Dr. Lipscomb's Complete, Unabridged Tutorial

Using sound in Flash to enhance the music learning experience

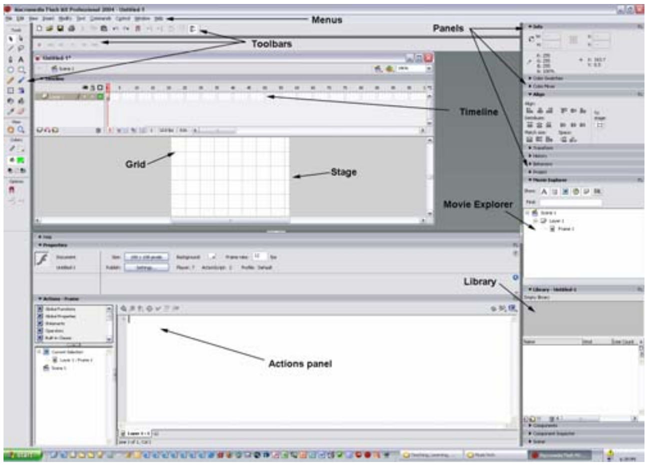
©2005 Created for the National Symposium on Music Instruction Technology

by Scott D. Lipscomb, Associate Professor
School of Music, *Northwestern University*.
lipscomb@northwestern.edu

Presentation made June 17th, 2005
Hartwick College, Oneonta, NY

Macromedia *Flash MX 2004* provides an authoring environment for the creation of truly interactive multimedia materials.[1] The free browser plug-in and player available from the Macromedia web site have established a near-ubiquitous presence on the Internet … over 97% of the computers used to surf the WWW have the plug-in installed, making this program one of the most reliable ways to ensure that your users will be able to utilize your interactive instructional materials to their fullest extent. *Flash* provides a marvelous balance between user-friendliness and complexity. It is possible to learn the program's most basic capabilities and put these to work almost immediately in the creation of interactive instructional materials. For those who are willing to spend the extra time to learn ActionScript, the scripting language that is part of *Flash*, the program's potential is almost limitless. One disappointing limitation is that *Flash* does not include MIDI capability, though it masterfully handles numerous types of digital audio files and substantially reduces file size of the web-ready movies using effective compression algorithms.

Though it does not work with MIDI, *Flash* offers many other attractive and useful capabilities. *Flash* allows users to import graphic images, video, sound files, and many other types of media created in other programs or downloaded from the Internet. Many types of objects (e.g., text, ovals, rectangles, lines, customized gradients, etc.) can be created directly in *Flash*, using the Tool palette. These basic shapes and objects can be combined to create professional looking graphics for use in your own animations. In addition, *Flash* comes with several "Common Libraries" to provide many essential tools. The "Buttons" library, for example, contains an impressive array of buttons, sliders, knobs, faders, and other useful objects that can be readily included in your movies. The "Learning Interactions" library contains templates – including complex interactive capabilities – for creating various types of quiz forms: true-false, multiple choice, fill in the



[1] The opening paragraphs of this paper are excerpted, in abridged form, from a forthcoming chapter (Lipscomb & Walls, in press) to be published in the *TI:ME Technology Guide*, 2nd edition.

blank, drag-and-drop, and others. In addition, the program comes with an impressive set of components with built-in functionality. Some of the most commonly-used Components – Macromedia's built-in, interactive objects – include the checkbox, radio button, progress bar, listbox, combo box, text input, and many others.

Like many Macromedia programs, *Flash* uses a "Timeline" metaphor to facilitate the creation of animations. This allows complex animations to be created without necessitating the frame-by-frame creation of each individual image. Instead, the user marks important locations (called "keyframes") along the Timeline and sets the desired location of each object in the movie at this point in time. *Flash* then automatically calculates the appropriate location for every object for every intervening frame, a process known as "tweening." Because *Flash* utilizes vector graphics instead of bitmaps (or raster graphics), the resulting files are much smaller and take less time for online visitors to download. The context-sensitive Properties window allows users direct access to the most common attributes of any object selected in the Work Area. Other panels (accessible from the "Windows" menu) allow the user to set alignment, transform & skew objects, mix colors, use pre-made Components, view all objects in the present movie, along with many other possibilities. For those who wish to move to more advanced levels of interactivity, *Flash* provides a powerful programming language called *ActionScript*, providing many of the same capabilities that were previously only the purview of much more complex and hard to learn programs like Java and C++.

There are two primary file types with which you must be familiar when working with Flash. When creating a Flash file (called a "movie") using the Flash interface, you work with an ".fla" file … the native format for Flash movies. When you "publish" a web-ready version of the movie, however, it is saved in ".swf" format (pronounced "swiff"). It is the ".swf" file that must be uploaded to your web site, *not* the ".fla" file. An internet browser (Internet Explorer, Netscape, Mozilla, Safari, Foxfire, etc.) with the Flash plug-in installed[2] will readily play back ".swf" file. In all likelihood, the browser will not know what to do if you link to an ".fla" file.

Work Flow in Flash

As you work with Flash, you will find that there are typically a series of steps involved with creating interactive multimedia content. The most common sequence of procedures is outlined below:

- draw or import graphics & sound files into Flash
- transform buttons, independent animations, & any other elements you intend to reuse into "symbols"
- place movie elements (vector graphics, bitmaps, symbols, etc.) on stage
- attach actions to buttons, movie clips, or frames on the timeline to make them interactive
    - in the steps outlined below, you will find that you are strongly encouraged to create two specialized layers in your Flash movies: an "action" layer (to

[2] You can download the free Flash plug-in from: http://www.macromedia.com/shockwave/download/download.cgi?P1_Prod_Version=ShockwaveFlash.

hold ActionScript elements) and a “sound” layer (providing a single location for your audio components)

- select a frame, symbol, stroke, fill, or text block on the stage to adjust its properties via different panels

