

Глава 7

Работаем с символами и их потомками

В этой главе изучаются функции программы Flash 8 по работе с символами и их потомками.

Символом называется вторичный объект, образованный из некоторых объектов документа произвольных типов или импортированной информации, который обладает двумя отличительными свойствами:

- символ хранится в рабочей библиотеке документа, а не в самом документе;
- из любого символа можно сформировать любое количество копий, которые размещаются в документе и называются *символьными объектами* или *потомками* символа (instances).

Во Flash 8 используются символы трех типов:

- *символ-графика*, или графический символ (graphic), — представляет собой одно из двух:
 - статический элемент, состоящий из одного кадра;
 - анимационный элемент, содержащий несколько кадров, поведение которого синхронизируется внешним анимационным процессом;
- *символ-кнопка* (button) — интерактивный элемент, реагирующий на воздействие мышью (см. главу 10);
- *символ-видеокадр* (movie clip) — анимационный элемент, создающий собственный автономный анимационный процесс.

Потомки символов (символьные объекты) обладают следующими свойствами:

- к любому потомку можно применять следующие операции, которые не будут воздействовать на образующий его символ или на другие потомки этого символа (см. ниже раздел “Обрабатываем символьные объекты”):
 - перемещать потомок в документе и его трансформировать;
 - создавать в любом потомке один цветовой эффект;
 - создавать в потомке типа кнопки или клипа до семи эффектов фильтрации, а также один эффект смешения цветов;
 - анимировать потомок с помощью раскадровки движения (см. раздел “Используем раскадровку движения” главы 9) путем изменения его координат и параметров трансформации, а также параметров созданных в нем эффектов фильтрации и цветового эффекта;
- допускается изменять исходное поведение потомка, определяемое типом образующего символа (любой потомок символа можно сделать объектом-графикой, объектом-кнопкой или объектом-клипом);
- при замене образующего символа параметры всех его существующих потомков сохраняются;
- потомку с текущим поведением кнопки или клипа можно присваивать командные действия, заданные на языке ActionScript в виде сценария.

Во Flash 8 символы и их потомки рекомендуется использовать в следующих случаях:

- с целью минимизации размера файла документа, а также файла результирующего видеоклипа в случае использования большого количества копий некоторой совокупности объектов, для чего из них следует образовать символы;
- для облегчения доступа к информации других документов Flash 8, которая может вам понадобиться при разработке документа;
- для обеспечения доступа к стандартным графическим, интерактивным и анимационным элементам, хранящимся в файлах стандартных библиотек программы Flash (ее различных версий);
- для регулярного обновления общей информации, используемой во многих видеоклипах формата Flash, работающих в Интернете (см. раздел “Подключаем общие элементы” главы 14);
- для создания в документе интерактивных и анимационных элементов типа кнопок и встроенных клипов;
- для создания сложных анимационных эффектов путем использования вложенных анимационных символов;
- для вставки сценариев в объекты документа, управляющих видеоклипком в зависимости от возникновения в них определенных управляющих событий. Для этой цели данные объекты представляются в документе потомками символов и каждому из них задается требуемое поведение (кнопки или клипа);
- для регулировки различных параметров объектов любых типов (их яркости, прозрачности или раскраски), а также возможного анимирования этих параметров, для чего данные объекты представляются потомками символов;
- для перемещения векторных форм по заданной траектории, для чего их следует преобразовать в символьный объект (этого можно также достичь путем преобразования данных форм в групповой объект или в объект рисования).

Создаем символы

В программе Flash 8 предусмотрены следующие способы создания в документе новых символов:

- путем формирования символа в рабочем окне документа или его импорта из файла другого документа Flash 8;
- из выделенных объектов текущего кадра документа;
- из выбранных ключевых кадров временной диаграммы документа;
- путем перетаскивания символа мышью из открытой рабочей библиотеки другого документа Flash-типа в библиотеку активного документа или в сам документ;
- перетаскиванием потомка символа из другого открытого документа в текущий документ;
- копированием через буфер обмена Windows в текущий документ потомка символа, выбранного в другом открытом документе;
- путем импорта в документ Flash 8 файла формата Flash (с расширением `.swf`), содержащего символьную информацию документа Illustrator CS2 (см. ниже раздел “Подготавливаем символы в Illustrator CS2”).

Создание символа в рабочем окне документа

Указанный способ является самым распространенным. Он позволяет сформировать практически любой символ таким же образом, как создается сам анимационный документ, а также импортировать требуемый символ из любого другого документа Flash 8, хранящегося в файле.

Для перехода в режим формирования нового символа используется команда **Insert⇒New Symbol** (Вставить⇒Новый символ). При ее выполнении открывается диалоговое окно **Create New Symbol** (Создать новый символ), в котором задаются параметры нового символа. Для этого окна предусмотрено два состояния: базовое (см. рис. 7.1) и расширенное (см. рис. 7.4).

В зависимости от способа создания нового символа и его дальнейшего использования количество настраиваемых в этом окне параметров может быть различным. Если символ будет создаваться непосредственно в окне документа, при этом его возможные параметры связи будут выбираться позже (см. раздел “Подключаем общие элементы” главы 14, подраздел “Общие символы”), то вам необходимо задать всего два основных параметра этого символа, представленные в обоих состояниях окна:

- его название (поле **Name**);
- тип символа, определяющий исходное поведение его будущих потомков: встроенный клип (переключатель **Movie Clip**), кнопка (переключатель **Button**) или графический символ (переключатель **Graphic**).

Рассмотрим дополнительные параметры символа. Для их настройки следует перевести окно **Create New Symbol** из базового состояния в расширенное, щелкнув на кнопке **Advanced** (для обратного перехода выполняется щелчок на той же кнопке, которая тогда будет иметь название **Basic**).

В области **Linkage** (Связь) окна задаются параметры связей символа в случае его использования в качестве общего или локального (см. раздел “Подключаем общие элементы” главы 14, подраздел “Общие символы”). Эти параметры вы можете указать и после создания символа, для чего его следует выбрать в рабочей библиотеке документа и применить к нему команду **Linkage** контекстного меню панели этой библиотеки.

В области **Source** (Источник) окна находятся следующие элементы настройки:

- кнопка **Browse**, предназначенная для выбора на диске файла документа Flash 8, из которого будет импортирован в данный документ требуемый символ;
- кнопка **Symbol**, используемая для выбора символа из заданного файла другого документа;
- флажок **Always update before publishing**, подключающий режим автоматического обновления в текущем документе импортированного символа при формировании результирующего клипа формата Flash в случае, если исходный образец этого символа был изменен в документе-источнике.

Порядок создания нового символа в рабочем окне документа состоит в следующем.

1. Выберите команду **Insert⇒New Symbol**, открыв диалоговое окно **Create New Symbol**.
2. Задайте в поле **Name** окна название будущего символа, которое не должно совпадать с именами существующих символов документа, а также его тип (один из трех переключателей).
3. Если создаваемый символ будет использован в качестве общего или локального (см. раздел “Подключаем общие элементы” главы 14, подраздел “Общие символы”), и вы хотите сразу же задать для него параметры связи, то сделайте это в области **Linkage** данного окна, для чего переведите его в расширенное состояние (кнопка **Advanced**).
4. Если вы будете импортировать символ из другого документа, то сделайте следующее. Выберите на диске файл документа-источника (кнопка **Browse**), выделите в дополнительном окне требуемый символ (кнопка **Symbol**), определитесь в отношении режима автоматического обновления импортируемого символа (флажок **Always update before publishing**) и щелкните на кнопке **OK**, завершив на этом выполнение данной инструкции.
5. В случае непосредственного формирования символа сделайте следующее. Закройте окно **Create New Symbol** щелчком на кнопке **OK**. При этом произойдет переход в режим

создания нового символа в рабочем окне документа. Об этом свидетельствует наличие в окне крестообразной метки точки регистрации, относительно которой следует разместить данный символ, а также название символа на панели инструментов Edit Bar (Панель редактирования).



Точка регистрации символа задает исходное положение точки трансформации потомка этого символа. Метка точки регистрации отображается как в самом символе, представленном в окне просмотра панели библиотеки, так и в любом его потомке символа, выделенном в документе (см. рис. 7.8).

6. Создайте символ в рабочем окне документа, используя для этого необходимые средства программы Flash 8 (в том числе и команду **Import to Stage** (Импорт в область кадра) подменю **Import** (Импорт) меню **File** (Файл), позволяющую импортировать в документ требуемую информацию, хранящуюся в файле).
7. Щелкните на названии текущей сцены документа, указанном слева на панели инструментов **Edit Bar**. При этом произойдет выход из режима создания нового символа (без формирования его потомка в документе).

На рис. 7.1 изображено диалоговое окно **Create New Symbol** (Создать новый символ), находящееся в базовом состоянии, в котором заданы основные параметры символа.



*Рис. 7.1. Вид окна **Create New Symbol** в базовом состоянии*

На рис. 7.2 рассмотрен пример создания в окне документа простого статического символа графического типа, представляющего собой фигуру овала с градиентной заливкой. Справа здесь изображена рабочая библиотека документа с данным символом. Обратите внимание на крестообразную метку точки регистрации, с которой был совмещен в данном случае геометрический центр созданного символа.

На рис. 7.3 приведен пример создания анимационного символа, имитирующего развевающийся флаг, путем импорта в документ информации из файла формата Flash.

На рис. 7.4 изображено окно **Create New Symbol**, находящееся в расширенном состоянии, в режиме импорта символа из файла другого документа. Об этом свидетельствует информация внизу окна, представляющая собой адрес и имя файла документа, из которого импортируется символ (первая строка), а также название этого символа (вторая строка).

На рис. 7.5 представлено диалоговое окно **Select Source Symbol** (Выбрать символ-источник), открываемое кнопкой **Symbol** окна **Create New Symbol**, в котором производится выбор символа в заданном документе-источнике.

На рис. 7.6 показана панель рабочей библиотеки документа с символом, импортированным из файла другого документа (параметры документа-источника и выбранного в нем символа представлены на рис. 7.4 и 7.5). Обратите внимание на два момента. Во-первых, тип данного символа (встроенный клип) совпадает с типом исходного символа, а не с типом нового (графический), заданным в окне **Create New Symbol** (см. рис.7.4). Во-вторых, вместе с выбранным символом были импортированы в документ и те исходные символы, из которых этот символ был образован (они находятся в папке **Supporting Graphic Files** рабочей библиотеки).

