образом, без сжатия каждая секунда видео обходится в миллионы бит данных.

Чтобы уменьшить этот объем до разумной величины, необходимо либо выбросить часть данных, либо воспользоваться сжатием. На вкладке **Profiles** страницы Encoding мастер предлагает ряд уже готовых наборов параметров для сжатия аудио и видео, подходящих для конкретных ситуаций. Если они вас не устраивают, то можете задать параметры самостоятельно на вкладках **Video** и **Audio**. Чтобы вообще выкинуть часть данных за счет обрезания клипа или уменьшения области просмотра, воспользуйтесь вкладкой **Crop and Resize**.

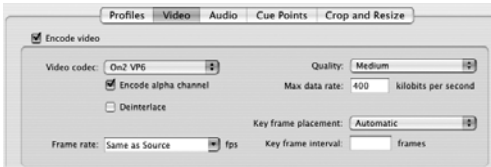


Рис. 16.20. На вкладке Video страницы Encoding можно задать параметры кодирования видеоданных.

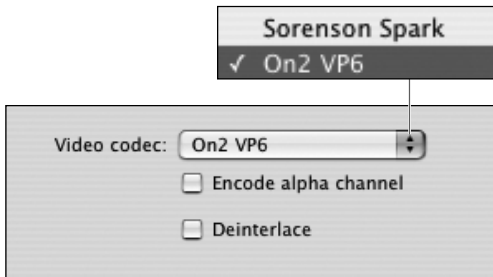


Рис. 16.21. Из списка Video Codec можно выбрать кодек, с помощью которого Flash будет кодировать видео. Кодеки Sorenson Spark и On VP6 создают видео без чередования кадров. Кодек On VP6 умеет также кодировать альфа-канал, если таковой присутствует в видеофайле.

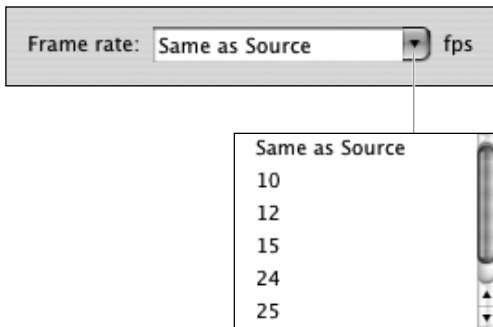


Рис. 16.22. По умолчанию применяется та же частота, что в оригинальном видеофайле (Same As Source — Как в источнике). Эту настройку рекомендуется оставить, если вы раньше не имели дела с видео.

## Чтобы самостоятельно задать параметры кодирования видео:

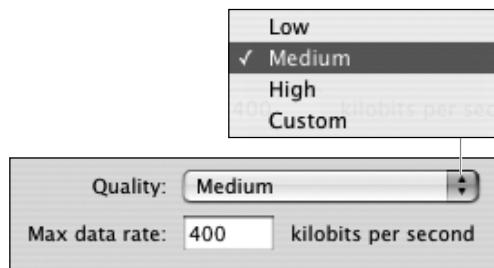
1. На странице Encoding нажмите кнопку **Video** (Mac) или перейдите на вкладку **Video** (Windows). Появляются параметры кодирования видео (рис. 16.20).
2. Чтобы Flash выполнял кодирование видео, отметьте флажок **Encode Video** (Кодировать видео). Если вы хотите кодировать только аудио, сбросьте этот флажок, но отметьте флажок **Encode Audio** (Кодировать аудио) на вкладке **Audio**.
3. Чтобы задать кодек — программу, которая собственно и выполняет кодирование, — выберите из списка **Video Codec** (Видеокодек) (рис. 16.21) один из следующих пунктов:
  - чтобы создать SWF-файл для воспроизведения в Flash-плеере версий 6 или 7, выберите пункт **Sorenson Spark**. Чтобы создать видео без чередования кадров, отметьте флажок **Deinterlace** (Сменить очередность);
  - чтобы создать SWF-файл для воспроизведения в Flash-плеере версий 8 или 9, выберите пункт **On2 VP6**. Чтобы включить в видео альфа-канал (если он присутствует), отметьте флажок **Encode Alpha Channel** (Кодировать альфа-канал). Если при этом нужно отключить чередование, отметьте флажок **Deinterlace**;
4. Чтобы задать частоту кадров, с которой Flash воспроизводит видеоклип, выберите значение из списка **Frame Rate** (рис. 16.22).

5. Чтобы установить максимальный битрейт (число бит, необходимых для кодирования одной секунды видео) (рис. 16.23), выполните любое из следующих действий:

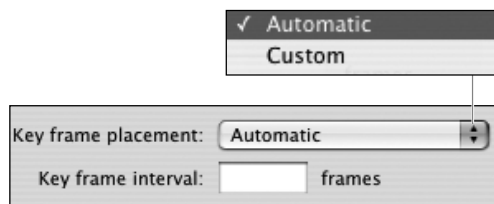
- выберите предустановленный битрейт (Low, Medium, High — Низкий, Средний, Высокий) из списка **Quality** (Качество); соответствующая числовая величина заносится в поле Max Data Rate (Максимальная скорость потока);
- введите значение в поле Max Data Rate. (На платформе Mac вы можете предварительно выбрать в списке **Quality** пункт Custom, а в Windows необходимо сначала ввести значение, после чего в списке будет автоматически выбран пункт Custom.) Чтобы зафиксировать новое значение, щелкните мышью вне поля ввода. (На платформе Mac нажатие клавиши **Enter** в момент, когда поле ввода активно, стирает значение в этом поле; в Windows нажатие **Enter** приводит к переходу на следующую страницу мастера, но введенное значение сохраняется.)

6. Чтобы определить, как часто кодек должен сохранять данные, необходимые для полной перерисовки кадра (*опорный кадр*), выберите из списка **Key Frame Placement** (Положение ключевого кадра) один из следующих пунктов:

- **Automatic** (Автоматически). Данные для создания опорного кадра сохраняются через каждые две секунды экранного времени. (Если исходный клип воспроизводится с частотой 30 кадров/с, то в этом режиме опорный кадр будет создаваться через каждые 60 кадров) (рис. 16.24);
- **Custom** (Заказной). Данные сохраняются с другим интервалом. В поле Key Frame Interval (Интервал между ключевыми кадрами) введите число кадров между опорными. Чем меньше интервал, тем больше будет создано опорных кадров, что приводит к увеличению размера ролика. (На платформе Mac вы можете предварительно выбрать в списке **Key Frame Placement** пункт Custom, а в Windows необходимо сначала ввести значение, после чего в списке будет автоматически выбран пункт Custom.)

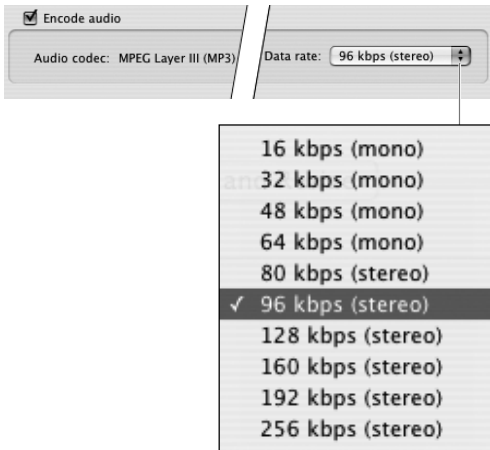


**Рис. 16.23.** Битрейт оказывает прямое влияние на размер SWF-файла и качество видеозображения. Чем выше битрейт, тем качественнее изображение и тем больше размер файла. Для больших файлов и низкой скорости соединения воспроизведение может приостанавливаться, поскольку данные не успевают поступать.



**Рис. 16.24.** Кодек может создавать опорные кадры видео. В каждом опорном кадре сохраняются все данные, необходимые, чтобы нарисовать изображение. В остальных кадрах сохраняются только изменения относительно предыдущего кадра.





**Рис. 16.25.** Мастер импорта видео использует только кодек MP3 для кодирования звуковой дорожки клипа. Чтобы аудиоданные кодировались, отметьте флажок **Encode Audio**. (Если хотите вообще отказаться от звуковой дорожки, сбросьте этот флажок.) Можно также задать битрейт для кодирования аудио.

### Чтобы самостоятельно задать параметры кодирования аудио:

1. На странице Encoding нажмите кнопку **Audio** (Mac) или перейдите на вкладку **Audio** (Windows). Появляются параметры кодирования аудио (рис. 16.25). Во Flash аудиоданные всегда кодируются в формате MP3.
2. Для кодирования звуковой дорожки клипа отметьте флажок **Encode Audio** (Кодировать аудио). Если этот флажок сброшен, то в импортированном клипе не будет звуковой дорожки.
3. В списке **Data Rate** (Скорость потока) выберите один из шести режимов стереофонического или один из четырех режимов монофонического звучания. Чем выше значение, тем качественнее кодируется звуковая дорожка.

## О профилях

Заданные параметры кодирования видео и аудиоданных можно сохранить для применения к другим видеоклипам. Набор сохраненных параметров называется *профилем*. На вкладке **Profiles** (Mac) или **Encoding Profiles** (Windows) страницы Encoding нажмите кнопку **Сохранить профиль кодирования** (значок с изображением дискеты). Открывается диалоговое окно Save Encoding Profile. Перейдите в папку, где хотите сохранить профиль, в поле Save As (Mac) или File Name (Windows) введите имя профиля и нажмите кнопку **Save** (Сохранить). Набор параметров сохраняется в формате XML.

Для загрузки сохраненного профиля нажмите кнопку **Загрузить профиль кодирования** на той же вкладке (значок папки). Открывается диалоговое окно Load Encoding Profile. Перейдите в папку, где хранится профиль, выберите его и нажмите кнопку **Open** (Открыть). Flash носит значения параметров в различные поля на вкладках **Video** и **Audio** страницы Encoding.

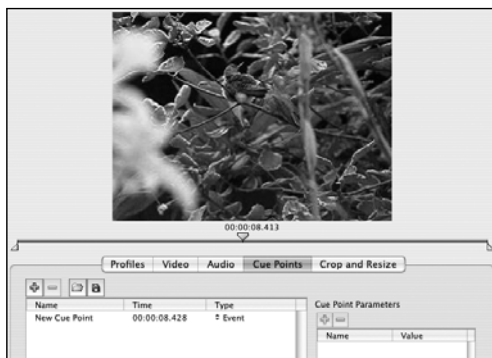
### Чтобы задать ключевые точки для прогрессивного или потокового видео:

1. На странице Encoding нажмите кнопку **Cue Points** (Mac) или перейдите на вкладку **Cue Points** (Windows). Появляются инструменты для создания ключевых точек (рис. 16.26).
2. Перетащите точку воспроизведения в то место, где должна располагаться ключевая точка.
3. Нажмите кнопку **Добавить ключевую точку** (значок плюс) (рис. 16.27).

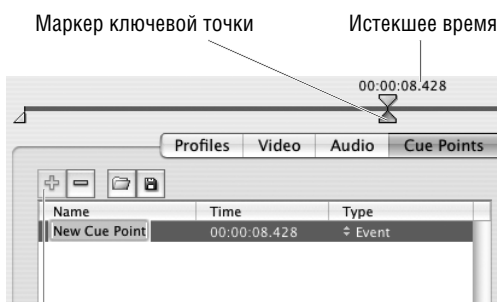
Flash помещает маркер ключевой точки на линейку, присваивает точке имя по умолчанию, добавляет данные о ней в список ключевых точек и активирует поле для ввода имени.

4. Введите имя ключевой точки.
5. В колонке Type (Тип) щелкните по стрелочке, чтобы раскрыть список, и выберите один из следующих пунктов:

- **Event** (Событие). Создается ключевая точка, которая может запускать ActionScript-сценарий.
- **Navigation** (Навигация). Для созданной ключевой точки можно написать сценарий, который заставит ее работать как закладку в видеоклипе. Он позволяет зрителю перейти сразу к кадру, соответствующему этой точке. Для таких ключевых точек кодек создает опорные кадры. Навигационные ключевые точки тоже могут запускать сценарий.



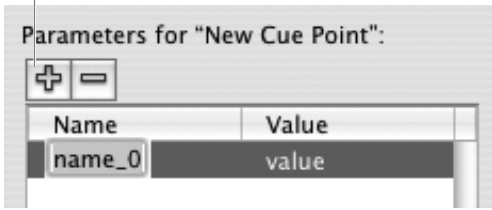
**Рис. 16.26.** На вкладке Cue Points страницы Encoding некоторые кадры можно пометить как ключевые точки. Затем из ActionScript-сценария можно перейти сразу к такому кадру или синхронизировать события с некоторыми участками видео.



Кнопка «Добавить ключевую точку»

**Рис. 16.27.** Щелкните по значку плюс, чтобы добавить ключевую точку в список. Маркер обозначает положение ключевой точки на линейке. Буксируя точку воспроизведения, вы можете выбрать ключевую точку визуально, наблюдая за изображением в области предварительного просмотра видео, или ориентируясь на истекшее время, которое отображается под ней. Время представлено в формате часы:минуты:секунды:миллисекунды.

Кнопка «Добавить параметры»



**Рис. 16.28.** Нажмите кнопку Добавить параметры, чтобы задать доступные из ActionScript-сценария параметры выбранной ключевой точки

6. Для задания параметров ключевой точки (доступных из сценария) выберите ее в списке и нажмите кнопку **Добавить параметры** (значок плюс) над списком параметров.

Flash заполняет колонки Name (Имя) и Value (Значение) по умолчанию (рис. 16.28). Чтобы изменить то или другое, дважды щелкните мышью по соответствующей колонке и введите новое значение.

**С**

Щелкните мышью в любом месте линейки, чтобы установить там точку воспроизведения.

**С**

Чтобы связать ключевую точку с определенным моментом времени, пользуйтесь клавишами со стрелками влево и вправо. Нажатие стрелки влево уменьшает время на одну миллисекунду, нажатие стрелки вправо увеличивает на столько же.

**С**

Для изменения существующей ключевой точки выберите ее в списке и перетащите маркер в то место на линейке, где она должна находиться.

**С**

Можно сохранить текущий список ключевых точек с параметрами в виде XML-файла. Затем этот файл можно загрузить для создания ключевых точек в тех же местах другого видеоклипа. Щелкните по значку с изображением дискеты, откроется диалоговое окно Save Cue Points File. Для загрузки XML-файла щелкните по значку папки и в диалоговом окне Load Cue Points File перейдите к сохраненному файлу.

### О качестве и битрейте

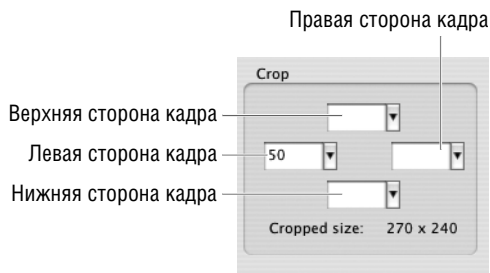
Параметры Quality и Max Data Rate тесно взаимосвязаны. Чтобы получить высокое качество, необходимо сохранять больше данных. Параметр Max Data Rate определяет, сколько данных будет в сжатом видео. Чем меньше битрейт, тем меньше данных кодек использует для кодирования одного кадра. Значение Low соответствует битрейту 150 килобит/с, значение Medium — битрейту 400 килобит/с, то есть более чем в два раза выше. Видео среднего качества загружается дольше низкокачественного, зато оно ближе к оригиналу.

### Чтобы изменить размер и продолжительность видео:

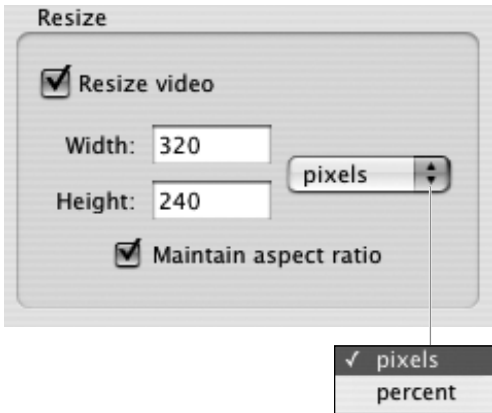
1. На странице Encoding нажмите кнопку **Crop and Resize** (Mac) или перейдите на вкладку **Crop and Resize** (Windows). Появляются инструменты для редактирования видео (рис. 16.29).
2. Чтобы обрезать ненужные участки видео, введите в разделе Crop (Кадрирование) значения, соответствующие четырем сторонам изображения (рис. 16.30). Пунктирная линия в области предварительного просмотра показывает, где именно кодек обрежет изображение.
3. Чтобы увеличить или уменьшить кадр видео, отметьте флажок **Resize Video** (Изменить размер видео) и выполните одно из следующих действий (рис. 16.31):
  - для ввода точных размеров выберите из списка пункт **Pixels** (Пиксели) и задайте значения в полях **Width** (Ширина) и **Height** (Высота);
  - чтобы уменьшить или увеличить кадр относительно исходных размеров, выберите из списка пункт **Percent** (Процент) и задайте значения в полях **Width** и **Height**;
  - чтобы сохранить исходные пропорции кадра, отметьте флажок **Maintain Aspect Ratio** (Сохранить пропорции).



**Рис. 16.29.** На вкладке Crop and Resize страницы Encoding можно кадрировать участок изображения, изменить размеры кадра и уменьшить продолжительность видео.



**Рис. 16.30.** Чтобы скадрировать изображение, задайте значение в соответствующих полях. Так, чтобы убрать 50 пикселей слева, введите число 50 в левое поле.



**Рис. 16.31.** Для уменьшения или увеличения кадра служат элементы в разделе **Resize**. Изменять можно абсолютные размеры (выберите в списке **Pixels**) или относительные, в процентах от размера оригинала (**Percent**).

4. Чтобы изменить продолжительность клипа, перетащите треугольнички по обе стороны линейки, так чтобы они отмечали новый начальный и конечный кадр.

В процессе перетаскивания продолжительность клипа, отображаемая под областью предварительного просмотра, изменяется. Полная продолжительность клипа, а также время до начального и после конечного кадра отображаются в разделе **Trim** (Обрезка).

**С**

Часто кадрировать изображение проще всего визуально. При щелчке по треугольничку справа от какого-нибудь из полей кадрирования появляется ползунок. Буксируйте его и наблюдайте за положением пунктирной линии в области предварительного просмотра, которая отмечает, где кадр будет обрезан.

**С**

Как правило, чтобы обрезать видеоклип, обязательно заходить на вкладку **Crop and Resize**. Если вы не собираетесь внедрять клип, то все инструменты — линейка, точка воспроизведения, ползунки начала и конца — присутствуют на всех вкладках страницы **Encoding**. Буксировка ползунков обрезает клип. У обрезания на вкладке **Crop and Resize** есть то преимущество, что можно видеть точное время начального и конечного кадра, а заодно и клипа в целом.

## Работа с внедренным видео

В зависимости от параметров, заданных мастеру при создании внедренного видеоклипа, Flash либо помещает экземпляр клипа на арену, либо просто оставляет его в библиотеке. Если экземпляр помещается на арену автоматически, то Flash может создать на временной шкале столько кадров, сколько необходимо для показа видео целиком, или разместить лишь такое число кадров видеоклипа, какое помещается в последовательности текущего опорного кадра. Если вы помещаете экземпляр внедренного клипа на временную шкалу самостоятельно, то должны сами создать последовательность опорного кадра необходимой длины.

### Чтобы поместить внедренный видеоклип на временную шкалу:

1. Откройте документ Flash и импортируйте в него видео, как описано в разделе «Импорт внедренного видео» выше.
2. Выберите на временной шкале опорный кадр, в котором должно начаться воспроизведение видео.



**Рис. 16.32.** Чтобы поместить внедренный видеоклип во Flash-ролик, перетащите экземпляр клипа с библиотечной панели на арену.

## О внедренном видео

Внедренные видеоклипы чем-то напоминают некоторые символы Flash, но тем не менее являются совершенно самостоятельными элементами. Подобно анимированным графическим символам, внедренные клипы воспроизводятся внутри кадров на главной временной шкале ролика и должны с ними синхронизироваться. Подобно аудиоклипам, внедренные видеоклипы могут содержать звук, хотя если он не был импортирован как интегрированная звуковая дорожка, то волновых форм на временной шкале вы не увидите. Как и любой символ, экземпляр видеоклипа можно перетащить с библиотечной панели на арену. В инспекторе свойств показывается информация о выбранных экземплярах внедренных видеоклипов. К выбранному экземпляру видеоклипа применимы многие обычные для Flash способы модификации объектов. Например, с помощью инструмента «Свободное преобразование» можно изменять размеры видеоизображения, поворачивать и перекашивать его. Если вы хотите, чтобы внедренный видеоклип воспроизводился в независимой временной шкале, или вам нужна та же степень контроля над внешним видом видеоклипа, какая предоставляется для символов-клипов (к примеру, возможность изменять яркость, тон или прозрачность), то следует поместить видеоклип внутри символа-клипа Flash.

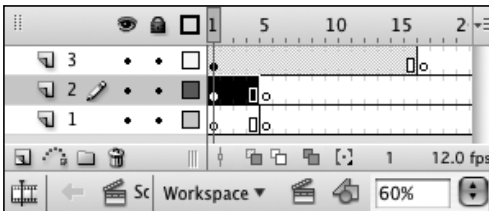


**Рис. 16.33.** Flash предупреждает, что вы пытаетесь добавить внедренный видеоклип, в котором больше кадров, чем в последовательности того опорного кадра, куда вы помещаете клип.

Перед добавлением внедренного видеоклипа



После добавления внедренного видеоклипа



**Рис. 16.34.** В третьем слое на этой временной шкале показано, что произошло после ответа Yes на предупреждение о слишком короткой последовательности опорного кадра. Flash добавил достаточно кадров, чтобы воспроизвести весь 16-кадровый клип. Во втором слое показан результат после ответа No: короткая последовательность опорного кадра обрезаает видео.

3. Перетащите копию внедренного видеоклипа с библиотечной панели на арену (рис. 16.32).

В отличие от символов-клипов, которые воспроизводятся на независимой временной шкале, внедренный видеоклип должен располагаться в кадрах главной шкалы ролика или шкалы содержащего его символа-клипа. Если последовательность выбранного опорного кадра содержит меньше кадров, чем видеоклип, выдается предупреждение и предлагается добавить необходимое число кадров (рис. 16.33).

4. Чтобы расширить последовательность опорного кадра, нажмите **Yes**.

Flash добавит столько кадров, сколько необходимо для воспроизведения клипа от начала до конца (рис. 16.34).

или

Если вы не хотите изменять число кадров, ответьте **No**. Flash все равно поместит видеоклип на арену, но не поместившиеся кадры показываться не будут.

### Предварительный просмотр внедренного видеоклипа

В среде разработки Flash кадры внедренных видеоклипов показываются в последовательности опорного кадра, содержащего клип. Вы можете просто воспроизвести ролик (команда **Control ⇒ Play**) или протянуть точку воспроизведения по всей временной шкале, чтобы посмотреть, как сменяются кадры видео. Однако, если во внедренном клипе есть звуковая дорожка, то для прослушивания звука необходимо воспользоваться одним из тестовых режимов. (Если на этапе импорта звуковая дорожка была преобразована в отдельный актив и помещена в библиотеку, то звук воспроизводится также при буксировке точки воспроизведения вдоль шкалы и при выполнении команды **Control ⇒ Play**). Тестовый режим необходим также для просмотра внедренного видео в контексте интерактивных элементов, например, символов-клипов или запрограммированных кнопок.