Overview of Interface Elements

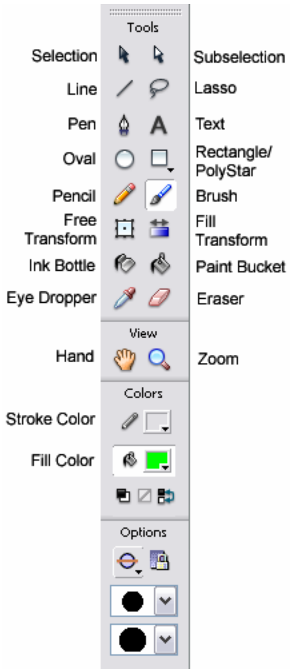
- Menus –many of the functions and capabilities of Flash are accessed by selecting a choice from the menu at the top of the screen
- Toolbars – allow you one-click access to many of the most often-used functions
    - You can reveal or hide any of the three main toolbars (main, controller, or edit bar) using the following menu selections: **`Window→Toolbars…`**
- Panels – any of the series of smaller windows within the Flash program interface; these panels allow you access to settings related to your movie and objects on the stage
    - You can reveal or hide any of the panels by selecting the appropriate option from the **`Window`** menu. The panels are grouped into three submenus: **`Design Panels`**, **`Development Panels`**, and **`Other Panels`**
    - Once visible, you can move the panels around by clicking on the top area of the panel and dragging it to the desired location
    - On the Windows platform (not Macintosh users), panels can be “docked” (i.e., virtually fastened) to the edge of the window, allowing you to organize your screen more easily; to “dock” or “undock” a chosen panel, grab the upper left corner of the panel where it has a grated appearance and move it either into our out of docked position
- Timeline – organizes and controls a document's content over time in layers and frames. Like films, Flash documents divide lengths of time into frames. Layers are like multiple film strips stacked on top of one another, each containing a different image that appears on the Stage. The major components of the Timeline are layers, frames, and the playhead.
    - layers in a document are listed in a column on the left side of the Timeline
    - frames contained in each layer appear in a row to the right of the layer name
        - the Timeline header at the top of the Timeline indicates frame numbers
    - the playhead indicates the current frame displayed on the Stage
    - the Timeline status display at the bottom of the Timeline indicates the selected frame number, the current frame rate, and the elapsed time to the current frame.
- Stage –the rectangular area where you place graphic content, including vector art, text boxes, buttons, imported bitmap graphics or video clips, and so on
    - the Stage in the Flash authoring environment represents the rectangular space in the Macromedia Flash Player where your Flash document is displayed during playback
    - using the **`View→Magnification`** submenu choices, you can zoom in and out to change the view of the Stage as you work

        - CTL+2 – handy shortcut to make the entire stage visible within the work area
- Library – the Library panel is where you store and organize symbols created in Flash, as well as imported files, including bitmap graphics, sound files, and video clips
    - the Library panel lets you organize library items in folders, see how often an item is used in a document, and sort items by type.
- Grid, Rulers, & Guides – Flash comes with rulers and guides that help you draw and lay out objects precisely. You can place guides in a document and snap objects to those guides, or turn on the grid and snap objects to it
    - grid – when the grid is displayed (**`View→Grid→Show Grid`**) in a document, it appears as a set of lines behind the artwork in all scenes. You can snap objects to the grid (**`View→Snapping→Snap to Grid`**), and you can modify the grid size and grid line color (**`View→Grid→Edit Grid`**)
    - rulers – when rulers are displayed (**`View→Rulers`**), they appear along the top and left sides of the document. You can change the unit of measure used in the rulers (**`Modify→Documents`**, then select an option from the pop-up menu at the upper right). When you move an element on the Stage with the rulers displayed, lines indicating the element's dimensions appear on the rulers
    - guides – when guides are visible (**`View→Guides→Show Guides`**) and rulers are displayed (see below), you can drag horizontal and vertical guides by clicking on the ruler and dragging the Guide line directly to the desired location on the Stage. You can also snap objects to guides (**`View→Snapping→Snap to Guides`**), and change guide color and determine how close objects must be to snap to a guide (**`View→Guide→Edit Guides`**)
        - guides appear only in the Timeline in which they were created
- Actions panel –lets you create and edit actions for an object or frame. Selecting a frame, button, or movie clip instance makes the Actions panel active. The Actions panel title changes to Button Actions, Movie Clip Actions, or Frame Actions, depending on what is selected.
    - *ActionScript* is the Flash scripting language used to assign actions to objects in your movie
- Movie Explorer – provides a snapshot of your entire Flash project, including its construction and the specific media elements used

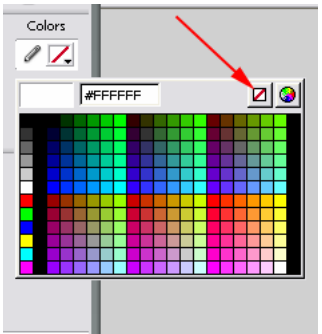
*Some Specific Tools (**`Window→Tools`**)*

The Tools panel contains a set of icons that allows you one-click access to many of Flash's powerful design tools. Notice the context-sensitive "Options" area at the bottom of this panel, which changes content based on which tool is selected

- Selection tool (black arrow) – used to select objects
    - clicking on the inside of a filled object selects only the inside ("fill")



    - clicking on the border of an object selects only the border (“stroke”)
    - you can also click-and-drag a rectangle to select a portion of any object or portions of many objects
        - to add more items to your current selection, hold the **`SHIFT`** key while you click on an object, border, fill, or draw a rectangle
    - to “unselect” objects – **`Edit→Deselect All`**
        - alternatively, you can simply select a different object or click on any blank area of the stage
- Subselection tool (white arrow) – allows you to reveal and manipulate the anchor points that define a line segment or curve
- Line tool – used to draw perfectly straight line segments
    - select the tool, then click at the location on the stage where you want your line to begin, drag the mouse pointer to the place where you want the line to end, then release the mouse.
    - the stroke color, line weight (in pixels), and the stroke style can be modified using the Properties window
- Lasso tool – for selecting multiple objects or portions of objects, you can draw a free form around as many objects – or portions of objects – as you like
    - you can use the **`SHIFT`** key to add to your selection as described above
- Pen Tool – a deceptively complex tool that allows you to draw straight lines, polygons, Bezier curves, and arches.
    - to create a straight line or polygon simply click and release the mouse button at the location on the stage where you want the line to begin, then click and release and the location where you want the line segment to end; this creates an “anchor point” at the beginning and end of the line segment
        - to create polygons or more complex objects, you can continue to click & release to create additional line segments
        - if you wish to create a “closed” object (i.e. to “close the path”), all you have to do is click and release one final time on the initial anchor point you created
    - once you have created a multi-sided object, using the Selection Tool, you can convert any of the sides made of straight lines into curves
        - click on the Selection Tool in the Tools panel
        - hover the mouse over one of the sides (a curved line will appear under the arrow cursor
        - click and drag the side until the desired shape is created
- Text Tool –to insert text into your Flash movie, click on this tool, then click the mouse at the location on the stage where you want your text to appear
    - set the text color, font face, style, etc. in the Properties window
    - IMPORTANT NOTE: beware of using uncommon fonts, since (unless the font is embedded in your movie [see me for details] the appearance might be dramatically different than intended for those who do not have the same font installed on their computer
- Oval Tool – allows you to draw ovals and circles, using a selected fill & stroke color
    - hold **`SHIFT`** key to create perfect circle