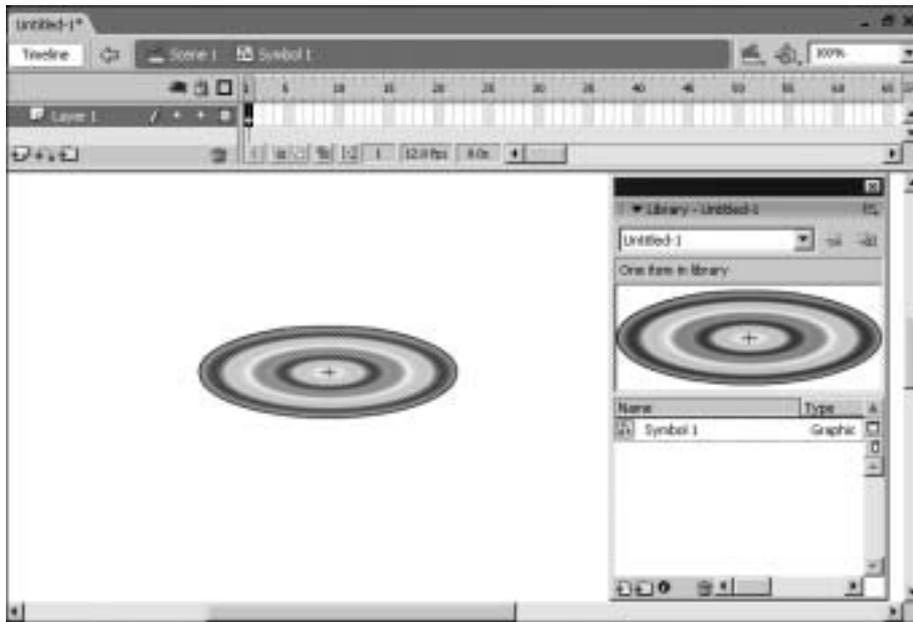


Рис. 7.2. Первый пример создания нового символа

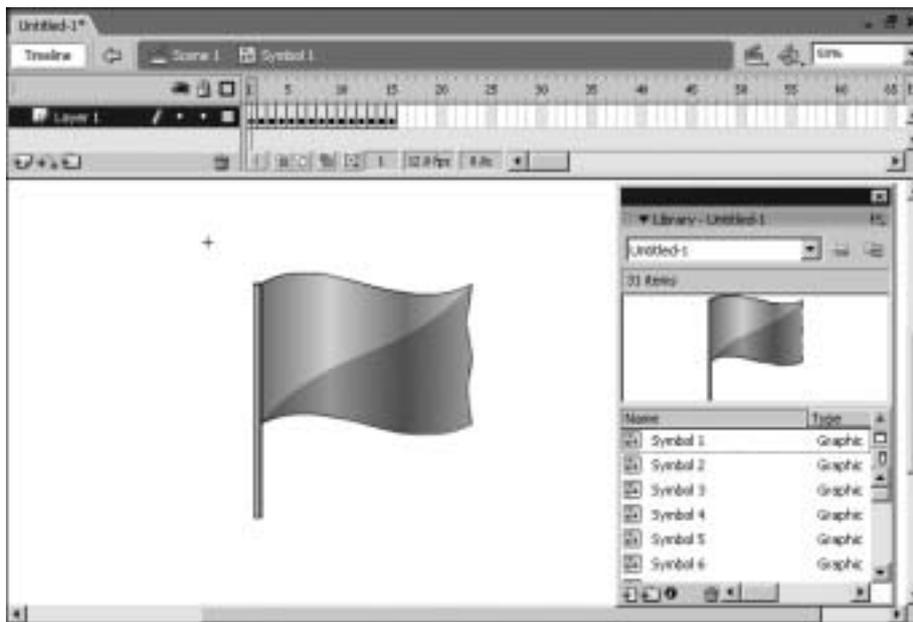


Рис. 7.3. Второй пример создания нового символа

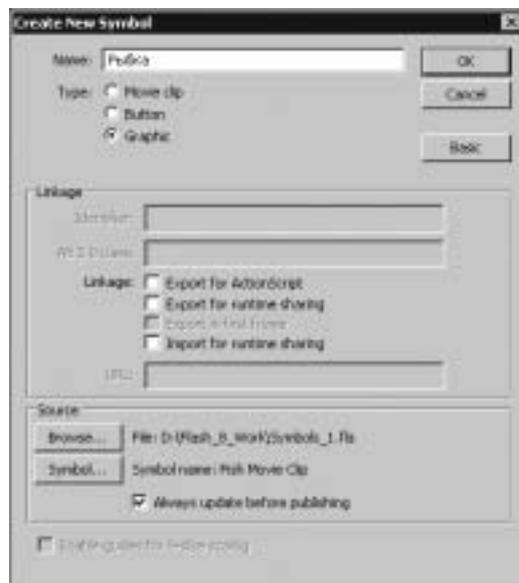


Рис. 7.4. Вид окна Create New Symbol в расширенном состоянии



Рис. 7.5. Окно Select Source Symbol

Преобразование в символы объектов документа

Программа Flash 8 позволяет формировать новые символы не только путем их непосредственного создания в разрабатываемом документе или импорта из другого документа, но и посредством преобразования в них существующих объектов документа. Предусмотрено две разновидности этой операции:

- создание одного символа из выделенных объектов текущего кадра документа;
- создание нескольких символов из содержимого выбранных ключевых кадров документа.

Создание символа из выделенных объектов документа

Данная операция выполняется с помощью команды **Modify**⇒**Convert to Symbol** (Изменить⇒Преобразовать в символ), а ее результатом является новый статический символ заданного типа.

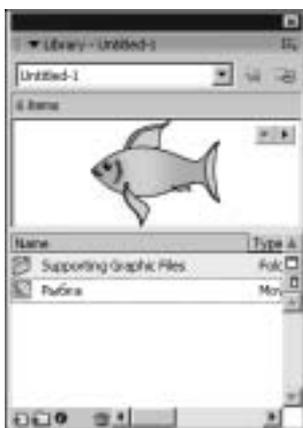


Рис. 7.6. Вид панели рабочей библиотеки документа с импортированным символом



Рис. 7.7. Окно Convert to Symbol

Порядок создания символа из выделенных объектов документа состоит в следующем.

1. Перейдите в нужный кадр документа и выделите в нем те объекты (они могут находиться в различных слоях), из которых будет образован символ.
2. Выберите команду **Modify**⇒**Convert to Symbol**, открыв ее диалоговое окно **Convert to Symbol** (Преобразовать в символ).
3. Введите название нового символа, которое не должно совпадать с именами существующих символов документа (поле **Name**).
4. Задайте тип символа (один из трех переключателей).
5. Выберите положение точки регистрации нового символа с помощью индикатора регистрации (он имеет название **Registration**), представляющего собой квадратный значок, по периметру и в центре которого расположены девять квадратных маркеров, доступных для манипуляции мышью (см. рис. 7.7).
6. Если создаваемый символ будет использован в качестве общего или локального, и вы хотите сразу же задать для него параметры связи, то сделайте это в области **Linkage** данного окна, переведя его в расширенное состояние (см. раздел “Подключаем общие элементы” главы 14, подраздел “Общие символы”).
7. Закройте окно **Convert to Symbol** щелчком на кнопке **OK**. В результате будет создан новый символ, потомок которого заменит в документе исходные выделенные объекты, расположившись в самом верхнем из тех слоев документа, в которых эти объекты находились.

На рис. 7.7 показано диалоговое окно **Convert to Symbol**, открытое в базовом состоянии.

На рис. 7.8 изображены фрагмент окна документа и рабочая библиотека документа после создания нового символа с параметрами, представленными на рис. 7.7, из фигуры звезды. Обратите внимание на положение крестообразной метки точки регистрации символа. Она является исходной для точки трансформации потомка символа, метка которой имеет вид колесика (она также изображена на этом рисунке).

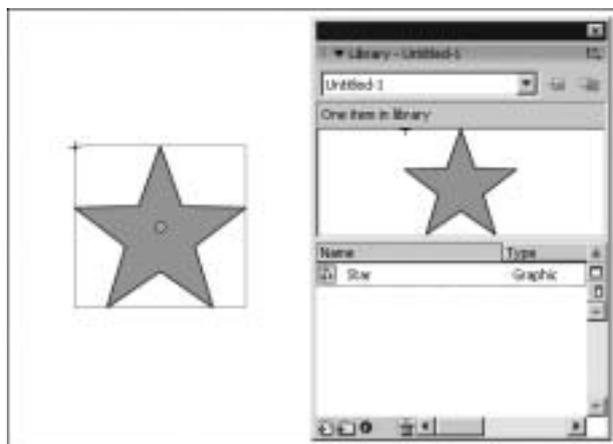


Рис. 7.8. Вид документа и рабочей библиотеки после создания символа из выделенного объекта документа

Создание символов из выбранных ключевых кадров документа

Указанная операция выполняется с помощью команды **Insert⇒Timeline⇒Create Motion Tween** (Вставить⇒Временная диаграмма⇒Создать раскадровку движения). В результате ее применения произойдет следующее:

- будет создано несколько статических символов графического типа (по количеству выделенных ключевых кадров), имеющих стандартные названия (**Tween <порядковый номер>**);
- произойдет формирование раскадровки движения в выделенных индикаторах кадров временной диаграммы документа (см. раздел “Используем раскадровку движения” главы 9);
- если в каком-либо слое документа было выделено несколько соседних ключевых кадров, то в них разместятся потомки того символа, который образуется из первого из этих кадров.

Порядок создания символов их выбранных ключевых кадров документа состоит в следующем.

1. На временной диаграмме документа выделите мышью (с помощью клавиши **<Ctrl>** или **<Shift>**) те ключевые кадры, из содержимого которых должны быть образованы символы.
2. Выберите команду **Insert⇒Timeline⇒Create Motion Tween**. При этом в рабочей библиотеке документа появятся новые символы графического типа, исходное содержимое выбранных ключевых кадров будет заменено потомками этих символов, а в индикаторах кадров для этих символов сформируется раскадровка движения.
3. Выполните, если необходимо, дополнительные операции обработки из приведенного ниже перечня:
 - переименуйте созданные символы в рабочей библиотеке документа (переход в режим редактирования названия символа производится двумя щелчками на этом названии);
 - для отмены сформированной раскадровки движения выберите команду **Insert⇒Timeline⇒Remove Tween** (Вставить⇒Временная диаграмма⇒Удалить раскадровку), находящуюся в меню **Insert** на месте расположения предыдущей команды **Create Motion Tween**;
 - для восстановления первоначального содержимого ключевых кадров (без удаления созданных символов) выберите команду **Edit⇒Undo** (Правка⇒Отменить).

Перетаскивание символа из библиотеки другого документа

Данная операция может быть выполнена двумя способами:

- перетаскиванием выбранного символа в библиотеку активного документа;
- перетаскиванием его в область рабочего окна документа.

Порядок перетаскивания символа из библиотеки другого документа таков.

