

Штенников Дмитрий Геннадьевич

MACROMEDIA FLASH. ОСНОВЫ ПРОГРАММИРОВАНИЯ. ПРОСТЕЙШИЕ СКРИПТЫ. УРОК 1

От редакции:

В № 5 журнала «Компьютерные инструменты в образовании» за 2003 год была опубликована вводная статья Д.Г. Штенникова по технологиям Macromedia Flash. В этом году журнал публикует продолжение – серию уроков по обучению работе с Macromedia Flash, подготовленную на основе курса, читаемого автором статьи в Санкт-Петербургском центре Федерации Интернет Образования.

ActionScript: событийно-управляемый язык, встроенный во Flash. Если в четвертой версии это был ограниченный набор команд, позволяющий осуществлять лишь основные действия и вводимый с помощью не очень удобного интерфейса, то новый ActionScript во Flash MX 2004 – это мощный язык с увеличенным набором команд, поддержкой всевозможных преимуществ объектно-ориентированного программирования: классов, наследования.

ActionScript делает ваши анимации и страницы интерактивными. Вы можете создавать реакции на события мышки или клавиатуры, можете выполнить какие-либо действия при проигрывании определенного кадра. Цель этой статьи – изложить основы ActionScript и показать, что этот язык может служить как для создания весь-



Вы можете создавать реакции на события мышки или клавиатуры...

ма внушительных программ, так и для выполнения элементарных действий, которые сделают вашу вебстраницу гораздо привлекательней.

ТЕРМИНЫ

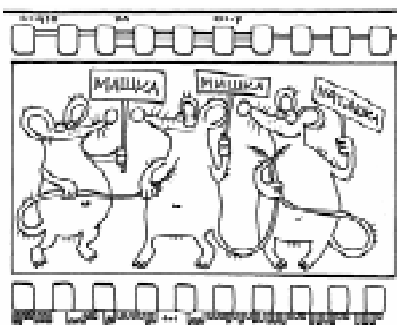
Вначале рассмотрим несколько терминов из области ActionScript:

Действия (Actions) – это инструкции, которые говорят Flash – ролику, что делать. От них произошло название языка – ActionScript (сценарий действий).

События (Events) – это действия, которые происходят, когда проигрывается мультимедиа. События, например, могут происходить, когда заканчивается загрузка какого-то кадра, когда мы достигаем какого-то кадра, когда пользователь нажимает клавишу на клавиатуре или курсор мышки оказывается над нашим объектом.



... класс дерева – растение.



Каждому экземпляру можно присвоить имя...

Выражения (Expressions) – это любая часть инструкции, которая порождает значение. Следующие примеры являются выражениями: $2 + 2$, $2 * 2$, $a + b$, $2 * \pi * r$, $(15 + k) * \text{random}(10)$.

Функции (Functions) – это блоки кода, которые можно многократно использовать. Функциям можно передавать значения и получать от них результат. Например, `number = get_color(15, 24)`. 15 и 24 являются аргументами (или параметрами) функции `get_color`, возвращаемое значение которой записывается в переменную `number`.

Классы (Classes) – это типы объектов. Например, класс дерева – растение. Во Flash есть некоторое количество predefined классов (очень похожих на классы JavaScript). Вы можете создавать свои классы или модифицировать существующие.

Экземпляры (Instances) – это буквально экземпляры определенных классов. Экземпляр – это уже конкретный реальный объект. Если класс – это определение объекта (экземпляра), то экземпляр – это уже конкретное воплощение, это класс в действии. Каждому экземпляру можно присвоить имя, чтобы через него обращаться к функциям или переменным объекта.

Обработчики (Handlers) – это специальные инструкции, которые обрабатывают

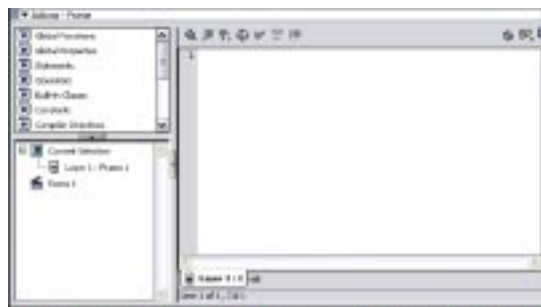


Рисунок 2.

события. Например, `onClipEvent` – обработчик действий, связанных с конкретным символом.

Операторы (Operators) – это элементы языка, которые вычисляют значения, исходя из одного или более аргументов. Например, оператор сложения (+) возвращает сумму двух значений, стоящих слева и справа от него.