## Работа с компонентом FLVPlayback

При импорте видео, загружаемого прогрессивно или потоком, мастер добавляет в библиотеку компонент FLVPlayback и помещает его экземпляр на арену. Вы можете заменить стандартный значок компонента кадром из видеоклипа, изменить исходный файл и задать другие параметры.

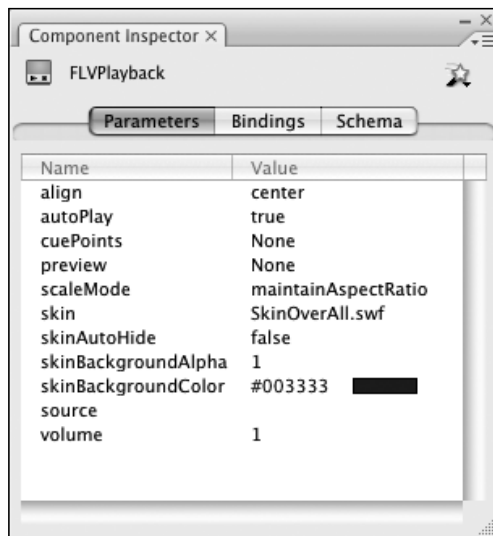
### Чтобы выбрать источник видео для экземпляра FLVPlayback:

1. Выберите экземпляр компонента FLVPlayback на арене.
2. Перейдите на панель инспектора компонентов. Если панель еще не открыта, выберите из меню пункт **Window** ⇒ **Component Inspector**.
3. Нажмите кнопку **Parameters** (Mac) или перейдите на вкладку **Parameters** (Windows). Ассоциированные с компонентом параметры отображаются в виде таблицы с двумя колонками (рис. 16.35).
4. Чтобы выбрать файл, который будет показываться в области видео во время воспроизведения, щелкните по параметру Source в колонке Name (Имя).

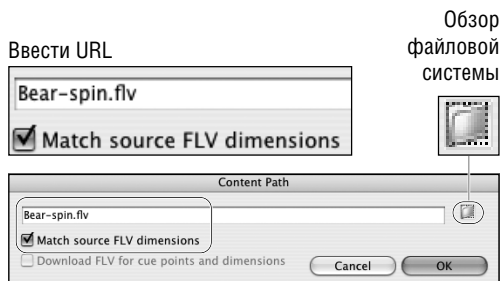
В колонке Value (Значение) появляется значок поиска (лупа).

5. Щелкните по значку поиска. Открывается диалоговое окно Content Path (Путь к содержимому).
6. Чтобы выбрать исходный видеофайл (он должен быть в формате FLV), выполните одно из следующих действий:

- введите URL файла (рис. 16.36);
- если файл находится в вашем компьютере, щелкните по значку папки справа от поля ввода. В диалоговом окне Browse for FLV File найдите нужный файл и нажмите кнопку **Open**. Flash поместит URL файла в поле ввода.

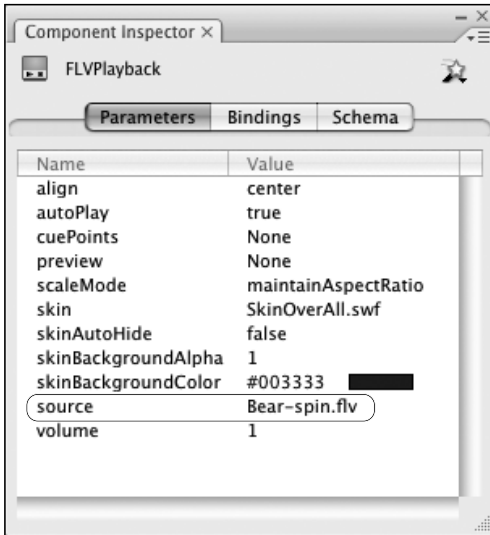


**Рис. 16.35.** Вкладка Parameters на панели инспектора компонентов содержит все параметры выбранного экземпляра компонента FLVPlayback.



**Рис. 16.36.** В диалоговом окне Content Path вы можете выбрать видеофайл, который компонент FLVPlayback будет демонстрировать конечному пользователю.





**Рис. 16.37.** Значение в колонке Value параметра Source — это имя файла, который будет демонстрироваться конечным пользователям, просматривающим ролик.

7. Для задания того, как Flash должен установить размер экземпляра компонента FLVPlayback, выполните одно из следующих действий:

- чтобы размер компонента соответствовал исходному видео, отметьте флажок **Match Source FLV Dimensions** (Подобрать измерения исходного FLV);
- чтобы подогнать размер видео под стандартные размеры компонента FLVPlayback (320 x 240 пикселей), сбросьте этот флажок.

8. Нажмите **ОК**.

Flash записывает в строку таблицы Source имя выбранного файла (рис. 16.37). В зависимости от режима, указанного на шаге 7, Flash также изменяет размеры экземпляра компонента FLVPlayback на арене. Теперь этот экземпляр ссылается на выбранный закодированный видеофайл.

**C**

Если тема, выбранная для экземпляра FLVPlayback, предусматривает наличие элемента управления громкостью, то можно задать начальное положение ползунка. Дважды щелкните по значению в строке Volume и введите число; 1 означает, что ползунок будет в крайней правой позиции (полная громкость), значения от 0.9 до 0.1 смещают ползунок влево (громкость уменьшается), а значение 0 соответствует крайней левой позиции, когда звук не слышен вовсе.

**C**

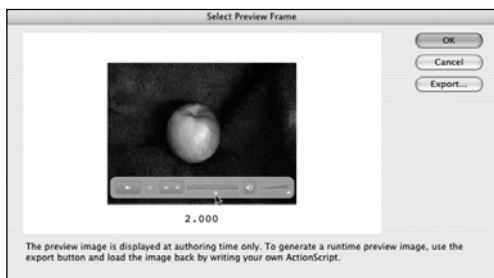
Чтобы панель управления появлялась только тогда, когда пользователь наведет курсор на окно воспроизведения видео, выберите значение True из списка в колонке Value строки SkinAutoHide.

### О компоненте FLVPlayback

Мастер импорта видео кодирует видеофайл и импортирует его в виде, пригодном для использования во Flash-ролике. Если задано развертывание в виде прогрессивного или потокового видео, то слово *импорт* не совсем точно. На самом деле мастер создает новый внешний файл, содержащий видеоданные (FLV-файл). Он хранится отдельно от FLA и SWF-файла, но между ними создается связь, опосредованная компонентом *FLVPlayback*. В процессе «импорта» мастер копирует этот компонент в библиотеку Flash-документа, помещает его экземпляр на арену и устанавливает параметр *Source* так, чтобы тот указывал на FLV-файл, содержащий кодированные видеоданные. Во время выполнения компонент *FLVPlayback* создает внутри ролика окно для демонстрации видео из этого файла.

Компоненты — это особый вид элементов Flash. (В главе 12 вы познакомились с компонентом-кнопкой.) В состав компонента входит сценарий на языке *ActionScript*, который управляет его поведением, но вообще-то поведение компонента можно модифицировать, и не прибегая к программированию; следует лишь изменить параметры на панели инспектора компонентов или на вкладке **Parameters** в инспекторе свойств. Компоненты пользовательского интерфейса, написанные на *ActionScript 3.0*, например кнопка, компилируются не полностью. Графические элементы, определяющие их внешний вид (тему), можно изменять точно так же, как любую другую графику. В отличие от интерфейсных компонентов, *FLVPlayback* откомпилирован полностью, просто так изменить его тему невозможно. Компонент *FLVPlayback* поставляется с набором готовых тем для создания панели управления окном видео. Задать стиль панели управления можно, выбрав тему для конкретного экземпляра компонента. (Можно и самостоятельно создавать для него новые темы, но это уже выходит за рамки настоящей книги.)

Подобно прочим активам, компонент *FLVPlayback* допускает повторное использование. Чтобы включить в ролик второе окно видео, перетащите еще один экземпляр *FLVPlayback* на арену и присвойте параметру *Source* путь к FLV-файлу, который хотите показывать. Если видеофайлы уже подготовлены в формате FLV, можно даже обойтись без мастера импорта. Зайдите на панель компонентов (чтобы ее открыть, можно выполнить команду **Window [Components]**). Раскройте папку *Video* и перетащите экземпляр *FLVPlayback* в документ. Параметр *Source* этого экземпляра пока пуст, но вы можете присвоить ему путь к FLV-файлу, как было описано выше.



**Рис. 16.38.** При наведении курсора на область предварительного просмотра видео появляется панель управления. Буксируя ползунок, выберите кадр, который экземпляр компонента FLVPlayback будет показывать на этапе разработки.

### Чтобы выбрать кадр видео, показываемый на этапе разработки:

1. Выполните шаги 1–3 из предыдущего упражнения.
2. В инспекторе компонентов дважды щелкните по колонке Value в строке Preview parameter, где по умолчанию выбрано значение None (Нет).

Открывается диалоговое окно Select Preview Frame (Выбор кадра контрольного просмотра). В зависимости от продолжительности видео может отображаться сообщение «Loading...» (Загрузка), пока Flash читает FLV-файл. Для предварительного просмотра видео воспроизводится один раз.

3. Наведите курсор на область предварительного просмотра. Появляется панель управления (рис. 16.38).
4. Буксируя треугольный ползунок, перейдите к конкретному кадру.

Под область просмотра показывается экранное время. Чтобы повторно воспроизвести видео, нажмите кнопку Play (Воспроизведение).

### Анимация компонента FLVPlayback

Компонент FLVPlayback аналогичен символу-клипу в том смысле, что это актив, хранящийся в библиотеке. Его можно модифицировать инструментом «Свободное преобразование», можно маскировать содержимое с помощью слоя-маски. На арене он выглядит как статическое изображение, а воспроизводится только во Flash-плеере. Но есть и важное отличие: экземпляр компонента FLVPlayback нельзя использовать в интерполяции движения. Вы можете поместить экземпляр компонента в два опорных кадра, изменить экземпляр во втором кадре и задать для последовательности первого опорного кадра свойство Tween, равное Motion. Но при тестировании ролика сразу же убедитесь, что интерполяция не работает. Окно видео не меняет местоположения, а изменения, произведенные во втором кадре, не видны.

Если вы хотите, чтобы окно видео, созданное компонентом, перемещалось по арене (или происходила еще какая-то анимация) в процессе интерполяции движения, то сначала поместите экземпляр компонента FLVPlayback в символ-клип. Теперь путем анимации самого символа-клипа вы можете достичь желаемого эффекта. (О процедуре создания символов-клипов см. главу 11, об интерполяции движения — главу 9.)

5. Когда нужный кадр видео появится в области предварительного просмотра, нажмите **OK**.

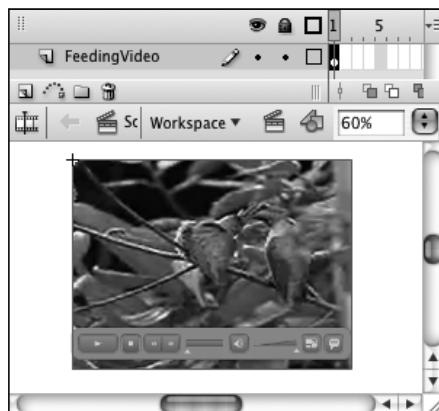
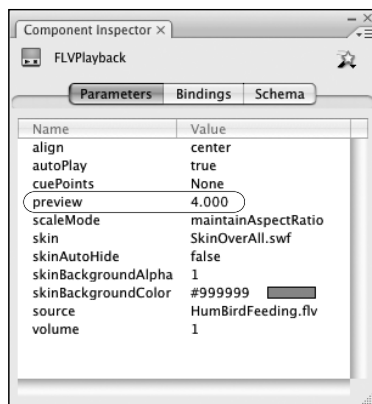
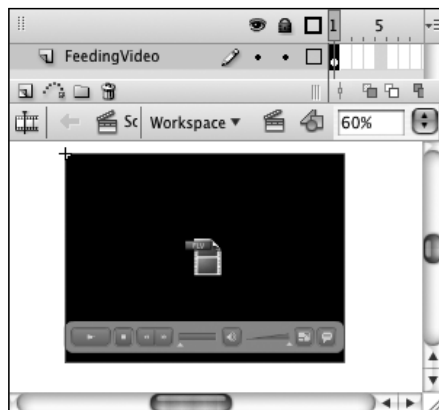
Выбранный кадр отображается в экземпляре компонента на арене, а в параметре **Preview** в инспекторе компонентов показывается время от начала клипа до этого кадра (рис. 16.39).

C

Вместо того чтобы использовать готовые темы панели управления, встроенные в **FLVPlayback**, вы можете сконструировать пользовательский интерфейс из других компонентов, предназначенных для управления воспроизведением видео. Задайте для экземпляра компонента **FLVPlayback** на арене параметр **skin** равным **None**. Перейдите в инспектор свойств и раскройте папку **Video**. Там есть целый ряд интерфейсных компонентов: **PlayButton**, **PauseButton**, **VolumeBar** и т.д. Перетащите на арену по экземпляру каждого элемента. Их можно расположить, где угодно, и настроить точно так же, как компонент-кнопку (см. главу 12).

C

С помощью инструмента «Свободное преобразование», панели преобразований или информации, а также вкладки **Properties** в инспекторе свойств можно изменить размер экземпляра **FLVPlayback**. Но, если вы выбрали для своего видео тему панели управления, будьте осторожны. Эта панель должна быть достаточно широкой, чтобы уместились все интерфейсные элементы (кнопки «Воспроизведение», «Пауза» и «Без звука», ползунок регулировки громкости и т.д.). Если вы сделаете компонент слишком узким, то панель управления может вылезти за пределы области просмотра видео.



**Рис. 16.39.** По умолчанию параметр **Preview** компонента **FLVPlayback** равен **None**, и на экране представлено стандартное изображение (верхний рисунок). Если в качестве значения задать промежуток времени от начала клипа до кадра (средний рисунок), то этот кадр и будет показываться на арене во время разработки (нижний рисунок).



**Рис. 16.40.** Для изменения набора элементов, управляющих воспроизведением видео, предназначено диалоговое окно Select Skin.

### Чтобы изменить тему панели управления:

1. Выполните шаги 1–3 из раздела «Чтобы выбрать источник видео для экземпляра FLVPlayback» выше.
2. В инспекторе компонентов дважды щелкните в колонке Value строки Skin.

Открывается диалоговое окно Select Skin (Выбрать тему), в котором показано все то же, что на странице загрузки темы в мастере импорта видео (рис. 16.40).

3. Выберите в списке **Skin** новую тему.

Детали см. в описании шага 8 первого упражнения в разделе «Импорт видео для прогрессивной или потоковой загрузки».

4. Если хотите, выберите с помощью элемента Color (Цвет) другой цвет фона для панели управления.

### Завершающие шаги при работе с компонентом FLVPlayback

Опубликованный Flash-ролик в виде SWF-файла выкладывается на Web-сервер (публикации посвящена глава 17). Если для доставки прогрессивного или потокового видео используется компонент FLVPlayback, то на сервер необходимо выложить еще два файла: FLV-файл, содержащий закодированное видео, и SWF-файл с темой панели управления (если вы ее включили). FLV-файл Flash помещает в ту же папку, где находится FLA-файл. И там же сохраняются опубликованные SWF-файлы самого ролика и темы панели управления.

В процессе разработки вы можете не раз публиковать свой ролик для целей тестирования. На этом этапе все необходимые файлы находятся в одной папке, и Flash может спокойно пользоваться относительными именами. Но перед тем, как публиковать окончательный ролик для доставки зрителям, убедитесь, что параметр source каждого экземпляра FLVPlayback содержит правильный URL исходного видеофайла на сервере (о том, как задавать этот параметр см. раздел «Чтобы выбрать источник видео для экземпляра FLVPlayback»).

## 5. Нажмите ОК.

Flash изменяет значение параметра Skin на вкладке **Parameters** в инспекторе компонентов и обновляет панель управления в экземпляре компонента FLVPlayback на арене.

**C**

Чтобы изменить цвет панели управления компонентом FLVPlayback, выберите его экземпляр на арене, перейдите в инспектор компонентов и нажмите кнопку **Parameters** (Mac) или перейдите на вкладку **Parameters** (Windows). Дважды щелкните по шестнадцатеричному значению цвета или по цветной полоске в колонке Value строки SkinBackgroundColor и выберите новый цвет в окне образцов.

**C**

Для панели управления можно задать прозрачность. Щелкните в колонке Value строки SkinBackgroundAlpha и введите значение от 1 до 0 (1 — полностью непрозрачный цвет; 0.5 — прозрачность 50%, 0 — полностью прозрачный).

**C**

Вместо этого можно дважды щелкнуть по значению параметра в колонке Value с тем же результатом.

**C**

Параметры экземпляра компонента FLVPlayback можно изменить на вкладке **Parameters** в инспекторе свойств. Зайдите на эту вкладку (если она закрыта, выберите из главного меню пункт **Window** ⇒ **Properties** ⇒ **Parameters**). На ней присутствует таблица параметров. Чтобы изменить значение некоторого параметра, нужно, как и выше, щелкнуть по названию параметра в колонке Name, а затем по значению в соответствующей колонке Value. В результате откроется диалоговое окно или инструмент для ввода нового значения.

**C**

За одним исключением, новые значения параметров, присвоенные им в инспекторе компонентов, запоминаются. Если, например, изменить параметр skin, то в следующий раз, когда вы поместите на арену экземпляр компонента FLVPlayback, у него будет такая же тема. То же самое верно в отношении цвета, громкости, прозрачности и других параметров. Только параметр source принимает значение, заданное в мастере импорта видео, или остается пустым, когда вы помещаете экземпляр на арену самостоятельно.

# ДОСТАВКА РОЛИКА АУДИТОРИИ

---

17

Когда создание графики, анимации и интерактивного интерфейса закончено, наступает время доставить свое творение публике. Необходимо опубликовать, или экспортировать документ Flash (FLA-файл) в формате, пригодном для воспроизведения. На выбор предоставляется несколько форматов. Из них формат SWF, ориентированный на программу Flash Player, дает гарантию, что зритель увидит всю анимацию и сможет взаимодействовать с интерактивным роликом. Частичную интерактивность обеспечивает формат QuickTime.

Одновременно с Flash CS3 можно установить и версию 9 приложения Flash Player. SWF-файлы можно просматривать прямо в Flash-плеере, который работает на вашем же компьютере. Но вызывать его можно и из других программ, в частности, из Web-браузера.

Ролик можно экспортировать в виде последовательности изображений в растровом (GIF, PNG) или векторном формате (например, в формате Adobe Illustrator). Можно опубликовать ролик и в формате Flash Video (FLV). Еще один вариант доставки — подготовить самовоспроизводящийся файл, так называемый *проектор*. Чтобы открыть и воспроизвести такой ролик, пользователю достаточно дважды щелкнуть по нему мышью. И, наконец, можно распечатать ролик целиком или по кадрам, если вы захотите отдать кому-то его материальную версию (например, для раскадровки).

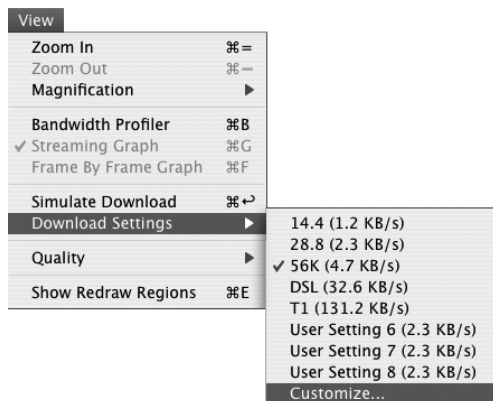
## Подготовка ролика для оптимального воспроизведения

При создании роликов, предназначенных для просмотра через Web, вы сталкиваетесь с проблемой качества и количества. Чем выше качество ролика, тем больше размер файла, что приводит к замедлению загрузки и воспроизведения. Размер файла увеличивается из-за растровых изображений (особенно анимированных), видеоклипов, звуковых файлов, наличия нескольких участков с одновременной анимацией, встроенных шрифтов, градиентов и использования отдельных графических элементов вместо символов и групп. Имеющаяся во Flash функция имитации потоковой доставки помогает найти места, в которых ролик «тормозит». Отчет о размерах и профилирование пропускной способности показывают, какие кадры могут привести к задержкам во время загрузки. По результатам вы сможете переделать или оптимизировать проблемные участки.

### Чтобы воспользоваться профилировщиком полосы пропускания:

1. Откройте документ Flash, предназначенный для просмотра через Web.
2. Выполните команду **Control** ⇒ **Test Movie** (Управление ⇒ Тестировать ролик) или **Test Scene** (Тестировать сцену). Flash экспортирует ролик и открывает его в Flash-плеере.
3. Из меню **View** (Просмотр) Flash-плеера выберите пункт **Download Settings** (Параметры загрузки) и задайте скорость передачи, при которой хотите протестировать ролик.

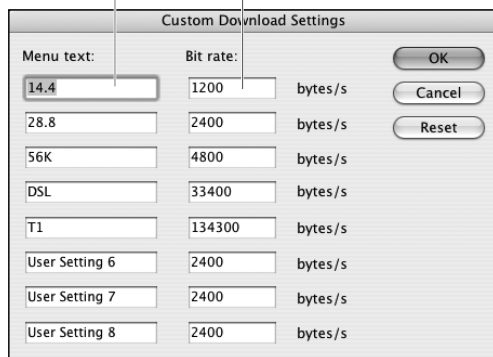
В меню предлагается восемь скоростей, но их можно настраивать. Для этого выполните команду **View** ⇒ **Download Settings** ⇒ **Customize** (Просмотр ⇒ Параметры загрузки ⇒ Настроить) (рис. 17.1). По умолчанию, Flash включает в список пять самых распространенных видов соединения: 14.4 Кбит/с, 28.8 Кбит/с, 56 Кбит/с, DSL и T1 — с параметрами, имитирующими реальные скорости передачи данных. Все эти настройки представлены в диалоговом окне Custom Download Settings (рис. 17.2). Чтобы создать собственный набор параметров, введите новые значения и нажмите **OK**.



**Рис. 17.1.** Для описания нестандартной скорости соединения в целях имитации воспроизведения через Web выберите из меню View пункт Download Settings ⇒ Customize.

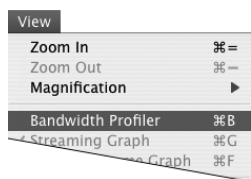
Это название отображается в меню

Имитируемая скорость соединения



**Рис. 17.2.** По умолчанию Flash предлагает настройки для имитации пяти стандартных скоростей соединения. В окне Custom Download Settings вы можете изменить их названия и параметры.