1. Откройте рабочую библиотеку документа-источника, в котором находится требуемый символ. Это можно сделать тремя способами:
 - откройте рабочую библиотеку данного документа командой **File⇒Import⇒Open External Library** (Файл⇒Импорт⇒Открыть внешнюю библиотеку);
 - откройте сам документ-источник командой **File⇒Open** (Файл⇒Открыть) и его рабочую библиотеку;
 - разверните рабочую библиотеку (щелчком на заголовке ее панели) данного документа-источника, если он находится в открытом состоянии.
2. Перетащите значок или миниатюру символа, выбранные в библиотеке документа-источника, в нижнюю часть панели рабочей библиотеки активного документа либо в сам документ. В первом случае будет сформирован такой же символ (без его потомка в документе), а во втором — символ и его потомок, помещенный в область текущего кадра документа. При этом может произойти следующее:
 - если в обрабатываемом документе уже есть символ с тем же названием, что и выбранный символ документа-источника, то на экране откроется диалоговое окно **Resolve Library Conflict** (Разрешить конфликт библиотек). В этом окне вы можете выполнить одно из трех: а) выбрать верхний переключатель и щелкнуть на кнопке **OK** (при этом старый одноименный символ не будет заменен на новый); б) выбрать нижний переключатель и щелкнуть на **OK** (будет заменен); в) щелкнуть на кнопке **Cancel** окна, после чего следует переименовать заменяемый символ и повторить данную операцию;
 - если выбранный символ документа-источника является общим или локальным, то после его копирования в текущий документ он станет локальным, содержащим все необходимые параметры связи (см. раздел “Подключаем общие элементы” главы 14, подраздел “Общие символы”);
 - если документ-источник открывался из файла, а не является новым, то скопированный из него символ сохранит связь с файлом исходного документа. Это означает, что если для данного символа установить флажок **Always update before publishing** в диалоговом окне **Symbol Properties** (Свойства символа) (см. рис. 7.9), то этот символ будет автоматически обновляться в документе-приемнике (при формировании файла видеоклипа) в случае изменения его прототипа в исходном документе.

Перетаскивание потомка символа из другого документа

Порядок перетаскивания потомка символа из другого документа таков.

1. Откройте во Flash 8 только документ-источник и документ-приемник.
2. Расположите окна этих документов рядом на экране, выполнив для этого команду **Window⇒Tile** (Окно⇒Мозаикой).
3. Перетащите мышью выбранный потомок некоторого символа из окна документа-источника в окно документа-приемника. В результате во втором документе будет сформирован такой же символ с потомком, первый из которых будет помещен в рабочую библиотеку, а второй — в область текущего кадра документа. При этом может произойти следующее:

- если в документе-приемнике уже есть символ с тем же названием, что и образующий символ для выбранного его потомка, то на экране откроется диалоговое окно **Resolve Library Conflict**, в котором вы должны выбрать один из трех возможных вариантов действий (см. выше подраздел “Перетаскивание символа из библиотеки другого документа”);
- если образующий символ выбранного потомка документа-источника является общим или локальным, то после его копирования в текущий документ он станет локальным, содержащим все необходимые параметры связи (см. раздел “Подключаем общие элементы” главы 14, подраздел “Общие символы”);
- если документ-источник открывался из файла, то скопированный из него символ сохранит связь с файлом исходного документа, при этом режим обновления этой связи будет отключен.

Копирование потомка символа через буфер обмена

Порядок копирования потомка символа через буфер обмена Windows состоит в следующем.

1. Откройте документ-источник и сделайте его активным.
2. Выделите в нем потомок требуемого символа и выберите команду **Edit⇒Copy** (**Правка⇒Копировать**).
3. Перейдите в документ-приемник, в который должен быть скопирован данный символ.
4. Выберите требуемый ключевой кадр документа, в который должен быть помещен потомок символа, скопированный в буфер обмена.
5. Выполните одну из двух команд меню **Edit** (**Правка**): **Paste in Center** (**Вставить по центру**) или **Paste in Place** (**Вставить по месту**). В первом случае данный потомок расположится в центр области текущего кадра, а во втором — в том месте, где в документе-источнике находился его прототип. Сам же образующий символ будет скопирован в рабочую библиотеку данного документа. При этом может произойти следующее:
 - если в документе-приемнике уже есть символ с тем же названием, что и образующий символ выбранного потомка документа-источника, то откроется диалоговое окно **Resolve Library Conflict** (**Разрешить конфликт библиотек**), в котором вы должны выбрать один из трех возможных вариантов действий (см. выше подраздел “Перетаскивание символа из библиотеки другого документа”);
 - если выбранный символ документа-источника является общим или локальным, то после его копирования в текущий документ он станет локальным, содержащим все необходимые параметры связи (см. раздел “Подключаем общие элементы” главы 14, подраздел “Общие символы”);
 - если документ-источник открывался из файла, то скопированный из него символ сохранит связь с файлом исходного документа, при этом режим обновления этой связи будет отключен.

Создав в документе новый символ, вы можете в любой момент изменить его параметры. Для этого выберите требуемый символ на панели рабочей библиотеке документа, раскройте ее контекстное меню и выберите команду **Properties** (**Свойства**). На экране откроется диалоговое окно **Symbol Properties** (**Свойства символа**), в котором и производится настройка параметров текущего символа. Это окно имеет такое же устройство, как и окно **Create New Symbol** (**Создать новый символ**), для которого предусмотрено два состояния: базовое и расширенное (см. рис. 7.1 и 7.4). Исключением является лишь наличие в окне **Symbol Properties** дополнительной кнопки **Edit** (**Редактировать**), с помощью которой подключается режим редактирования выбранного символа (рис. 7.9).

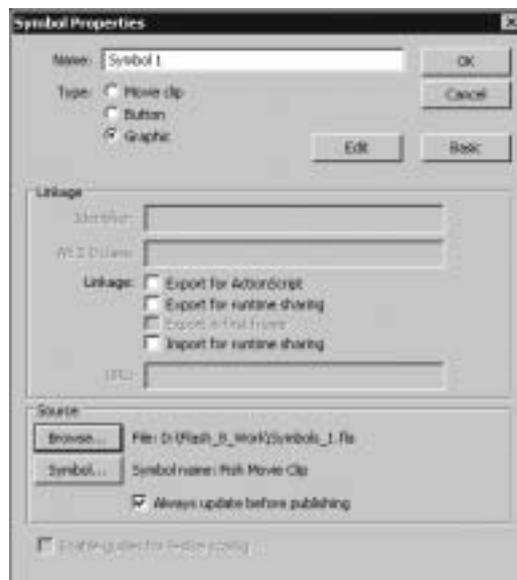


Рис. 7.9. Вид диалогового окна Symbol Properties в расширенном состоянии

На рис. 7.10 показан пример изменения параметров символа с помощью команды Properties. В данном случае выполнялась замена выбранного в документе символа-графики на символ-клип из стандартной библиотеки графических символов программы Flash 5 (параметры этого символа представлены внизу рис. 7.9). Здесь изображено: слева — фрагмент окна документа с панелью рабочей библиотеки до применения команды Properties, а справа — после.

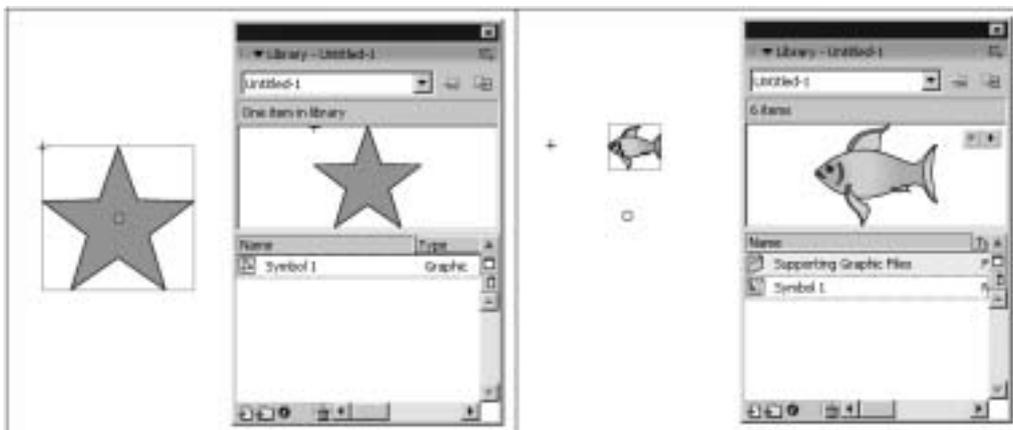


Рис. 7.10. Пример замены одного символа другим

Редактируем символы

Операция редактирования (обработки) символов любого документа Flash 8 производится в рабочем окне этого документа, находящегося в открытом состоянии. Для перехода в режим редактирования требуемого символа, относящегося к одному из открытых документов, вы можете воспользоваться четырьмя способами:

Создаем символы

- дважды щелкнуть на значке символа, выбранного на панели рабочей библиотеки данного документа;
- щелкнуть на названии символа в списке символов активного документа, который раскрывается кнопкой, находящейся в правой части панели инструментов Edit Bar (Панель редактирования);
- дважды щелкнуть в рабочем окне активного документа на потомке нужного вам символа;
- выполнить одну из трех команд контекстного меню, раскрываемого правой кнопкой мыши при установке указателя на потомке данного символа: Edit (Редактировать), Edit in Place (Редактировать по месту) или Edit in New Window (Редактировать в новом окне).

Если редактируемый символ состоит из других символов или групповых объектов, то для перехода в режим обработки нужной вам части символа сделайте на этой части требуемое количество двойных щелчков, разгруппировав ее полностью. При этом в левой части панели инструментов Edit Bar появится соответствующее количество надписей с идентификационными значками обрабатываемых частей исходного символа (см. рис. 2.5), означающих принадлежность каждой из них к тому или иному его составному символу или групповому объекту.

На рис. 7.11 показан пример редактирования сложного анимационного символа-клипа Fish Movie Clip, который был скопирован в документ из стандартной библиотеки символов программы Flash 5. Как видите, область градиентной заливки, выделенная для обработки, входит в символ Fish Graphic, являющийся составной частью вложенного символа Fish Animated Graphic.

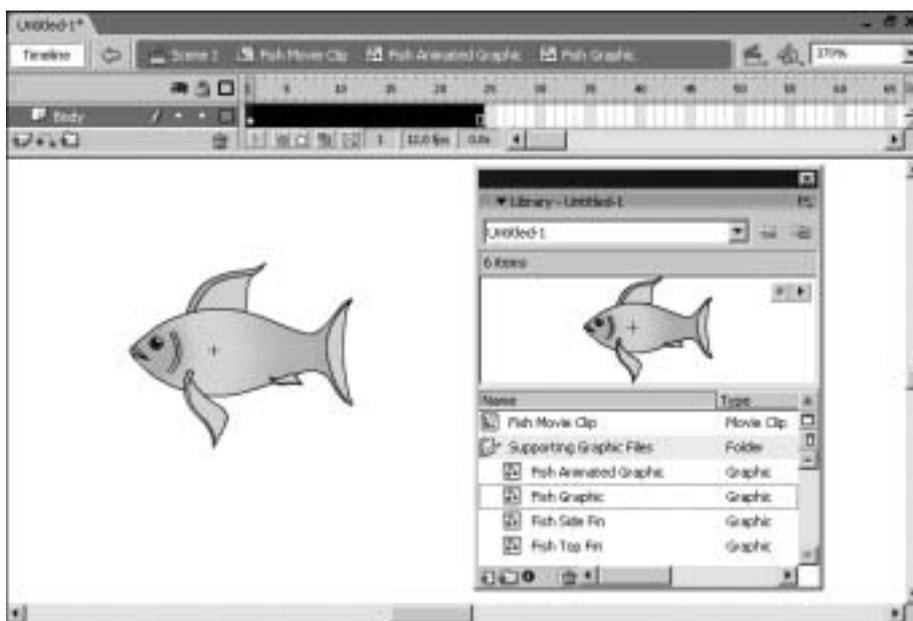


Рис. 7.11. Пример редактирования анимационного символа

Обрабатываем символьные объекты

Любой символьный объект документа, являющийся потомком некоторого символа, может быть обработан без воздействия на сам символ. Перечислим операции обработки символьных объектов, предусмотренные в программе Flash 8, указав при этом используемые элементы настройки инспектора свойств:



Если при ссылке на некоторый элемент настройки инспектора свойств не указывается вкладка, где этот элемент находится, то это означает, что данная вкладка является основной.

