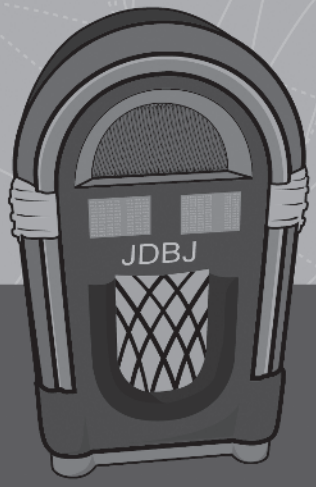
FLASH WORKSPACE



Interface Overview

WHEN YOU OPEN FLASH for the first time, you will most likely see a central white drawing area ("the Stage") surrounded by all sorts of panels and buttons. It may seem a little daunting, but Flash has only four main areas: the Stage, Timeline, toolbars, and panels.

Flash is available in Windows and Macintosh flavors. The only real major difference between these is that Windows supports dockable panels; otherwise, getting to know Flash in Windows is the same as getting to know it on the Macintosh.

In this chapter, we introduce the main areas of the Flash interface, looking at how the interface works in general.

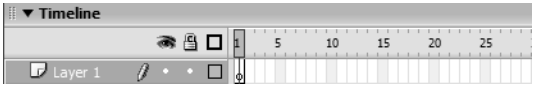
1.1 Overview: Creating Flash Animation

➡ 4.1 Timeline Overview

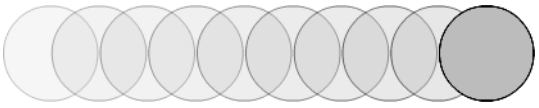
➡ 10.2 Animation Concepts in Flash

For those coming to Flash from an HTML or Photoshop background, the Flash interface can seem a little alien. The reason for this is that web design and Photoshop have a *print-based* mindset. Flash follows a very different *animation-based* mindset. Let's see how the interface works to create animation.

Start Flash and select the default user interface (Window **>** Panel Sets **>** Default Layout). In the middle of the screen is the Stage. This is currently a blank white rectangle; an empty Stage. The content that you add here is what the user sees when you put your Flash site onto the Web. At the top of the screen is the *Timeline*.



Think of the Timeline as a *film reel*, each rectangle corresponding to one frame in your movie. Above frame 1 is a pink rectangle, the *playhead.* The playhead always sits above the current frame, and during movie playback, it will move forward through the Timeline, left to right.



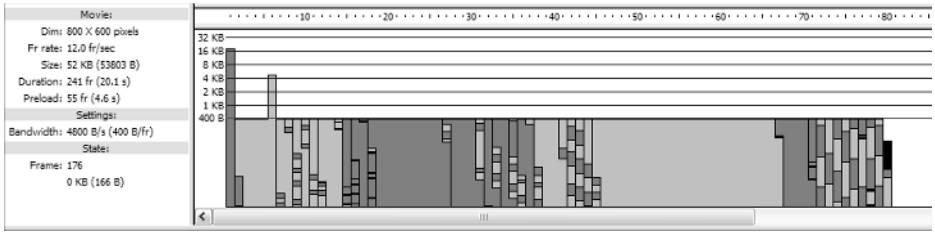
So how does that cause animation? Well, for each frame in the Timeline, you will have a different corresponding Stage view as the animated content moves slightly per frame. Assume your Stage looks as shown in the top image for frames 1–10; as the playhead moves between these frames, the dot will appear to move from left to right, causing animation, as seen in the bottom image.

Although there are different ways of making your "consecutive Stage views" look different (you can either use Timeline animation or move things around between frames using ActionScript, Flash's coding language), the principle remains the same: a playhead that moves quickly through a succession of frames, each associated with a slightly different Stage. When viewed over time, this process causes animation.

1.2 Overview: Testing Flash Content

When you create your animations in the Flash authoring environment, you are creating the *source* file, the FLA. The file that the user will see is a compiled version of the FLA, the SWF file. The compiled SWF content can only be viewed via the Flash Player and *not* in the authoring environment. To compile and view the final compiled SWF content, you need to move from the authoring environment into the *test environment* window (Ctrl-/⌘-Enter).

At the bottom of the Test window is the Stage area. At the top is a graph, the Bandwidth Profiler, which tells you something about how big (in file size) your animation will be, how long it will take to load into the user's browser, and whether it will play over the Web as intended. To the right you will (by default) see the Projects panel if you are using Flash MX Professional 2004.