    - it is possible to create an object with a transparent stroke, so it appears to have no border
        - click on the Stroke color selector in the "Colors" area of the Tool panel, then select the "no stroke" item (square with a red line drawn through it) from the pop-up pallette
- Shape Tool(s) – to alternate between the Rectangle Tool and the PolyStar Tool, click and hold on the visible toolbar item until a pop-up appears, then select the desired tool
    - Rectangle Tool – creates a rectangle or square
        - hold **SHIFT** key to create perfect square
        - notice the "round rectangle radius" button in the "Options" area that allows you to create rectangles with rounded corners
    - PolyStar Tool – creates a five-sided polygon
- Pencil Tool – allows the creation of freeform lines with the option of using one of several "assistance modes":
    - experiment with the straighten, smooth, and ink assistance modes available in "Options" area
        - smooth mode – softens the curve of the line you draw
        - straighten mode – transforms the line into a series of straight-line segments and standard curves
            - additionally, in this mode, Flash performs shape recognition to the lines you draw so that if it detects something approximating a simple geometric shape such as a rectangle, oval, or triangle, it converts your drawing into whichever shape it detects
        - ink mode – bypasses the modifications of smooth & straighten mode, allowing for the most freeform drawing with minimal correction by Flash
- Brush Tool – another drawing tool that allows you (in the "Options" area) to set the brush size and shape
    - in addition to "Paint Normal" (which functions exactly as you would expect a pen to draw), using the mode selection drop-down box in the upper left corner of the "Options" area, you can select and use one of several complex drawing modes; to prepare to use one of these other modes, you must follow the steps below:
        - using the Selection Tool, select an object (or objects on the stage)
        - click on the brush tool and set the desired fill color, brush size, and brush shape
        - select one of the following drawing modes:
            - Paint Fills – draws over any area of the screen (including the stage itself) that can be considered a "fill," without affecting any of the areas defined as a "stroke"
            - Paint Behind – draws behind the selected objects, whether stroke or fill
            - Paint Selection – draws *only* in the selected area

        - Paint Inside – draws *only* inside the area of the page or object in which you initiate the brush stoke, affecting the fill area only (not the stroke area)
- Free Transform Tool – allows you to scale, rotate, compress, stretch, or skew lines and shapes
- Fill Transform Tool – for objects with a gradient color transition, this tool allows you to alter the shape and direction in which the colors transition
- Ink Bottle Tool – uses the Stroke pop-up to modify the outline of lines or objects, like the Paint Tool uses the Fill pop-up to change the fill of objects. The tool recognizes and works only on lines whether they form an outline of an object or are standalone lines created with the Line Tool or Pencil Tool.
    - In addition to changing the color of lines, you can also modify their thickness and style
    - you can change the properties (color, style and thickness) of several lines (and outlines) by selecting all of them and clicking on the selection with the ink bottle modifier
    - the Ink Bottle tool acts ONLY on lines, so if both outline and fill are selected, only the outline will be changed.
- Paint Bucket Tool – allows you to change the fill color of any object(s)
    - for gradient fills, the location that you click within the object determines the lightest portion of the fill, changing the apparent light source and shadowing
- Eyedropper Tool - allows you to select the attributes of a shape such as fill and stroke color, and line weight and style, and then transfer them to other shapes. This tool detects whether it is a stroke or a fill you are selecting, and then changes into the Ink Bottle (when selecting strokes) or the Paint Bucket (when selecting fills). The Eyedropper provides a quick means for storing and transferring attributes between editable shapes.
    - Click the Eyedropper tool in the Toolbar. The pointer becomes an eyedropper.
    - Position the eyedropper over the stroke or fill of a shape on the Stage.
        - strokes: when the mouse cursor is positioned over a stroke, a tiny pencil appears next to the tool. When you click on the stroke the Eyedropper becomes an Ink Bottle.
        - fills: when the mouse cursor is positioned over a fill, a tiny paint brush appears next to the tool. When you click on the fill the Eyedropper becomes a Paint Bucket.
    - Click on another shape's stroke or fill to transfer the selected attributes. The stroke or fill updates to the new color.
- Erase Tool – allows you to erase part or all of any object(s) on the stage, using any of a number of eraser shapes; analogous to the Brush tool described above, the Eraser Tool has several complex modes for deleting content
    - Erase Normal
    - Erase Fills
    - Erase Lines
    - Erase Selected Fills

  - Erase Inside

Layers – Keeping Your Graphics & Other Symbols Separate

Layers can be considered analogous to the transparent sheets of acetate stacked on top of each other by the cartoon studio animators. In the areas of a layer that don't contain content, you can see through to content on the layers below. Layers assist you in organizing content contained in your Flash movie. For example, you can keep background art, navigational buttons, sounds, and ActionScript command on separate layers. Additionally, you can create and edit objects on one layer without affecting objects on another layer.

- adding additional layers to your movie (**`Insert→Timeline→Layer`** or use the "New Layer" icon below the Timeline)
  - Note: *I recommend keeping all media on separate layers* (e.g., a sound layer, a buttons layer, an actions layer, etc.)
  -
- moving layers is as easy as clicking on the layer you wish to relocate in the Timeline and dragging it up or down to a new location
- hiding layers – click in the area below the eye icon on a given layer to toggle between visible and invisible
- locking layers – once you have completed work on a layer, to ensure that no change is made inadvertently, the ability to lock the layer is very useful
  - click in the area below the lock icon on a given layer to toggle between locked and unlocked
- frame labels – labels located throughout your movie that allow you to visually identify salient moments within the movie; these labels are also available to ActionScript, allowing users to navigate interactively to a specific point in the file. To add a label,
  - in the Timeline, select the frame to which you wish to add a label
  - if it is not visible already, open the Properties Window (**`Window→Properties`**)
  - in the Frame field, type a name (keep it brief)
  - in the Timeline, deselect the frame and you will see that the frame now contains a flag icon, identifying it as a "label"

Complex Graphics on a Single Layer

When Lines Intercept & Shapes Interact

When you create objects directly on the stage using the tools available in Flash (e.g., lines, ovals, rectangles, etc.), these objects do *not* maintain a separate identity when they intersect or overlap.

Try this example:

- Select the "Oval Tool" and, in the "Colors" section of the Tool window, choose a Stroke color and Fill Color
  - before proceeding, in the Properties window, change the Stroke Height property to 5

  - you can also change the Stroke Style, if you like to change the appearance of the shape's outline
- Draw a large oval on the stage
- Select the Arrow Tool and click somewhere in the middle of the oval you just created
- Notice how the fill of the oval is selected, but the outline is not
  - selected portions appear speckled and slightly lighter in color than unselected areas
  - if you want to test this out, simply press the **`DELETE`** key and notice how the filled portion of the oval disappears, while the outline (the "stroke") remains in place; if you did delete the fill, "Undo" the last action (**`Edit→Undo`** or **`CTL+Z`**)
- Make sure that the Oval Tool is selected, set the Stroke Color to "no stroke" (press the button in the upper righthand corner of the Stroke Color Palette … the white square with the crossed out by a red line), and choose a different fill color
- Draw a small oval inside the first oval that you drew
- Select the Arrow Tool, select the small oval you just drew and then, using the arrow tool, click & drag the small oval outside of the large oval
  - notice that the area where the small oval was drawn is now no longer filled with the fill color of the first oval
  - as long as an object is selected, you can move it around the stage as much as you like and it will not cause any alteration to other objects; however, as soon as it is unselected (or another object is selected), it merges with other graphic elements on the same layer
- If you want objects to retain their separateness when overlapping, you must place them on separate layers

Reusing Objects

Converting Graphics to Symbols & Symbol Instances

One of the most powerful capabilities of Flash is the capability to reuse objects (called "symbols.". Reusing objects also allows you a means of ensuring that your movie size stays relatively small … if you have 250 oval shapes drawn on your stage, they can all be based on the same "oval" symbol. Individual ovals (called "instances" of the oval symbol) can be altered in several ways:

- Library panel (**`Window→Library`** or **`CTL+L`**) for viewing all objects contained in the current movie
  - To edit a symbol, simply double-click on the object in the Library; editing the symbol itself changes *all instances* based upon that symbol
- Properties window (**`Window→Properties`**) for editing individual instances; can change brightness, tint, or transparency (alpha)
- Scaling (**`Modify→Transform→Scale`**)
- Rotation (**`Modify→Transform→Rotate and Skew`**)
- of course, all of the techniques can be applied and/or changed over time by using keyframes in the movie Timeline

To convert a graphic image into a symbol, simply select the image and choose **`Modify→Convert to Symbol`**. Once you have created a symbol, it is ready to reuse anywhere in your movie. You can access all symbols used in your Flash movie via the Library window (**`Window→Library`**) and add them to your movie by clicking on the desired item in the Library window and dragging it to the Stage.