- перемещение и трансформация выбранного символьного объекта (см. раздел “Перемещаем объекты” и “Трансформируем объекты” главы 8);
- замена символа, образующего данный объект, на любой другой символ документа без изменения параметров объекта — кнопка **Swap** (Поменять) в левой части инспектора;
- изменение текущего поведения символьного объекта на один из трех возможных образцов поведения:
 - графики — пункт **Graphic** в списке **Instance Behavior** (Поведение потомка);
 - кнопки — пункт **Button** в данном списке;
 - встроенного клипа — пункт **Movie Clip** в данном списке;
- для символьного объекта типа кнопки или клипа:
 - присвоение имени объекту — поле **Instance Name** (Имя потомка) в левой части инспектора;
 - режим представления объекта в видеоклипе растровым изображением, что в ряде случаев позволяет ускорить воспроизведение клипа за счет устранения многократной перерисовки на экране статического символьного объекта — флажок **Use runtime bitmap caching** справа внизу инспектора;
- настройка параметров анимационного символьного объекта с поведением графики (см. раздел “Обрабатываем анимационные символы” главы 9);
- создание в символьном объекте следующих возможных эффектов (см. ниже):
 - для любого объекта — одного из четырех цветовых эффектов;
 - для объекта типа кнопки или клипа — до семи эффектов фильтрации, а также одного из предусмотренных во Flash 8 эффектов смешения цветов;
- отмена созданных ранее эффектов.

Создаем цветовые эффекты

Flash 8 позволяет создавать в любых символьных объектах четыре цветовых эффекта (по одному на объект). Выбор этих эффектов производится в списке **Color** (Цвет) основной вкладки инспектора свойств, а настройка их параметров — на той же вкладке. Перечислим эти эффекты, указав используемые элементы настройки вкладки **Properties** (свойства) инспектора свойств:

- *эффект изменения яркости* объекта — пункт **Brightness** (Яркость) списка **Color**; ползунок **Brightness Amount**, задающий уровень яркости (в процентах);
- *эффект нанесения оттенка* на объект — пункт **Tint** (Оттенок) списка **Color**; раскрывающаяся цветовая панель и три ползунка **RGB**, используемые для выбора дополнительного цвета; ползунок **Tint Amount**, регулирующий оттенок этого цвета (в процентах);
- *эффект увеличения прозрачности* объекта — пункт **Alpha** (Альфа) списка **Color**; ползунок **Alpha Amount**, регулирующий уровень непрозрачности (в процентах);
- *эффект изменения раскраски и прозрачности* объекта — пункт **Advanced** (Расширенный) списка **Color**; кнопка **Setting** (Настройка), открывающая диалоговое окно **Advanced Effect** (Расширенный эффект) для настройки параметров эффекта;

Для отмены существующего цветового эффекта следует выбрать в списке **Color** пункт **None** (Отсутствует).



Параметры созданного в объекте цветового эффекта могут быть анимированы с помощью раскдровки движения (см. раздел “Используем раскдровку движения” главы 9).

На рис. 7.12 представлены два потомка непрозрачного символа графического типа Clock Pencil, которым было присвоено поведение клипа. В правом из этих потомков был создан эффект увеличения прозрачности с уровнем непрозрачности в 50%, что позволило увидеть сквозь него градиентное изображение, расположенное в нижнем слое документа.

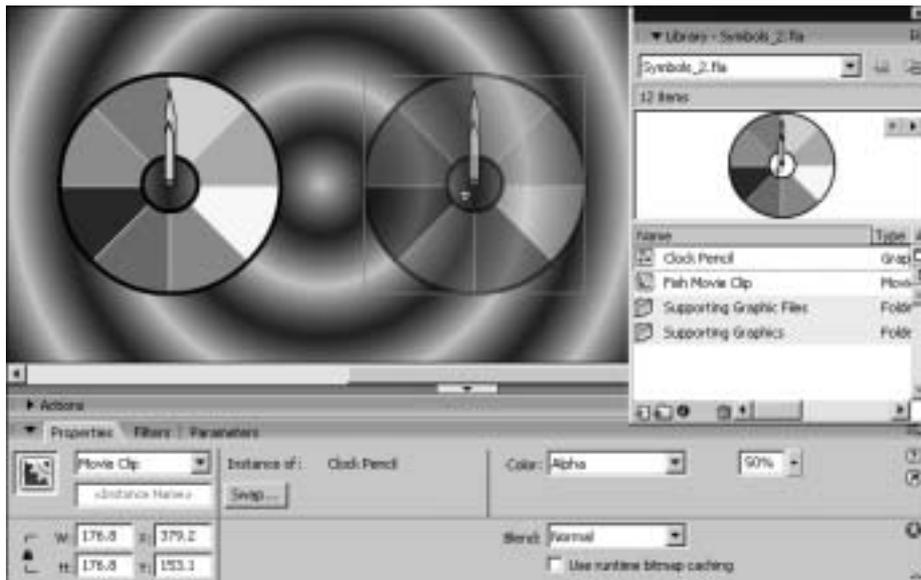


Рис. 7.12. Пример создания в символьном объекте эффекта увеличения прозрачности

На рис. 7.13 рассмотрен пример замены образующего символа для созданного ранее его полупрозрачного потомка (с эффектом увеличения прозрачности). Как вы видите, параметры прозрачности потомка не изменились после замены образующего символа.

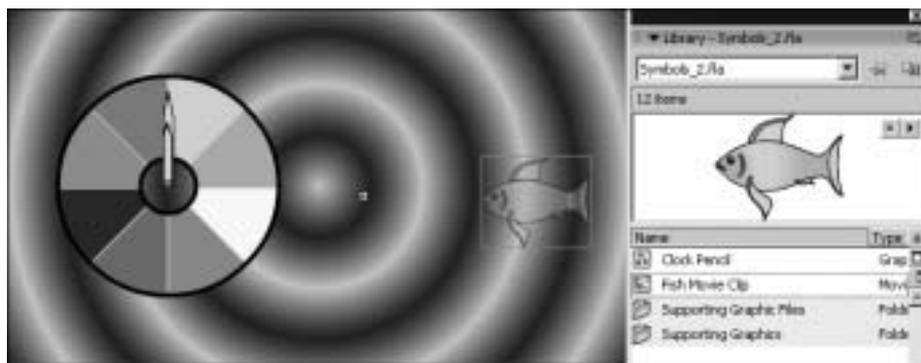


Рис. 7.13. Пример замены образующего символа для символьного объекта

Особенности эффекта изменения расцветки и прозрачности

Суть данного эффекта заключается в следующем. Для каждого пикселя изображения символического объекта вычисляются новые значения его яркостей в трех цветовых каналах RGB-типа, а также уровня непрозрачности в зависимости от исходных значений этих параметров и тех управляющих величин, которые задаются в диалоговом окне **Advanced Effect** (Расширенный эффект), раскрываемом из инспектора свойств.

Расчет этих значений производится программой по следующим формулам:

$$Red = X_1/100 \times R + Y_1;$$

$$Green = X_2/100 \times G + Y_2;$$

$$Blue = X_3/100 \times B + Y_3;$$

$$Alpha = X_4/100 \times A + Y_4.$$

Здесь: R , G и B — исходные значения канальных яркостей пикселя в красном, зеленом и синем каналах соответственно; Red , $Green$ и $Blue$ — конечные значения яркостей в этих каналах; A — исходное значение уровня непрозрачности данного пикселя, задаваемое в абсолютных единицах (в диапазоне от 0 до 255); $Alpha$ — конечное значение уровня непрозрачности; X_1 , Y_1 , X_2 , Y_2 , X_3 , Y_3 , X_4 и Y_4 — значения параметров, задаваемые в окне **Advanced Effect** с помощью ползунков и полей (см. рис. 7.14).

Поскольку конечные значения параметров пикселей вычисляются путем умножения их исходных значений на одни величины с последующим добавлением других величин (все эти величины регулируются пользователем и могут быть как положительными, так и отрицательными), обеспечивается широкий диапазон изменения расцветки и прозрачности пикселей обрабатываемого объекта. В частности, вы можете сделать полностью прозрачными или непрозрачными частично прозрачные пиксели исходного объекта или создать негативное изображение объекта (см. ниже).



Если вы задаете параметры эффекта путем ввода их значений в поля окна **Advanced Effect**, а не регулировкой соответствующих ползунков, то, чтобы проконтролировать действие текущих параметров на обрабатываемый объект без закрытия данного окна, вам следует щелкнуть мышью в каком-либо другом поле по отношению к тому полю, в котором вводился последний параметр.

На рис. 7.14 показан пример создания в символическом объекте эффекта изменения расцветки и прозрачности. Здесь изображено: слева — объект без эффекта; в центре — объект с эффектом (он выделен); справа — окно **Advanced Color** с параметрами данного эффекта; внизу — инспектор свойств. Образующий символ, потомки которого представлены на рисунке, состоит из кружков (красный, зеленый и синий), каждый из которых разделен пополам, и одна из этих половинок (более светлая) была сделана частично прозрачной с уровнем непрозрачности 30%. Как видите, у объекта с эффектом непрозрачные половинки кружков остались почти без изменения, а частично прозрачные стали полностью прозрачными.

Создаем эффекты фильтрации

Новой функцией программы Flash 8 является создание семи эффектов фильтрации в символических объектах типа кнопок и клипов, а также в текстовых объектах документа.



Эффекты фильтрации Flash 8 не имеют ничего общего с эффектами фильтрации, создаваемыми в растровых изображениях. По своему действию они очень напоминают эффекты слоев, которые можно создавать в объектах растровых документов с помощью графических программ Photoshop CS2 и ImageReady CS2 (см. раздел "Подготавливаем изображения в других программах" главы 5, подраздел "Photoshop CS2").