Here is a bandwidth profile for an 80+ frame animation. The frame number is listed along the top, and the bar graph shows the bandwidth each frame needs to play on time. As you can see, frame 1 requires about 16K to be loaded within one frame (about 1/12 of a second, assuming the default frame rate) to play on time. A 56K modem would give you (worst case) around 400 bytes a second, and anything above the 400 B line on the profiler would result in the frame content not loading in time, causing a pause.

You can see this happening in real time if you press Ctrl-/⌘-Enter a second time. The movie will start again, this time simulating what would happen on the Web. A green band appears signifying the number of frames that has loaded into the browser, and the pointer (shown at frame 21) shows the position of the playhead. As long as the next frame is always loaded (i.e., painted green) before the playhead gets to it, the Flash movie will play without any delays caused by low bandwidth. This process of starting to play a timeline before it is fully loaded is called *streaming content*. The opposite process (waiting until everything is loaded before starting) is called *preloading content*.

➠ 26.1 Using the Test Environment

➠ 26.4 Using the Debugger Panel

Test Movie
Ctrl Enter

⌘ Enter

When testing your Flash productions, you should *always* test content destined for the Web via the browser every so often: Although the Flash test environment is good, it cannot model all the quirks of the browser, which is what all users will be using to view your Flash web content.

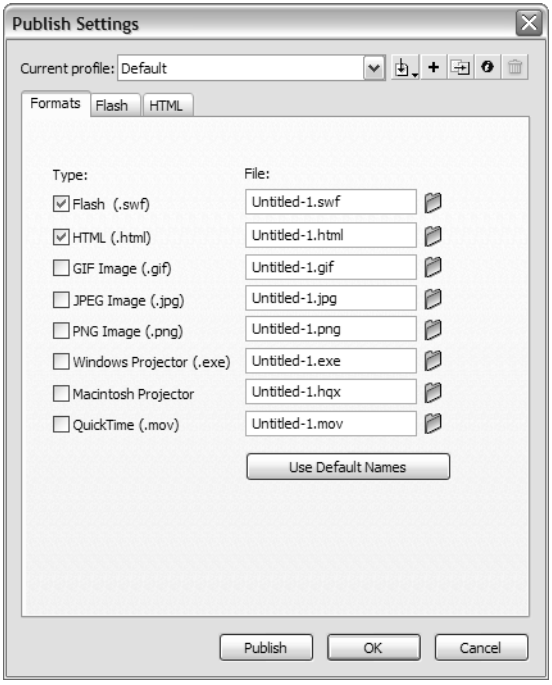
When you use the test environment, a Shockwave for Flash (.swf) and (if your Flash also contains ActionScript) a Shockwave diagnostic (.swd) file are created. The SWF is a compiled and compressed version of the FLA file and is what the browser needs to display your Flash content. The SWD file contains diagnostic information created by the test environment. It is kept separate from the SWF file for security reasons (the SWD is used to debug ActionScript and contains a full uncompiled listing of your code, so it could be used to steal your scripts and uncover any security measures you are using). In most cases, you can delete the SWD after testing.

1.3 Overview: Publishing Flash Content

➠ 27 Publishing and Deploying Flash Content

Publish
Ctrl F12
Shift F12

Once you have a working animation together, you need to deploy it onto the Web. For this you need an HTML file (`.html`) and a Shockwave for Flash file (`.swf`). Although some browsers can view the Flash SWF file directly, for maximum compatibility, you will usually use an HTML file that references the SWF file. To create these two files you need to select File **>** Publish Settings.



The Formats tab allows you to select the output formats. Usually, you only need to select the first two, Flash (SWF) and HTML. Clicking the Publish button (top-right corner) creates the HTML and SWF files. To see the final Flash presentation in a browser, select File **>** Publish Preview **>** Default (HTML) or File **>** Publish Preview **>** HTML.