Animation Basics

Creating Keyframes

- **`Insert→Timeline→Blank Keyframe`** – create brand new content in new frame
- **`Insert→Timeline→Keyframe`** – duplicates contents of previous keyframe

To add a Motion Tween animation, follow these steps:

- In the layer containing the object(s) you wish to animate, simply click on the frame where you want your animation to begin
- When you create a new movie, the first frame of Layer 1 automatically has a keyframe; otherwise, you can add one (**`Insert→Timeline→Keyframe`**)
  - o if your objects are not yet on the stage, *ensure that a keyframe on the appropriate layer of your movie is selected*, then drag the object(s) from the Library onto the Stage at the location you would like for them to appear
- Click on the final frame where you would like the animation to end, then select **`Insert→Timeline→Frame`**
- Right-click (Win) or Control-click (Mac) the first keyframe of this animated sequence to open the pop-up menu and select “Create Motion Tween”
  - o at this point a dashed line will connect the first frame of your animation to the last in the Timeline … this lets you know that the process is not yet complete
- Click once again on the final frame of your animation and move the object(s) to the location where you would like it/them to be at the conclusion of this animated sequence
  - o now the dashed line should have changed to an arrow pointing from the first frame of your Motion Tween animation to the last

In addition to the tweening motion, you can change other parameters of the object as it moves from one keyframe from the next. Try these ideas to change an object on the stage as it moves from one location to the next using a Motion Tween:

- *fade out*: click on the second keyframe, select the object, then set the Alpha property (in the Properties window) to 0
- *grow or shrink*: click on the second keyframe, then use the Free Transform Tool or the Transform Panel (**`Window→Transform`**) to resize the object
- *rotate*: click on the timeline ***between*** the two keyframes in the layer containing the tweened object, then select either CW (clockwise) or CCW (counter clockwise) form the “Rotate” drop-down box in the Properties window. You can

also change the number in the "times" textbox to determine how many times the object will rotate as the playhead moves from one keyframe to the next

Creating a more complex Flash movie …

Using Common Library Elements

- `File→New`
- To create an empty movie, select "Flash Document" from the "General" tab contained in the New Document dialog window that appears
- In the Properties window, set …
  - frame rate to "12" fps
  - set background color to a color of your liking
- then, click on the "Size" button and set …
  - stage width = 100
  - stage height = 75
  - set ruler unit to "pixels"
- then, click on the "Settings" and review settings for Flash, Formats, & HTML, using the tabs at the top of the "Publish Settings" dialog window
- save the file as "playButton.fla"
- `Window→Other Panels→Common Libraries→Buttons`
  - find a set of buttons you like and drag an instance of a Play button to the Stage
- test the movie (`Control→Test Movie`)
  - notice how the `mouseOver`, `mouseDown`, and `mouseOut` events respond to your interaction … nothing happens when you press the buttons, but we'll get to that!!
- close the window containing the test of your movice
- notice that the button symbol consists of several layers of images
    - in the Timeline, click on the layers one at a time and you will see a bounding box on the stage identify which portion of the image is represented in that layer
    - hide them one at a time by clicking on the appropriate layer directly underneath the eye icon, so you can see the effect of eliminating one or more row(s) of the button
  - simply by "testing" the movie, a web-ready **SWF** file was created (or "published") and can be inserted directly into an HTML page
    - when you create hyperlinks to Flash movies or embed Flash movies within your web site, you *must* link to the SWF file (web-ready format) – *not* the FLA file (Flash's native format). This is extremely important, since your web browser will not be able to play the FLA file
  - Flash drawing tools create *vector graphics* …
    - on the stage, resize the Play button (`Modify→Transform→Scale`), making it about 4 times as large as its original size

      - you can also set the width & height values by typing them directly into the Properties window (must be expanded)
    - notice that, even though the image is significantly larger, it does not distort
      - this is the difference between working with *vector graphics* rather than *bitmap* (or *raster*) graphics
  - if you want to insert this Flash movie into a web page using **Dreamweaver**, …
    - "publish" the Flash movie (File→Publish…) and save the SWF (web-ready) file in the root folder or one of the subfolders (perhaps the "movies" folder?) of your web site
    - start Dreamweaver and either create a new blank HTM file or open the existing file into which you wish to place the movie
    - select **`Insert→Media→Flash`**
      - browse to find the SWF file you created when you published in the step above
    - a placeholder for the Flash object is placed on your HTML page and can be moved around and manipulated just like other objects on the web page
    - to see the movie – instead of the placeholder – click on the "Play" button in the Properties window
      - if you do not see the "Play" button, make sure that your Properties window is expanded by clicking on the down-pointing triangle in its lower right corner
      - after pressing the Play button, the movie plays as if it were in a browser window
    - click on the "Stop" button in the Properties window to return the Flash object to "Edit" mode
    - to confirm the effect of using *vector graphics* within Dreamweaver …
      - resize the movie, making it about 4 times as large as its original size
      - click on the "Play" button in the Properties window again
        - voila! … due to the wonders of *vector graphic* images, you will see no distortion or pixilation
    - close Dreamweaver and **return to Flash**
- double-click on your Play Button on the Stage (or click on your button select **`Edit→Edit Symbols`**)
  - you have now entered Symbol Edit mode; notice that there is a new label ("Play") in the upper left corner of the Timeline area next to the "Scene 1" label
  - click through the various button "states" (Up, Over, Down, Hit) to see how the image is changed; Buttons are a special type of symbol that uses this 4-element timeline

        - the "Hit" state is not one that you see while your button is in use, but its function is very important … it simply identifies the portion(s) of the image that are "active"
    - if you were creating your own button, you would simply select the frame for the desired state and edit the appearance of the button object on stage to reflect the way you would like for it to appear in that state
- create a second movie entitled "stopButton.fla"
    - you can do this on your own by modifying the steps above

Sound in Flash

Importing Sound & Using It in Your Movie
(use: 01soundCrash_template.fla)