Все операции с эффектами фильтрации, относящимися к выбранному объекту документа одного из указанных выше типов, выполняются на вкладке **Filters** (Фильтры) инспектора свойств. Перечислим эти операции, указав при этом используемые средства данной вкладки инспектора:

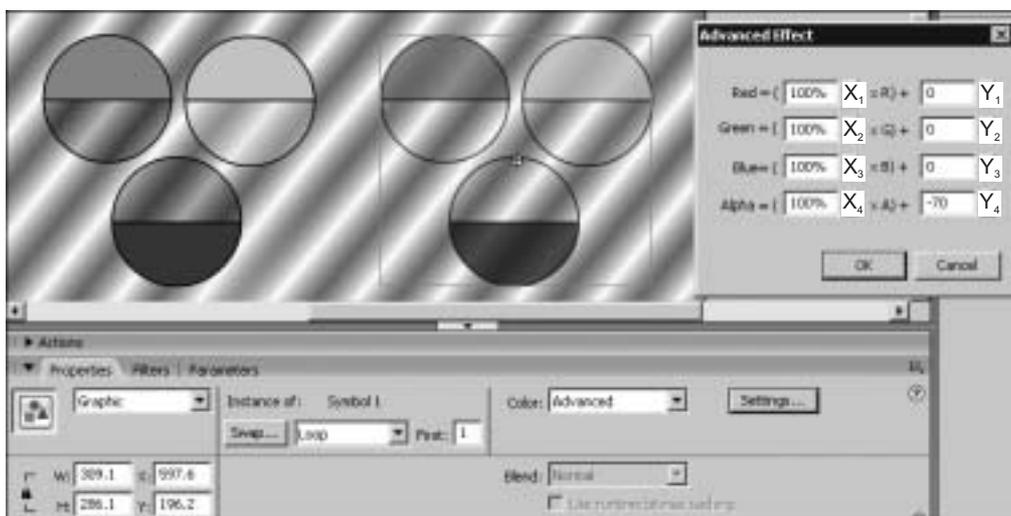


Рис. 7.14. Пример создания в символьном объекте эффекта изменения расцветки и прозрачности

- создание в объекте до семи эффектов фильтрации, выбор которых производится в контекстном меню, раскрываемом кнопкой со знаком “+”, находящейся в левом верхнем углу вкладки. Название выбранного вами эффекта помещается в список созданных эффектов, расположенный под этой кнопкой. Эффекты представлены в меню следующими пунктами:
 - Drop Shadow — эффект тени (см. рис. 7.15);
 - Blur — эффект размытия (см. рис. 7.16);
 - Glow — эффект свечения (см. рис. 7.17);
 - Bevel — эффект скоса (см. рис. 7.18);
 - Gradient Glow — эффект градиентного свечения (см. рис. 7.19);
 - Gradient Bevel — эффект градиентного скоса (см. рис. 7.20);
 - Adjust Color — эффект регулировки цвета (см. рис. 7.21);
- управление подключением созданных эффектов щелчками мыши перед их названиями (зеленая метка в виде галочки означает подключенный эффект, а красная метка в виде перекрестия — отключенный);
- изменение порядка применения к объекту заданных эффектов путем перетаскивания мышью их названий в списке;
- выделение некоторого эффекта щелчком мыши с целью его удаления с помощью кнопки со знаком “-”, расположенной справа от кнопки “+”;
- выделение подключенного эффекта (с зеленой галочкой), чтобы настроить его параметры с помощью элементов вкладки, которые при этом появляются.

Мы не будем здесь описывать элементы настройки параметров эффектов фильтрации, поскольку вы в них легко сможете разобраться в процессе создания того или иного эффекта.



Параметры созданных в объекте эффектов фильтрации могут быть анимированы с помощью раскадровки движения (см. раздел “Используем раскадровку движения” главы 9).

На рис. 7.15–7.21 показаны примеры создания всех семи эффектов фильтрации (по одному в каждом случае) в символьном объекте круглой формы с шаблонной заливкой. На каждом из этих рисунков изображено: слева — исходный объект без эффекта; справа — объект с эффектом; внизу — вкладка **Filters** инспектора свойств с параметрами созданного эффекта. На некоторых рисунках оба объекта изображены на темном фоне, чтобы сделать более выразительным тот или иной эффект.

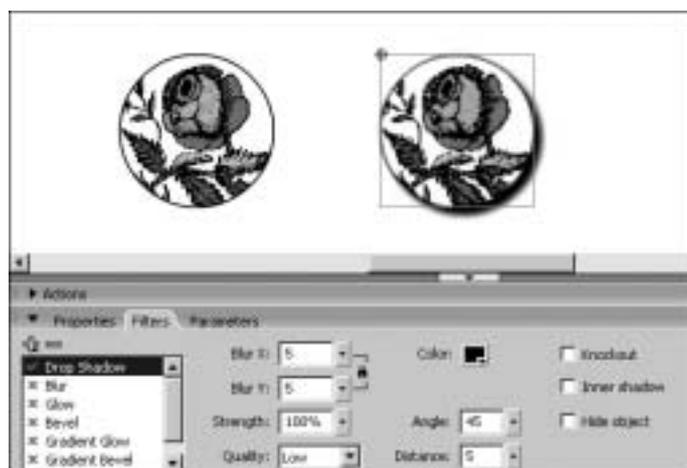


Рис. 7.15. Пример создания эффекта тени

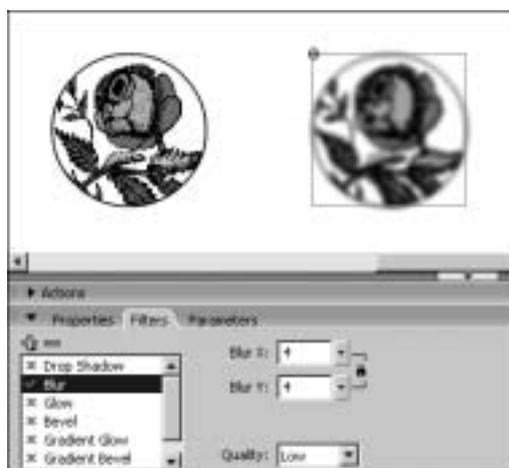


Рис. 7.16. Пример создания эффекта размытия

Следует отметить, что наибольшую выразительность в оформлении объектов с помощью эффектов фильтрации можно достичь при создании в выбранном объекте не одного, а нескольких эффектов. Чтобы сформированную вами удачную комбинацию эффектов можно было многократно использовать при оформлении различных объектов документа, во Flash 8 предусмотрена возможность сохранения таких комбинаций, называемых *предустановками*, под заданными именами.

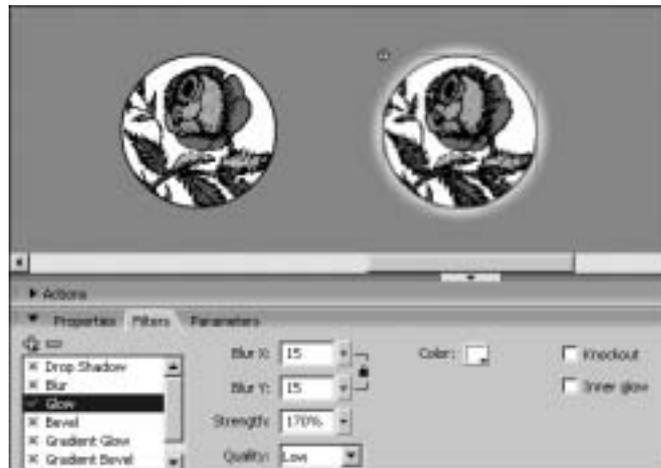


Рис. 7.17. Пример создания эффекта свечения

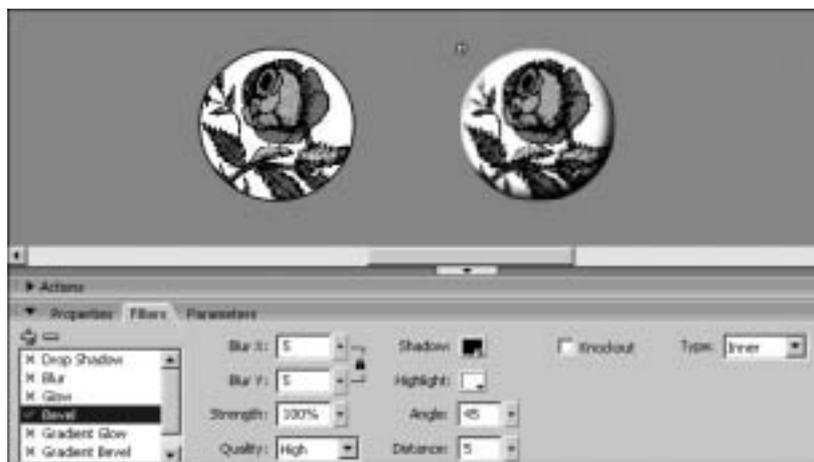


Рис. 7.18. Пример создания эффекта скоса



В программах растровой графики Photoshop CS2 и ImageReady CS2 также предусмотрено многократное использование заданных комбинаций эффектов слоев, являющихся аналогами эффектов фильтрации Flash 8. Только они там называются не предустановками, а стилями.

Операция сохранения созданной предустановки выполняется с помощью команды **Save As** (Сохранить как) подменю **Presets** (Предустановки) контекстного меню вкладки **Filters** инспектора. В ее диалоговом окне **Save Preset As** (Сохранить предустановку как) введите имя предустановки, в качестве которого рекомендуется использовать названия входящих в ее состав эффектов фильтрации, после чего щелкните на кнопке **ОК**.

Для применения к некоторому объекту документа нужной вам предустановки достаточно выполнить команду с ее названием, которая будет находиться внизу подменю **Presets** (Предустановки) меню вкладки **Filters**.

На рис. 7.22 изображено окно **Save Preset As** в момент присвоения имени **Shadow_Bevel** новой предустановке, объединяющей два эффекта фильтрации: тени (**shadow**) и скоса (**bevel**).



Рис. 7.19. Пример создания эффекта градиентного свечения

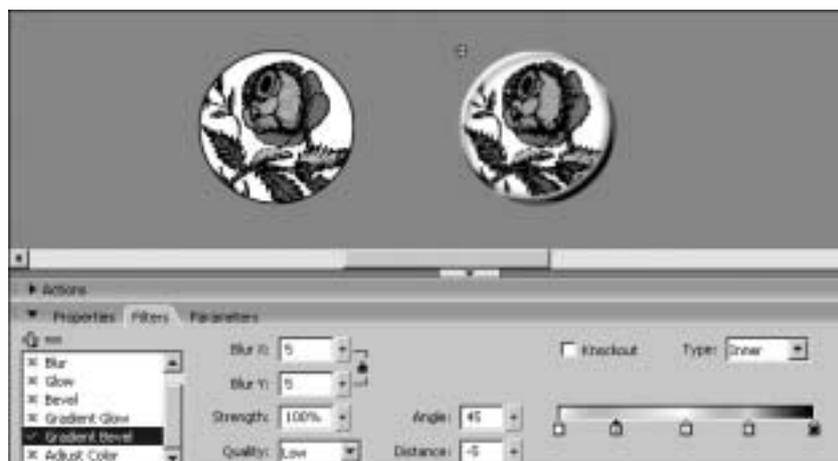


Рис. 7.20. Пример создания эффекта градиентного скоса

На рис. 7.23 изображены два символьных объекта разных форм, которые были оформлены указанной выше предустановкой.

Создаем эффекты смешения цветов

Еще одной новой функцией Flash 8 является создание 13 эффектов смешения цветов, которые применяются к символьным объектам с поведением кнопок и клипов. Суть каждого из этих эффектов состоит в том алгоритме, по которому происходит смешение цветов (а иногда и уровней непрозрачности) соответствующих пикселей изображения символьного объекта с данным эффектом и нижележащих изображений документа.