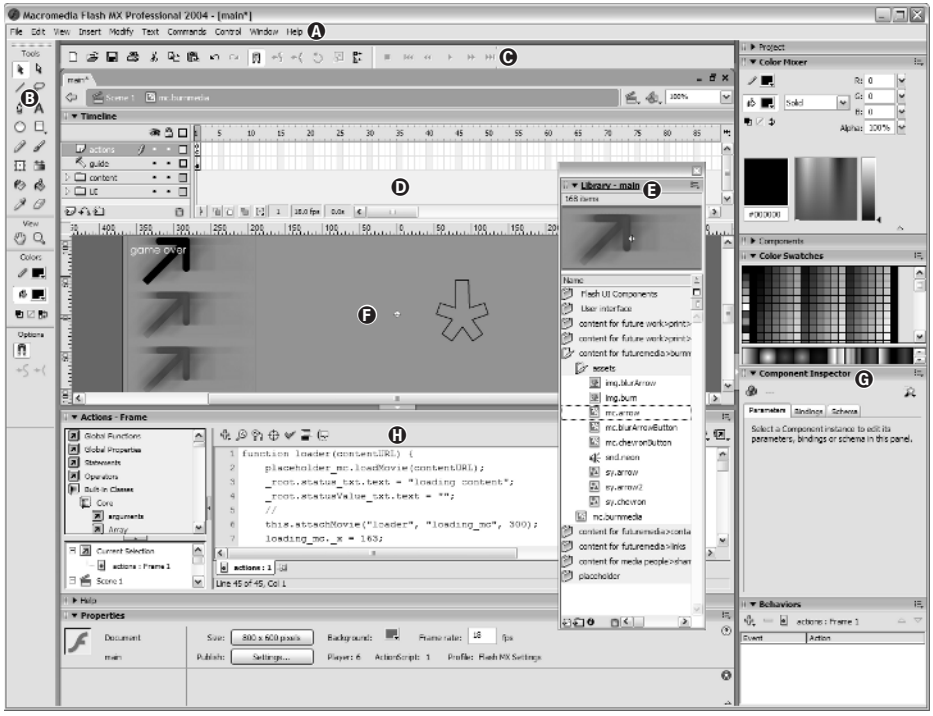
To actually deploy your content onto the Web, the easiest option is to use a modern browser with built-in FTP (file transfer protocol) abilities, and this includes current versions of Internet Explorer. Simply enter the name of your FTP (e.g., `ftp://myUserID@mydomain.com/`) as the web address and drag and drop the HTML and SWF files into the browser window to upload.

When uploading content to the server, you need to upload the HTML and SWF files, and not the FLA. If you see an unexpected empty white stage when you view the HTML file, it is usually because the browser is not finding the SWF file; make sure you have uploaded the correct file and put it in the correct place.

FTP functions are built-in to Internet Explorer 6.*x* and higher, and it seems to be one of the easiest options for uploading Flash content to your server.

1.4 Windows Authoring Interface

The Windows interface incorporates a system of dockable panels, something that is not available on the Macintosh.



A Menu bar Click any of the menu headings to open the corresponding drop-down menus.

B Tools panel Click these icons to select tools and tool options. The tools are used to create vector-based graphics and text. Flash can also handle bitmaps, but these have to be imported into the environment; they cannot be created within Flash.

C Toolbars Provide a quick way of accessing common functions. Although the toolbars do not appear in the default configuration, you are strongly recommended to have them showing.

D Timeline Used to create sequences of animation frames, or to attach frame-based scripts.

E Library panel One of the most used panels. Used to organize and store assets, including Flash graphic symbols, video, sound, and bitmaps.

F Stage The viewable area of the final Flash movie. Assets are also created on the Stage before being moved to the Library panel (E).

G Side docked panel area The default area that most panels will first appear.

H Lower docked panel area The default area that the Actions and Help panels and the Property inspector will appear.

➠ 1.6 Working with Panels (Windows)

Although the recommended minimum screen resolution is 1024×768, most designers use something considerably higher; 1280×1024 is a realistic minimum for the purposes of efficiency. Flash was initially designed as an operating system for a pen- or tablet-based computer, and it does seem to work best when combined with a pen or tablet, particularly when using the drawing tools.

The toolbars do not appear by default, but they are extremely useful. To make them appear, select Window > Toolbars and check Main, Status, and Controller in the submenu that appears.