- **`File→New`**, then select "Flash Document" from the "General" tab contained in the New Document dialog window that appears
- **`File→Import→Import to Library`**, then browse on your hard drive to select a sound file
- **`Window→Library`** (or **`CTL+L`**)
    - notice the sound file has been added to your movie as a Symbol
    - if you click on the "Information" icon at the bottom of the Library window, you can determine the appropriate level & type of compression for your sound file
- **Recommended**: add a separate layer labeled "sound"
- IMPORTANT: click on the placeholder frame where you want your sound file to play, then select **`Insert→Timeline→Keyframe`**
- You must *always* add a keyframe to the frames in your timeline where you want a specific event – like initiating playback of a sound file – to occur
- with the frame containing the newly added keyframe selected on the appropriate layer of the timeline, *drag sound file from Library onto the stage*
    - notice that a representation of the sound wave appears in the frame to confirm that a sound has been added
- while the frame with the sound file is selected, notice the Properties window now contains additional options for manipulating the sound
    - from the "Effect" drop-down box, select Custom and notice how you can control the playback of your sound file; to get to this same window, you can simply click on the "Edit" button to the right of the "Effect" drop-down
    - learn about the different "Sync" types (Event, Start, Stop, & Stream) to ensure that you get the most out of using sound in your movies
- **`Control→Test Movie`**

Adding Sound to a Button
(use: 02soundButtons_template.fla)

- **`File→New`**, then select "Flash Document" from the "General" tab contained in the New Document dialog window that appears

- Open the Buttons Common Library panel (**`Window→Other Panels→Common Libraries→Buttons`**)
- select the button you want to use from the Common Library and drag an instance onto the Stage
- Double-click on the button to enter "Symbol Edit" mode; alternatively, you can select **`Edit→Edit Symbols`**
    - Just below the labels on the left side of the Timeline, you should now see the name of the button upon which you are operating to the right of the "Scene 1" label
    - Note that the timeline for button symbols is different than the movie timeline, containing only four frames … one representing each button state
- Decide which states (up, down, and/or over) to which you wish to attach sounds
    - IMPORTANT: If the frame where you intend to insert the sound does not contain a keyframe, you must add one now (**`Insert→Timeline→Keyframe`**)
- **`File→Import`**
    - import an appropriate sound for each of these states
- Add a layer and label it "sound"
    - If your movie's Library window (*different from* the Common Library window) is not open, **`Window→Library`** (or **`CTL+L`**)
- In the "sound" layer, click on the frame representing the state to which you wish to add a sound, then drag the appropriate sound from the Library
    - repeat this step for every state for which you wish to add a sound
- **`Control→Test Movie`**

- To create a button that will allow the user to stop sound playback …
    - select an appropriate button from the Common Library (or create your own)
    - drag an instance of the button from your movie's Library to the stage
    - click on the Stop button instance
    - add the following code to the "Actions" window:

```
on (release) {
      stopAllSounds();
}
```

    - **`Control→Test Movie`**

Control a Sound Using Frame Navigation

(use: 03playSound_frameNavigation_template.fla)

This method takes advantage of the "Start" and "Stop" sound Sync modes, initiating and terminating playback simply by navigating to an appropriate frame in the movie using the **`gotoAndStop()`** command when one of the buttons is pressed.

- **`File→New`**, then select "Flash Document" from the "General" tab contained in the New Document dialog window that appears
- import the sound file you wish to play into your movie's Library (**`File→Import→Import to Library…`**)

- select a "Play" and "Stop" button from the Common Library or create your own
- create a "button" layer, select the keyframe in frame 1 of that layer, then place an instance of each button on the Stage
- create an "actions" layer, select the keyframe on frame 1, then type the following into the "Actions" window to avoid automatic playback of your movie, turning complete control over to user interactions:

  ```
  stop();
  ```
- create a "labels" layer and insert two frame labels that will be used to navigate within your movie
  - click on the keyframe in frame 1 of the "labels" layer and type "stop" into the Frame Label textbox in the Properties window
  - insert a keyframe (**`Insert→Timeline→Keyframe`**) at frame 15 of this same layer and type "play" into the Frame Label textbox in the Properties window
  - before proceeding, insert a frame (**`Insert→Timeline→Frame`**) at frame 15 on both the actions & labels layers
- add a "sound" layer to your movie and add a *keyframe* (**`Insert→Timeline→Keyframe`**) at frame 15 of this layer
- click on the keyframe in frame 1 of the "sound" layer and drag the sound file from your movie's Library to the stage, then click on the same keyframe a second time and select "Stop" from the "Sync" drop-down box in the Properties window
  - entering the frame containing this keyframe causes the sound file, if playing, to halt playback
- now, click on the keyframe in frame 15 of the "sound" layer and drag the same sound file you dragged to frame 1 from your movie's Library to the stage, then click on the keyframe in frame 15 a second time and select "Start" from the "Sync" drop-down box in the Properties window
  - entering the frame containing this keyframe causes the sound file, if not already playing, to initiate playback
- the final step is to attach some basic ActionScript to the "release" event for each of the two buttons, causing the playback head of your movie to relocate to the appropriate frame in your movie:
  - click on the "Play" button and type the following text into the "Actions" window:

    ```
    on (release)
    {
       gotoAndStop("play");
    }
    ```
  - click on the "Stop" button and type the following text into the "Actions" window:

    ```
    on (release)
    {
       gotoAndStop("stop");
    }
    ```
- **`Control→Test Movie`**

Using Flash's Sound Object to Instantiate & Control Playback
(use: 04soundObject_template.fla)

This method for integrating sound into your Flash movies utilizes the capabilities of the `Sound` object provided with Flash. In this example, we are going to control sound playback completely through the use of ActionScript, so we need to take a few steps at the outset to inform Flash about our sound file and how we will refer to it. This is accomplished by creating a "Linkage."

- `File→New`, then select "Flash Document" from the "General" tab contained in the New Document dialog window that appears
- import the sound you want to play into the movie's Library (`File→Import→Import to Library…`)
- right-click (Windows) or Control-click (Mac) on the name of the file in your Library and select "Linkage…" from the pop-up men that appears
    - remember, if you do not see the Library window, you can always make it visible by selecting `Window→Library` (or `Control+L`)
- in the Linkage Properties dialogue box, …
    - type "eurotech" into the Identifier textbox
        - this is the name we will use in ActionScript to refer to this specific sound file
    - place a check in the "Export for ActionScript" box
    - place a check in the "Export in first frame" box

We have now taken all of the preliminary steps necessary to make the sound file accessible to our ActionScript code, so we can now begin the process of instantiating a Sound object and attaching a sound file to it, after we accomplish some basic set-up tasks.