Выбор эффектов смешения цветов производится в раскрывающемся списке Blend (Смешение) основной вкладки инспектора свойств. Перечислим эти эффекты, указав их названия в данном списке:

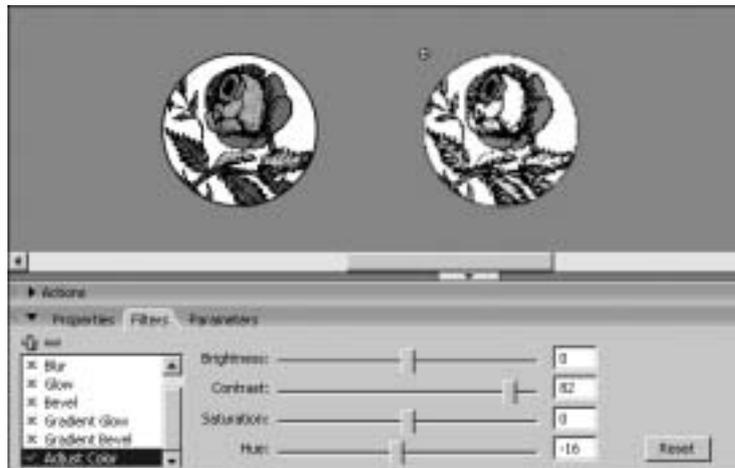


Рис. 7.21. Пример создания эффекта регулировки цвета



Рис. 7.22. Вид окна Save Preset As в процессе работы

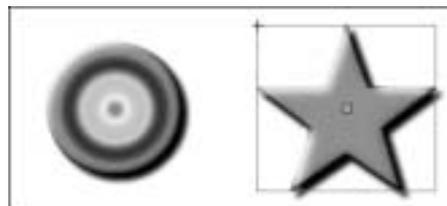


Рис. 7.23. Пример оформления объектов предустановкой Shadow_Bevel

- **Normal** — эффект отсутствует (обычный режим смешения цветов, заданный по умолчанию, когда отображается только наносимое изображение, если оно непрозрачно, либо смесь наносимого и фонового изображений в противном случае);
- **Layer** — эффект слоя, который применяется к потомкам символов, состоящих из двух или более вложенных перекрывающихся символов с эффектами смешения. Он состоит в ограничении области воздействия эффектов смешения, созданных во вложенных символах, лишь на область их перекрытия (см. рис. 7.24);
- **Darken** — эффект затемнения, состоящий в следующем: пикселю результирующего изображения присваиваются меньшие из значений канальных яркостей двух соответствующих пикселей наносимого и фонового изображений;
- **Multiply** — эффект умножения, состоящий в следующем: цветовые параметры пикселей результирующего изображения вычисляются путем перемножения значений канальных яркостей соответствующих пикселей наносимого и фонового изображений и последующего деления на 255 (конечное изображение всегда будет более темным, чем исходное);

- **Lighten** — *эффект осветления*, состоящий в следующем: из двух значений канальных яркостей пикселей наносимого и фонового изображений присваивается большее значение;
- **Screen** — *эффект экранирования*, состоящий в следующем: для пикселя результирующего изображения цвета вычисляются следующим образом: вначале перемножаются инвертированные (дополнительные) значения канальных яркостей соответствующих пикселей наносимого и фонового изображений, после чего производится инвертирование полученных величин (суммарное изображение становится более светлым);
- **Overlay** — *эффект перекрытия*, который представляет собой следующую комбинацию эффектов *Multiply* и *Screen*: если значение канальной яркости пикселя фонового изображения не превышает 128, то используется первый режим, в противном случае — второй;
- **Hard Light** — *эффект жесткого света*, состоящий в следующем: имитируется подсветка фонового изображения источником света, лучи которого раскрашены цветами наносимого изображения (фоновое изображение значительно осветляется в светлых участках наносимого изображения и затемняется в темных);
- **Add** — *эффект добавления*, состоящий в следующем: величины яркости соответствующих пикселей наносимого и фонового изображений суммируются (результатирующее изображение становится более светлым, чем любое из исходных);
- **Subtract** — *эффект вычитания*, состоящий в следующем: значения яркости соответствующих пикселей наносимого и фонового изображений суммируются, после чего из полученной величины вычитается число 255 (конечное изображение становится более темным, чем любое из исходных);
- **Difference** — *эффект разницы*, состоящий в следующем: величина яркости пикселя результирующего изображения вычисляется как разность между большим и меньшим значениями соответствующих пикселей наносимого и фонового изображений;
- **Invert** — *эффект инверсии*, состоящий в следующем: происходит инвертирование того фрагмента нижележащего изображения, который находится под данным объектом;
- **Alpha** — *эффект прозрачности*, который применяется к самому верхнему из вложенных символов, входящих в состав образующего символа для объекта с эффектом *Layer*. Эффект состоит в следующем: участки других вложенных символов, которые находятся под верхним символом с эффектом *Alpha*, становятся частично прозрачными с уровнем непрозрачности, заданным для верхнего символа путем создания в нем дополнительного цветового эффекта увеличения прозрачности (пункт *Alpha* списка *Color* и ползунок *Alpha Amount* в инспекторе свойств);
- **Eraser** — *эффект стирания*, который применяется к самому верхнему из вложенных символов, входящих в состав образующего символа для объекта с эффектом *Layer*. Эффект состоит в следующем: те участки других вложенных символов, которые находятся под верхним символом с эффектом *Eraser*, становятся полностью прозрачными.

На рис. 7.24 рассмотрен первый пример совместного использования эффектов смешения цветов. Здесь представлены два вида документа, содержащего два потомка-клипа символа **Symbol 3**, состоящего из двух других перекрывающихся символов квадратной формы: **Symbol 1** и **Symbol 2** (верхний). Для левого из этих потомков эффект смешения цветов отсутствует (пункт *Normal* в списке *Blend* инспектора), а для правого, выделенного, создан эффект слоя (пункт *Layer* в данном списке). Для верхнего вложенного символа **Symbol 2** был задан эффект инверсии (пункт *Invert* в списке *Blend*). В верхней части рисунка документ показан в исходном состоянии, а в нижней части — в режиме редактирования символа **Symbol 3** (с выделением его верхнего вложенного символа). Как показано в верхней половине рисунка, в левом потомке символа **Symbol 3** происходит инвертирование того участка нижележащего изображения, который расположен под всем его верхним вложенным символом с эффектом инверсии, а в правом потомке — только под областью перекрытия двух вложенных символов.

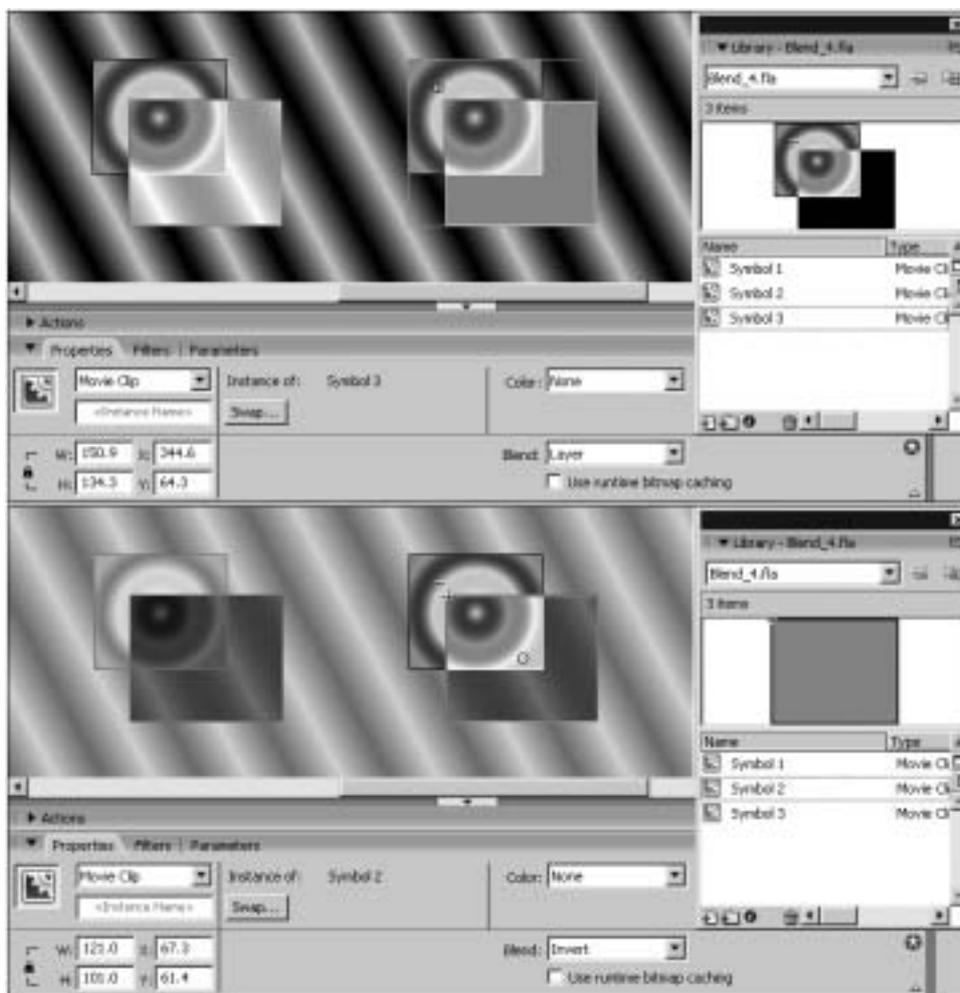


Рис. 7.24. Пример совместного использования эффектов смешения цветов *Layer* и *Invert*

На рис. 7.25 приведен второй пример совместного использования эффектов смешения цветов. Он отличается от предыдущего примера лишь тем, что в верхнем вложенном символе *Symbol 2* был создан не эффект инверсии, как на рис. 7.24, а следующие два эффекта: цветовой эффект увеличения прозрачности (с уровнем непрозрачности в 50%) и эффект прозрачности, являющийся эффектом смешения цветов. Как видите, в левом потомке символа *Symbol 3*, не имеющем эффекта смешения цветов *Layer*, вложенный символ *Symbol 2* стал полностью прозрачным, а в правом потомке с таким эффектом стала частично прозрачной область перекрытия двух вложенных символов, что и следовало ожидать.

Создаем негативное изображение объекта

В ряде случаев вам может понадобиться представить в негативе изображение некоторого объекта документа. Данная операция может быть выполнена во Flash 8 следующими двумя способами, для чего исходный объект должен быть предварительно преобразован в потомок символа (если он таким не является):

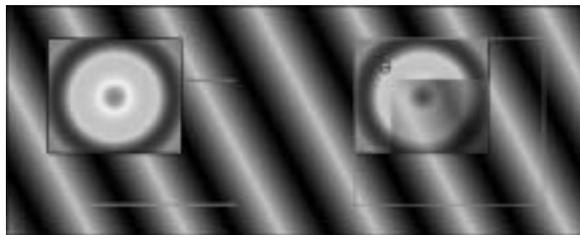


Рис. 7.25. Пример совместного использования эффектов смешения цветов Layer и Alpha

- примените к исходному символьному объекту цветовой эффект изменения расцветки и прозрачности (пункт **Advanced** в списке **Color** инспектора свойств) с заданными значениями его параметров (см. рис. 7.26);
- создайте еще один потомок данного символа, присвойте ему поведение клипа, задайте для него эффект инверсии (пункт **Invert** в списке **Blend** инспектора), а также расположите его поверх исходного потомка (см. рис. 7.27).