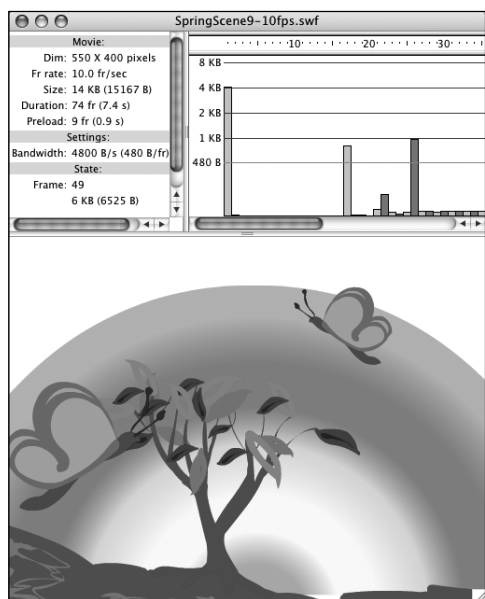




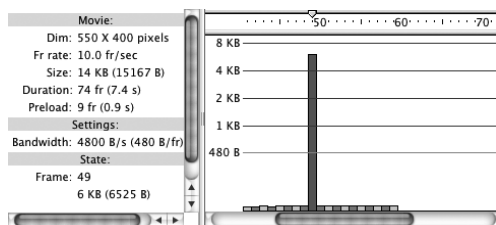
**Рис. 17.3.** Для просмотра диаграммы объема данных в каждом кадре выберите в окне Flash-плеера пункт меню View ⇒ Bandwidth Profiler.

Скорость загрузки не выходит за рамки заданной частоты кадров

Будет задержка при воспроизведении



**Рис. 17.4.** Профиль пропускной способности в верхней части окна Flash-плеера показывает, сколько данных содержит каждый кадр. Столбики на диаграмме соответствуют отдельным кадрам.



**Рис. 17.5.** В режиме покадровой диаграммы высота столбика отражает количество данных в кадре.

4. Выберите в меню **View** пункт **Bandwidth Profiler** (Профилировщик пропускной способности) (рис. 17.3).

В верхней части области тестирования ролика Flash строит диаграмму объема передаваемых данных в контексте временной шкалы (рис. 17.4). Столбики показывают, сколько байтов передается на каждый кадр. Нижняя красная линия показывает, сколько данных можно передать достаточно быстро для того, чтобы выдержать заданную частоту кадров. Если какой-то кадр содержит больше данных, то будут возникать задержки.

### Чтобы просмотреть содержимое каждого кадра по отдельности:

1. Из меню **View** Flash-плеера выберите пункт **Frame by Frame Graph** (Покадровая диаграмма) или нажмите комбинацию клавиш **⌘-F** (Mac) либо **Ctrl-F** (Windows).

На диаграмме профилировщика пропускной способности Flash будет показывать по одному столбику для каждого кадра. Числа на верхней линейке соответствуют номерам кадров (рис. 17.5). Высота столбика отражает количество данных в кадре.

2. Выберите столбик.

В окне профиля отображаются сведения о выбранном кадре и ролике в целом.

## Чтобы визуализировать поток кадров:

1. Из меню **View** Flash-плеера выберите пункт **Streaming Graph** (Потоковая диаграмма) или нажмите комбинацию клавиш **⌘-G** (Mac) либо **Ctrl-G** (Windows).

Flash представляет кадры в виде столбиков, окрашенных в светло- и темно-серый цвет. В этом варианте видно, сколько времени занимает загрузка каждого кадра (рис. 17.6). Линейка градуирована в единицах времени, зависящих от заданной частоты кадров. (Например, в ролике с частотой 10 кадров/с одно деление соответствует 0.1 с.) Там, где кадры содержат мало данных, в одну единицу времени может уместиться несколько столбиков. Наоборот, объемные кадры могут занимать несколько единиц времени.

2. Выберите столбик.

В окне профиля отображаются сведения о данном кадре и ролике в целом (рис. 17.7).

## Чтобы вывести индикатор загрузки:

- Запустив профилировщик пропускной способности, выберите из меню пункт **View** ⇒ **Simulate Download** (Просмотр ⇒ Имитация загрузки) или нажмите комбинацию клавиш **⌘-Return** (Mac) либо **Ctrl-Enter** (Windows).

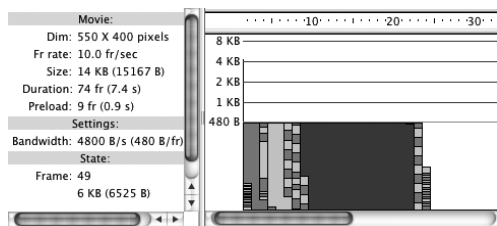
Пока в тестовом окне воспроизводится анимация, Flash подсвечивает числа на временной шкале зеленым цветом, показывая, какой кадр сейчас загружается.

## Чтобы выйти из профилировщика пропускной способности:

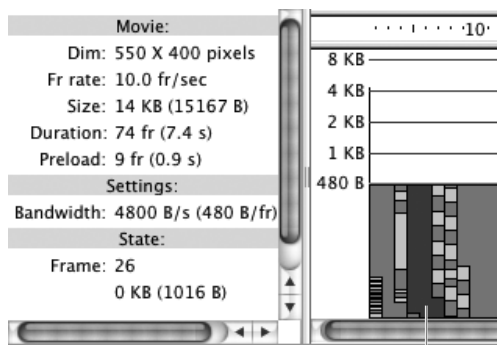
- Снова выберите из меню **View** пункт **Bandwidth Profiler**, чтобы прекратить профилирование.

## C

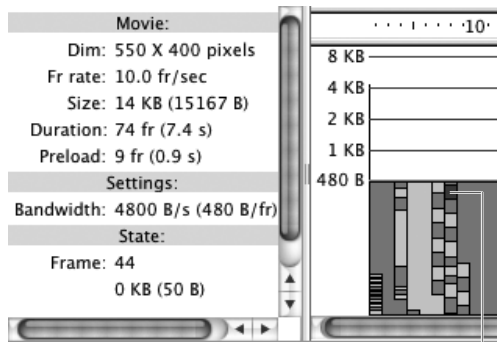
Чтобы получить информацию о количестве данных в каждом кадре в текстовом виде, отметьте флажок **Generate Size Report** (Сформировать отчет о размере) на вкладке **Flash** в диалоговом окне **Publish Settings** (Параметры публикации). В процессе публикации Flash создает файл, в котором показано, сколько байтов занимает каждый кадр ролика.



**Рис. 17.6.** В режиме вывода потоковой диаграммы ширина каждого столбика показывает, сколько времени будет загружаться соответствующий кадр при заданной скорости соединения и частоте кадров. В данном случае кадр 49 содержит 6 Кб; при скорости 56 Кбит/с и частоте 10 кадров/с он будет загружаться 1.3 с.



Кадр 26



Кадр 44

**Рис. 17.7.** В окне профиля слева от диаграммы приведена информация о ролике. В разделе **State** (Состояние) указан выбранный кадр (подсвеченный красным) и объем данных в нем. Как видите, кадр 26 (верхний рисунок) содержит 1016 байтов, а кадр 44 (нижний рисунок) — всего 50 байтов.

### Замечание о доступности

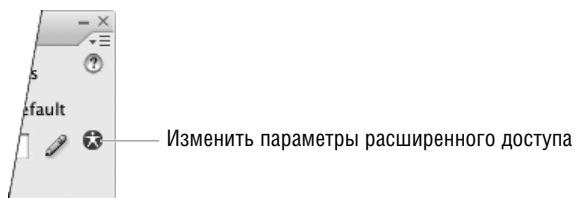
Размышляя о том, как лучше доставить свой Flash-ролик до аудитории, не забывайте, что среди зрителей могут быть и люди, не способные взаимодействовать с сайтом обычным образом. По мере того как сеть Web становится все более интересной визуально, люди с ослабленным зрением, которые тоже хотели бы иметь доступ ко многим ресурсам, оказываются в невыгодном положении.

Наше общество постепенно проникается осознанием того, что некоторые виды деятельности и ресурсы не рассчитаны на людей с ограниченными возможностями. Web-дизайнеры должны использовать Flash не только для создания сайтов, способных захватить внимание зрителя и блеснуть графикой, анимацией, интерактивностью, звуком и видео, но и таких сайтов, которые несут информацию широкому кругу людей. Помните, что некоторые пользователи не могут увидеть сайт или услышать его звуковое сопровождение. Есть люди, не способные работать с мышью, им приходится переходить от одного элемента к другому с помощью клавиши табуляции.

Учитывая необходимость обеспечить доступность сайтов, Flash позволяет оформить содержимое так, что его смогут воспринимать программы чтения с экрана, основанные на технологии Microsoft Active Accessibility (MSAA). (Когда разрабатывалась версия Flash CS3, технология MSAA была реализована только для Windows и поддерживалась лишь в браузере Internet Explorer.) Программы чтения с экрана информируют о содержимом страницы, например, произносят названия кнопок и содержимое текстовых полей. Встроенные во Flash средства обеспечения доступности позволяют снабжать объекты описаниями для считывателя, пресекать попытки считывателя описать определенные объекты (например, чисто декоративные символы-клипы) и назначать клавиши для манипулирования объектами.

Принципы построения по-настоящему доступного сайта слишком многочисленны и сложны, чтобы пытаться рассмотреть их в этой книге. Но вы можете оценить инструменты создания доступных объектов на панели Accessibility. Выберите из меню пункт **Window** ⇒ **Other Panels** ⇒ **Accessibility** (Окно ⇒ Другие панели ⇒ Расширенный доступ) или щелкните по значку с изображением человека в правом нижнем углу вкладки **Properties** в инспекторе свойств (рис. 17.8). На панели появляются параметры расширенного доступа для выбранных объектов.

Компания Adobe включила описание основ Web-дизайна, обеспечивающего доступность сайта, в состав справки (см. тему Using Flash ⇒ Best Practices ⇒ Accessibility Guidelines (Использование Flash ⇒ Рекомендованная практика ⇒ Рекомендации по обеспечению доступности)). Дополнительную информацию можно найти на сайте Adobe Accessibility Resource Center (<http://www.adobe.com/resources/accessibility/>).



**Рис. 17.8.** Щелчок по значку расширенного доступа (левый рисунок) на вкладке свойств документа в инспекторе свойств открывает панель расширенного доступа (правый рисунок). Она позволяет сделать выбранные объекты ролика доступными (или недоступными) программам чтения с экрана.



## Публикация

Чтобы ваш ролик смогли увидеть зрители, его необходимо опубликовать или экспортировать. Функция Publish ориентирована на представление материалов в Web и на создание сразу нескольких файлов в форматах для различных программ просмотра. У функции Export параметры аналогичны, но за раз она создает файл только в одном формате (см. врезку «Публикация и экспорт»).

### Чтобы задать формат публикации:

1. Откройте документ, который собираетесь опубликовать.
2. Выберите из меню пункт **File** ⇒ **Publish Settings** (Файл ⇒ Параметры публикации) или нажмите комбинацию клавиш **Option-Shift-F12** (Mac) либо **Ctrl-Shift-F12** (Windows) (рис. 17.9).

Открывается диалоговое окно Publish Settings. В верхней части находится список **Current Profile** (Текущий профиль) и кнопки для работы с профилями. Если вы работаете с новым документом и никогда раньше не создавали профилей, то единственным элементом в списке является Default (По умолчанию). Если открытый файл был подготовлен в предыдущей версии Flash, то в имени профиля отражен номер этой версии. Профиль представляет собой набор параметров, управляющих различными аспектами публикации. Для сохранения параметров можно создавать новые профили. Но пока просто оставьте текущий профиль.

3. Нажмите кнопку **Formats** (Mac) или перейдите на вкладку **Formats** (Форматы) (Windows) (рис. 17.10).
4. Выберите один из восьми форматов.

Поддерживаются следующие форматы: Flash (.swf), HTML (.html), Изображение GIF (.gif), Изображение JPEG (.jpg), Изображение PNG (.png), Windows Projector (.exe), Macintosh Projector и QuickTime с Flash-дорожкой (.mov).

5. Чтобы задать параметры для выбранного формата, перейдите на соответствующую ему вкладку (Windows) или нажмите соответствующую кнопку (Mac) (подробности описаны в упражнениях ниже).

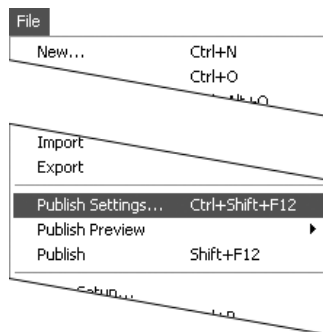


Рис. 17.9. Для задания параметров публикации ролика выберите из меню пункт **File** ⇒ **Publish Settings**.

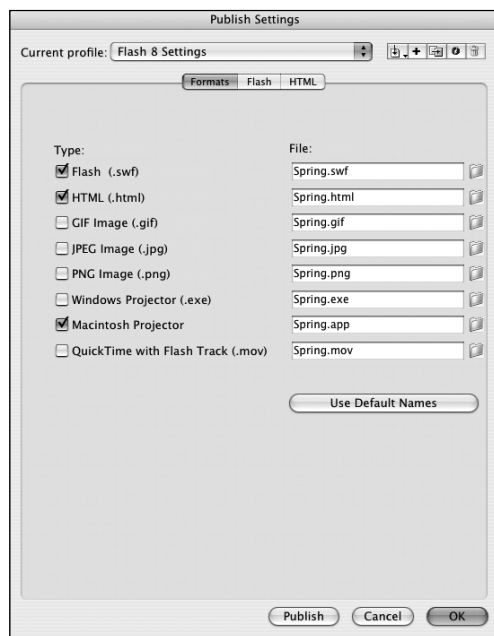


Рис. 17.10. На вкладке **Formats** в диалоговом окне Publish Settings можно задать параметры публикации сразу в нескольких форматах (не более семи). Восьмой формат предназначен для создания HTML-документа, необходимого для отображения опубликованных файлов в браузере.

6. Для сохранения параметров в текущем файле нажмите **ОК**.

Flash будет применять эти параметры при каждом выполнении команды **Publish** (Опубликовать) или **Publish Preview** (Просмотр публикации) для данного документа. Они же будут действовать при входе в тестовый режим (командой **Test Movie** (Тестировать ролик) или **Test Scene** (Тестировать сцену) из меню **Control**).

**C**

Если открыта вкладка свойств документа в инспекторе свойств, то для открытия диалогового окна **Publish Settings** достаточно нажать кнопку **Settings** (Параметры).

### Публикация и экспорт

При выполнении команды **Publish** заданные параметры публикации применяются для создания всех элементов, необходимых для отображения ролика в Web: SWF-файла для Flash-плеера, HTML-файла и JavaScript-файла. Последние два работают совместно, организуя среду, необходимую для отображения SWF-файла в окне браузера. Кроме того, в JavaScript-файле есть код, который проверяет, что на компьютере пользователя установлена версия Flash Player, позволяющая просматривать содержимое ролика. Команда **Publish** может создавать и файлы в других форматах: GIF, JPEG, PNG и QuickTime, а также HTML-код, необходимый для их отображения в браузере. Альтернативные форматы позволяют частично просматривать анимацию и взаимодействовать с роликом даже тем пользователям, у которых нет Flash-плеера. Наконец, с помощью команды **Publish** можно создавать автономные файлы-проекторы.

Команда **Export Movie** транслирует документ Flash только в один формат. Варианты выбора графических форматов такие, такие же, как при публикации: GIF, JPEG и PNG. Но расположение некоторых параметров в диалогах экспорта и публикации различается, а для ряда форматов количество параметров публикации больше. Например, в окне параметров публикации можно удалить из GIF-файла градиенты (чтобы уменьшить размер), тогда как при экспорте такой возможности нет. Экспорт в формат QuickTime существенно отличается от публикации. Публикация в этом формате разрешается только, если содержимое подготовлено для Flash Player версии 5 или младше, но зато сохраняются элементы интерактивности, например, кнопки и гиперссылки. При экспорте в формат QuickTime вся интерактивность теряется, но принимается анимация, созданная для любой версии Flash Player. Диалоговые окна для экспорта в формат QuickTime дают возможность контролировать результат гораздо точнее, чем в случае публикации. Экспорт в формат QuickTime работает следующим образом: SWF-файл воспроизводится в Flash-плеере, при этом все кадры анимации (в том числе и созданные из сценария) захватываются и выводятся в соответствующие кадры MOV-файла.

Еще одно различие между публикацией и экспортом заключается в том, что параметры публикации Flash сохраняются вместе с FLA-файлом для повторного использования. Параметры же экспорта нужно задавать каждый раз, даже если вы экспортируете тот же самый FLA-файл в тот же самый формат.

### Чтобы опубликовать ролик:

1. Откройте Flash-файл, который собираетесь опубликовать.
2. Чтобы начать публикацию, выполните одно из следующих действий:

- выберите из меню пункт **File** ⇒ **Publish Settings** (Файл ⇒ Параметры публикации). Открывается диалоговое окно Publish Settings. Можете выполнить те же шаги, что в предыдущем упражнении, чтобы задать новые параметры или принять текущие. Затем нажмите кнопку **Publish** (Опубликовать) (рис. 17.11);
- выберите из меню пункт **File** ⇒ **Publish** (Файл ⇒ Опубликовать) или нажмите **F12**. Открывается диалоговое окно Publishing, в котором имеется индикатор выполнения и кнопка отмены (рис. 17.12).

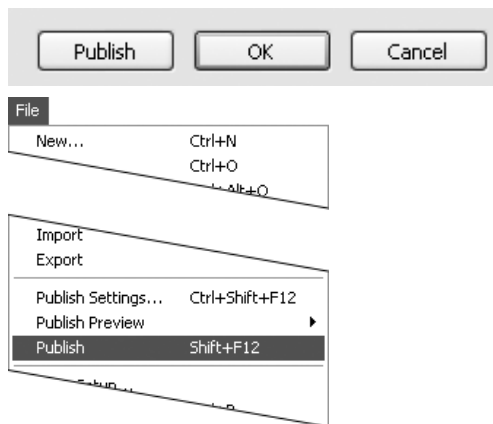
Применяя параметры, хранящиеся в самом документе, Flash создает новые файлы в каждом из заказанных форматов.

C

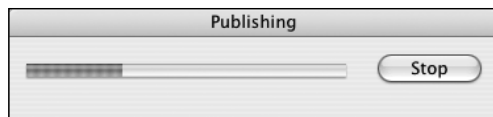
По умолчанию опубликованные файлы хранятся в той же папке, что и исходный файл Flash. Но можно указать и другое место. На вкладке **Formats** в окне параметров публикации щелкните по значку справа от имени файла и задайте путь в диалог Select Publish Destination.

C

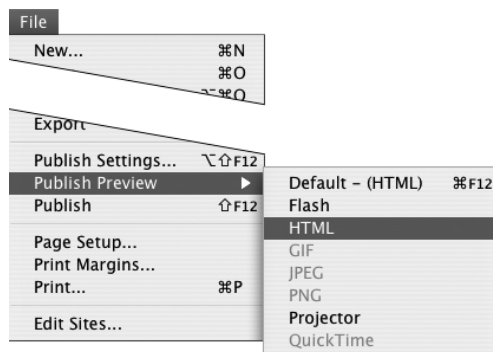
Открыть браузер и посмотреть, как в нем выглядит ролик, можно за одну операцию. Выполните команду **File** ⇒ **Publish Preview** (Файл ⇒ Просмотр публикации) (рис. 17.13). Выберите формат. Flash опубликует файл в этом формате с применением текущих параметров и откроет его в окне браузера.



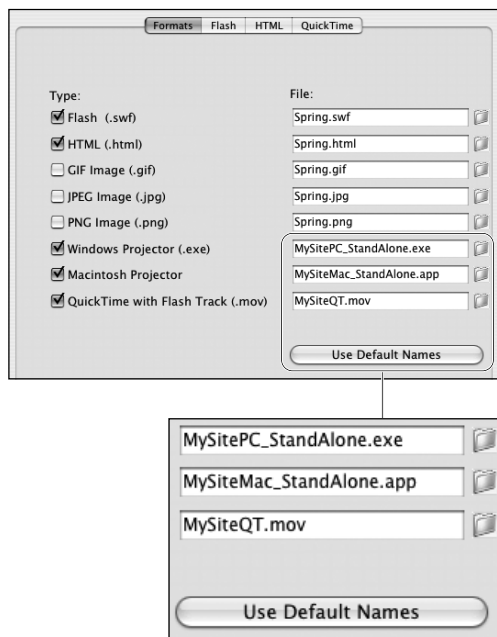
**Рис. 17.11.** Чтобы опубликовать ролик, нажмите кнопку Publish в окне Publish Settings (верхний рисунок) или выполните команду File ⇒ Publish (нижний рисунок).



**Рис. 17.12.** Чтобы отменить публикацию, нажмите кнопку Stop (Mac) или Cancel (Windows). Если раньше вы уже публиковали этот файл, то окно может присутствовать очень недолго, так что для отмены просто не хватит времени.



**Рис. 17.13.** В подменю File ⇒ Publish Preview отображаются все форматы, присутствующие в окне Publish Settings. Flash публикует ролик в выбранном формате и откроет его в браузере.



**Рис. 17.14.** В окне Publish Settings можно задать имена файлов по своему усмотрению, только не забывайте указывать правильное расширение. Чтобы восстановить имя по умолчанию, нажмите кнопку Use Default Names.

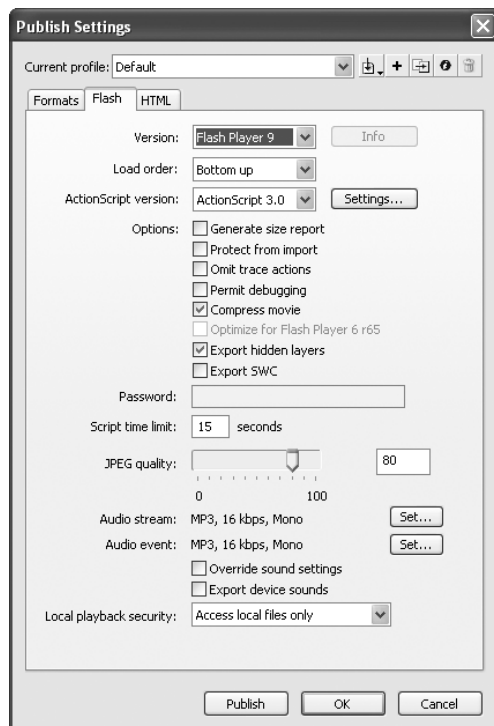
- Для предварительного просмотра ролика в браузере можно пользоваться командой **Publish Preview**. Однако, если вы тестируете SWF-ролик, который хранится на локальном диске, но содержит ссылки на внешние URL, то возможны проблемы с безопасностью. Поэтому на стадии тестирования следует либо изменить настройки безопасности в диалоговом окне Publish Settings (из списка **Local Playback Security** (Безопасность при локальном воспроизведении) выберите значение Access Network Only (Доступ только к сети)), либо назначить тестируемому файлу специальные разрешения (см. врезку «Замечание о настройках безопасности в Flash Player 9» ниже).
- Один из форматов в меню **Publish Preview** принимается по умолчанию. Чтобы опубликовать файл в этом формате, нажмите **#F12** (Mac) либо **F12** (Windows). Если вы часто тестируете не в формате SWF (а, например, в виде анимированного GIF-файла), то задайте параметры так, чтобы публиковать файл только в этом формате. Тогда этот формат будет считаться умалчиваемым, и для тестирования достаточно будет нажать пару клавиш.
- По умолчанию Flash присваивает опубликованным файлам имена путем добавления того или иного расширения к имени исходного файла (.gif — для файлов в формате GIF, .png — для формата PNG и т.д.). Если вам это не подходит, введите другое имя в диалоговом окне Publish Settings. Чтобы восстановить имя по умолчанию, нажмите кнопку **Use Default Names** (Использовать имена по умолчанию) (рис. 17.14).
- Команды **Publish** и **Publish Preview** не дают возможности поименовать опубликованные файлы; берутся те имена, которые заданы в окне Publish Settings. Если вы хотите опубликовать несколько версий ролика с разными параметрами, примите меры к тому, чтобы не затереть предыдущий файл. Переименуйте уже опубликованный файл, переместите его в другую папку или задайте другое имя на вкладке **Formats** в окне Publish Settings.