1.5 Windows Test Interface

- 2.9 Control Menu
- 26.1 Using the Test Environment
- 26.4 Using the Debugger Panel

Simulate download (toggle)
Ctrl Enter
⌘ Enter

Test movie (from authoring enviroment)
Ctrl Enter
⌘ Enter

Toggle bandwidth profiler
Ctrl B

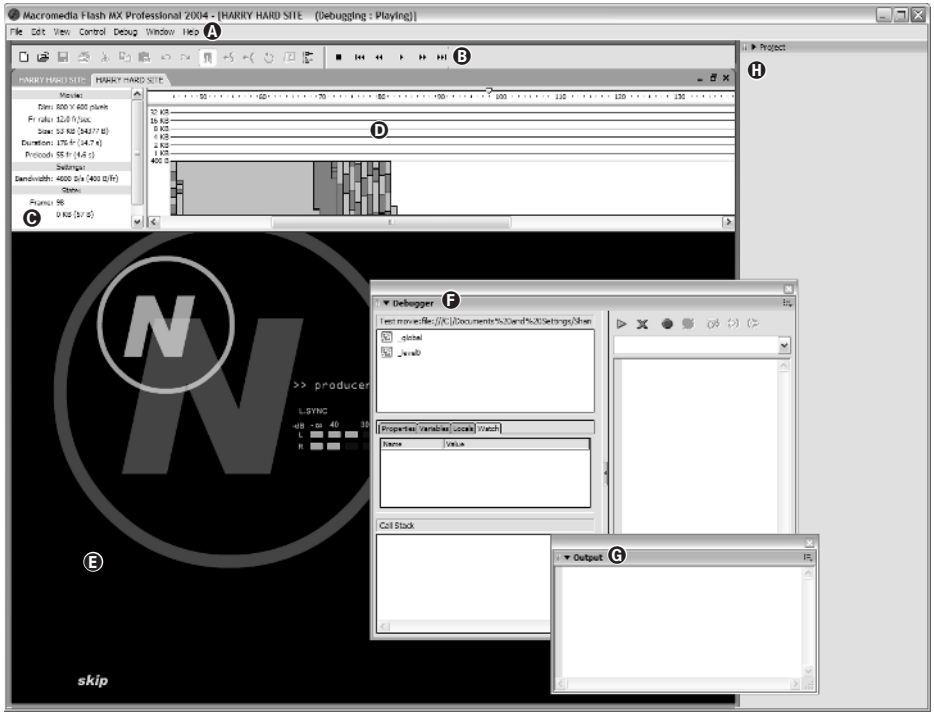
Toggle Output window
F2

Toggle Debugger window
Shift F4

Exit test mode
Ctrl F4
⌘ F4

To make the toolbars appear, select Window > Toolbars and check Main, Status, and Controller in the submenu that appears.

Although somewhat hidden when you first start Flash, the test environment is integral to the Flash production workflow. It is used to run Flash content via the Flash Player and to investigate the bandwidth profile of the movie under test. It is also used via the debugger and output windows to test ActionScript.



A Menu bar Click any of the menu headings to open the corresponding drop-down menus.

B Toolbars Provide a quick way of accessing common functions. Although the toolbars do not appear in the default configuration, you are strongly recommended to have them showing.

C Bandwidth Profiler; info Provides basic information regarding the movie under test.

D Bandwidth Profiler; graph Provides per-frame bandwidth information for the movie under test.

E Stage The viewable area of the final Flash movie. The Stage will be shown rendered and animated in real time via the Flash Player.

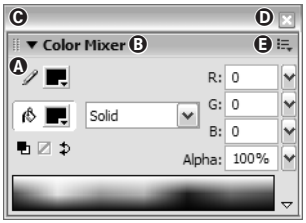
F Debugger Debugs scripts via a number of features, including the ability to watch or change variables in real time, as well as code breakpoints.

G Output window Allows scripts to output information to the test environment via the ActionScript `trace()` action. Trace actions are usually (by default) ignored outside the test environment (i.e., when you deploy the Flash presentation online, view it on a browser or via the standalone Flash Player). In the test environment, the output window will appear whenever a trace action is encountered.

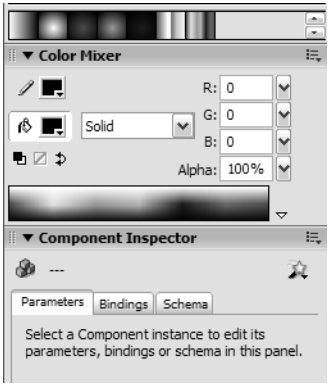
H Docking area The Projects panel will appear to the right of the test environment for Flash MX 2004 Professional.