Порядок создания негативного изображения первым способом таков.

1. Если обрабатываемый вами объект не является символьным, то сделайте его таким, выполнив для этого команду **Modify**⇒**Convert to Symbol** (**Изменить**⇒**Преобразовать в символ**).
2. Примените к данному объекту эффект изменения расцветки и прозрачности, задав в дополнительном диалоговом окне **Advanced Effect** инспектора свойства параметры, представленные на рис. 7.26. В результате исходные значения яркостей (**R**, **G** и **B**) для каждого пикселя обрабатываемого объекта будут заменены на их дополнительные величины: $(255 - R)$, $(255 - G)$ и $(255 - B)$, соответствующие негативному изображению.

На рис. 7.26 изображено: слева — исходный символьный объект; посередине — тот же объект, но с эффектом негатива, созданным первым способом; справа — окно **Advanced Effect** с параметрами данного эффекта. Образующий символ для этих объектов был сформирован из растрового изображения цветка.

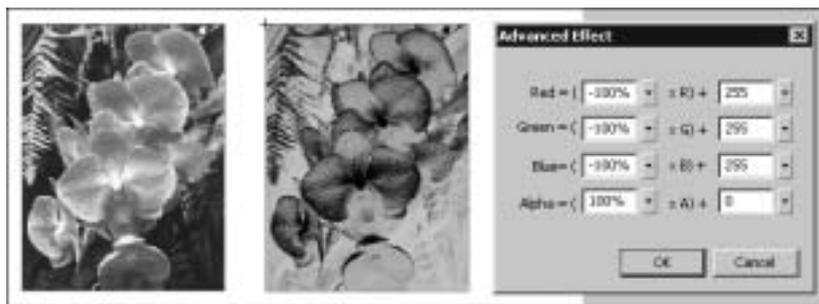


Рис. 7.26. Пример создания первым способом негативного изображения объекта

На рис. 7.27 зафиксирован момент создания вторым способом негативного изображения того же объекта, что и на предыдущем рисунке. Объект без эффекта здесь также находится слева.

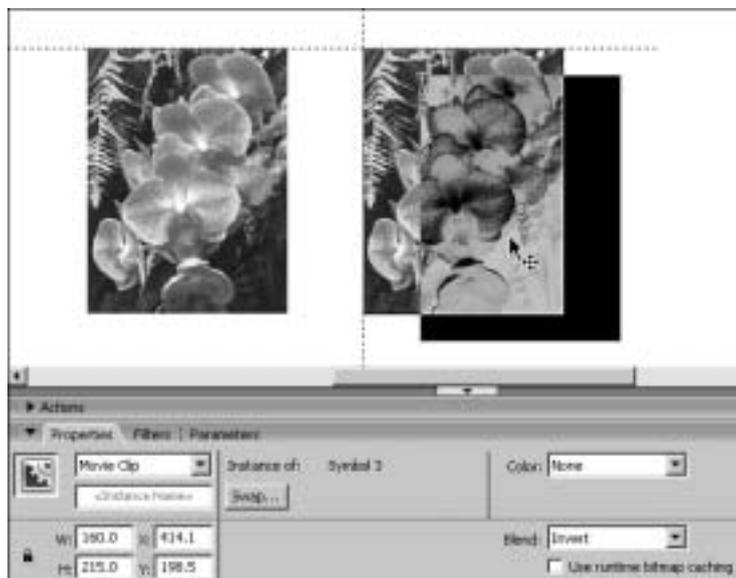


Рис. 7.27. Пример формирования вторыми способом негативного изображения объекта

Подготавливаем символы в Illustrator CS2

Если в разрабатываемом документе Flash 8 вам понадобятся статические символы или целые наборы их предварительно обработанных потомков, то лучше всего их создавать не в рассматриваемой программе, а в программе векторной графики Adobe Illustrator CS2. Ведь эта программа обладает многими преимуществами в отношении обработки символьной информации по сравнению с Flash 8, к числу которых относятся:

- широкие возможности художественной обработки векторной графики, из которой обычно образуются символы;
- наличие развитых средств создания и обработки символов, их отдельных потомков и их целых наборов, к числу которых относятся:
 - палитра Symbols (Символы);
 - восемь инструментов, предназначенных для создания и обработки наборов потомков;
 - двадцать пять стандартных библиотек символов (2 основные и 23 дополнительные), предоставляющих широкий их выбор по самым разнообразным тематикам;
- простота переноса символьной информации из Illustrator CS2 во Flash 8 (через файл формата Flash), когда в область документа-приемника загружаются все исходные потомки, а в рабочую библиотеку документа — все исходные символы, породившие эти потомки (в ряде случаев могут также формироваться дополнительные символы и растровые изображения).

Таким образом, в программе Illustrator CS2 вы можете не только создавать нужные вам статические символы, но и формировать и обрабатывать там целые наборы их потомков, которые без изменений будут перенесены в нужный документ Flash 8.

Если вы будете вести подготовку символьной информации в Illustrator CS2, примите во внимание следующие особенности ее переноса в документ Flash 8:

- в файле формата Flash (с расширением `.swf`), через который осуществляется перенос символов и потомков во Flash 8, сохраняется (в режиме экспорта) вся информация исходного документа Illustrator CS2;
- информация, импортированная из документа-источника, будет располагаться в левой верхней части области текущего кадра документа Flash 8;
- для переноса во Flash 8 некоторого символа необходимо в исходном документе Illustrator CS2 сформировать хотя бы один потомок этого символа;
- при обработке в Illustrator CS2 потомков и их наборов следующими шестью инструментами: Symbol Sprayer (Распылитель символов), Symbol Shifter (Регулятор смещения символов), Symbol Scruncher (Регулятор сдавливания символов), Symbol Sizer (Регулятор размеров символов), Symbol Spinner (Регулятор вращения символов) и Symbol Screener (Регулятор экранирования символов), данные символы будут перенесены во Flash 8 без их размножения (см. рис. 7.32);
- в случае обработки в Illustrator CS2 потомков символов инструментами Symbol Stainer (Регулятор цвета символов) и Symbol Styler (Регулятор стиля символов) во Flash 8 произойдет размножение исходных символов за счет формирования дополнительных символов, а для второго инструмента — также и растровых изображений (см. рис. 7.33).



Если та или иная операция, относящаяся к рассматриваемой внешней программе, может также выполняться и непосредственно во Flash 8, то перед ее определением мы будем ставить метку ^(F1).

Перечислим операции с символами и потомками, выполнение которых предусмотрено в Illustrator CS2:

- ^(F1) создание одного потомка символов (см. ниже) — палитра Symbols (Символы) или палитра открытой дополнительной библиотеки символов;
- создание набора потомков (см. ниже) — инструмент Symbol Sprayer (Распылитель символов);
- ^(F1) различные операции обработки отдельных потомков, к числу которых относятся:
 - перемещение потомка по области документа — инструмент Selection (Выделение);
 - трансформация потомка;
 - оформление потомка заданным стилем;
- различные операции обработки набором потомков, к числу которых относятся:
 - перемещение потомков из некоторого набора в заданном направлении — инструмент Symbol Shifter (Регулятор смещения символов);
 - сведение потомков набора в выбранную точку документа или их разведение в разные стороны — инструмент Symbol Scruncher (Регулятор сдавливания символов);
 - изменение размеров потомков набора — инструмент Symbol Sizer (Регулятор размеров символов);
 - изменение угла ориентации потомков набора — инструмент Symbol Spinner (Регулятор вращения символов);
 - изменение раскраски потомков набора путем смешения их цветов с текущим цветом заливки — инструмент Symbol Stainer (Регулятор цвета символов);
 - регулировка прозрачности потомков набора — инструмент Symbol Screener (Регулятор экранирования символов);
 - дополнительное оформление потомков набора — выбранным стилем (инструмент Symbol Styler (Регулятор стиля символов));



Для каждого из последних семи инструментов предусмотрено использование клавиши <Alt>, изменяющей на противоположное направление воздействия инструмента на потомки обрабатываемого набора.

- ^(F1) создание нового символа (см. ниже) — кнопка **New Symbol** (Новый символ) палитры **Symbols**;
- ^(F1) редактирование символа (см. ниже) — палитра **Symbols**.

Опишем вкратце операции по созданию символов и их потомков в Illustrator CS2.

Создание потомка символа. Открывается (при необходимости) требуемая дополнительная библиотека символов, хранящаяся в файле, для чего используется команда с ее названием из меню **Window**⇒**Symbol Libraries** (Окно⇒Библиотеки символов). В палитре **Symbols** или в палитре открытой дополнительной библиотеки выделяется щелчком мыши значок нужного символа, после чего он перетаскивается мышью в нужное место документа, где и произойдет формирование потомка выбранного символа.

Создание набора потомков. Открываются требуемые библиотеки символов. В документе формируется потомок первого по порядку символа из тех, которые будут использоваться. Выбирается инструмент **Symbol Sprayer** и двойным щелчком на его кнопке открывается диалоговое окно **Symbolism Tools Options** (Параметры инструментов символизма), в котором задается диаметр круглой области формирования новых потомков (ползунок **Diameter**), после чего окно закрывается щелчком на кнопке **OK**. Затем указатель помещается рядом с исходным потомком, нажимается кнопка мыши и путем перетаскивания его по заданной траектории образуется требуемое количество потомков выбранного символа. Для включения в создаваемый набор потомков и представителей других символов, для каждого из них выполняются следующие действия: выделите значок символа, нажмите кнопку мыши в нужном месте области расположения текущего набора потомков и, перетаскивая указатель, сформируйте требуемое количество потомков данного символа.

Создание нового символа. В документе создается и выделяется один или несколько объектов произвольных типов, из которых будет образован новый символ. Затем в палитре **Symbols** выполняется щелчок на кнопке **New Symbol** (она вторая справа), в результате чего и будет сформирован новый символ. После этого исходные объекты удаляются из документа (клавиша).

На рис. 7.28 показан пример формирования в Illustrator CS2 отдельных потомков трех разных символов. Здесь представлены те средства программы, которые обычно используются при работе с символьной информацией: слева — открытая дополнительная панель блока инструментов с восемью инструментами для работы с символами и справа — палитра **Symbols**, содержащая основную библиотеку символов.

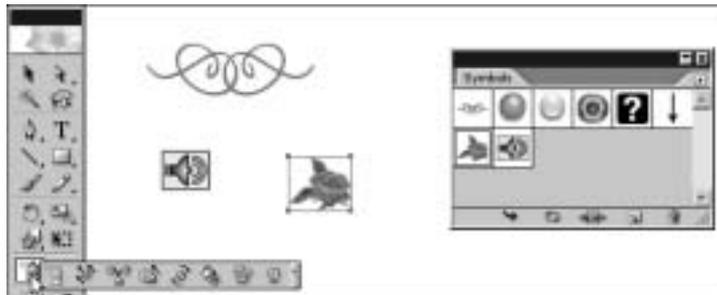


Рис. 7.28. Пример формирования в Illustrator CS2 отдельных символов

На рис. 7.29 изображены палитры четырех интересных дополнительных библиотек символов Illustrator CS2. Знакомство с содержимым этих библиотек, а также основной библиотеки, находящейся в палитре Symbols, облегчит вам выбор нужного символа.