## Настройки Flash-плеера

Вместе с Flash устанавливается автономный Flash-плеер. Это приложение открывается при двойном щелчке по значку SWF-файла. (Если плеер уже запущен, то для воспроизведения другого SWF-файла пользуйтесь командой меню **File ⇒ Open.**) Чтобы подготовить ролик к воспроизведению в автономном Flash-плеере, выберите в редакторе любую из команд **Publish** или **Export**. Параметры у них практически одинаковы.

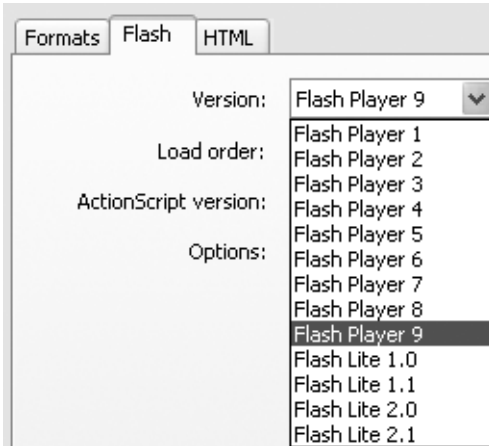
### Чтобы опубликовать SWF-файл для Flash-плеера:

1. Открыв документ, выберите из меню пункт **File ⇒ Publish Settings**.  
В диалоговом окне Publish Settings выберите профиль публикации или оставьте текущие параметры без изменения.
2. Перейдите на вкладку **Formats** (Форматы).
3. Отметьте только флажок **Flash (.swf)**. Если хотите, можете ввести другое имя SWF-файла в поле справа от флажка. Не забудьте включить расширение .swf.
4. Перейдите на вкладку **Flash** (рис. 17.15).
5. Задайте параметры Flash, как описано в последующих упражнениях.
6. Нажмите кнопку **Publish**.



**Рис. 17.15.** На вкладке Flash в окне Publish Settings перечислены параметры публикации ролика в формате SWF.





**Рис. 17.16.** Укажите, для какой версии Flash-плеера публикуется ролик. При выборе старых версий некоторые возможности Flash CS3 становятся недоступны, зато повышается охват аудитории.

### Чтобы выбрать версию Flash-плеера:

- На вкладке Flash в окне Publish Settings выберите версию Flash-плеера из списка **Version** (Версия) (рис. 17.16).

Варианты таковы: версии Flash Player от 1 (раньше назывался FutureSplash Animator) до 9, а также Flash Lite 1.0, 1.1, 2.0 и 2.1 (плееры для мобильных устройств, например телефонов). При публикации для Flash-плеера версии ниже, чем 9, некоторые возможности Flash CS3 теряются.

**С**

Прежде чем приступать к написанию сценариев на языке ActionScript, задайте на вкладке **Flash** самую раннюю версию Flash-плеера, в которой должен работать ваш ролик. В инструментарию на панели действий есть информация о том, какие действия не будут работать в указанной версии плеера.

**С**

При выборе версии Flash Player 6 становится доступным флажок **Optimize for Flash Player 6 r65** (Оптимизировать для Flash Player 6 r65). Отметьте его, если хотите воспользоваться усовершенствованиями, которым была подвергнута версия 6 для повышения производительности (они вошли составной частью в выпуск 65 этой версии и во все последующие).

**С**

При выборе любой из версий Flash Player 6, 7, 8, 9 становится доступным флажок **Compress Movie** (Сжать ролик). Сжатие уменьшает размер файла, содержащего много текста или кода на языке ActionScript.

### Чтобы задать режим рисования кадров ролика:

- На вкладке **Flash** в окне Publish Settings выберите из списка **Load Order** порядок загрузки слоев из ролика (рис. 17.17).

### Чтобы выбрать версию языка ActionScript (только для Flash Player 6–9):

- На вкладке **Flash** в окне Publish Settings выберите из списка **ActionScript version** номер версии языка: 1.0, 2.0 или 3.0 (рис. 17.18).

Этот параметр сообщает Flash, какая версия языка была использована при написании сценариев, чтобы компилятор правильно обрабатывал код. Список доступен только, если выбрана версия Flash-плеера не младше 6.

**С**

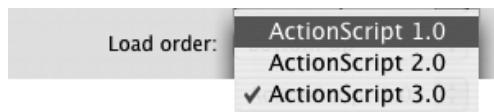
Для работы компонентов Flash язык ActionScript необходим. Очень важно, чтобы версия ActionScript, указанная в окне Publish Settings соответствовала версии компонентов, используемых во FLA-файле. Если документ содержит компоненты для одной версии ActionScript, а ролик опубликован для другой, то компоненты могут не работать или даже не появиться в опубликованном ролике.

**С**

На этапе разработке на панели компонентов представлены лишь те компоненты, которые предназначены для работы с выбранной версией ActionScript. Так, чтобы увидеть на этой панели компоненты для ActionScript 3.0, установите в окне Publish Settings публикацию для этой версии языка.



**Рис. 17.17.** Из списка Load Order выбирается порядок, в котором Flash рисует слои в каждом кадре ролика. Если воспроизведение через Web осуществляется медленным, Flash начинает рисовать отдельные слои по мере загрузки. Режим Top Down (Сверху вниз) означает, что нужно сначала посылать (и, стало быть, отображать) верхний слой. В режиме Bottom Up (Снизу вверх) сначала посылается нижний слой.



**Рис. 17.18.** На этапе публикации SWF-файла языки ActionScript 1.0, 2.0 и 3.0 обрабатываются компилятором по-разному. Не забудьте правильно указать версию, которой вы пользуетесь или собираетесь воспользоваться в сценариях.

Options: ☒ **Generate size report**

Frame #	Frame Bytes
1	4320
2	22
3	9

Spring Report.txt

Frame #	Frame Bytes	Total Bytes	Scene
1	4320	4320	Scene 1 (AS 2.0 Classes Export Frame)
2	22	4342	
3	9	4351	
4	9	4360	
5	9	4369	
6	9	4378	
7	9	4387	
8	9	4396	
9	9	4405	

Frame #	Frame Bytes	Total Bytes	Scene
72	68	15846	
73	69	15855	
74	70	15864	
75	71	15873	
76	72	15882	
77	73	15891	
78	74	15900	
79	75	15909	
80	76	15918	
81	77	15927	
82	78	15936	
83	79	15945	
84	80	15954	
85	81	15963	
86	82	15972	
87	83	15981	
88	84	15990	
89	85	15999	
90	86	16008	
91	87	16017	
92	88	16026	
93	89	16035	
94	90	16044	
95	91	16053	
96	92	16062	
97	93	16071	
98	94	16080	
99	95	16089	
100	96	16098	

**Рис. 17.19.** Отметьте флажок **Generate Size Report** (верхний рисунок), чтобы получить текстовый файл, в котором описано распределение данных по кадрам ролика (нижний рисунок).

**Чтобы вывести количество данных в каждом кадре ролика:**

- На вкладке **Flash** в окне **Publish Settings** отметьте флажок **Generate Size Report** (Сформировать отчет о размере) (рис. 17.19). Flash создает текстовый файл, в котором перечислены все кадры ролика и указано, сколько данных каждый из них содержит. (Текст отчета печатается также на панели вывода.) Этот отчет помогает найти кадры, которые тормозят воспроизведение ролика. Вы можете оптимизировать или удалить часть содержимого из таких кадров.

**С** В отчете о размере присутствует также детальная информация о количестве данных в каждом символе. Неиспользуемые символы в отчете тоже отражены, но их размер нулевой.

**С** В зависимости от размеров панели вывода и особенностей отчета вы можете увидеть только пустое место. В таком случае для ознакомления с отчетом увеличьте окно и прокрутите его вверх.

**Чтобы защитить плоды своих трудов:**

- На вкладке **Flash** в окне **Publish Settings** отметьте флажок **Protect from Import** (Защитить от импорта) (рис. 17.20). В этом случае зрители не смогут преобразовать SWF-файл обратно в документ Flash (FLA-файл).

**С** Защитить ролик от импорта можно и условно. Введите пароль в поле **Password**. Тогда импортировать SWF-файл смогут лишь те, кто знает пароль.

Options: ☐ **Generate size report**  
☒ **Protect from import**  
☐ **Omit trace actions**  
☐ **Permit debugging**  
☒ **Compress movie**  
☐ **Optimize for Flash Player 6 r65**  
☒ **Export hidden layers**  
☐ **Export SWC**

**Рис. 17.20.** Отметьте флажок **Protect from Import**, чтобы воспрепятствовать обратному преобразованию SWF-файла в FLA-файл.

## Чтобы установить параметры трассировки и отладки:

- На вкладке **Flash** в окне Publish Settings отметить следующие флажки:

**Omit Trace Actions** (Пропустить действия трассировки) запрещает действиям трассировки печатать что-либо в окне вывода на этапе отладки (рис. 17.21).

**Permit Debugging** (Разрешить отладку) разрешает удаленную отладку ActionScript-сценариев (рис. 17.22). Этот параметр позволяет вам и другим пользователям, имеющим отладочную версию программы Flash Player, отлаживать SWF-файл во время воспроизведения через Интернет.

**C** Если флажок **Permit Debugging** поднят, то обязательно следует вводить пароль (рис. 17.22). Это не позволит неавторизованным лицам получить доступ к сценарию, но даст возможность удаленно отлаживать его тем, кому это разрешено.

## Чтобы сжать SWF-файл (Flash Player версий 6, 7, 8, 9):

1. На вкладке **Flash** в окне Publish Settings выберите из списка **Version** версию Flash Player 6, 7, 8 или 9.
2. Отметьте флажок **Compress Movie** (Сжать ролик) (рис. 17.23).

Эта возможность имеется только в версиях 6–9 Flash-плеера. Сжатие ролика не оказывает влияния на качество JPEG-изображений и на параметры сжатия аудио, о которых пойдет речь в следующих упражнениях.

### ☒ Omit trace actions

**Рис. 17.21.** Если флажок Omit Trace Actions отмечен, то Flash исключает все действия трассировки из SWF-файла, оставляя в окне вывода только отладочную печать, не связанную с трассировкой. Если в сценариях много трассировочных действий, то в этом режиме размер файла слегка уменьшается.

- ☒ Omit trace actions
- ☒ Permit debugging
- ☒ Compress movie
- ☐ Optimize for Flash Player 6 r65
- ☒ Export hidden layers
- ☐ Export SWC

Password:

**Рис. 17.22.** Если отмечен флажок Permit Debugging, то для защиты ролика, который открыт для удаленной отладки, нужно ввести пароль.

Version:

Load order:

ActionScript version:

Options: ☐ Generate size report

☒ Protect from import

☒ Omit trace actions

☒ Permit debugging

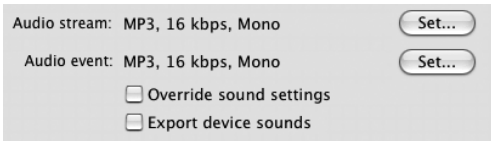
☒ Compress movie

Отметить для сжатия

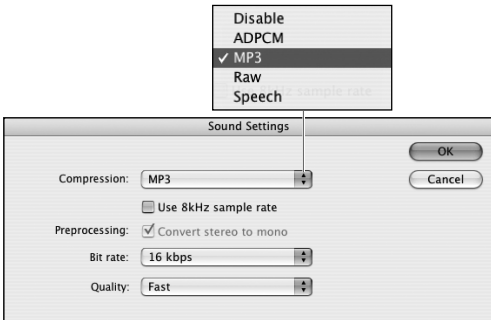
**Рис. 17.23.** В случае публикации для Flash-плеера версий 6–9 можно задать режим сжатия ролика. Это позволяет сократить размер файла, особенно если в нем много текста или сценариев.



**Рис. 17.24.** Чтобы задать степень сжатия всех присутствующих в ролике растровых изображений в формате JPEG, введите значение в поле JPEG Quality или воспользуйтесь ползунком. Величина 0 означает наибольшее сжатие (и самое низкое качество, так как сжатие приводит к потере качества); значение 100 соответствует наименьшему сжатию (и самому высокому качеству).



**Рис. 17.25.** Для потокового и синхронизированного с событиями аудио необходимо задавать частоту дискретизации и степень сжатия по отдельности. Нажмите кнопку Set, чтобы задать параметры для каждой категории звуков.



**Рис. 17.26.** Выберите метод сжатия из списка Compression. Становятся доступны также другие параметры, относящиеся к выбранному методу. Чтобы вообще отключить звук, выберите пункт Disable.

### Как выбирать частоту дискретизации

Частота дискретизации измеряется в килогерцах (Кгц). Запись на музыкальные компакт-диски производится с частотой 44 Кгц. Для мультимедийных CD-ROM стандартной является частота 22 Кгц. Для музыкальных клипов в Flash-роликах, которые воспроизводятся через Web, часто бывает достаточно частоты 11 Кгц. А для коротких звуков, в том числе произносимых слов, можно обойтись даже меньшей частотой дискретизации.

### Чтобы применить сжатие к изображениям в формате JPEG:

- На вкладке **Flash** в окне Publish Settings выполните одно из следующих действий:
  - подвигайте ползунок JPEG Quality (Качество JPEG);
  - введите значение в поле JPEG Quality (рис. 17.24).

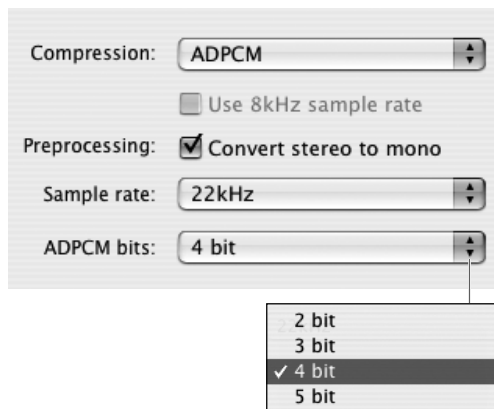
Этот параметр управляет степенью сжатия растровых изображений в формате JPEG, включенных в ролик.

- Flash не применяет сжатие JPEG к изображениям в формате GIF, поскольку этот формат по определению не допускает потери качества.
- Можно задать степень сжатия и для отдельных изображений в библиотеке документа. Выберите нужное изображение на библиотечной панели. Из меню панели выберите пункт **Properties** (Свойства) — откроется диалоговое окно свойств растрового изображения. В списке **Compression** (Сжатие) есть два пункта: Lossless (PNG/GIF) (Без потерь) и Photo (JPEG) (Фото) (последний означает сжатие с потерей качества).

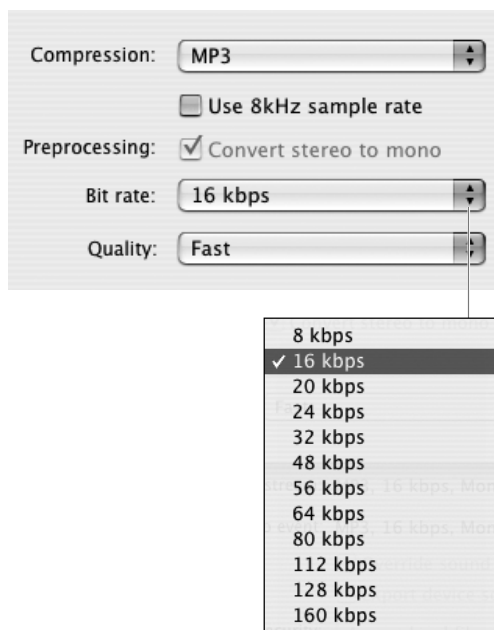
### Чтобы задать степень сжатия и частоту дискретизации для всех аудиороликов:

1. На вкладке **Flash** в окне Publish Settings нажмите кнопку **Set** (Задать), расположенную справа от поля Audio Stream (Поток аудио) или Audio Event (Событие аудио) (рис. 17.25). Открывается диалоговое окно Sound Settings (Параметры звука). Во Flash звуки разбиты на две категории: потоковые и синхронизированные с событиями (подробнее см. главу 15). Задавать сжатие необходимо для каждой категории в отдельности, но сама процедура и параметры ничем не отличаются.
2. Из списка **Compression** (Сжатие) (рис. 17.26) выберите одно из следующих значений:

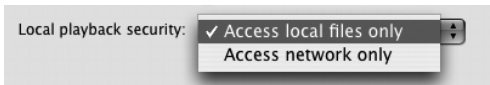
- **Disable** (Отключить) убирает звук из опубликованного файла.
- **ADPCM**. Этот метод сжатия лучше всего подходит для роликов, содержащих короткие звуки, синхронизированные с событиями, например, хлопок в ладоши или щелчок при нажатии кнопки. (Обычно такой параметр задается для звуков из категории Audio Event.) Появляются параметры, относящиеся к методу ADPCM. Из списка **ADPCM Bits** (Разрядность ADPCM) выберите одно из четырех значений, определяющих степень сжатия звука (рис. 17.27). В режиме сжатия ADPCM можно также задавать частоту дискретизации и преобразовать стереофонический звук в монофонический.
- **MP3**. Этот способ больше подходит для роликов, содержащих в основном продолжительную звуковую дорожку. (Обычно такой параметр задается для звуков из категории Audio Stream.) Появляются параметры, относящиеся к методу MP3. Из списка **Bit Rate** (Скорость передачи в битах) выберите один из 12 битрейтов (рис. 17.28). Из списка **Quality** (Качество) выберите значение Fast (Быстрый), если нужно протестировать ролик, или значение Medium (Средний) либо Best (Наилучший) для окончательной публикации, поскольку в этих режимах обеспечивается более высокое качество.
- **Raw** (Без преобразования). В этом режиме вы можете регулировать размер файла путем задания частоты дискретизации и преобразования стереофонического звука в монофонический.
- **Speech** (Речь) позволяет задать степень сжатия для звуков, состоящих из произносимых слов. Из списка выберите частоту дискретизации.



**Рис. 17.27.** Список ADPCM Bits позволяет регулировать степень сжатия звука в ролике. Максимальное сжатие (и самое низкое качество) получается, если выбрать значение 2-Bit. Минимальное сжатие (и самое высокое качество) дает значение 5-Bit.



**Рис. 17.28.** В режиме сжатия MP3 при битрейте менее 20 Кбит/с стереофонический звук преобразуется в монофонический. Если битрейт составляет 20 Кбит/с и выше, то можно публиковать стереозвук или преобразовывать его в моно.



**Рис. 17.29.** Безопасность во Flash CS3 обеспечивается путем ограничения возможности SWF-ролика манипулировать данными, хранящимися в разных системах. Если установлен режим **Access Local Files Only**, то опубликованный SWF-ролик может копировать или записывать данные только в файлы, находящиеся в локальной системе. В режиме **Access Network Only** разрешено копировать и записывать только в файлы, находящиеся в сети, но не в локальной системе.

### Чтобы управлять доступом к локальным и сетевым ресурсам:

- Чтобы определить, из каких файлов SWF-ролик может копировать данные, выберите значение из списка **Local Playback Security** (Безопасность при локальном воспроизведении) (рис. 17.29):

**Access Local Files Only** (Доступ только к локальным файлам). В этом режиме SWF-ролик может обращаться к файлам, находящимся в той же системе, что и он сам, но не к файлам из Интернета. В частности, запрещено загружать данные из XML-файлов, находящихся в Интернете, или отправлять в Интернет данные формы.

**Access Network Only** (Доступ только к сети). В этом случае SWF-ролик может обращаться только к файлам, находящимся в Интернете, но не на локальном компьютере.

### Звуки, записанные цифровым способом

Цифровая запись соотносится со звуком точно так же, как кинофильм с движением. В обоих случаях фиксируются мгновенные снимки непрерывного потока в дискретные моменты времени. Впоследствии, воспроизводя эти снимки в том же порядке, мы воссоздаем исходный поток. В кинофильме снимок — это один кадр, а в цифровой записи — фрагмент звука.

Процесс записи можно представлять себе как фиксацию формы сигнала путем наложения на него сетки и копирования волновых форм в точках пересечения с сеткой. Линии вдоль горизонтальной оси определяют *частоту дискретизации* — как часто звук захватывается. Линии вдоль вертикальной оси определяют *битрейт* — насколько подробно фиксируется информация об амплитуде сигнала. Чем выше частота и битрейт (чем мельче ячейки сетки), тем реалистичнее звучание при воспроизведении. К сожалению, повышение реалистичности означает и рост размера файла.

Относящиеся к звуку параметры в окне **Publish Settings** позволяют гибко создавать версии ролика с разными частотами дискретизации и битрейтами, не изменяя исходные звуки. Для версии, поставляемой на CD-ROM, можно позволить себе больший размер файла и соответственно более высокое качество, чем при распространении через Web. Но в любом случае для достижения оптимального баланса между качеством и размером файла прослушивайте, что получилось.

### Замечание о настройках безопасности в Flash Player 9

Flash CS3 и Flash Player версий 8 и 9 позволяют создавать и воспроизводить ролики, которые загружают файлы на сервер или с сервера во время выполнения. Это наделяет Flash куда более широкими возможностями, чем раньше, но потенциально они могут нанести ущерб. В программу Flash Player 8 и 9 встроены средства обеспечения безопасности, гарантирующие, что SWF-файл не сможет совершать злонамеренные действия на компьютерах конечных пользователей. По умолчанию система безопасности настроена так, SWF-ролик, исполняемый локально (на одном компьютере) не может получить доступ к файлам, размещенным где-либо в Интернете (и наоборот). Чтобы предоставить такую возможность, вы должны дать Flash-плееру специальные разрешения.

Когда в окне Publish Settings вы указываете, что ролик публикуется для Flash Player версии 8 или 9, вступает в силу механизм обеспечения безопасности. Если перед выкладыванием ролика на сервер вы публикуете его локально и пытаетесь запустить в браузере, то результатом щелчка по кнопке или ссылке, который ведет к установлению соединения с Интернетом, может оказаться предупреждение о том, что Flash Player прервал потенциально небезопасную операцию.