- setting up our movie
    - create three layers and name them: labels, actions, & buttons
    - insert a frame (`Insert→Timeline→Frame`) at frame 10 on the actions & buttons layers
    - insert a *keyframe* (`Insert→Timeline→Keyframe`) at frame 10 on the "labels" layer
    - using the same technique described previously, add the following frame labels to:
        - frame 1: "initialize"
            - the purpose of this frame is simply to carry out some basic initialization steps that only need to be carried out one time at the very beginning of the movie, then cause the playback head to navigate to a later frame where user interaction will occur
        - frame 10: "playback"
    - add the initialization code
        - click on the keyframe in frame 1 of the "actions" layer
        - type the following text into the "Actions" window:

```
var currLocation = 0;
var mySound = new Sound();
```

```
mySound.attachSound("eurotech");
gotoAndStop("playback");
```

- allow me to explain the purpose of this code
    - first, I create a variable called `currLocation` that is used to track the current position (in milliseconds) within the sound file[3]
    - second, I create a Sound object called `mySound`
    - I then use the `attachSound()` method to attach the sound file in the movie's Library – remember that we set the Identifier field to "eurotech" when we created the "Linkage" above? – so that any command sent to the `mySound` object acts on the "eurotech" sound file
    - finally, I use the `gotoAndStop()` command to send the movie's playback head to the "playback" frame label

- add buttons to the stage for controlling sound file playback
    - select a Play, Pause, and Stop button from the Common Library (or create your own
    - select the keyframe in frame 1 of the "buttons" layer and drag an instance of each button to the desired location on the Stage
    - select the Play button and type the following code into the "Actions" window:[4]

```
on (release) {
   mySound.start(currLocation / 1000);
}
```

    - select the Pause button and type the following code into the "Actions" window:

```
on (release) {
   currLocation = mySound.position;
   mySound.stop();
}
```

    - select the Stop button and type the following code into the "Actions" window:

```
on (release) {
   mySound.stop();
   currLocation = 0;
}
```

- **`Control→Test Movie`**

[3] Using this value allows me to cause the buttons to behave exactly as the user expects from her/his experience with CD and DVD players. Notice, as you study the ActionScript below that the Start button calls the `start()` method using the value of `currLocation` (divided by 1000; see below) as the starting location. When a users presses the Pause button, the current position within the sound file is loaded into this variable before playback is halted, so that when the Start button is pressed, playback is initiated from the point where the sound file left off. If, however, the Stop button is pressed, the value of `currLocation` is set to "0", so that when the play button is pressed, playback starts from the very beginning of the sound file.

[4] There is an inconsistency in the manner in which time is referenced from within the Sound object. All of the objects methods use milliseconds as the unit of time *except* the `start()` method which – for some strange reason – uses *seconds* instead!! Beware of this discrepancy when you are creating your own movies that incorporate the built-in Sound object. This is the reason that the `currLocation` value must be divided by 1000.

Using Streaming Sound
(use: 05_I-iv-IV-V_template.fla)

When you want to ensure that audio-visual synchronization is maintained within a Flash movie, your best bet is to use the "Stream" Sync mode, as described below. To see how this works in a very basic way, follow the steps below. After the general concept is understood, you can take a look at – and get more meaning from – the set of musical form templates available on my web site.

- **File→New**, then select "Flash Document" from the "General" tab contained in the New Document dialog window that appears
- import the sound you want to stream into the movie's Library (**File→Import→Import to Library…**)
- in the movie's Timeline, create a new layer and name it "sound"
- select frame 1 on the "sound" layer, then drag the sound file from your movie's Library onto the stage
- click on frame 1 of the "sound" layer a second time, then select "Stream" from the Sync drop-down box in the Properties window
- now insert a frame (**Insert→Timeline→Frame**) at frame 60 of the "sound" layer
    - notice that the representation of the digital audio file continues from frame 1 until the end of the sound file (which can be *thousands* of frames, depending on the frame rate of your movie and the duration of your sound file)
    - also notice that when you click and drag the playback head (the red rectangle about the Timeline), you can hear the content of the sound file as it passes by
- by creating other layers in your movie to contain objects, animations, interactive elements, you can synchronize, as precisely as you like, the appearance and/or movement of objects on the stage with your audio sound file.

There is a great deal of potential inherent in this method for use in the music classroom, as demonstrated by the form templates available from my web site. Simply point your web browser to: http://www.lecafeamericain.net/faculty.htm, then click on "stuff" to navigate to the appropriate page containing a list of available downloads. At the time of NSMIT 2005, Flash templates were available for the 12-bar blues, AABA, and I-vi-IV-V progression.[5] On that same page, you will also find BubbleMachine™, a tool – free to all educators – for the creation of bubble charts to allow point-and-click navigation and interactive exploration of any musical composition that has been saved in MP3 format. The form templates require that the user has *Flash MX 2004* (or later), but the BubbleMachine™ program requires only the free Flash Player. Documentation and basic tutorials are also available from the web site.

Starting and Stopping Sound using Actions in Flash

- **File→New**, then select "Flash Document" from the "General" tab contained in the New Document dialog window that appears

[5] There is also a template for the Sonata Form that was created in *Director*, if you are interested.

- In the Properties window, set …
    - frame rate to “12” fps
    - set background color to a color of your liking
- then, click on the “Size” button and set …
    - stage width = 250
    - stage height = 75
    - set ruler unit to “pixels”
- Save the file as “soundButtons.fla”
- **`Window→Other Panels→Common Libraries→Buttons`**
    - find a set of buttons you like and drag an instance of a Play button and a Stop button onto the stage; rearrange so that they are spread evenly across the stage & at about the same vertical position
        - you can use the Grid (**`View→Grid→Show Grid`**) and/or Guides (**`View→Guides→Show Guides`**), if you like, to assist with alignment
            - alternatively, once both objects are on the Stage, you can use the “Align” commands (**`Modify→Align→[`*`select an option`*`]`**) to neatly arrange your objects on stage
            - these same options are also available via the Align panel (**`Window→Design Panels→Align`**)
        - feel free to readjust the movie properties (height & width) to your satisfaction
- Test the movie (**`Control→Test Movie`**)
    - notice how the **`mouseOver`**, **`mouseDown`**, and **`mouseOut`** events respond to your interaction … nothing happens when you press the buttons, but we’ll get to that!!
    - as you now know, you can rescale either or both buttons (**`Modify→Transform→Scale`**) without causing the image to pixelate
- Import a sound file (**`File→Import→Import to Library`**)
    - Set “File Type” to “All Sound Formats,” then import a sound file into your project
- Open the Library Window (**`Window→Library`**) and notice that both of your buttons & a sound file are listed in the Window
    - in this window, you can create subfolders to organize the “assets” (Flash’s word for all of the elements used) in your project … for this project, since it is so small, it is not necessary to create any subfolders
- Organize Layers for your Scene
    - name the layer with the buttons appropriately: “buttons”
    - in a more complex movie, you would add other layers … naming them helps you as you design
- set up the Stop button, so it will cause all sounds to stop [*the easy way*]
    - Click on the Stop button to select it
    - Open the Actions Window (**`Window→Actions`**)
    - Type the following into the Actions window:

```
on (release){stopAllSounds();}
```

- Add sound to the Play button [a little bit harder]
    - on the Stage, double-click on the Play button to enter Symbol Edit mode
        - notice that next to the "Scene 1" label just above the left side of the Timeline area, there is now a label with the name of the symbol used to create this button; you are now able to edit the button symbol; to return to the Scene when you are finished editing, simply click on the "Scene 1" label … but don't do that just quite yet
    - add a Layer (**`Insert→Timeline→Layer`** or click the "New Layer" icon) and name it "sound"
    - in the "sound" layer, click on the frame under "Down" [this is the button state – when it is pressed – that you want to begin playback of the sound file], then insert a keyframe (**`Insert→Timeline→Keyframe`**)
    - while the keyframe in the sound layer of the Play button is selected, add the desired sound by simply dragging it from the Library window on to the Stage
    - click on the keyframe in the sound layer again and select "Event" from the Sync drop-down list, since you want sound file playback to be initiated by the **`mouseDown`** *event*
        - you can also change the "Loop" value from "1" to the number of times you would like your sound file to play; "99" is good when you want to give the illusion of infinite playback
- Click on the "Scene 1" label (upper left of the Timeline area) to leave Symbol Edit mode and return to the main movie timeline
- Test your movie (**`Control→Test Movie`**)
    - clicking the Play button should cause the sound file to start playing back (multiple copies, if you click it more than once)
    - clicking the Stop button should cause all sound files being played to stop

Using Different Types of Animation in Flash

Frame-by-Frame Animation

Why do this, when you can use tweening???