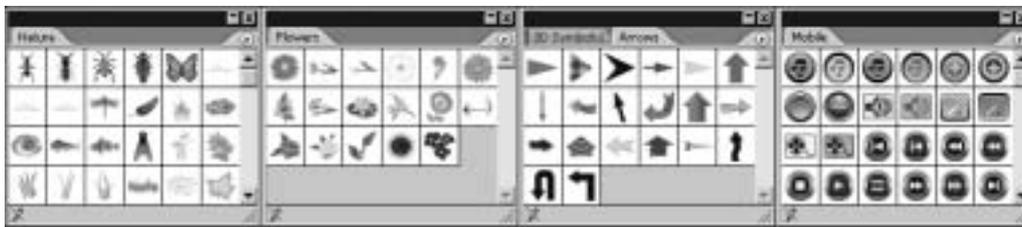


Рис. 7.29. Дополнительные библиотеки символов Illustrator CS2

На рис. 7.30 приведен пример формирования инструментом Symbol Sprayer набора потомков некоторого символа. Слева здесь зафиксирован момент формирования данного набора (при нажатой кнопке мыши), а справа — его окончательный вид (после отпущения кнопки мыши).

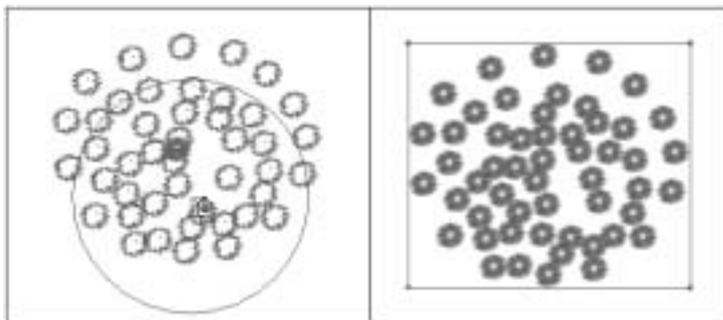


Рис. 7.30. Пример формирования в Illustrator CS2 набора потомков символов

Перенос во Flash 8 потомков вместе с символами

Порядок выполнения данной операции состоит в следующем.

1. Создайте в документе Illustrator CS2 нужные вам потомки или их наборы, обработав их должным образом.
2. Удалите из документа всю лишнюю информацию.
3. Сформируйте файл формата Flash, сохранив в нем содержимое документа. Для этого выберите команду **File⇒Export** (Файл⇒Экспорт) и в раскрывшемся диалоговом окне **Export** выполните следующие действия: выберите папку на диске (рабочая область окна, список **Папка** и кнопка справа от него), задайте в списке **Тип файла** пункт **Macromedia Flash (*.SWF)**, введите название файла (поле **Имя файла**) и щелкните на кнопке **Сохранить**.
4. Перейдите в программу Flash 8, откройте там нужный вам документ и активизируйте в нем требуемый ключевой кадр.
5. Выберите команду **File⇒Import⇒Import to Stage** (Файл⇒Импорт⇒Импорт в область кадра) Flash 8, импортировав в данный документ информацию из файла формата Flash, сформированного в Illustrator CS2. При этом произойдет следующее. В область текущего кадра активного документа загрузятся все потомки символов исходного документа Illustrator CS2, которые расположатся в левой верхней части текущего кадра.

При этом в рабочую библиотеку документа будут загружены сами эти символы, а также, возможно, дополнительные символы и растровые изображения, которые были образованы в процессе импорта информации.

Рассмотрим четыре примера переноса символов и потомков, подготовленных в программе Illustrator CS2, в документ Flash 8.

На рис. 7.31 изображено рабочее окно документа Flash 8 (с панелью рабочей библиотеки), в который была импортирована символьная информация через файл формата Flash из документа Illustrator CS2, представленного на рис. 7.28. В данном случае все потомки трех символов, а также сами эти символы были перенесены без изменений, при этом никакого размножения символов не произошло.

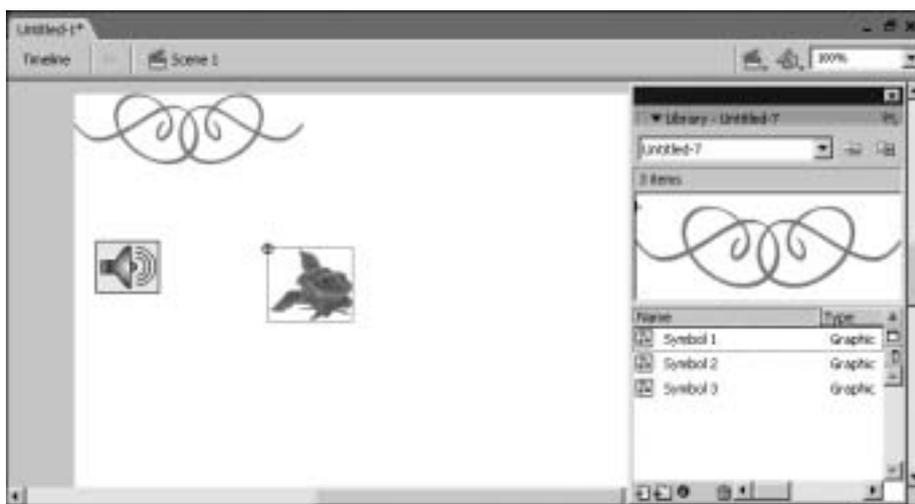


Рис. 7.31. Первый пример переноса символьной информации из Illustrator CS2 во Flash 8

На рис. 7.32 рассмотрен пример переноса во Flash 8 набора потомков, который был сформирован в Illustrator CS2 инструментом Symbol Sprayer и обработан пятью инструментами: Symbol Shifter, Symbol Scruncher, Symbol Sizer, Symbol Spinner и Symbol Screener. В этом случае также не произошло размножения исходного символа.

На рис. 7.33 приведен пример переноса в рассматриваемую программу набора потомков символа, который был сформирован в Illustrator CS2 и обработан указанными выше пятью инструментами, а также инструментом Symbol Stainer (Регулятор цвета символов). Как видите, в связи с использованием указанного инструмента произошло размножение исходного символа.

Вопросы для самопроверки

1. Что представляет собой символ, и какие три типа символов используются во Flash 8?
2. Что такое потомки символов, и каково их альтернативное название?
3. Какими свойствами обладают потомки символов?
4. В каких случаях рекомендуется использовать во Flash 8 символы и потомки?
5. Какие существуют способы создания новых символов?
6. Как можно перейти в режим редактирования выбранного символа, а также как происходит выход из этого режима?

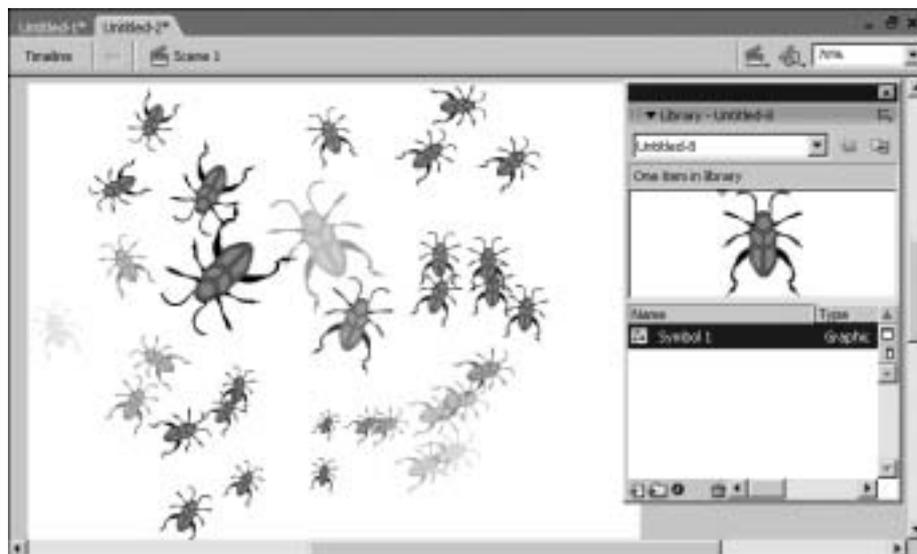


Рис. 7.32. Второй пример переноса символьной информации из Illustrator CS2 во Flash 8

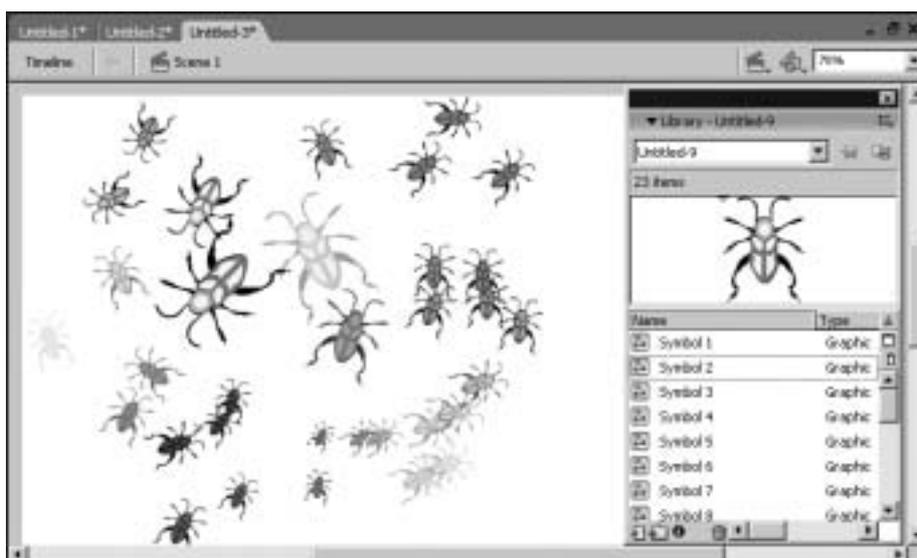


Рис. 7.33. Третий пример переноса символьной информации из Illustrator CS2 во Flash 8

7. Какие операции допускается применять во Flash 8 по обработке символьных объектов?
8. Какие цветовые эффекты могут создаваться в символьных объектах и с помощью каких средств программы?
9. Что такое эффекты фильтрации Flash 8, и в каких объектах их можно создавать?
10. Сколько всего эффектов фильтрации предусмотрено в данной программе, и как происходит с ними работа?

11. Что такое эффекты смешения цветов, сколько их, в каких объектах они создаются и с помощью каких средств программы?
12. Какими двумя способами можно создать негативное изображение объекта?
13. В каких случаях рекомендуется использовать программу векторной графики Illustrator CS2 для подготовки символов, которые будут использоваться в разрабатываемом вами документе Flash 8?
14. Какими основными возможностями обладает внешняя программа Illustrator CS2 по обработке символов и потомков?
15. Каким образом осуществляется перенос символов и потомков из Illustrator CS2 во Flash 8?