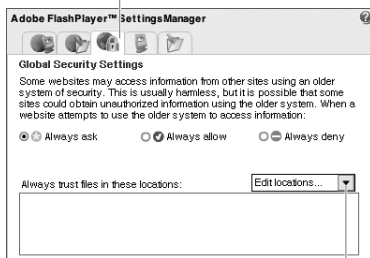
Если тестируемому SWF-ролику не нужно обращаться одновременно к локальной системе и к Интернету, то решить проблему можно путем изменения настроек в окне Publish Settings. На вкладке **Flash** выберите из списка **Local Playback Security** (Безопасность при локальном воспроизведении) значение **Access Network Only** (Доступ только к сети). (Отметим, что при прогоне ролика в одном из тестовых режимов предупреждений от системы безопасности не будет. В тестовом режиме Flash автоматически доверяет всем локальным файлам, к которым производится доступ.)

Если после изменения настройки безопасности вы все равно получаете предупреждение, придется дать SWF-файлу специальные разрешения на доступ к Интернету. В диалоговом окне Adobe Flash Player Security (рис. 17.30) нажмите кнопку **Settings** (Настройки). Flash откроет в окне браузера страницу документации с сайта компании Adobe ([http://www.macromedia.com/support/documentation/en/flashplayer/help/settings\\_manager04a.html](http://www.macromedia.com/support/documentation/en/flashplayer/help/settings_manager04a.html)). С этой страницы есть ссылка на менеджер настроек Adobe Flash Player (рис. 17.31).



**Рис. 17.30.** Flash Player 9 пресечет попытку локально исполняемого SWF-ролика обратиться за файлами в Интернет, если только вы не зададите правильные настройки безопасности. Вы увидите предупреждение о том, что текущие настройки запрещают выход в сеть.

### Вкладка Global Security Settings



Нажать для добавления доверенных файлов или папок

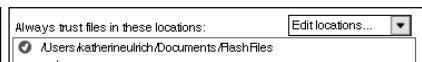
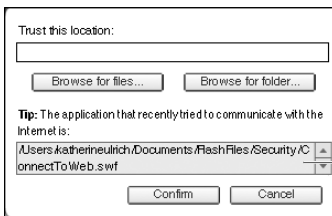
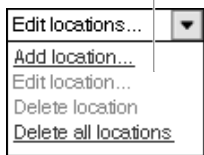
**Рис. 17.31.** На вкладке Global Security Settings в менеджере настроек Adobe Flash Player можно создать список доверенных файлов или папок для Flash Player 9. Любые SWF-файлы, попавшие в этот список, могут обращаться к файлам в Интернете.



Для создания списка доверенных файлов или папок перейдите на вкладку Global Security Settings (Глобальные настройки безопасности) в менеджере. Flash-плеер позволяет любому SWF-файлу, указанному в этом списке, обращаться в Интернет, даже если он выполняется локально. Для добавления файлов в список доверия или удаления из него воспользуйтесь выпадающим меню **Edit Locations** (Редактировать местоположение) (рис. 17.32).

Если вы часто тестируете, то имеет смысл создать специальную папку Trusted, в которой будут храниться только ваши подлежащие тестированию SWF-файлы. Не позволяйте Flash-плееру доверять папкам в вашей системе, в которых могут оказаться SWF-файлы из других источников, поскольку они теоретически могут сделать что-нибудь опасное, например, скопировать информацию, хранящуюся на вашем компьютере, и отправить ее неизвестно куда.

Нажать для добавления доверенных файлов



Список подтвержденных доверенных файлов

Скопировать содержимое поля Tip и вставить его в поле Trust

**Рис. 17.32.** Выпадающее меню Edit Locations позволяет управлять списком доверенных файлов (левый рисунок). Щелкнув по ссылке Add Location, вы откроете окно (средний рисунок), в котором можно добавить новый файл или папку в список доверия. Если вы открывали менеджер глобальных настроек безопасности нажатием кнопки Settings в окне предупреждения (рис. 17.30), то в поле Tip отображается путь к файлу, с которым вы работали. Скопируйте его в поле Trusted (или с помощью любой из кнопок Browse найдите файл или папку, которую собираетесь добавить) и нажмите кнопку Confirm, чтобы подтвердить добавление доверенного файла.

### Еще о работе со звуком

Flash автоматически применяет параметры сжатия звука, заданные в окне Publish Settings, ко всем звуковым файлам, для которых не заданы другие параметры. Если избирательно назначить некоторым звукам более высокое качество, то можно и сохранить приемлемый размер файла, и добиться приличного звучания там, где это необходимо.

Чтобы индивидуально задать параметры сжатия, выберите звуковой файл на библиотечной панели и нажмите кнопку с изображением буквы *i* в нижней части панели либо выберите пункт **Properties** из меню панели. Открывается окно Sound Properties (Свойства звука). Список **Compression** (Сжатие) содержит те же параметры, что и на вкладке **Flash** в окне Publish Settings.

Если вы зададите для отдельных звуковых файлов специальные настройки экспорта звука, то Flash применит их на этапе публикации к этим и только этим звукам. Ко всем остальным звукам в ролике применяются настройки, заданные в окне Publish Settings.

Можно сделать так, что Flash будет игнорировать специальные настройки и публиковать все звуки, применяя параметры, заданные в окне Publish Settings. Для этого отметьте на вкладке **Flash** в этом окне флажок **Override Sound Settings** (Переопределить параметры звука). Эта возможность может оказаться полезной, например, когда вы хотите получить низкокачественную версию ролика, подготовленного для записи на CD-ROM.

## Публикация HTML-документа для просмотра файлов в Flash-плеере

*HTML-документ* — это набор команд браузеру о том, как отображать Web-страницу. Функция Publish создает HTML-документ, сообщающий браузеру, как показывать опубликованные файлы (в формате SWF, GIF, JPEG, PNG или QuickTime с Flash-дорожками — в соответствии с набором форматов, указанных в окне Publish Settings). Кроме того, при публикации создается JavaScript-файл, который, работая совместно с HTML-документом, делает Flash-ролик доступным конечным пользователям (см. врезку «Об активном содержимом»).

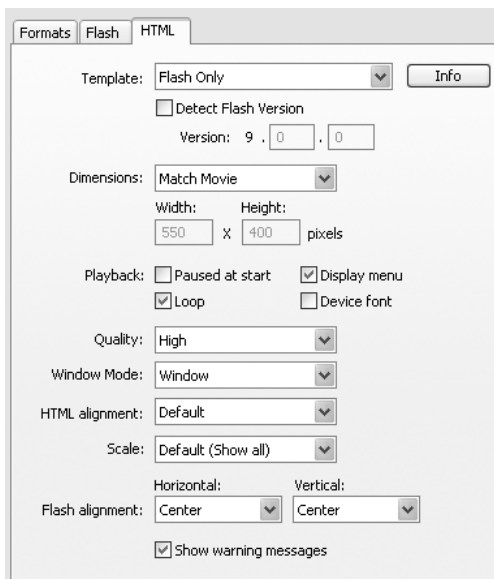
Требуемый HTML-документ создается путем подстановки имен в шаблон. В состав дистрибутива Flash входит 11 шаблонов, но вы можете создавать и собственные.

### Об активном содержимом

В результате судебной тяжбы по поводу патента компании EOLAS в 2006 году корпорация Microsoft начала изменять некоторые аспекты работы браузера Internet Explorer. Изменения касаются *активного содержимого*, внедренного в HTML-документы с помощью стандартных тегов <object> и <embed> (Flash — это один из видов активного содержимого). Теперь требуется, чтобы посетитель сайта самостоятельно активировал такое содержимое, щелкнув по нему. Рассмотрим какой-нибудь сайт с Flash-анимацией. Если внедрить SWF-файл обычным способом, то пользователь, впервые зашедший на этот сайт с помощью Internet Explorer, увидит статическое изображение. При наведении на него курсора появится подсказка, сообщающая, что для активации содержимого по нему нужно щелкнуть. Чтобы обойти это препятствие на пути к интерактивной работе, команда Publish в Flash CS3 создает два файла, необходимых для отображения содержимого в браузере: HTML и JavaScript (последний называется AC\_RunActiveContent.js). Код в этом сценарии динамически формирует HTML-разметку (по существу, выводит теги <object> и <embed>); при таком подходе SWF-ролик активируется немедленно. Конечные пользователи видят и могут взаимодействовать с Flash-роликом сразу же, как он появляется в окне браузера.

Для тех пользователей, которые отключили JavaScript, в разметке есть также стандартные теги <object> и <embed>, заключенные внутри пары тегов <noscript>. Разметка между тегами <noscript> интерпретируется браузером только в случае, когда JavaScript недоступен.

Комбинация JavaScript и HTML работает во всех браузерах, которые поддерживают Flash Player. Но при использовании этого метода необходимо, чтобы SWF-файл, HTML-документ и сценарий AC\_RunActiveContent.js находились в одной папке на сервере, где размещен ваш сайт.



**Рис. 17.33.** На вкладке HTML в окне Publish Settings представлены параметры отображения Flash-ролика в браузере.

### Чтобы опубликовать HTML-документ для отображения Flash-ролика:

1. Открыв документ Flash, который вы собираетесь опубликовать, выберите из меню пункт **File** ⇒ **Publish Settings**. В окне Publish Settings выберите профиль публикации или оставьте текущие параметры без изменения.
2. Нажмите кнопку **Formats** (Mac) или перейдите на вкладку **Formats** (Windows).
3. Отметьте флажок **HTML (.html)**. При этом автоматически отмечается также флажок **Flash (.swf)**.
4. Нажмите кнопку **HTML** (Mac) или перейдите на вкладку **HTML** (Windows) (рис. 17.33).

На вкладке представлены параметры отображения Flash-ролика в браузере. Выбранные значения при публикации подставляются в нужные места шаблона.

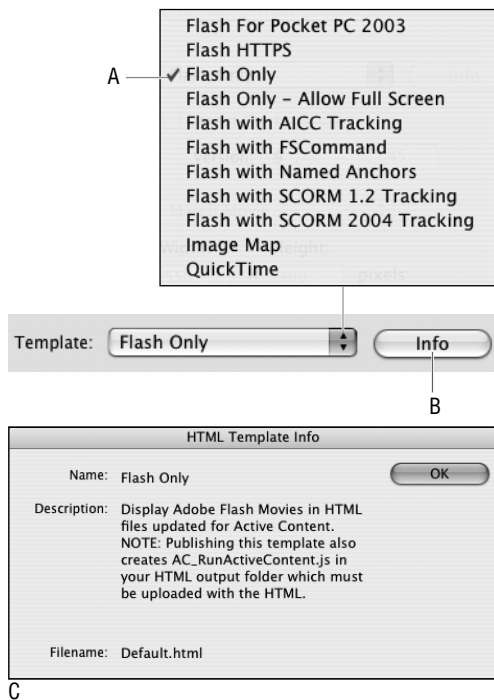
### Чтобы создать HTML-документ только для показа Flash:

- Из списка **Template** (Шаблон) выберите пункт **Flash Only** (рис. 17.34).

Это самый простой шаблон. В нем используется комбинация HTML и JavaScript для показа Flash-ролика зрителям, имеющим именно ту версию плеера, которая была выбрана на вкладке **Flash** в окне **Publish Settings** (см. врезку «Об активном содержимом» выше). Другие зрители ничего не увидят. (В остальных шаблонах есть возможность представить ролик по-другому; например, по шаблону **Image Map** создается разметка, в которой пользователям, не имеющим нужной версии плеера демонстрируются альтернативные изображения или файлы.)

**С**

Если вы не помните, для чего предназначен конкретный HTML-шаблон, выберите его из списка **Template** и нажмите кнопку **Info** (Информация) справа от списка. Flash выведет краткое описание вместе с инструкциями по выбору альтернативных форматов, если это имеет смысл.



**Рис. 17.34.** Выберите шаблон **Flash Only** (A), если хотите создать разметку для показа только Flash-ролика, отказываясь от каких-либо альтернативных изображений. Нажмите кнопку **Info** (B), чтобы ознакомиться с описанием шаблона (C).

### Об HTML-шаблонах

Традиционно для отображения SWF-файла применяется тег `<object>` в случае браузера Internet Explorer (Windows) и `<embed>` — для всех остальных браузеров. (Помимо этого, Flash может включать тег `<img>` для отображения файла в другом формате, например, JPEG или анимированный GIF. Если вы создали именованные анкеры во Flash-ролике и выбрали шаблон Flash with Named Anchors, то будут также созданы теги `<a>` для навигации внутри браузера.)

Команда Publish формирует теги `<object>` и `<embed>`, работая рука об руку с HTML-шаблонами, которые представляют собой просто пустографки, и JavaScript-сценарием. К числу предоставляемых параметров относятся ширина и высота окна ролика, качество изображения (степень сглаживания) и расположение ролика относительно окна браузера. У любого параметра на вкладке HTML в окне Publish Settings есть эквивалентная переменная шаблона. *Переменной шаблона* называется слово, начинающееся со знака доллара \$. Выбранное значение параметра подставляется вместо соответствующей переменной в шаблон. Например, если задана ширина ролика 500 пикселей, то оно будет подставлено вместо переменной `$W`, чтобы в HTML-разметке было описано окно именно такой ширины. Если быть точным, то переменные шаблона — это параметры функции в JavaScript-сценарии. На этапе публикации ролика Flash создает HTML-код, который вызывает эту функцию, чтобы динамически сформировать теги `<object>` и `<embed>` (см. врезку «Об активном содержимом» выше).

В HTML-шаблонах есть код не только для отображения Flash-ролика, но и для показа его версий в форматах JPEG, GIF или PNG на случай, если у зрителя нет подходящей версии Flash-плеера.

В ходе публикации Flash сохраняет копию HTML-шаблона, присваивая ей имя вашего ролика и расширение самого шаблона. (Например, файлы шаблонов, поставляемые вместе с Flash, имеют расширение `.html`.) Файл шаблона можно отредактировать, как любой текстовый файл, изменив HTML-разметку.

Можно расширить возможности команды Publish, написав собственные HTML-шаблоны. Если вы хотите, чтобы шаблон попал в список на вкладке **HTML**, файл должен включать заголовки (используйте переменную `$T`). Шаблоны должны быть помещены в папку HTML, которая находится в папке Configuration. Дополнительную информацию о местонахождении этой папки см. на врезке «Папка Configuration» в главе 1.

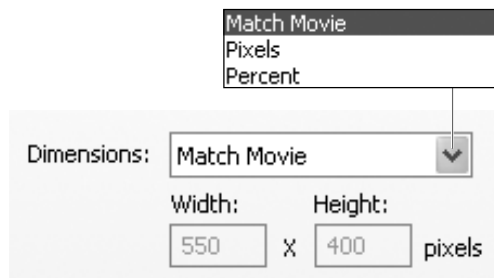
### Чтобы задать размеры окна отображения ролика:

- Для задания ширины и высоты прямоугольника, в котором ваш ролик будет отображаться в браузере, выберите одно из следующих значений в списке **Dimensions** (Размер) на вкладке **HTML** в окне Publish Settings (рис. 17.35):

**Match Movie** (По ролику). Используются размеры самого ролика (заданные в окне свойств документа).

**Pixels** (Пиксели). Размеры в пикселях вводятся в поля Width (Ширина) и Height (Высота).

**Percent** (Проценты). Размеры задаются в процентах от размеров окна браузера. Введите в поля Width и Height значения от 1 до 100.



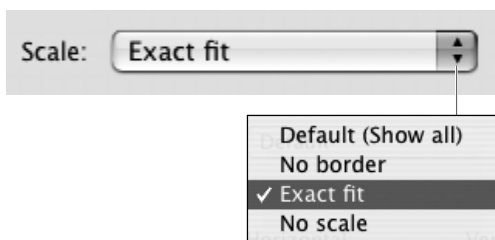
**Рис. 17.35.** Выберите способ задания размера окна отображения ролика (области, в которой браузер демонстрирует Flash-ролик).

### Размещение ролика в окне браузера

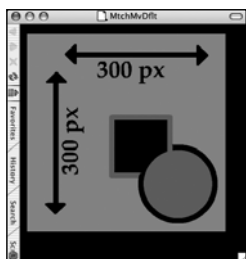
При публикации HTML-документа для демонстрации ролика на Web-странице применяется следующая терминология:

- *окно браузера* содержит всю Web-страницу;
- *окно отображения ролика* (movie-display window) — это область, в которой Flash-плеер показывает ролик. Это окно создается JavaScript-сценарием, который динамически выводит HTML-теги `<object>`, `<embed>` и `<img>`;
- внутри окна отображения ролика находится собственно *окно ролика* (movie window), которое соответствует арене Flash-документа.

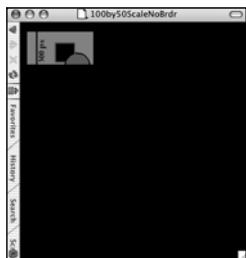
У всех трех окон свои размеры, и вы должны сообщить Flash, куда помещать окна и что делать, если, например, пропорции отличаются или пользователь изменяет размеры окна браузера. Чтобы проинструктировать браузер обо всех деталях, следует задать параметры на вкладке HTML в окне Publish Settings. Если ширина или высота окна отображения ролика отличаются от ширины/высоты исходного Flash-документа, то нужно указать Flash, как масштабировать ролик по размерам окна.



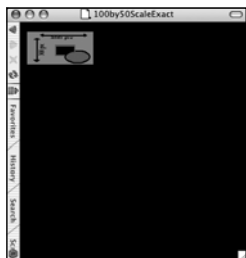
**Рис. 17.36.** Способ масштабирования определяет, как Flash должен разместить ролик внутри заданного окна отображения. Задавать масштабирование нужно только в том случае, когда размер окна отображения отличается от размера самого ролика, — например, задан в процентах от размера окна браузера.



Размер: По ролику  
Масштабирование: Без масштабирования



Размер: 100x50 пикселей  
Масштабирование: Без рамки



Размер: 100x50 пикселей  
Масштабирование: Точное соответствие

**Рис. 17.37.** Этот ролик размером 300x300 пикселей выглядит совершенно по-разному при разных сочетаниях размера и способа масштабирования. На рисунках показано, как были заданы размеры окна отображения и способ масштабирования. (Цвет фона браузера сделан черным, чтобы окно отображения было отчетливо видно.)

### Чтобы ролик масштабировался по размеру окна отображения:

- Из списка **Scale** (Масштабировать) (рис. 17.36) выберите одно из следующих значений:

**Default (Show All)** (По умолчанию (показать все)). Сохраняет исходные пропорции (отношение ширины к высоте) и масштабирует ролик так, чтобы он целиком уместился в заданном прямоугольнике (рис. 17.37). (Имейте в виду, что по краям масштабированного ролика может при этом появиться пустое место.)

**No Border** (Без рамки). Сохраняет исходные пропорции и масштабирует ролик так, чтобы он целиком заполнял заданный прямоугольник (часть ролика может оказаться за пределами прямоугольника и будет обрезана).

**Exact Fit** (Точное соответствие). Изменяет ширину и высоту ролика в соответствии с заданными параметрами, даже если это потребует искажения пропорций.

**No Scale** (Без масштабирования). Сохраняет истинные размеры ролика. При изменении размера окна браузера изображение может быть обрезано.

**С**

Если указать, что размер окна отображения ролика равен 100% от ширины и высоты окна браузера, то в некоторых браузерах вне зависимости от того, насколько большим будет сделано окно браузера, всегда появляются полосы прокрутки. Если сделать ширину и высоту равными 95% (или меньше) от размеров окна браузера, то зритель гарантированно сможет увеличить окно браузера так, чтобы избавиться от полос прокрутки.

### Чтобы задать положение окна ролика внутри окна отображения:

- Чтобы описать положение окна ролика относительно окна отображения ролика выполните любое из следующих действий на вкладке **HTML** в разделе **Flash Alignment** (Выравнивание Flash):

- из списка **Horizontal** (По горизонтали) выберите значение **Left** (Влево), **Center** (По центру) или **Right** (Вправо);
- из списка **Vertical** (По вертикали) выберите значение **Top** (Верхний край), **Center** (По центру) или **Bottom** (Нижний край).

Flash позиционирует ролик внутри окна отображения (рис. 17.38).

### Чтобы задать параметры воспроизведения:

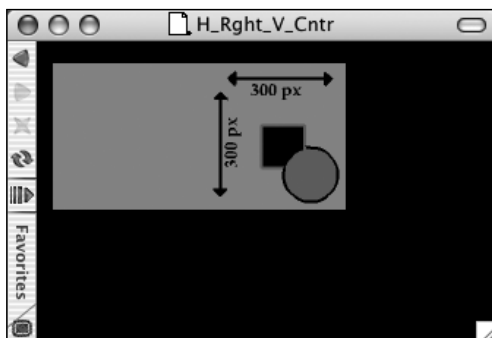
- В разделе **Playback** (Воспроизведение) на вкладке **HTML** (рис. 17.39) отметьте следующие флажки в любом сочетании:

**Paused at Start** (Приостановить после запуска). Если этот флажок отмечен, то пользователь должен будет запустить ролик вручную (нажав кнопку или выбрав команду **Play** из контекстного меню);

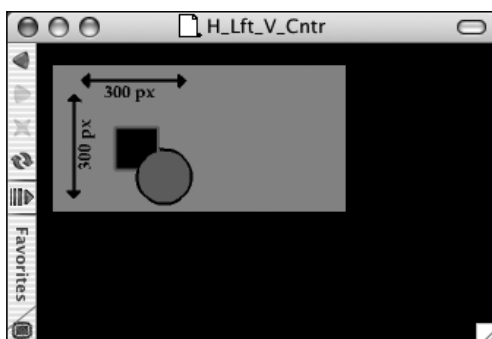
**Loop** (Цикл). По завершении ролик начинается сначала.

**Display Menu** (Отображать меню). Создается контекстное меню, содержащее команды и режимы воспроизведения.

**Device Font** (Шрифт устройства). Ускоряет воспроизведение в системе Windows. В этом случае операционной системе разрешено подставлять системные шрифты взамен отсутствующих на компьютере пользователя. Подстановка возможна только в статических текстовых блоках, для которых на этапе разработки было разрешено использовать шрифты устройства.

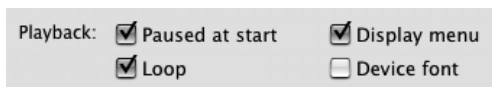


По горизонтали: Вправо; По вертикали: По центру



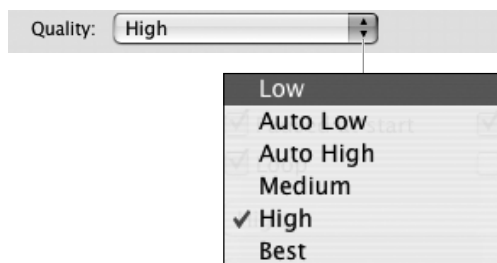
По горизонтали: Влево; По вертикали: По центру

**Рис. 17.38.** Списки **Horizontal** и **Vertical** в разделе **Flash Alignment** позволяют позиционировать ролик в окне отображения, если размеры этого окна отличаются от размеров самого ролика. Сравните результаты применения двух разных настроек для ролика размером 300х300 пикселей, отображаемого в окне 200х100 пикселей (в списке **Scale** выбрано значение **Default**). Светло-серый прямоугольник — это окно отображения, которое автоматически закрашивается тем же цветом, что и фон ролика.