Переменные (Variables) – это идентификаторы, которые могут хранить значения. Например, `a = 5`; или `name = "Michael"`.

На рисунке 1 представлен интерфейс программы Macromedia Flash MX 2004.

ПАНЕЛЬ ДЕЙСТВИЙ (ACTIONS)

На рисунке 2 представлена панель Actions. В левой колонке панели видны под-

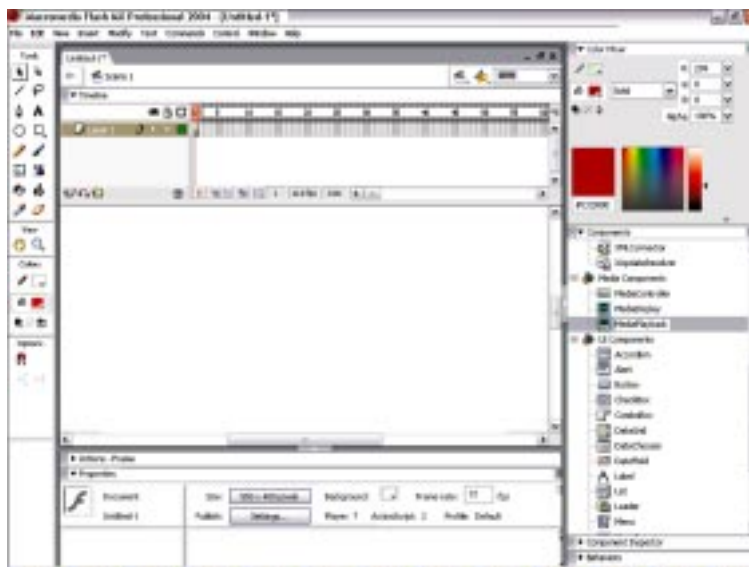


Рисунок 1.

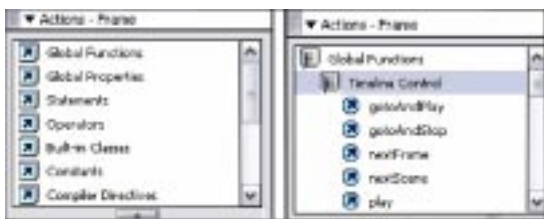





Рисунок 3.


разделы акций и навигационная панель по мультику (рисунок 3).


Если щелкнуть левой клавишей мыши по значку , то раздел раскроется , появятся подразделы и далее сами инструкции (акции). Под списком инструкций находится мини панель навигации по мультику. По этой мини панели легко ориентироваться. Особенно это относится к тому, где именно вы будете писать ваш сценарий (скрипт).


В верхней части панели Actions имеется меню из кнопок:



Кнопка  предназначена для выбора той или иной инструкции, которые разбиты по категориям,

 – для поиска того или иного сочетания букв или цифр внутри скрипта (рисунок 4),

 – для замены символов, при этом, как и в Word, существует возможность замены как в конкретном месте внутри скрипта, так и во всех частях скрипта (рисунок 5),

 – для вставки пути, при этом возможны два варианта: либо относительные ссылки (Relative), либо абсолютные (Absolute). В примере на сцене находится два мультика с именами «a» и «b». В случае относительной адресации адреса кли-

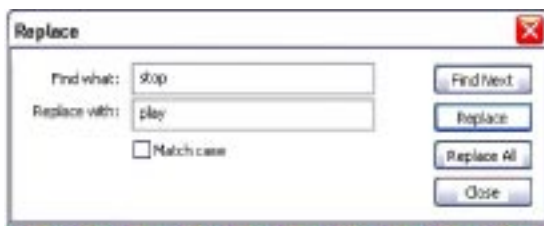




Рисунок 5.





Рисунок 4.


пов будут: a, b; абсолютной адресации `_root.a`, `_root.b` (рисунок 6),


 – для проверки написания синтаксиса (спеллинг),

 – для автоформатирования скрипта, позволяющего легче разобраться в инструкциях и найти ошибки,

 – для режима демонстрации подсказок в процессе написания,

 – система встроенной помощи,

 – выбор опций по отладке скриптов,

 – варианты просмотра.

Панель действий служит для отображения и ввода ActionScript-программ. В некоторых версиях существуют два режима работы с панелью – нормальный (для начинающих свою нелегкую работу с программированием во Flash) и экспертный. В экспертном режиме список команд – это простое поле для ввода текста. В нормальном же режиме мы не можем напрямую редактировать команды. Для этого используется панель параметров.

Добавить инструкцию можно, нажав на кнопку «+» или выбрав соответствующую инструкцию в списке элементов языка. Кроме того, для всех действий во Flash



Рисунок 6.