1.6 Working with Panels (Windows)

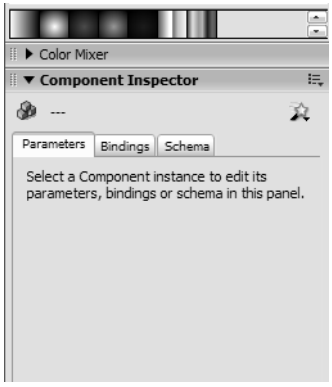
Flash for Windows uses a system of dockable panels. Although they look different, they all have the same basic features.



- **A** Drag this icon to dock or undock the panel to the sides, top, or bottom of the screen or to another panel.
- **B** Click the arrow icon or title text to toggle maximized or minimized views of the panel.
- **C** Drag this bar to move the panel without docking.
- **D** Click this button to close the panel. (To close a docked panel, click the panel menu icon and select Close Panel.)
- **E** Click this icon to access the panel menu.



Docked, maximized



Docked, minimized

Undocked, minimized

➠ 5 Panels

Toggle all panels on/off
Tab

You can also minimize, maximize, or close a panel by right-clicking the panel title area and selecting from the contextual menu. This menu also allows you to select Help on the current panel.

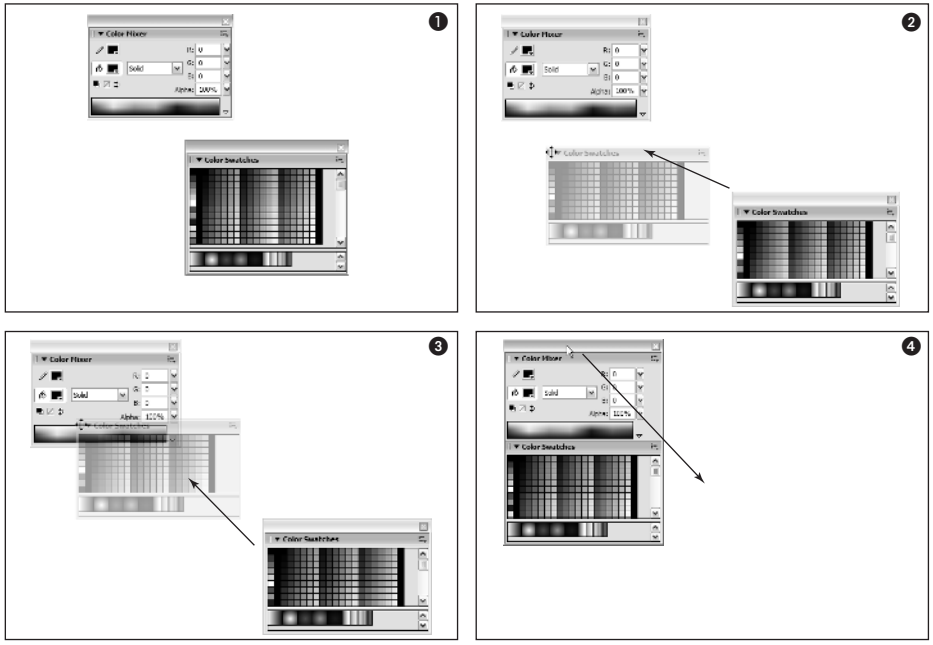
You can open and close all panels via the Window menu.

Continues

1.6 Working with Panels (Windows) *(continued)*

➠ 2.11 Window Menu

To undock a panel, click-drag the knurled area on its title bar and drop it outside the docking area. To redock it, simply do the reverse.



In addition to docking panels to the sides of the screen, you can also dock panels into floating groups. Click-drag a panel by the knurled area (1) until it is over the panel you want to dock to (2, 3). The target panel will also inherit a bold outline when this happens. Release, and the dropped panel will dock onto the target panel. You can move the panel group by using the window's drag bar at the top of the groups (4).

Note that the Timeline and toolbar are also panels, although there is usually little reason to dock them away from their default positions. A useful trick is to temporarily minimize the Timeline panel when you are using the Stage, thus giving you a bigger view of the Stage.

Most panels will remember where they were last closed and reappear in the same position if you reopen them.