**Рис. 17.39.** На вкладке **HTML** имеются флажки для управления воспроизведением. Можно указать, должно ли воспроизведение начинаться в окне браузера сразу или нужно дождаться действия пользователя; будет ли ролик прокручиваться циклически или только один раз; следует ли дать пользователям возможность управлять роликом с помощью контекстного меню. На платформе Windows можно повысить скорость воспроизведения за счет применения шрифтов устройства.





**Рис. 17.40.** Параметр Quality регулирует баланс между качеством изображения и скоростью воспроизведения ролика.

### Чтобы задать степень антиальясинга и сглаживания:

- Из списка **Quality** (Качество) на вкладке **HTML** (рис.17.40) выберите одно из следующих значений:

**Low** (Низкое). Антиальясинг отключается.

**Auto Low** (Низкое, авто). Воспроизведение начинается с отключенным антиальясингом, но если Flash обнаруживает, что компьютер и соединение пользователя способны поддерживать антиальясинг без уменьшения частоты кадров, то он включается.

**Auto High** (Высокое, авто). В начале воспроизведения антиальясинг включен, но выключается, если частота кадров падает ниже заданной.

**Medium** (Среднее). Flash отказывается от сглаживания растровых изображений, но антиальясинг в ряде случаев применяет.

**High** (Высокое). Антиальясинг применяется ко всему, но растровые изображения сглаживаются только в отсутствие анимации.

**Best** (Наилучшее). Антиальясинг и сглаживание применяются ко всем растровым изображениям.

### Печать из Flash-плеера

По умолчанию команда **Print** (Печать) в контекстном меню Flash-плеера распечатывает все кадры ролика. Можно разрешить печать только некоторых опорных кадров, пометив их в исходном документе. Чтобы сделать опорный кадр печатаемым, выберите его и на вкладке **Properties** в инспекторе свойств введите #p (буква *p* должна быть маленькой) в поле **Frame Label** (Метка кадра). Чтобы при воспроизведении ролика в Flash-плеере было доступно контекстное меню, на вкладке **HTML** в окне **Publish Settings** необходимо отметить флажок **Display Menu** (см. раздел «Чтобы создать HTML-документ только для показа Flash» выше). Чтобы отключить печать из Flash-плеера, выберите любой опорный кадр в документе и введите в поле **Frame Label** на вкладке свойств кадра строку !#p. В этом случае при воспроизведении опубликованного кадра пункт **Print** в контекстном меню будет недоступен.

**Чтобы задать прозрачность:**

- Из списка **Window Mode** (Оконный режим) (рис. 17.41) выберите одно из следующих значений:

**Window** (Окно). Ролик воспроизводится в отдельном окне отображения внутри Web-страницы.

**Opaque Windowless** (Непрозрачный безоконный). Ролик воспроизводится прямо в окне браузера.

**Transparent Windowless** (Прозрачный безоконный). Ролик воспроизводится прямо в окне браузера.



**Рис. 17.41.** Для зрителей, имеющих подходящий браузер, вы можете создать эффект прозрачности, в которой элементы объемлющей Web-страницы видны сквозь прозрачные области ролика. Для этого на вкладке **HTML** в окне **Publish Settings** выберите из списка **Window Mode** режим **Transparent Windowless**.

### Еще об оконных режимах

В режиме **Window** окно отображения ролика блокирует элементы Web-страницы, оказавшиеся в слоях под роликом, и зачастую блокирует или мешает работе также с элементами, расположенными над роликом. В этом режиме Flash получает полный контроль над областью экрана, в которой находится окно отображения, поэтому производительность оптимальна. В режиме **Opaque Windowless** элементы Web-страницы, расположенные выше ролика, видны, а те, что расположены под ним, — не видны. В режиме **Transparent Windowless** элементы страницы, просвечивают сквозь те области ролика, в которых нет объектов; в этом случае воспроизведение анимации может замедляться. Кроме того, клавиатурные команды по-разному обрабатываются в режимах **Opaque Windowless** и **Transparent Windowless**, а в некоторых браузерах могут наблюдаться проблемы с позиционированием активных областей кнопок. Поэтому, решив выбрать любой из этих режимов, уделите повышенное внимание тестированию.

Не все браузеры (и не все версии Flash Player) могут создавать прозрачный фон для ролика. Поддержка режима **Transparent Windowless** появилась, начиная с версии 6.0.65.0 для Windows и 6.0.67.0 для Mac. На платформе Mac OS X (в режиме Mac Classic браузеры вообще не поддерживают режим **Transparent Windowless**) поддержка реализована в версиях Internet Explorer (IE) 5.1 и 5.2, а в Windows — Internet Explorer 5.0, 5.5 и 6.0. Среди других браузеров, поддерживающих этот режим на обеих платформах, упомянем Netscape 7.0 и старше, Opera 6 и старше, Mozilla 1.0 и старше и AOL/CompuServe.

Заголовок шаблона, отображаемый в списке

**\$TTFlash Only**

```

$TTFlash Only
$D{DefaultDetection.html}
$DS
Display Adobe Flash Movies in HTML files updated for Active Content. NOTE:
Publishing this template also creates AC_RunActiveContent.js in your HTML output
folder which must be uploaded with the HTML.
$DF
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
$CS
<title>$TI</title>
<script language="javascript">AC_FL_RunContent = 0;</script>
<script src="AC_RunActiveContent.js" language="javascript"></script>
</head>
<body bgcolor="$BG">
    
```

**<body bgcolor="\$BG">**

Переменная, определяющая цвет фона

Новый заголовок шаблона

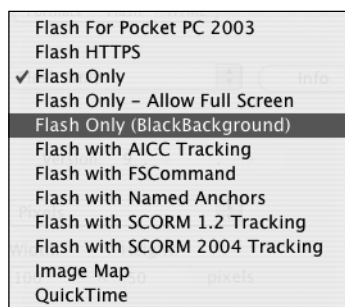
**\$TTFlash Only (BlackBackground)**

```

$TTFlash Only (BlackBackground)
$D{DefaultDetection.html}
$DS
Display Adobe Flash Movies in HTML files updated for Active Content. NOTE:
Publishing this template also creates AC_RunActiveContent.js in your HTML output
folder which must be uploaded with the HTML.
$DF
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
$CS
<title>$TI</title>
<script language="javascript">AC_FL_RunContent = 0;</script>
<script src="AC_RunActiveContent.js" language="javascript"></script>
</head>
<body bgcolor="000000">
    
```

**<body bgcolor="000000">**

Новое значение переменной



**Рис. 17.42.** В шаблоне, принимаемом по умолчанию, цвет фона Web-страницы устанавливается таким же, как в ролике. Копию этого шаблона можно модифицировать (верхний рисунок). Измените теги Title и задайте конкретный цвет (средний рисунок). Новый заголовок появляется в списке шаблонов на вкладке HTML в окне Publish Settings (нижний рисунок).

**Чтобы видеть предупреждения об отсутствующем альтернативном контенте:**

- Отметьте флажок **Show Warning Messages** (Показывать предупреждения) в нижней части вкладки **HTML**, если хотите, чтобы Flash уведомлял о том, что в выбранном шаблоне есть теги для отображения альтернативного контента (например, GIF-файла), но вы не включили генерацию соответствующего файла на вкладке **Formats**.

**C**

В шаблоне, принимаемом по умолчанию, цвет фона Web-страницы автоматически устанавливается таким же, как в ролике. Если вы хотите, чтобы цвет был другим, создайте модифицированный шаблон (рис. 17.42). Откройте шаблон по умолчанию и сохраните его копию под другим именем. В первой строке замените строку **\$TTFlash Only** примерно такой: **\$TTFlash Only (BlackBackground)**, чтобы Flash опознал новый шаблон и включил его в список шаблонов. В теге **<BODY bgcolor="\$BG">** замените **\$BG** шестнадцатеричным кодом другого цвета (например, код черного цвета равен **000000**). Не забудьте поместить новый шаблон в подпапку HTML папки Configuration (подробнее о том, где находится эта папка, см. врезку «Папка Configuration» в главе 1). Новый шаблон появится в списке только после перезагрузки Flash.

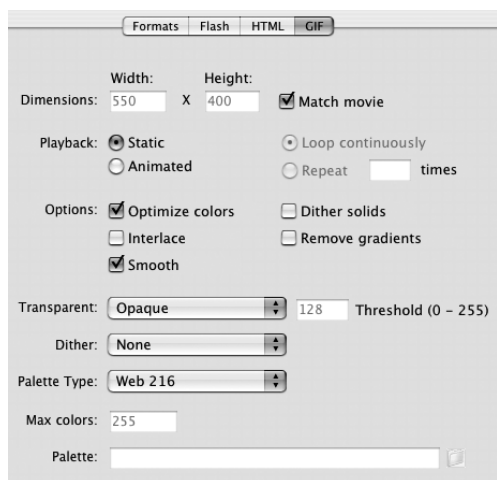
## Использование альтернативных графических форматов

Хотя у большинства зрителей имеется подключаемый модуль Flash Player, необходимый для просмотра Flash-роликов, некоторые могли его и не установить. Для таких пользователей можно показать сайт хотя бы в урезанном виде, предложив браузеру загрузить альтернативные изображения. Например, если вы использовали Flash для создания простого баннера, то можно заодно создать и анимированный GIF, который будет демонстрироваться в случае отсутствия Flash-плеера. Flash позволяет опубликовать альтернативные изображения в форматах GIF, JPEG, PNG и QuickTime. Разумеется, чтобы сообщить браузеру о наличии таких файлов, необходима соответствующая HTML-разметка. В подготовке нужной разметки могут помочь HTML-шаблоны Image Map и QuickTime, а также функция определения установленной версии Flash-плеера (см. раздел «Определение версии» ниже).

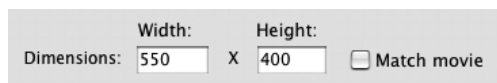
Хотя для каждого формата необходимы специфические параметры публикации, общие принципы опубликования всех четырех альтернатив одинаковы. В этом упражнении мы рассмотрим задание параметров для GIF-файлов.

### О шаблоне Image Map

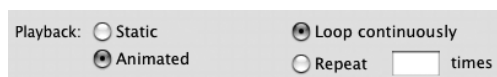
Если Flash-файл содержит символы-кнопки, с которыми ассоциированы ссылки на URL, то можно преобразовать содержимое в *карту изображения* — файл в формате GIF (или PNG либо JPEG), в который встроены активные области-ссылки, ведущие на другие сайты. Чтобы Flash создал карту изображения, в которой активные области размещены там же, где символы-кнопки в исходном ролике, отметьте на вкладке **Formats** в окне Publish Settings флажки HTML и GIF (или PNG либо JPEG). В списке **Template** на вкладке **HTML** выберите шаблон Image Map. Нажмите кнопку **Publish**. Хотя одновременно с HTML-документом Flash всегда создает и SWF-файл, при выборе шаблона Image Map в генерируемый HTML-файл включается разметка, требующая от браузера загрузить GIF-файл (PNG, JPEG) вместо SWF. В карте отображения, показанной браузером, пользователь увидит графическое содержимое, в котором кнопки будут активными ссылками. (Отметим, что по шаблону Image Map активные области создаются только для символов-кнопок, но не для компонентов-кнопок и не для символов-клипов, работающих как кнопки. При этом в сценарии на языке ActionScript 2.0, ассоциированном с символом-кнопкой, может использоваться действие `getURL`. Flash не умеет создавать карты изображения из символов-кнопок, с которыми ассоциирован сценарий на языке ActionScript 3.0.)



**Рис. 17.43.** Нажмите кнопку GIF (Mac) или перейдите на вкладку GIF (Windows), чтобы задать параметры создания статического или анимированного GIF-файла, в котором будет храниться альтернативное изображение.



**Рис. 17.44.** Чтобы ввести размеры публикуемого GIF-изображения, сбросьте флажок Match Movie. Чтобы размеры GIF-изображения были такими же, как у исходного ролика, отметьте флажок Match Movie.



**Рис. 17.45.** Чтобы продемонстрировать анимацию Flash-ролика (правда, без звука и без интерактивности) зрителям, у которых нет Flash-плеера, отметьте переключатель Animated в разделе Playback на вкладке GIF.

## Чтобы опубликовать GIF-файл:

1. Откройте документ и выберите из меню пункт **File** ⇒ **Publish Settings**. В диалоговом окне выберите профиль публикации или оставьте текущие параметры без изменения.
2. На вкладке **Formats** отметьте флажок **GIF Image (.gif)**.
3. На вкладке **GIF** (рис. 17.43) задайте параметры GIF-файла, как описано в следующих упражнениях.
4. Нажмите кнопку **Publish**.

## Чтобы задать размеры изображения в формате GIF:

- В разделе Dimensions на вкладке **GIF** (рис. 17.44) выполните одно из следующих действий:
  - чтобы самостоятельно задать размеры GIF-изображения, сбросьте флажок **Match Movie** (По ролику) и введите значения в поля Width (Ширина) и Height (Высота);
  - чтобы размеры GIF-изображения были такими же, как у исходного ролика, отметьте флажок **Match Movie**.

## Чтобы создать статический или анимированный GIF:

- В разделе Playback (Воспроизведение) на вкладке **GIF** (рис. 17.45) задайте следующие параметры:

**Static** (Статический). Создается статический GIF.

**Animated** (Анимированный). Создается анимированный GIF. При этом становятся доступны элементы управления, описывающие параметры анимации. Чтобы анимация повторялась циклически, отметьте переключатель **Loop Continuously** (Повторять непрерывно). Чтобы повторять анимацию заданное число раз, отметьте переключатель **Repeat** (Повторить) и введите число повторений. Flash экспортирует все кадры ролика в виде анимированного GIF-файла.

## C

По умолчанию в статический GIF включается первый кадр ролика. Чтобы использовать другой кадр, создайте в соответствующей позиции временной шкалы опорный кадр и выберите его. На вкладке свойств кадра в инспекторе свойств введите в поле Frame Label строку #Static (рис. 17.46).

### Чтобы соблюсти баланс между размером, скоростью загрузки и внешним видом:

- В разделе Options (Параметры) на вкладке GIF (рис. 17.47) отметьте следующие флажки в любом сочетании:

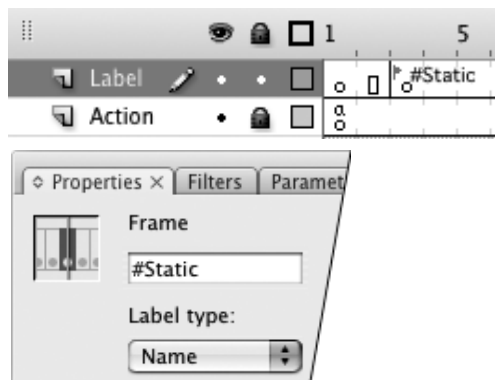
**Optimize Colors** (Оптимизировать цвета). Удаляет неиспользуемые цвета из таблицы цветов в GIF-файле.

**Interlace** (Чередовать). GIF-файл сначала отображается с низким разрешением, а по мере загрузки недостающих частей принимает нормальный вид. (Параметр Interlace следует использовать только для статических GIF-файлов.)

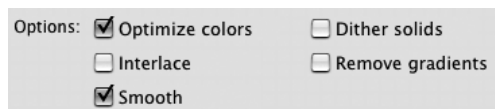
**Smooth** (Сгладить). Flash применяет антиалиасинг для сглаживания изображений, составляющих анимированный GIF. Если этот флажок сброшен, размер файла уменьшается.

**Dither Solids** (Дизеринг чистых цветов). Применяет метод, указанный в списке **Dither** (Дизеринг) к чистым цветам, а также к градиентам и растровым изображениям.

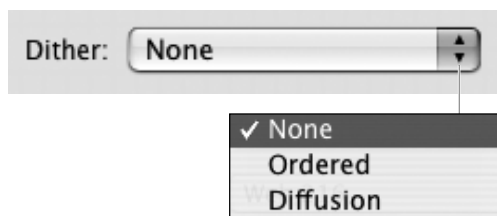
**Remove Gradients** (Удалить градиенты). Сокращает размер файла за счет преобразования градиентных заливок в сплошные (при этом в качестве цвета заливки берется начальный цвет градиента).



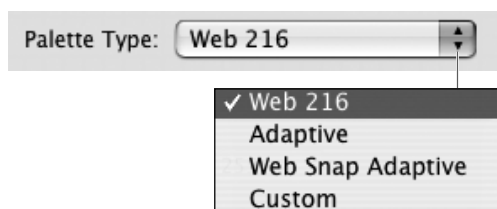
**Рис. 17.46.** Наличие строки #Static в поле метки кадра говорит Flash о том, что именно этот кадр следует включить в статический GIF-файл.



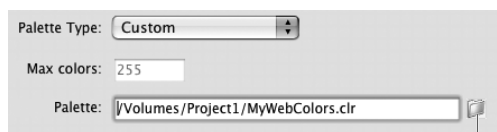
**Рис. 17.47.** В разделе Options на вкладке GIF можно задать параметры, уменьшающие время, в течение которого зритель видит пустой экран. (Флажок Remove Gradients имеется в диалоговом окне Publish Settings, но отсутствует в окне Export GIF).



**Рис. 17.48.** Для формата GIF можно задать один из трех режимов подстановки цветов, отсутствующих в текущей таблице цветов.



**Рис. 17.49.** Выберите палитру для оптимизации цветов в публикуемом GIF-файле. Можно определить собственную палитру либо воспользоваться безопасной для Web или адаптивной палитрой, предлагаемой Flash.



Кнопка «Обзор»

**Рис. 17.50.** При выборе значения Custom из списка Palette Type необходимо указать путь к файлу палитры в поле Palette. Нажмите кнопку «Обзор» (значок папки), чтобы открыть диалоговое окно для выбора файла.

### Чтобы задать порядок обработки цветов, отсутствующих в текущей палитре:

1. Из списка **Dither** (Дизеринг) на вкладке **GIF** (рис. 17.48) выберите одно из следующих значений:

**None** (Нет). Заменяет отсутствующий цвет ближайшим из текущей палитры.

**Ordered** (Упорядоченный). Имитирует отсутствующий цвет повторяющимся орнаментом из других цветов в текущей палитре.

**Diffusion** (Диффузия). Имитирует отсутствующий цвет случайным узором из цветов, входящих в палитру Web 216 (для этого режима необходимо также выбрать на шаге 2 значение Web 216 из списка **Palette Type**).

2. Из списка **Palette Type** (Тип палитры) выберите таблицу цветов для GIF-файла (рис. 17.49).

Предлагаются следующие варианты: Web 216 (стандартная палитра из 216 цветов, считающихся безопасными для Web), Adaptive (Адаптивная) (только цвета, присутствующие в документе, но не более 256), Web Snap Adaptive (Адаптивная с привязкой к Web-цветам) (модифицированная палитра Adaptive, в которой не безопасные для Web цвета заменены ближайшими безопасными) и Custom (Заказная) (таблица цветов задается на шаге 4).

3. Если выбран тип палитры Adaptive или Web Snap Adaptive, то в поле Max Colors (Максимум цветов) укажите, сколько цветов вы готовы использовать.

Этот параметр позволяет еще ограничить размер таблицы цветов и тем самым дополнительно уменьшить размер файла.

4. Если выбран тип палитры Custom, то загружается определенная вами палитра (рис. 17.50).

Нажмите кнопку с изображением папки. В диалоговом окне найдите файл с палитрой, выберите его и нажмите кнопку **Open** (Открыть).

## Определение версии

Подключаемый модуль Flash Player распространен весьма широко, но не у всех пользователей установлена последняя его версия. Чтобы проверить, имеется ли у зрителя подходящий плеер для воспроизведения вашего ролика, воспользуйтесь одним из двух шаблонов, которые генерируют HTML-разметку с определением версии (см. врезку «Об определении версии»).

### Чтобы определить версию Flash Player:

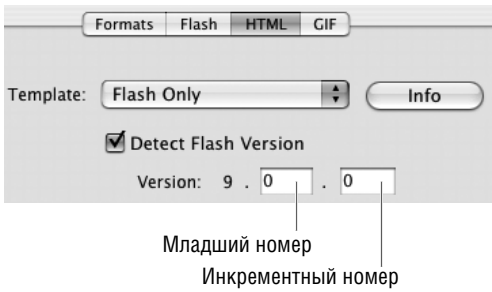
1. Открыв документ Flash, выберите из меню пункт **File** ⇒ **Publish Settings**. В диалоговом окне выберите профиль публикации или оставьте текущие параметры без изменения.
2. Перейдите на вкладку **Flash**, где находятся параметры публикации SWF-файла.
3. Из списка **Version** выберите версию Flash Player 4 или старше (рис. 17.51).
4. Перейдите на вкладку **HTML**.
5. Из списка **Template** выберите пункт Flash Only или Flash HTTPS.

Это единственные шаблоны, в которых автоматически определяется номер версии.



**Рис. 17.51.** Чтобы в опубликованном документе автоматически определялся номер версии Flash-плеера, в списке Version необходимо выбрать версию Flash Player 4 или старше.





**Рис. 17.52.** Если отмечен флажок **Detect Flash Version**, становятся доступны для ввода младшего и инкрементного номера выпуска. Введите полный номер версии плеера, который должен быть установлен на компьютере конечного пользователя, чтобы можно было просматривать ролик.

6. Отметьте флажок **Detect Flash Version** (Определить версию Flash).

Если вы хотите определять конкретные выпуски версии Flash-плеера, введите их номера в соответствующие поля (рис. 17.52).

7. Для подтверждения введенных значений нажмите **ОК**.

При публикации документа с возможностью определения версии Flash создает HTML-файл, который отображает ролик в окне браузера. В этот файл включен JavaScript-код для определения версии Flash-плеера на компьютере пользователя.

### Об определении версии

Определение версии — это процедура проверки того, какая версия Flash Player работает на компьютере конечного пользователя. Три HTML-шаблона из числа поставляемых вместе с Flash CS3 — **Flash Only**, **Flash Only - Allow Full Screen** и **Flash HTTPS** — генерируют код, необходимый для этой цели. Если пользователь пытается воспроизвести ролик на компьютере, где установлен необходимый плеер, то браузер выведет содержимое. Если же плеер более старый или отсутствует вовсе, браузер выведет страницу со ссылкой на сайт компании Adobe, откуда можно загрузить нужную версию. (Если страница создана по шаблону **Flash HTTPS**, то ссылка будет вести на HTTPS-версию сайта Adobe.)

Прочие HTML-шаблоны не определяют версию. Для создания разметки, не требующей Flash-плеера и, следовательно, не нуждающейся в определении версии, предназначены шаблоны **Image Map** и **QuickTime**. JavaScript-код, включенный во все остальные шаблоны, конфликтует с кодом, используемым в шаблонах **Flash Only** и **Flash HTTPS** (который внедряет SWF-файл в HTML-документ). Если вы захотите воспользоваться каким-нибудь из них, то все же сможете рассчитывать на помощь в определении версии от команды публикации, но должны будете создать небольшой дополнительный FLA-файл, который перенаправит пользователя на основной ролик. Опубликуйте этот файл по одному из шаблонов, умеющих определять версию. Если у пользователя установлена подходящая версия, то этот файл отправит его на настоящий ролик, в противном случае зритель получит ссылку на сайт Adobe.