Shape-Tweened Animation

Shape Tweening allows you to morph one graphic image or shape into another.

- **`File→New`**, then select "Flash Document" from the "General" tab contained in the New Document dialog window that appears
- draw a square of any color with no outline in the lower lefthand corner of the Stage
    - use the Rectangle tool on the Drawing toolbar
    - hold the **`SHIFT`** key down to create a perfect square (same technique works for drawing perfect circles with the Oval tool)
- select frame 25 on the same layer of the timeline as your square
- **`Insert→Timeline→Blank Keyframe`**

- we are going to use text in our movie but, before adding text, you should *always* embed the font(s) you use as part of your movie, so users who do not have that specific font installed will still see the text as you intend them to
    - click on the context menu in the upper right corner of the Library window and select "New Font…"
    - type a name for your embedded font and select the font on your system upon which you wish to base the embedded font
    - if you want, you can set the style of the font as bold and/or italic
- while the playback head is still on frame 25, select the Text tool and click on the stage in the area where you want the text to appear
- set the following properties in the Properties window
    - choose your embedded font (in the font list, all embedded fonts are followed by an asterisk
        - hint: if you want to make your embedded font easy to find, start it with an underscore ("_") character and it will appear at the top of the font list on the Windows platform and at the end of the font list on the Mac
        - Flash comes with several fonts that are intended to be universal (_sans, _serif, and _typewriter), but the method above ensures that the font will appear as you intend
    - click on the downward facing arrow to the right of the Font Size text box, then use the pop-up slider to set the font size to 20
    - choose any color, but different than the color of your square
- click in the upper righthand corner of the Stage and type a capital "V"
- select the Arrow tool and the text you just typed will be automatically selected
- **`Modify→Break Apart`**
    - this turns the text into a graphic shape that is no longer editable as a text object; a necessary step to accomplish the "morphing" effect we want to see
- click on Frame 1 to move the playback head to that frame
- in the Properties window, set the following properties:
    - choose "Shape" from the Tweening drop-down menu
    - set the Blend Type to "Angular"
        - "distributive" works best for blending smooth, curvy shapes
        - "angular" works best with sharp corners and straight sides
    - leave the Easing setting at its default value
        - "easing in" causes the animation to move slower at the beginning and gradually increasing in speed
        - "easing out" causes the animation to start fast and decelerate as it continues
        - set to the middle, speed will remain constant
- if you want to add "shape hints" to alter the resulting animation, simply:
    - click on the first frame of the shape tween, then select **`Modify→Shape→Add Shape Hint`** (or **`CTRL+SHIFT+H`**)
    - move the shape hint (the numbered circle_ to the location on the that you want to control

  - click on the keyframe located at the end of the shape tween and move the shape hint to the point in the ending shape that should correspond to the first point you marked
  - add additional shape hints as necessary
  - test your movie to see the results

Motion-Tweened Animation

- **`File→New`**, then select "Flash Document" from the "General" tab contained in the New Document dialog window that appears
- Click the Oval tool on the Drawing toolbar and draw a medium sized circle (no outline, you pick a color) in the lower left corner of the Stage
  - use the **`SHIFT`** key if you want to make a perfect circle
- Select white as the fill color and draw a small white circle slightly on – but slightly off-center - the black circle (look a little like a billiard ball)
  - remember that once your shapes overlap on the same layer, you can no longer edit them separately; if you need to keep them separate, you must place the shapes on separate layers
- Click on Frame 25 in the timeline on the same layer as the black & white circles
- **`Insert→Timeline→Frame`**
- Right-click (Win) or Control-click (Mac) the keyframe on Frame 1 and select "Create Motion Tween" from the pop-up menu
  - since motion tween works only with specific items (symbol instances, groups, & text blocks), this command automatically converts content of any other type (like our "ball") into a symbol instance
  - open the Library window and confirm that a symbol has been added called "Tween 1"
  - notice also that the "tween" you created is not yet complete (identified by the dashed line instead of an arrow pointing from the initial keyframe to the next) … we still have to define the *end* of our tweening motion
- move the playback head to Frame 25 and click the Arrow tool in the Drawing toolbar
- drag the black ball to the middle of the right side of the Stage, adding a keyframe automatically to Frame 25 (notice the completed tweening arrow that is now displayed on the timeline between the two keyframes)
- you can move the playback head back & forth manually to see your animation

Customize the Motion Tween (getting in a little deeper)

Let's make the ball appear to grow smaller and disappear by the end of the tween

- move the playback head to Frame 25
- click the Arrow tool on the Drawing toolbar and select the ball
- **`Modify→Transform→Scale and Rotate…`**
- type "40" into the "Scale" box
  - as a result, the object will be scaled to 40% of its original size at this point in the animation

- while the ball symbol in frame 25 is selected, in the Properties window, choose "alpha" from the "Color" drop-down list and enter "0" in the percentage box
    - setting the alpha value to "0" results in making the ball completely transparent, so it will seem to disappear
- click on keyframe in Frame 1 in the timeline and, in the Properties window, set the following properties
    - Scale: checked
    - Rotate: CW (clockwise)
    - Times: 2
    - Orient to path: unchecked
    - Easing: -100 (all the way to the bottom of the slider)
    - Synchronize: unchecked
    - Snap: Checked

Motion Tweening Along a Path

- Select the layer that contains our motion-tweened animation and press the "Add Motion Guide" icon (just to the right of the "New Layer" icon at the lower left side of the timeline)
    - a guide layer will always appear *above* the layer it controls and the controlled layer will appear indented under the name of the motion guide layer
- Click on the motion guide layer if it is not already selected as the current layer
- Select the Pencil tool from the Drawing toolbar
    - in the "Options" area of the Tool window, choose "Smooth" as the assistance mode
- Starting near the center of the ball, draw a curvy line on the stage ending up near the middle of the right side of the stage
- **`Control→Test Movie`**

Saving Animations as Graphic Symbols or Movie Clip Symbols

Animated graphic symbols and movie clips are two kinds of animated symbols, with differences that – at first – may seem subtle and hard to distinguish

- Graphic Symbol
    - tied to the timeline of any movie in which it is placed
    - no sounds or interactivity retained, even if present in original copied frames
- Movie Clip Symbol (like a film loop in Director)
    - runs on its own, independent timeline
        - when in development environment you see only a static image (must Test Movie to see animation)
        - when testing movie, continues to run animation, even if playback head is stopped, as long as movieClip exists in frame where it stops