имеются последовательности клавиш, с помощью которых это можно сделать гораздо быстрее. Они приведены справа от каждого действия в меню кнопки «+». Например, чтобы добавить функцию **stop()**, нужно нажать Esc+st (последовательно: «Esc», затем «s», затем «t»).

Удалить инструкцию можно, просто клавишей Delete.

Часто рекомендуется не начинать сразу же пользоваться экспертным режимом, если у вас нет опыта программирования на С-подобных языках (C++, Java, JavaScript, C#). У нормального режима есть большое достоинство, делающее его незаменимым для новичков – в этом случае гораздо меньше шансов ошибиться с синтаксисом языка. Но некоторые конструкции лучше и легче сразу почувствовать и научиться писать правильно.





Удалить инструкцию можно, просто клавишей Delete.

КНОПКИ (BUTTON)

Первое, что хочется, когда начинаешь изучать интерактивность Flash, – сделать что-нибудь, что бы откликалось на действия пользователя, «оживить» ваше творение, добавить обратную связь. Самый простой способ сделать это – кнопки. Поэтому с них-то и начнем.

Возможно, вы помните, что во Flash существуют символы (Symbol). И среди типов символов существует специальный тип символа для создания кнопок – **Button**.

Чтобы создать кнопку, необходимо произвести следующие действия:

Выбрать инструмент рисования (например, прямоугольник (Rectangle Tool) ) и нарисовать его на сцене. Выбрать инструмент «черная стрелка» (Selection Tool) 



Самый простой способ сделать это – кнопки.

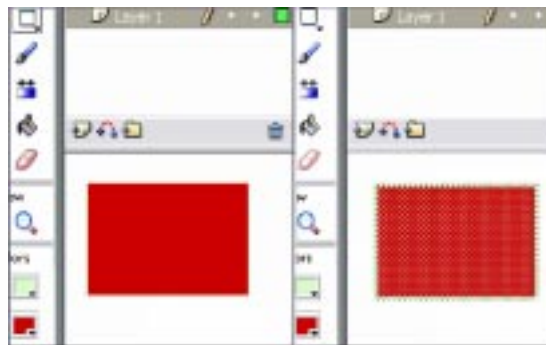


Рисунок 7.

и выделить нарисованный прямоугольник (рисунок 7).

После этого выберите пункт меню **Insert → Convert to Symbol** (или нажмите горячую клавишу F8). В появившемся диалоговом окне (рисунок 8) выберите пункт **Button**.

Все, кнопка готова. Будем считать, что вы уже научились создавать кнопки, теперь научимся отслеживать нажатия на эти кнопки.

Кнопки в Macromedia Flash обладают обширным списком событий, на которые они могут реагировать:

press – клавиша мышки нажата, когда курсор находится в пределах кнопки;

release – клавиша мышки отжата, когда курсор находится в пределах кнопки;

releaseOutside – клавиша мышки отжата, когда курсор находится вне пределов кнопки;

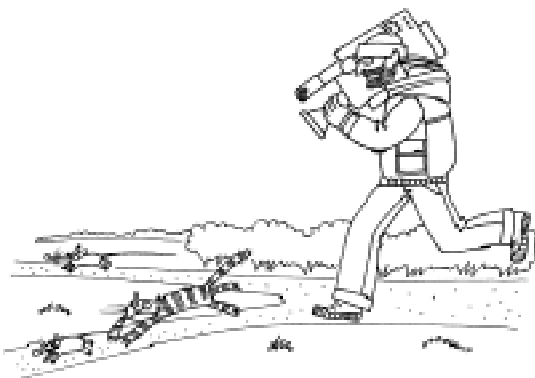
rollOver – курсор мыши входит в пределы кнопки;

rollOut – курсор выходит за пределы кнопки;

dragOver – курсор входит в пределы кнопки, при этом была нажата кнопка и нажата клавиша мыши;



Рисунок 8.



К сожалению, Flash «понимает» только левую клавишу мыши.

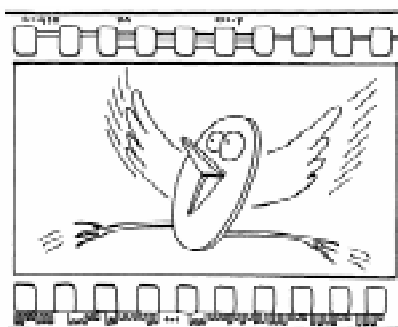
dragOut – курсор выходит за пределы кнопки, при этом была нажата кнопка и нажата клавиша мыши;

keyPress («клавиша») – была нажата «клавиша». Список клавиш можно посмотреть в справке по Flash (объект Key) или использовать панель параметров для ввода нужной клавиши.