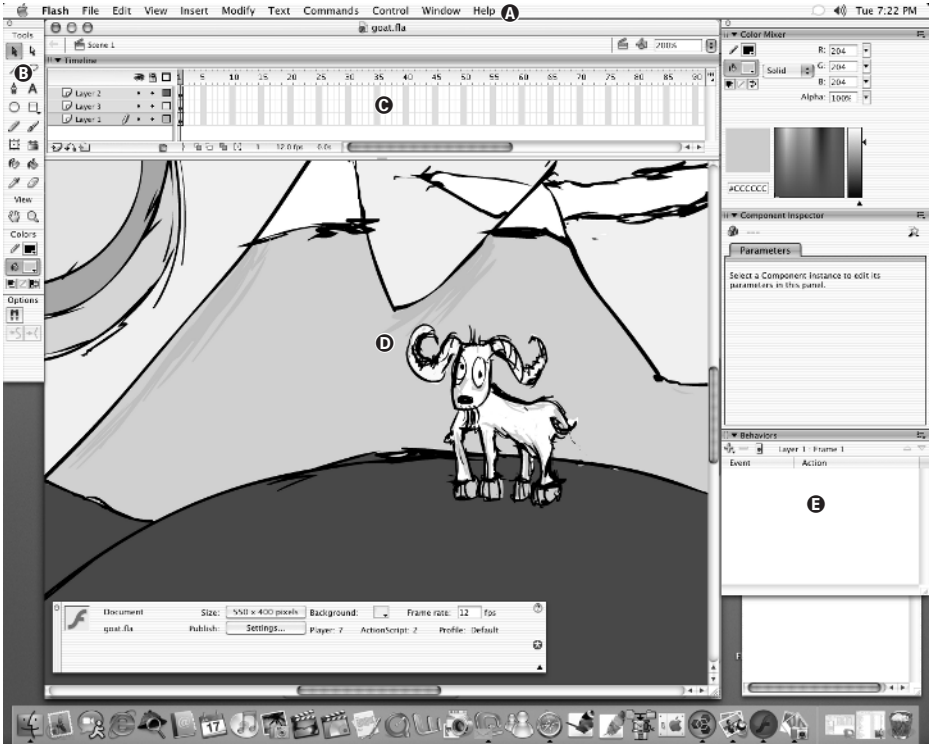
Although you can dock to any of the four edges of the screen, there is usually no reason to dock panels far from their default positions.

INTERFACE DIFFERENCES BETWEEN "STANDARD" AND PROFESSIONAL

The following differences are apparent between the interfaces of the two versions of Flash: Flash MX Professional 2004 contains a full-screen ActionScript editor, a slide-based editing environment, and the Projects panel. It also contains more file options in the File > New (and Start panel) options and more default components in the Components panel (not all of which work with Flash MX 2004).

1.7 Mac OS X Authoring Interface

The OS X authoring interface does not support the docking system used by Windows versions, but it is otherwise identical, and all the major interface areas (Tools panel, menus, Timeline, Stage) are in the same relative positions as the PC version.



A Menu bar Click any of the menu headings to open the corresponding drop-down menus.

B Tools panel Click these icons to select tools and tool options. The tools are used to create vector-based graphics and text. Flash can also handle bitmaps, but these have to be imported into the environment; they cannot be created within Flash.

C Timeline Used to create sequences of animation frames or to attach frame-based scripts.

D Stage The viewable area of the final Flash movie. Assets are also created on the Stage, before being moved to the Library panel (E).

E Panels Mac panels are always floating and undocked given that there is no docking system in the Mac version.

➠ 1.9 Working with Panels (Mac)

Although the recommended minimum screen resolution is 1024×768, most designers use something considerably higher; 1280×1024 is a realistic minimum for the purposes of efficiency. Flash was initially designed as an operating system for a pen- or tablet-based computer, and it does seem to work best when combined with a pen or tablet, particularly when using the drawing tools.

The toolbars do not appear by default, but they are extremely useful. To make them appear, select Window > Toolbars and check Main, Status, and Controller in the submenu that appears. They are not shown in the image, which is the default appearance.

1.8 Mac OS X Test Interface

- 2.9 Control Menu
- 26.1 Using the Test Environment
- 26.4 Using the Debugger Panel