        - if number of frames over which animation extends changes, continues to run animation smoothly, without jumping from mid-animation to the beginning
    - retain sounds and interactivity

To convert existing animation into a graphic symbol or movie clip:

- Create an animation using any method outlined above
- Select all frames in all layers you wish to convert
- **`Edit→Timeline→Copy Frames`**
- **`Insert→New Symbol`**
- Type a name for your symbol
- Select either “Graphic” (to create an animated graphic) or “Movie Clip” (to create a movie clip), then click on the OK button
    - Your view will then change from Movie Editing mode to Symbol Editing mode
- Select Frame 1 in the symbol timeline, and choose **`Edit→Timeline→Paste Frames`**
- To return to Movie Editing Mode, click the current scene name in upper lefthand corner of the window
- Open the movie library (**`Window→Library`**) and notice that your new symbol has been added to the list

To insert graphic animation or movie clip into your movie

- select the frame in which you want your graphic animation or movie clip to appear
- drag a copy of the symbol onto the Stage
- in the timeline, select the last frame in which you want your animation to appear
    - remember that, in determining the number of frames, you must keep in mind that animated graphics depend on the movie timeline, while movie clips do not
- choose **`Insert→Timeline→Frame`**, causing Flash to add in-between frames from the first to last frames selected
- **`Control→Test Movie`**

Completed movie has been saved as “movieClipVsAnimatedGraphic.fla”.

ActionScript – the Basics

For interactivity, Flash requires three things

1. an event that triggers an action
    a. mouse events (press, release, releaseOutside, rollover, rollout, dragOver, and dragOut)
    b. keyboard events (occur when a key on the computer keyboard is pressed)
    c. frame events (*must* be placed at keyframes and are triggered when the playback head reaches that frame in the timeline)

   d. movie clip events (load, enterFrame, unload, mouseDown, mouseUp, mouseMove, keyDown, keyUp, data)
2. an action triggered by that event
   a. these actions are attached using the Actions window (**`Window→Actions`**) to create ActionScript
3. the target object that performs the action or is affected by it … can be either:
   a. the current movie and its timeline
   b. other movies and their timelines (e.g., movie clip instances)
   c. external applications (e.g., internet browsers)

It is highly recommended that you add a separate layer to your movie that is used exclusively for Frame Actions and label it "actions."

Adding actions to a frame:

- select the appropriate frame in the "actions" layer of the movie timeline
- select the desired action by doing one of the following:
  - click on the plus sign ("**`+`**") in the upper left corner of the Actions window and navigate through action categories to the desired action
  - as you become more familiar with the scripting language, you will find that by far the fastest way to add actions is to type the ActionScript directly into the Actions window
- to add multiple actions to a frame, simply follow the previous step until all desired actions have been added

Editing the actions list:

- to change the order of actions, cut and paste within the Actions window
  - select the desired action
- to remove an action from the list:
  - select the desired action, then press the **`DELETE`** key

Other Topics of Interest

Mask Layers

Allow you to focus attention on a limited area of the stage

- single layer
- multiple layers

To create a mask layer:

- **`File→New`**
- To create an empty movie, select "Flash Document" from the "General" tab contained in the New Document dialog window that appears
- Create several different colored objects on the Stage
- Name this layer of your movie "objects" and lock it
- Add another layer above the objects layer and name it "mask"

- Open the layer properties window (**`Modify→Layer…`**) and change the layer type from “Normal” to “Mask”
- click on the “objects” layer and open the layer properties window, then change the layer type from “Normal” to “Masked” and accept the change by clicking on the OK button
    - notice that making this change has caused the label of the “objects” layer to be indented and preceded by an arrow, representing the fact that this layer is now controlled by the “mask” layer
- select the “mask” layer again, choose the Oval Tool, set the Stroke Color to “no stroke,” and set the Fill Color to black
- Draw a circle in the middle of the stage, covering several objects on the “objects” layer below
- **`Control→Test Movie`** (or **`CTL+ENTER`**)

Notice how *only* the portion of the “objects” layer that shows through when the movie is played is the section over which the black circle was drawn. That black circle acts as the “mask.”

Guide Layers

Allow you to create a “path” in one layer to control the action of a symbol in a “guided” layer; see “Motion Tweening Along a Path” later in this tutorial for an example.

Turning Bitmaps into Vector Graphics

One of the reasons that Flash is able to create such interesting animations and still maintain relatively small file sizes is its use of vector graphics instead of bitmap graphics. Vector graphics describe images using lines and curves, called vectors, that also include color and position properties. In this case, the image is described by points through which lines pass, creating each object’s outlines. The color of a given object is determined by the color of the outline and the color of the area enclosed by the outline. Bitmap graphics describe images using colored dots, called pixels, arranged in a grid. In bitmap graphics, the image is described by the specific location and color value of each pixel in the grid, creating an image in much the same manner as a mosaic.

To convert a bitmap graphic into a vector graphic:

- place a copy of the bitmap image photograph on the Stage
- select the bitmap
- **`Modify→Bitmap→Trace Bitmap`**
    - experiment with the color threshold, minimum area, curve fit, and corner threshold settings as you observe the results of the conversion process

More ActionScript Examples

To pause a movie:

- select the frame in the “actions” layer of the movie timeline where you want your movie to pause

- in the Actions Window, select `Basic Actions→Stop`
- deselect the frame in your movie timeline to complete the process
    - confirm that an action has been added by noticing that a small “a” now appears in the intended frame of the movie timeline in the “actions” layer

Using GoTo actions to navigate within your movie:

- select the frame in the “actions” layer of the movie timeline where you want your movie to branch to a different location
- in the Actions Window, select `Basic Actions→Go To`
- Specify the new location:
    - in the “Scene” field, either type the Scene name or select the appropriate name from the drop-down list
    - specify the frame to which Flash should navigate
        - from the “Type” drop-down menu, choose one of the frame types, allowing you to refer to a specific frame number, frame label, the next/previous frame, or expression
        - if you selected “Frame Number,” “Frame Label,” or “Expression” in the previous step, you must provide the appropriate information in the “Frame” field
    - the “Go to and Play” check box should be checked if you want your movie to begin playing immediately when it arrives at the new location; otherwise, it will pause in the arrival frame

A PreLoader movie

- add a scene to your movie by clicking on the “+” in the lower right of the Scene Panel (`Window→Design Panels→Scene`)
    - name it “preLoader” and move it to the top of the Scene list by clicking on it and dragging
- create the animation you want your user to see while the movie loads
    - an example has been provided for you (“`preLoader.fla`”)
- add an “action” layer to your preLoader scene, then add the following Actions
    - in frame 1, add an “ifFrameLoaded” action (`Basic Actions→If Frame is Loaded`)
        - set the “Scene” field to the specific scene you want to begin playing when the movie has loaded
        - set the Frame Number or Frame Label field to the appropriate value … this is the amount of your movie that must be loaded before the “preLoader” scene will end and start playback of your main movie
    - in the last frame of your “preLoader” animation “Go To” action
        - set the “Scene” field to “<current scene>”
        - set the “Type” field to “Frame Number” and the “Frame” field to “1” … this will cause your preLoader scene to repeat until the portion of the following scene you specified above has been loaded