К сожалению, Flash «понимает» только левую клавишу мыши. Правая используется для вызова контекстного меню (щелкните правой клавишей на каком-нибудь Flash мультике). Способов отлавливать во Flash среднюю клавишу или «колесико» (mouse wheel) я пока не встречал; думаю, что их не существует, такая возможность существует в JavaScript, который вы можете комбинировать с Flash.

Перехватываются эти события с помощью директивы **on()**. Синтаксис ее таков:

```
on (событие)
{
... // Другие инструкции
}
```



Для того чтобы оживить вашу кнопку...

Например, переход на страничку.

```
on (release)
{
  gotoURL («http://www.yandex.ru»);
}
```

Вот так просто можно перехватить все события, связанные с кнопкой. Ну, а как их использовать, – это дело исключительно вашего воображения.

АНИМИРОВАННАЯ КНОПКА

Вы можете также создать не просто абстрактный прямоугольник, но анимированную кнопку. Для того чтобы оживить вашу кнопку, необходимо сделать следующее:

Нарисовать прямоугольник и превратить его в кнопку (это вы уже умеете).

Дважды щелкнуть левой клавишей мышки (ЛКМ) и перед вами появится следующая картинка (см. рисунок 9).

Двойным щелчком ЛКМ был произведен переход в режим редактирования кнопки, и то, что вы видите, несколько не походит на ту временную линейку (Timeline), которую вы видели ранее. Такая временная линейка присуща только кнопкам. Как вы видите, у кнопки есть 4 состояния:

Up – кнопка не нажата, курсор мышки находится вне кнопки;

Over – кнопка не нажата, курсор мышки находится над кнопкой;

Down – кнопка нажата, курсор мышки находится над кнопкой;

Hit – область реакции кнопки.

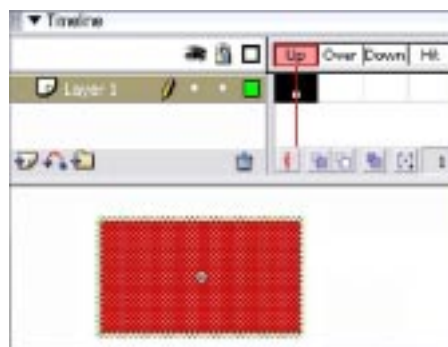


Рисунок 9.



Рисунок 10.

Первые три состояния могут быть анимированы, по умолчанию, все состояния равны состоянию Up. Чтобы создать все состояния кнопки, необходимо 3 раза нажать на клавишу F6. Таким образом, у вас появляется панель времени (рисунок 10).

То есть после вышеуказанных действий вы находитесь в состоянии Hit. Обычно анимируют состояние Over, и мы не будем отходить от этой традиции. Щелкните ЛКМ в состоянии Over (рисунок 11).

При этом произойдет автоматическое выделение прямоугольника, и вам остается лишь преобразовать его в Movie Clip нажатием клавиши F8 и выбрать в диалоговом окне опцию **Movie Clip** (MC) (рисунок 12).



Рисунок 12.



Рисунок 11.

Теперь необходимо перейти в режим редактирования MC. Для этого достаточно щелкнуть два раза ЛКМ по вновь созданному MC. И теперь вы видите хорошо известную вам временную линейку и можете создавать анимацию.

Обратите внимание: чтобы вернуться опять на основную сцену, необходимо щелкнуть ЛКМ по ссылке **Scene 1** над временной линейкой (рисунок 13).

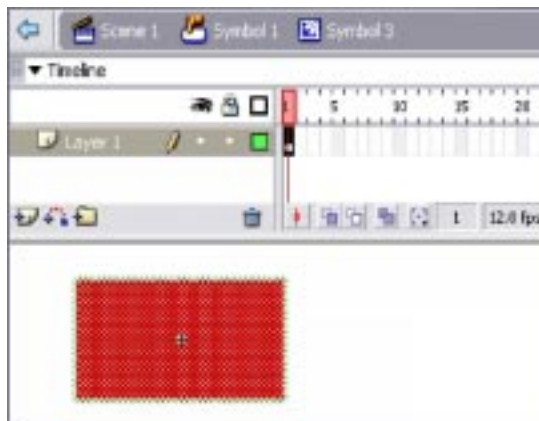


Рисунок 13.

*Штенников Дмитрий Геннадьевич,
Санкт-Петербургский
Государственный Университет
Информационных Технологий
Механики и Оптики.*



Наши авторы, 2004.
Our authors, 2004.