## Печать из Flash

На стадии редактирования документа Flash вы можете распечатывать кадры как отдельные страницы или помещать несколько кадров на одну страницу в виде раскадровки. Сколько кадров будет в каждой строке, определяете вы сами, а Flash соответственно подбирает размеры. Для задания этих параметров применяется команда **Print Margins** (Mac) или **Page Setup** (Windows).

### Чтобы печатать по одному кадру на странице:

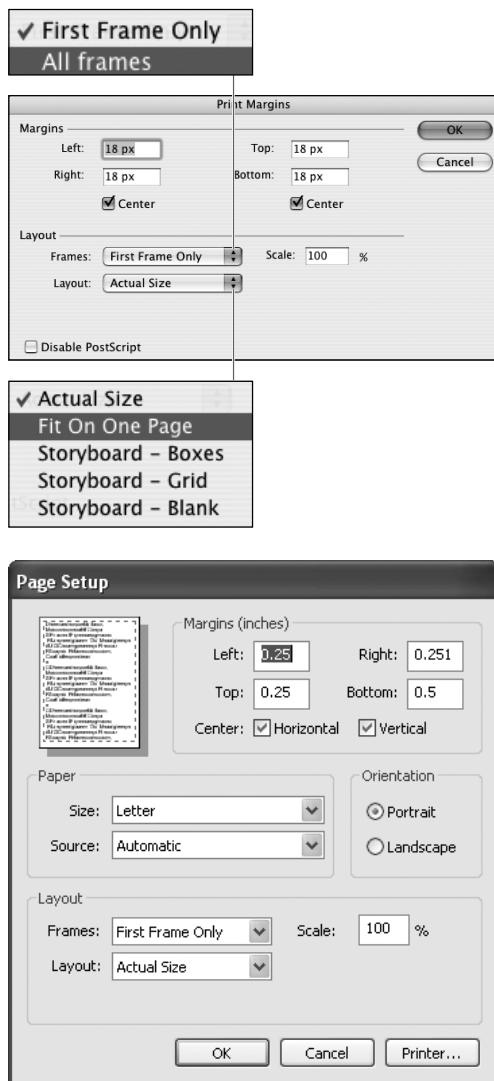
1. Открыв документ, выберите из меню пункт **File** ⇒ **Print Margins** (Mac) или **File** ⇒ **Page Setup** (Файл ⇒ Параметры страницы) (Windows). Открывается диалоговое окно Print Margins или Page Setup (рис. 17.53).
2. Из списка **Frames** (Кадры) выберите пункт **All Frames** (Все кадры).
3. Из списка **Layout** (Макет) выберите пункт **Fit on One Page** (Подогнать к размеру страницы).
4. Нажмите **OK**.
5. Выполните команду **File** ⇒ **Print** (Файл ⇒ Печать). Открывается диалоговое окно Print.
6. Введите номер начального и конечного кадра в поля **From** (Страницы с) и **To** (по).
7. Нажмите кнопку **Print** (Mac) или **OK** (Windows).

C

Если на платформе Macintosh принтер не поддерживает PostScript, не забудьте отметить флажок **Disable PostScript** (Отключить PostScript) в диалоговом окне Print Margins.

C

Чтобы напечатать только первый кадр из каждой сцены, на шаге 2 выберите из списка пункт **First Frame Only** (Только первый кадр).



**Рис. 17.53.** С помощью параметров в диалоговом окне Print Margins (Mac, верхний рисунок) или Page Setup (Windows, нижний рисунок) настраивается печать кадров на этапе разработки: по одному на странице или в виде раскадровки.



**Рис. 17.54.** В списке Layout имеется три варианта макета раскадровки: Boxes, Grid и Blank.

Кадры в прямоугольной рамке



Кадры в ячейках сетки



Кадры сами по себе



**Рис. 17.55.** При печати документа Flash в режиме раскадровки кадры располагаются внутри прямоугольной рамки, в ячейках сетки или сами по себе.

### Чтобы напечатать уменьшенные изображения в виде раскадровки:

1. Выполните шаги 1 и 2 из предыдущего упражнения.
2. Из списка **Layout** (Макет) (рис. 17.54) выберите одно из следующих значений:

**Storyboard — Boxes** (Раскадровка — Прямоугольники). Каждый кадр заключается в прямоугольную рамку.

**Storyboard — Grid** (Раскадровка — Сетка). Кадры печатаются в ячейках сетки.

**Storyboard — Blank** (Раскадровка — Пустая). Печатаются только графические элементы из каждого кадра.

В любом случае предлагается ввести также параметры макета.

3. В поле **Frames** (Кадры) введите число кадров, печатаемых в одной строке. Flash может напечатать в строке до 128 кадров.
  4. В поле **Frame Margin** (Поле кадра) укажите, сколько пустого места оставлять между соседними кадрами.
  5. Нажмите кнопку **Print** (Mac) или **OK** (Windows).
  6. Выполните команду **File** ⇒ **Print**. Открывается диалоговое окно Print.
  7. Если вы хотите распечатать только часть раскадровки, введите номера начальной и конечной страниц в поля **From** и **To** соответственно.
  8. Нажмите кнопку **Print** (Mac) или **OK** (Windows).
- Flash печатает уменьшенные изображения, применяя заданные параметры (рис. 17.55).



Чтобы под каждым кадром печатались номер сцены и номер кадра, отметьте флажок **Label Frames** в диалоговом окне **Page Setup** (Windows). Этот флажок имеется и в окне **Print Margins** (Mac), но метки не печатаются, даже если его отметить.

### Экспорт из Flash в другие форматы

Помимо публикации Flash-ролика для конечных пользователей, его можно также экспортировать. Для экспорта можно выбрать ролик целиком или только один кадр. Flash поддерживает экспорт в различные форматы, не указанные в окне Publish Settings. Чтобы экспортировать содержимое открытого документа, выберите из меню пункт **File ⇒ Export ⇒ Export Image** (Файл ⇒ Экспорт ⇒ Экспортировать изображение) для экспорта одного кадра или **File ⇒ Export ⇒ Export Movie** (Файл ⇒ Экспорт ⇒ Экспортировать ролик) для экспорта всего ролика. В зависимости от выбранного формата предлагается задать те или иные параметры.

На обеих платформах Mac и Windows поддерживаются следующие форматы экспорта: Adobe Illustrator Sequence и Adobe Illustrator Image (.ai); анимированный GIF, последовательность GIF и один GIF-файл (.gif); DXF Sequence и AutoCAD DXF (.dxf); Encapsulated PostScript 3.0 с предпросмотром (.eps); Flash Document (.swf) и Flash Video (.flv); последовательность JPEG и один JPEG-файл (.jpg); последовательность PNG и один PNG-файл (.png); QuickTime (.mov). На платформе Windows можно также экспортировать последовательность Bitmap и один Bitmap-файл (.bmp), последовательность Enhanced Metafile и один расширенный метафайл (.emf), WAV-аудио (.wav), Windows AVI (.avi) и последовательность Windows Metafile и один метафайл (.wmf). На платформе Mac можно также экспортировать последовательность PICT и один PICT-файл (.pct).

### О проекторах

Помимо публикации ролика для воспроизведения в браузере, можно публиковать еще *проекторы*, представляющие собой автономные приложения для отображения содержимого Flash, в которые встроен Flash Player. Чтобы запустить проектор, пользователь должен дважды щелкнуть по значку файла (кружочек с изображением стилизованной буквы F). Проекторы — это удобный способ передать ролик конкретному человеку; например, можно отправить другу электронное письмо с приложенной Flash-открыткой. Проекторы платформенно-зависимы, но на любой платформе можно создать проектор как для Mac, так и для Windows.

Для создания проектора откройте окно Publish Settings (команда **File ⇒ Publish Settings**). На вкладке **Formats** выберите тип проектора, отметив флажки **Windows Projector** и/или **Macintosh Projector**. При следующей публикации документа Flash создает проекторы для указанных платформ наряду со всеми прочими заказанными файлами.

Если вы ведете разработку на платформе Windows, то при публикации проектора для Macintosh будет создана папка с расширением .app. В этой папке хранится вся информация, необходимая для создания автономного проектора для Mac. Пользователи Macintosh этой папки никогда не увидят, при передаче на Mac она преобразуется в файл Flash-проектора. Но, если вы хотите отправить проектор по электронной почте, то эту папку необходимо сжать (например, с помощью программы WinZip).

# ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

---

## А

### Абзац 109

- выделение 109
- выравнивание 110
- задание межстрочного интервала 112
- задание отступа первой строки 112
- задание полей 111
- свойства 109

### Активное содержимое 534, 536, 537

### Актив

- определение 199
- организация 204
- сортировка 200, 204

### Альтернативный контент 543

### Анимация

- антиальясинг 105, 541
- вращательная 341
- графики 266
- добавление опорных кадров 250
- игры в пинг-понг 314, 325, 464
- изменение порядка кадров 321
- изменение скорости 279
- интерполяция движения 259, 314
- интерполяция форм 291
- использование сцен 308
- компонента FLVPlayback 511
- копирование движения 429
- пламени свечи 319
- покадровая 225
- предварительный просмотр 248, 343
- при наведении на кнопку 362
- прыгающего мяча
  - интерполяцией движения 260, 271
  - интерполяцией форм 292
- покадровая 246
- радуги 339

### с переменной скоростью 257

- сглаживание 250
- сложная 307
- сохранение в виде графического символа 325
- текста 106, 160, 318
- традиционная 225, 235, 246
- цветовые эффекты 265
- человечка 326

### Анимированный GIF-файл 441

### Анимированный графический символ

- в слоях-масках 340
- включение в ролик 329
- зацикливание 331
- значок 327
- и размер файла 331
- и символы-клипы 328

### Анимированная открытка 552

### Анимированный фильтр 345

### Арена

- задание размера 34
- задание цвета фона 36
- изменение масштаба 45
- перемещение анимированной графики 254
- просмотр состояний кнопки 358
- расположение элементов на 450

### Атрибут

- абзаца 109
- заливки 72
- обводки 74
- текста 102

## Б

### Баннеры 25, 35, 379, 544

### Безье

- инструменты 62, 147, 151, 154
- кривые 144

- манипуляторы 88, 149, 150
- Библиотека 200
  - активы 199
  - импорт файлов 450
  - общие 202
  - организация активов 204
  - открытие 200, 202
  - папки 204
  - перемещение из одной папки в другую 205
  - просмотр 201
  - терминология 199
  - удаление символов 222
- Битрейт 500, 503, 531
- Буфер обмена
  - операции 122
  - параметры 20

## **В**

- Векторная графика 338, 441
- Видео 487
  - внедренное, редактирование 496
  - для мобильных устройств 488
  - задание ключевых точек 502
  - задание параметров кодирования 491, 493, 498
  - изменение размера и продолжительность видео 504
  - импорт внедряемого 494
  - импорт для прогрессивной или потоковой загрузки 489
  - кадрирование 504, 505
  - методы доставки 490
  - элементы управления воспроизведением 492
- Вложенные объекты 421
- Внедренное видео
  - и другие символы Flash 506
  - и размер файла 488
  - импорт 494
  - помещение на временную шкалу 506
  - предварительный просмотр 507
  - установка новой точки начала 497
- Волновая форма 463, 469, 477, 481
- Восковка 252
  - задание числа соседних кадров 253
  - и интерполяция движения 314
  - и интерполяция форм 296
  - и контрольные точки 302
  - и режим редактирования нескольких кадров 254

- Временная шкала 30
  - внешний вид 30
  - выбор кадров 236
  - добавление промежуточных кадров 234
  - добавление слоев и папок 177
  - задание свойств слоя 182
  - изменение порядка слоев 184
  - изменение размера 226
  - копирование движения 429
  - копирование и вставка кадров 239
  - назначение 30
  - открытие и закрытие 228
  - перемещение кадров 241
  - показ и сокрытие 31, 228
  - помещение внедренного видео 506
  - просмотр кадров 227, 228, 229
  - свойства слоев 176
  - стыковка и отстыковка 31, 227
  - удаление слоев и папок 178
  - управление с помощью кнопок 421
  - цветовое кодирование кадров 260
- Выравнивание абзаца 110

## **Г**

- Главный символ
  - дублирование 221
  - редактирование 220
  - удаление 222
- Гладкая точка
  - модификация 150
  - перемещение 144, 149
  - преобразование в угловую точку 151, 153
  - преобразование угловой точки 152
- Градиент
  - добавление на панель образцов 69
  - заливание контура 97
  - и быстрое действие 98
  - и размер файла 98
  - и режимы рисования 93
  - изменение центра 140
  - окрашивание с помощью 91
  - поворот 143
  - радиальный 68, 141
  - создание 66
  - управление переполнением 143
- Графические форматы файлов 441, 544
- Графический элемент. См. также Форма
  - анимация 266
  - в нескольких слоях 175
  - в одном слое 163
  - выделение 115

добавление обводок и заливок 94  
задание атрибутов заливки 72  
задание атрибутов обводки 74  
задание цвета и градиента 64  
зеркальное отражение 130  
изменение размера 124  
импорт из других программ 441  
инструменты для работы 62  
искажение 134  
копирование и вставка 122, 189  
модификация 113  
ориентация относительно пути 277  
отслеживание во Flash 129  
перекос 132  
перемещение вдоль пути 273  
перемещение по прямой 271  
поворот 131, 268  
повторное использование 199  
позиционирование 128  
преобразование в символы 206  
преобразование экземпляров символа в 223  
простые, создание 61  
распределение по слоям 192  
текст как 100  
управление порядком в стопке 171  
управление с помощью кнопок 432  
Графический планшет 90  
Группа  
    вложенная 170  
    определение 166  
    редактирование 169  
    создание 166

## Д

Двойной щелчок мышью 116  
Действие  
    выбор 381  
    группировка 400  
    идентификация кадров 385  
    помещение в отдельный слой 385  
    тестирование 401  
Десятичная доля дюйма 35  
Дизеринг 546, 547  
Дискретизации частота 529, 531  
Доверенный файл 532  
Документ Flash  
    задание единиц измерения 35  
    открытие нескольких 23  
    открытие существующего 23  
    просмотр библиотеки 201

создание нового 22  
создание по шаблону 26  
уплотнение 24  
Доступ к файлам, управление 531  
Дочерний объекты 421

## Е

Единица измерения 35

## З

Заливка  
    атрибуты 72  
    взаимодействие с комбинированными формами 165  
    геометрических форм 80  
    добавление к контурам 97  
    добавление обводки 94  
    изменение в неактивном слое 188  
    изменение центра градиента 140  
    конвертирование обводок в 160  
    модификация 137  
    определение 61  
    поворот 143  
    пути 147, 165  
    создание произвольной сплошной заливки 89  
    текстовая 160  
    типы 72  
    трансформация 295  
Замкнутый путь 151  
Звук 461

    акустическая обратная связь 468  
    в библиотеке 199, 463  
    в составе видео 495  
    воспроизведение 463  
    добавление в кадры 464  
    добавление к кнопке 468  
    задание степени сжатия и частоты дискретизации 529, 533  
    записанные цифровым способом 531  
    изменения точки начала и конца звучания 486  
    импорт 462  
    источники 466  
    независимые и синхронизированные 470  
    одновременно нескольких 473  
    организация 467  
    остановка 478  
    параметры кодирования 498, 501  
    параметры сжатия 528, 533

перекрывающиеся экземпляры 472  
повтор 480, 486  
помещение в отдельный слой 467  
поток 470, 475  
просмотр формы сигнала 463  
редактирование 461, 482  
синхронизированные с событиями 470  
    и потоковое аудио 475  
    и синхронизированные с началом 474  
одновременный запуск нескольких 473  
остановка 478  
    перекрывающиеся 472  
    порядок обработки 470  
тестирование 470  
удаление неслышных кусков 486  
управление с помощью кнопок 481  
форматы файлов 463  
Зеркальное отражение 130

## **И**

Изменение размера  
    видео 504  
    временной шкалы 226  
    градиента 141  
    графических элементов 124  
    зоны стыковки 54  
    инспектора свойств 58  
    компоненты FLVPlayback 512  
    объекта-рисунка 126  
    окна панели 50  
    примитивной формы 126  
    текстового поля 101  
    шрифта 105  
Изображение. См. также Графические  
    элементы  
    для интерполяции движения 259  
    для покадровой анимации 246  
    импорт 452  
    наложение 344  
    растровые. См. Растровые изображения  
    синхронизация со звуком 475  
Импорт 441  
    в библиотеку и на арену 450  
    видео 489  
    графики из других программ 441  
    звука 462  
    путей 452  
    растровых изображений 442  
    сценариев 380  
    текста 451

Инспектор свойств  
    вкладки 58  
    задание атрибутов обводки 74  
    задание атрибутов текста 102  
    задание параметров сглаживания кисти 93  
    задание размеров элемента 127  
    задание цвета заливки 73  
    изменение размера 58  
    назначение 58  
    позиционирование графических  
        элементов 128  
Инструменты  
    ведро с краской 88, 96, 137  
    выделения 115, 144  
    геометрические фигуры 62, 80  
    для работы с узлами 84, 144  
    добавить узловую точку 154  
    карандаш 84, 85  
    кисть 84  
    lasso 115  
    ластик 159, 163  
    многоугольник 79, 80  
    овал 79, 81  
    овальный примитив 76, 79, 81, 155, 157  
    перо 84, 85, 151  
    подсказки 63  
    преобразование градиента 92, 93, 140, 141  
    преобразования 62  
    прямоугольник 78, 81  
    прямоугольный примитив 76, 78, 81, 155  
    свободное преобразование 124, 133,  
        214, 512  
    спецвыделение 144, 147, 154  
    текст  
        внешний вид 100  
        выбор текста 103  
        и инспектор свойств 102  
        назначение 99  
        форма курсора 100  
    удалить узловую точку 153  
    чернильница 94, 139  
Инструменты рисования 38, 54, 62, 77  
Интерактивная среда 353  
Интерактивность  
    добавление с помощью кнопок 353  
    и версии ActionScript 379  
    определение 383  
    программирование 379, 380, 383, 397  
Интерактивные объекты 411, 421  
Интерполяция  
    битая 260



- движения 259
  - анимация графики с изменяющимся размером 266
  - анимация прозрачности 265
  - анимация прыгающего мяча 260
  - анимация фильтров 352
  - для слоев-масок 340
  - добавление опорных кадров 264
  - задание свойства Tween для последовательности кадров 262
  - и вращение форм 268
  - и графические контейнеры 259
  - и интерполяция форм 291
  - и компонент FLVPlayback 511
  - изменение скорости анимации 279
  - копирование 285
  - нескольких элементов 314
  - ориентация относительно пути 277
  - параметр Ease 281
  - перемещение графических элементов вдоль пути 273
  - перемещение графических элементов по прямой 271
  - подготовка опорных кадров 261
  - цветовые эффекты 265
- форм 291
  - анимация прыгающего мяча 292
  - движущиеся формы 304
  - добавление обводок и заливок 94
  - заливки 295
  - и интерполяция движения 291
  - и пути 304
  - и слои 298
  - нескольких форм 297, 319
  - примитивных 303
  - простые линии 295
  - сгруппированные формы и символы 294
  - сложные формы 299
- Искажение 334
- К**
- Кадрирование
  - видеоизображений 504, 505
  - форм 174
- Кадр 234
  - выбор 236
  - действия 385
    - добавление 386
    - и пользовательские классы 384
    - написание сценариев 384
    - организация 385
    - создание отдельного слоя 385
  - добавление звука 464
  - изменение порядка 321
  - команды 230
  - копирование и вставка 239
  - манипулирование в слоях 239, 311
  - метки
    - именование 377
    - создание 406
    - сопоставление состояниям 376
    - установка точки воспроизведения 407
  - отображение соержимого 226
  - печать 541, 550
  - преобразование в опорные 233
  - просмотр на временной шкале 226, 229
  - редактирование нескольких 254
  - создание промежуточных 234, 235
  - удаление 243
  - установка точки воспроизведения 405
  - цветовое кодирование 260
  - экспорт 552
- Касательный манипулятор 88
- Кели 225, 235
- Кернинг 108
- Кисть, параметры сглаживания 93
- Ключевое слово
  - ActionScript 383
  - в метаданных 33
- Ключевая точка 502
- Кнопка. См. также Компоненты-кнопки, Символы-кнопки
  - виды интерактивности 353
  - включение в ролик 358
  - добавление звука 468
  - наведение курсора 362, 408
  - определение 355
  - открытие Web-страницы 426
  - повторное использование 358
  - получение извещений о нескольких событиях 409
  - предварительный просмотр 358
  - программирование 390
  - программирование клипов как 417
  - создание символа-клипа 374
  - управление временной шкалой 421
  - управление графическими объектами 432
  - управление звуком 481
- Кнопка-клип
  - достоинства 374
  - именование 374

- наделение поведением обычной кнопки 417
- состояния 374
- Кодек 499, 500, 501, 503
- Колонка
  - на библиотечной панели 203
  - текста 103
- Комбинированная форма
  - в неактивном слое 187
  - взаимодействие 164, 165, 167
  - выбор 116
  - и графические объекты других видов 76
  - и рядов в стопке 171
  - и слои 164, 171
  - изменение размера 126
  - конвертирование в другие виды форм 161
  - конвертирование объектов-рисунков в 161
  - модификация 144, 187
  - объединение 174
  - отмена изменений 132
  - преобразование букв в 162
- Комбинированный режим рисования
  - взаимодействие комбинированных форм 164
  - и градиенты 93
  - и другие режимы 76
  - назначение 61
  - предотвращение взаимодействий 167
- Комментарий 388, 391, 398
- Компилятор 402
- Композиция слоев 449
- Компонент
  - FLVPlayback. См. FLVPlayback, компонент
  - и другие библиотечные активы 366
  - именование 366
  - модификация поведения 510
  - назначение 364
  - повторное использование 510
  - типы 364
  - кнопки. См. Компонент-кнопка
- Компонент-кнопка
  - включение в ролик 365
  - модификация 368
  - предварительный просмотр 367
  - программирование интерактивности 413
- Контейнеры
  - графические 259
  - и интерполяция движения 259
  - объекты-рисунки 170
- Контрольная точка 299, 300
- Контурная форма 78, 94, 96
- Корректирующий слой 447
- Кран, модификатор 159
- Кривая
  - изменение формы 145, 149
  - превращение прямолинейного сегмента в 146
- Курсор-перекрестье 65, 78, 100
- Л**
- Линейка 38
- Линия. См. также Обводка
  - атрибуты 74
  - преобразование прямых в кривые 146
  - режимы рисования 84
  - трансформация 295
- М**
- Метаданные 32, 33
- Метод 389
- Миллиметр 35
- Н**
- Наложение, режимы 344
- Направляющая
  - задание параметров 40
  - перемещение 40
  - показ и сокрытие 38
  - привязка к 43
- Настройка безопасности 531, 532
- О**
- Обводка
  - атрибуты 74
  - взаимодействие с заливками 167
  - добавление к заливке 94
  - конвертирование в заливку 160
  - модификация 137
  - ограничение масштабирования 76
  - определение 61
  - параметры сглаживания 93
  - рисование произвольной 85
  - стили 74, 76
  - толщина 74
  - трансформация 297
- Обработчик событий
  - интерактивный компонент-кнопка 413
  - несколько действий в одном 404
  - один обработчик для нескольких событий 415
  - определение 397

**Объект-рисунки**

- в неактивном слое 187
- выбор 116
- изменение размера 126
- инструменты создания 77
- конвертирование в другие типы форм 161
- конвертирование комбинированных форм в 161
- модификация 144, 170, 187
- объединение нескольких 173
- отмена изменений 132
- предотвращение взаимодействия 167

**Овал, углы 83****Окно отображения ролика 538**

- задание размеров 538
- и режимы окна 542
- масштабирование ролика по размеру 539
- параметры воспроизведения 540
- терминология 538
- управление положением ролика в 540

**Окно свойств документа 32**

- внешний вид 33
- задание единиц измерения 35
- задание размера арены 34
- задание цвета фона 36
- задание частоты кадров 36, 256
- и метаданные SWF-файла 32
- открытие 33
- сохранение параметров 36

**Оконечный узел 88****Окончание линии 75****Опорный кадр**

- в традиционной анимации 235
- вставка и преобразование 233
- вставки и добавление 232
- для анимации прыгающего мяча 246
- добавление в интерполяцию движения 264
- добавление в существующую анимацию 250
- добавление звука 464
- дублирование содержимого 232
- и промежуточные кадры 235
- описание вращения 268
- определение 225
- отображение на временной шкале 227
- печать 541
- подготовка для интерполяции движения 261
- преобразование промежуточных кадров в 323
- присоединение сценария 390

создание 230, 247

удаление 243

**Определение версии 548**

Охватывающий прямоугольник 93, 129

Ошибка времени выполнения 403

**П****Панель**

автоматическое сворачивание 57

внешний вид 47

группировка 50, 51, 53

действий

внешний вид 381

доступ 380

изменение размера области сценария 393

инструментарий 381

настройка 382

изменение положения 50

изменение размера 50

инспектора компонентов 364, 368, 508, 514

компонентов 364, 365, 366, 412, 510, 526

компоненты 47

открытие и закрытие 48

плавающие 47

пристыкованные и плавающие 47, 56

разделение стопки 53

расширенного доступа 519

сворачивание и восстановление 49, 57

стопка 52

стыковка 47

управления, темы 512

**Папка**

Configuration 25, 27

в библиотеке 204

доступ 27

слоев. См. Слои

**Параметр**

ActionScript 382

автоматического сворачивания 57

выделения 115

допуска 84

импорта 444, 450

категории 20

ключевой точки 503

кодирования

загрузка 501

казанные 499

и битрейт 503

и качество изображения 503

- предопределенные 498
- сохранение 501
- компонентов 58, 59, 514
- направляющей 40
- операции отмены 224
- подсказок 63
- последовательности опорного кадра 236
- привязки 43
- рисования 84
- сетки 39
- функции 400
- Пароль 528
- Перекрестье, точка регистрации 127, 129, 207, 211, 212
- Переменные шаблона 537
- Печать
  - из Flash 550
  - из Flash Player 541
  - кадров 541, 550
  - раскадровки 551
  - ролика 515
- Пиксель 35, 42
  - экранный 35
- Поведение
  - загрузка 21
  - и ActionScript 380
  - и управление звуком 481
- Подсказка
  - к коду
    - включение/выключение 384
    - и имена экземпляров 398
    - механизм работы 391, 396
    - настройка 382
  - по параметрам 392, 393
- Покадровая анимация 225
  - для слоя-маски 340
  - задание частоты кадров 256
  - использование восковок 252
  - когда применять 246
  - комбинирование с интерполяцией 323
  - определение 246
  - предварительный просмотр 248
  - прыгающего мяча 246
  - редактирование нескольких кадров 254
  - с переменной скоростью 257
  - сглаживание 250
- Пользовательский класс 384
- Поле 111
- Порядок в стопке 171
- Последовательности опорного кадра 227, 236, 242
- Привилегии, доступ к папке 27
- Привязка 41
  - включение и выключение 41
  - задание параметров 43
- Примитивная форма
  - выбор 116
  - и другие графические объекты 76, 79
  - изменение размера 126
  - инструменты для работы 61
  - интерполяция 303
  - конвертирование в комбинированную форму 161
  - конвертирование в объект-рисунок 161
  - модификация 144, 155
  - объединение 173
  - отмена изменений 132
  - предотвращение взаимодействий 167
- Программа чтения с экрана 519
- Прогрессивная загрузка
  - и компонент FLVPlayback 513
  - подготовка видео для 489, 513
  - принцип работы 488
- Проектор 515, 552
- Прозрачность
  - градиента 68
  - и импорт из FreeHand 460
  - интерполяция движения 265
  - панели управления 514
  - цвета 66, 216, 265
  - экземпляра символа 216
- Промежуточный кадр. См. Кадр
- Пропорции 125, 127, 368, 504, 538
- Протокадры 227
- Профиль кодирования 498, 501
- Профилировщик полосы пропускания 516
- Публикация
  - HTML-документа для просмотра SWF-файлов 534
  - SWF-файла 524
  - альтернативные графические форматы 544
  - задание формата 520
  - замечания о доступности 519
  - и версии ActionScript 525, 526
  - и экспорт 521, 552
  - имена файлов по умолчанию 523
  - командой Publish 522, 534
  - местоположение файлов 522
  - отмена 522
  - подготовка ролика для 516
  - проекторов 552
- Пункт 35, 112

Путеводитель 274

## Путь

- абсолютные и относительные 421
- добавление точек 154
- замкнутые и открытые 82
- замыкание 151
- и интерполяция форм 304, 306
- импорт 452
- модификация 113, 155
- определение 88
- ориентация элементов 277
- продолжение 151, 154

## Р

Рабочее пространство 28, 37

Радиальный градиент 68

Раскадровка 235, 515, 551

Распознавание формы 84, 85

Растровое изображение

- и библиотеки Flash 199
- кэширование 338
- редакторы 442
- сглаживание 541
- сжатие 529
- экспорт ролика 515

Растровый текст 106

Расширение от сторонних фирм 21

Редактируемая форма 61, 295, 445, 451

## Режим

- редактирования символов 211, 212, 220, 327
- рисования объектов 61, 77, 93

Родительский объект 421

## Ролик

- варианты доставки 515
- включение кнопок 358
- включение экземпляров символов 213
- добавление видео 487
- добавление звука 461
- добавление компонентов-кнопок 412
- добавление опорных кадров 230
- доступность зрителю 519
- задание частоты кадров 256
- открытие библиотеки 200
- печать 541
- подготовка для оптимального воспроизведения 516
- предварительный просмотр 248
- публикация 513, 520
- размер файла 516
- ручное воспроизведение 248

сцены 308

удаление кадров 243

управление воспроизведением 248, 386, 422

экспорт 515, 521, 552

## С

Сглаживание

- кисти 93
- покадровой анимации 250
- растровых изображений 541

Сегментация 164

Сетка

- задание параметров 39
- и произвольные формы 84
- показ и сокрытие 38
- привязка 41, 43

Сжатие

- ADPCM 530
- JPEG 529
- MP3 530
- параметры 446, 498, 528, 533

Символ. См. также Экземпляры символов

- вставки 191
- графический
- и другие типы символов 208, 328
- преобразование анимации в 325
- статические 327

достоинства 206, 213

дублирование 221

и размер SWF-файла 203, 213

изменение цвета 265

именование 207, 210

назначение 199

обновление счетчиков использования 203

определение 199

предотвращение взаимодействий 167

преобразование графических  
элементов в 206

расщепление 167, 203, 223

редактирование 220

создание 210

сохранение анимации в виде 325

типы 208, 210

точка регистрации 207, 209, 212

точка трансформации 207, 209

трансформация 214

удаление 222

уменьшение числа слоев 326

Символ-клип 332

9-дольное масштабирование 334

- в слое маске 340
- задание прозрачности 435
- и другие типы символов 208, 328
- и события мыши 411
- как интерактивные объекты 411
- откомпилированные 364
- перемещение по экрану 436
- просмотр 337
- размещение 336
- сохранение анимации в виде 332
- Символ-кнопка
  - анимация 362
  - анимированный, создание 362
  - и другие типы символов 208
  - и события мыши 411
  - кадры состояния 355
  - как интерактивные объекты 411
  - меняющий форму, создание 359
  - простейший, создание 355
- Синтаксис
  - подсветка 383
  - проверка правильности 401
  - языка ActionScript 387, 389
- Скорость соединения 516
- Сложные задачи анимации 307
  - анимированные графические символы 329
  - анимированные маски 339
  - изменение порядка кадров 321
  - интерполяция нескольких форм 319
  - использование символов-клипов 336
  - использование сцен 308
  - использование фильтров 345
  - комбинирование интерполяции с покадровой анимацией 323
  - манипулирование кадрами в нескольких слоях 311
  - несколько интерполяций движения 314
  - сохранение анимаций в виде в виде символа-клипа 332
  - сохранение анимаций в виде графических символов 325
- Слой 163
  - активация 187
  - блокировка и разблокировка 176, 181, 183, 467
  - взаимодействие комбинированных форм в одном слое 164, 167
  - вырезание и вставка элементов 189
  - для действий 385
  - для звуков 467
  - задание свойств 180
  - и интерполяция форм 298
  - именование 180, 182
  - корректирующие 447
  - манипулирование кадрами 239, 311
  - маски 195, 339, 511
  - направляющие 176, 193, См. Направляющий слой
  - организация 176, 184
  - папки 176
    - организация слоев 184
    - открытие и закрытие 186
    - переименование 182
    - перемещение слоя в 185
    - переупорядочение 184
    - показ и сокрытие 182, 186
    - создание 177
    - удаление 178
    - управление 180
  - перемещение в папку 185
  - переупорядочение 184
  - порядок в стопке 171, 175
  - представление в Flash 176
  - размещение элементов в разных слоях 175, 385, 467
  - распределение элементов 192
  - создание 177
  - удаление 178
  - управление 180
- Событие
  - выбор 408
  - интерактивных объектов 411
  - определение 397
  - отмена регистрации 440
- Событие мыши 396, 411, 419
- Состояние кнопки
  - кадры для 355
  - предварительный просмотр 358
  - создание 356, 374
  - сопоставление метод кадрам 376
- Спрайт 421, 422
- Среда разработки 19, 28, 99
- Страница приветствия 19, 20, 26
- Стыковка
  - временной шкалы 31, 227
  - зоны
    - добавление панелей в 55
    - изменение размеров 54
    - создание 54
    - удаление панелей из 56
  - панелей 28, 50

**Сценарий**

- добавление кода в существующий 428
- задание параметров 382
- и команда Save As 391
- импорт 380
- интерактивные кнопки 420
- использование комментариев 388, 391, 398
- использование функций 400
- кнопки-клипы 420
- модификация 404
- обработки нескольких событий 415
- определение 380
- ошибки 402
- перенос по словам 387
- повторное использование 396
- предотвращение несанкционированного доступа 528
- проверка синтаксиса 401
- программирование интерактивности 397
- с действиями в кадре 384
- с пользовательскими классами 384
- создание гиперссылок 425
- тестирование 401
- управление воспроизведением ролика 386, 422
- условные предложения 394, 398
- установка точки воспроизведения на заданный кадр 405

**Сцена**

- добавление в ролик 308
- изменение порядка 310
- определение 308
- переименование 310
- подводные камни 310
- тестирование интерактивности 403
- удаление 309

**Т****Текст 99**

- анимация 160
- антиальясинг 105, 106
- вертикальный и горизонтальный 103
- выбор 103, 109
- задание межстрочного интервала 112
- задание отступа 112
- задание полей 111
- задание размера шрифта 105
- задание стиля 107
- задание цвета 107
- задание шрифта 103
- импорт 451

- интерполяция 318
- модификация 109
- ориентация 103
- перенос 100, 101
- разрешение копировать 104
- расщепление 160, 162
- создание блока 100
- статический 99

**Текстовое поле**

- выбор 103
- динамические 99
- для ввода данных 99
- изменение размера 101, 126
- перемещение 101
- перенос по словам 101
- создание 100
- типы 99

**Тема компонентов 372****Тень, эффект 345****Точка**

- регистрации 127, 129, 207, 212
- схода 134
- трансформации 127, 128, 207, 209

**Трекинг 102, 108****У****Угловая точка**

- перемещение 144, 148
- преобразование в гладкую точку 152
- преобразование гладких точек 153
- создание 146
- удаление 151

**Узел**

- выбор 148
- добавление 151
- инструменты для работы 144
- определение 88
- просмотр 147
- создание 151
- удаление 153

**Уплотнение файлов 24****Условное предложение 394, 398****Ф****Фильтр 345**

- анимированные, подводные камни 352
- изменение параметров 347
- наборы параметров 351
- снятие 350

**Фокальная точка 141**

- Формат файла
  - графические 441, 544, 552
  - звуковые 463
  - при публикации 520
  - ролика 515, 552
- Форма. См. также Графический элемент
  - виды 61, 160
  - геометрические 78
  - модификация 144
  - отмена выбора 144
  - предотвращение взаимодействий 167
  - произвольные 84
    - рекомендуемые параметры 84
    - рисование карандашом 85
    - рисование пером 85
    - создание кистью 89
    - трансформация прямоугольника в 296
  - редактируемые 61, 445, 451
- Фотография
  - отсканированные, импорт 441
- Функция 397, 400

## **Ц**

- Цвет
  - безопасные в Web 71, 547
  - выделения 115
  - добавление на панель образцов 69
  - загрузка сохраненного набора 71
  - заливки 64, 72
  - наборы 70
  - обводки 64
  - прозрачность 66, 265
  - пространство 64, 456
  - сетки 39
  - системный диалог для выбора 67
  - создание 64, 70
  - текста 107
  - экземпляра символа 214
- Центральный манипулятор 140
- Цифровая запись 531

## **Ч**

- Частота кадров
  - анимации в Интернете 257
  - в кинофильме 257
  - задание 36, 256
  - и градиенты 98
- Чередование 546
- Чувствительность к контакту 119

## **Ш**

- Шаблон
  - Flash 25
  - HTML 536, 537, 543, 549
  - звука 482
- Шестнадцатеричное кодирование цвета 64
- Шрифт 103
  - рендеринг 105, 106
  - символы 199
  - устройства 104, 106

## **Э**

- Экземпляр. См. также Символ
  - замена 218
  - изменение оттенка 217
  - изменение цвета 214
  - изменение яркости 215
  - именование 398
  - терминология 199
- Экземпляр символа. См. также Символ
  - включение в ролик 213
  - замена 218
  - модификация 214
  - определение 213
  - преобразование в графические элементы 223
- Экспорт ролика 515, 521, 552
- Элемент
  - графического интерфейса 353, 365, 412
  - управления цветом 73
- Эффект затухания
  - для градиентов 68
  - для звуков 482

## **Символы**

- <embed>, тег 534
- <noscript>, тег 534
- <object>, тег 534
- 9-дольное (неискажающее)
  - масштабирование 334

## **А**

- Accessibility Resource Center, Adobe 519
- АСТ, формат 70
- ActionScript, язык 379
  - версии 379
  - действия в кадре 328, 384, 385
  - добавление звука 461
  - знаки препинания 389



и версии Flash Player 525  
и компонент FLVPlayback 512  
и компоненты 364  
и маски 340  
имена экземпляров 398  
интерактивные компоненты-кнопки 412  
использование кнопок для управления  
    графическими объектами 432  
как работает 380  
ключевые слова 383  
комментарии 388, 391, 398  
копирование движения 429  
наделение клипа поведением обычной  
    кнопки 417  
обработка наката на кнопку 408  
обработка нескольких событий 415  
организация ссылок на Web-страницы 427  
отладка 528  
перенос по словам 387  
присоединение к опорным кадрам 390  
синтаксис 389  
события и их обработчики 397, 404  
стили написания сценариев 384  
темы компонентов 372  
тип документа 22  
управление воспроизведением  
    ролика 386, 422  
управление временной шкалой с помощью  
    кнопок 421  
управление кнопками 328, 390  
установка точки воспроизведения на  
    заданный кадр 405  
установки 20, 382  
Adobe Exchange, сайт 21  
Adobe Fireworks 338, 344, 450, 457  
Adobe Freehand 441, 459  
Adobe Illustrator 441, 450, 552  
    задание параметров импорта 450  
    импорт содержимого 453  
    несовместимость 456  
Adobe Illustrator Image, формат 552  
Adobe Illustrator Sequence, формат 552  
Adobe Photoshop 441, 444  
ADPCM, метод сжатия 530  
AIFF, формат 463  
AOL/CompuServe 542  
AutoCAD DXF, формат 441  
AU-файлы 463  
AVI-файлы 552

**B**

Bitmap, формат 552  
BMP-файлы 441

**C**

Click, событие 408, 411, 437  
CLR, формат 70, 71  
CMYK, пространство цветов 67, 456  
Configuration, папка 25, 27

**D**

DoubleClick, событие 411  
DXF, формат 552

**E**

EMF-файлы 441  
EnterFrame, событие 437, 440  
EOLAS, патент 534

**F**

Flash Player 524  
    вкладка Global Security Settings 532  
    выбор версии 525  
    запуск 524  
    и 9-дольное масштабирование 334  
    и ActionScript 525, 526  
    и антиальясинг 106  
    и прозрачный фон 542  
    настройки безопасности 531, 532  
    определение версии 521, 548  
    печать из 541  
    подготовка видео для 489  
    подготовка ролика для 524  
    просмотр всех кадров 249  
FLA-файлы 494, 513, 515  
FLVPlayback, компонент 508  
    анимация 511  
    выбор источника видео 508  
    и интерполяция движения 511  
    и размер ролика 488  
    изменение параметров 514  
    определение 510  
    просмотр на этапе разработки 511  
    сравнение с символом-клипом 511  
    темы панели управления 512, 513

**G**

GetURL, действие 544  
GIF-файлы 71, 441, 515, 552

**Н**

HSB, пространство цветов 64

HTML

документы

активное содержимое 534

определение 534

публикация 520, 521, 535

задание цвета 64

и JavaScript 534, 537

метаданные 32

теги 534, 537

шаблоны 537, 543, 549

**I**

Internet Explorer 534, 537, 542

**J**

JavaScript 521, 534, 536, 537, 549

JPEG, формат 441

**M**

Mac OS X 542

Macintosh

и PostScript 550

команда Print Margins 550

проекторы 552

системные диалоги для выбора цвета 67

экспорт кадров и роликов 552

MacPaint 441

MouseMove, событие 411

MOV-файлы 488

Mozilla 542

MP3

кодек 501

сжатие 530

формат 463

MSAA, технология 519

**N**

Netscape 542

**P**

PICT-файлы 441, 552

PNG, формат 441, 450, 457, 515

PNTG, формат 441

PostScript 550, 552

PSD-файлы 444

задание параметров импорта 444

импорт из 447

композиция слоев 449

переименование слоев 449

**Q**

QTIF-файлы 441

QuickTime

поддерживаемые графические

форматы 441

поддерживаемые звуковые форматы 463

публикация в формате 488, 515, 521

с Flash-дорожкой 520

экспорт для 521, 552

**R**

RGB, пространство цветов 64, 456

RollOut, событие 411

RollOver, событие 411

**S**

SGI-файлы 441

SPL-файлы 441

Stop, действие 386

SWF-файлы

выкладывание на Web-сервер 513

где сохраняется 249

замечания о размере 104, 203, 213

и внедренное видео 494

и компонент FLVPlayback 513

и символы 203, 213

и шрифты устройства 104

метаданные 32

настройки безопасности 531, 532

просмотр 515

публикация 524

сжатие 528

**T**

TGA-файлы 441

TIFF-файлы 441

Trace, действие 404, 405, 528

**W**

WAV, формат 463

Web 216, палитра 547

WMF, формат 441

**X**

XML-код 430, 431



Издательский дом «ДМК-пресс» предлагает для продажи в сети ваших магазинов **универсальный держатель EASY-READ™ – приспособление, которое помогает читать книги и журналы в любом месте: дома, на работе, на пляже, в походе.** Иными словами – везде, где угодно.

Держатель устойчив на любой поверхности. Уникальным достоинством держателя является возможность расположения его не только на столе, стене, полке, но и на **мониторе компьютера.**

**Универсальный держатель EASY-READ™:**

- Не имеет аналогов в мире
- Чрезвычайно легкий – всего 85 граммов
- Компактный – в сложенном виде занимает место менее дамского зонтика
- Изготовлен из современных экологически чистых материалов
- Пригоден для документов любого размера, вплоть до формата А3
- Предназначен для оснащения школ, офисов, библиотек, интернет-кафе
- Невысокая цена, изящный чехол
- Удобная и полезная вещь для ежедневного использования.
- Прекрасный подарок любому ценителю всего необычного.



**Условия сотрудничества:**

- Каждой торговой точке розничной сети наша компания предоставляет красочную и эффектную витрину-стенд, привлекающую внимание покупателей, что даст возможность самостоятельно ознакомиться с достоинствами держателей **EASY-READ™.**
- Адреса розничных магазинов будут размещены на сайте нашей компании и войдут в список точек, в которых можно приобрести держатели **EASY-READ™.**



Подробнее с данным продуктом можно ознакомиться на сайте [www.easy-read.ru](http://www.easy-read.ru).

Приобрести оптом и в розницу держатель можно приобрести в компании «Альянс-книга» [www.aliants-kniga.ru](http://www.aliants-kniga.ru), тел. (495) 258-91-94, 258-91-95



Книги издательства «ДМК Пресс» можно заказать в торгово-издательском холдинге «АЛЪЯНС-КНИГА» наложенным платежом, выслав открытку или письмо по почтовому адресу: **123242, Москва, а/я 20** или по электронному адресу: **orders@alians-kniga.ru**.

При оформлении заказа следует указать адрес (полностью), по которому должны быть высланы книги; фамилию, имя и отчество получателя. Желательно также указать свой телефон и электронный адрес.

Эти книги вы можете заказать и в Internet-магазине: **www.alians-kniga.ru**.

Оптовые закупки: тел. (495) 258-91-94, 258-91-95; электронный адрес **books@alians-kniga.ru**.

Кетрин Ульрих

## **Интерактивная Web-анимация во Flash**

Главный редактор *Мовчан Д. А.*  
dm@dmk-press.ru

Перевод с английского *Слинкина А. А.*

Верстка *Ребенок Ю. Л.*

Графика *Ребенок Ю. Л.*

Дизайн обложки *Мовчан А. Г.*

Подписано в печать 17.07.2010. Формат 70×100  $\frac{1}{16}$ .

Гарнитура «Миниатюра». Печать офсетная.

Усл. печ. л. 27,2. Тираж 500 экз. Зак. №

Web-сайт издательства: [www.dmk-press.ru](http://www.dmk-press.ru)