Simulate download (toggle)
⌘ Enter

Test movie (from authoring environment)
⌘ Enter

Toggle Bandwidth Profiler
⌘ B

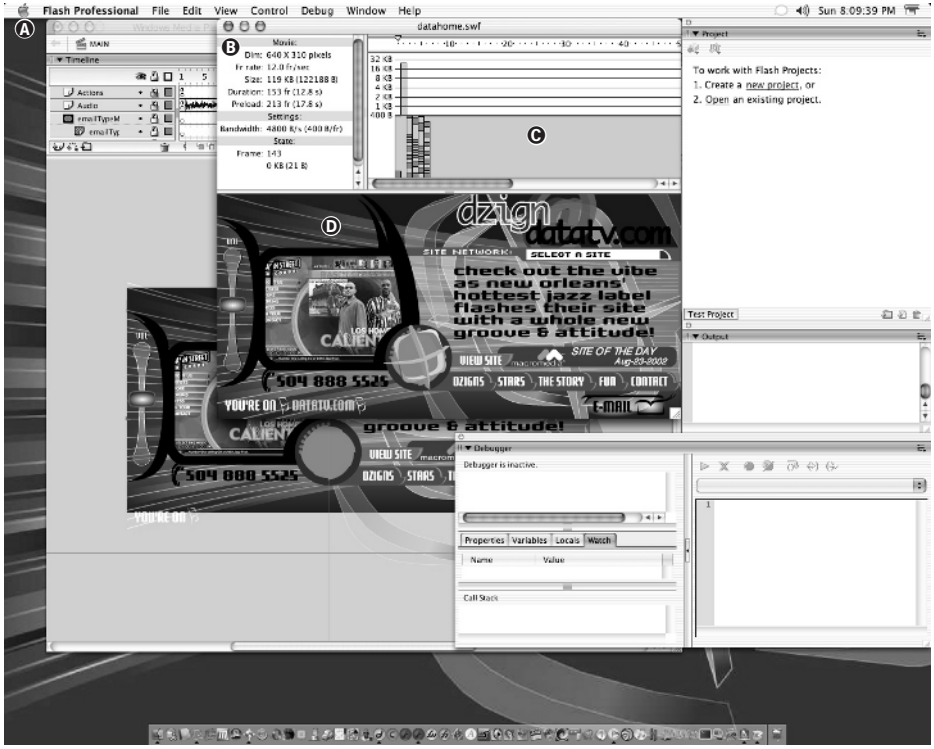
Toggle Output window
F2

Toggle Debugger window
Shift F4

Exit test mode
⌘ F4

To make the toolbars appear, select Window > Toolbars and check Main, Status, and Controller in the submenu that appears.

The Mac test environment is substantially the same as the Windows-based version but without docking panels. The interface is again substantially the same, with all major portions in the same place.



Ⓐ **Menu bar** Click any of the menu headings to open the corresponding drop-down menus.

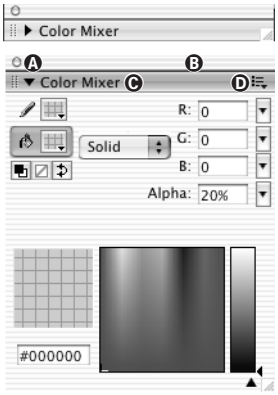
Ⓑ **Bandwidth Profiler; info** Provides basic information regarding the movie under test.

Ⓒ **Bandwidth Profiler; graph** Provides per-frame bandwidth information for the movie under test.

Ⓓ **Stage** The viewable area of the final Flash movie. The Stage will be shown rendered and animated in real time via the Flash Player.

1.9 Working with Panels (Mac)

Mac panels are much simpler than their Windows-based cousins: they don't have the docking features, although you can maximize or minimize them. To maximize or minimize the panel, click the panel's top bar.



Ⓐ Click the title to close the panel.

Ⓑ Drag this bar to move the panel.

Ⓒ Click the arrow icon or title text to toggle minimized (top) or maximized (bottom) views of the panel.

Ⓓ Click this icon to access the panel menu.

➠ 5 Panels

➠ 2.9 Window Menu

Toggle all panels on or off
Tab

You can also minimize, maximize, or close a panel by ⌘-clicking the panel title area and selecting from the contextual menu. This menu also allows you to select Help on the current panel.

You can open and close all panels via the Window menu.

Note that the Timeline and toolbar are also panels, although there is usually little reason to dock them away from their default positions. A useful trick is to temporarily minimize the Timeline panel when you are using the Stage, thus allowing you a bigger view of the Stage.

Most panels will remember where they were last closed and reappear in the same position if you reopen them.

1.10 Interface Objects

➠ 3.3 Controller Toolbar

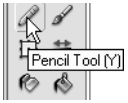
The menu icon will disappear when you minimize a panel because the panel is will be deemed to be inactive.

The Minimum/Maximum feature (which switches between two levels of information detail or window views) should not be confused with Maximized and Minimized panels (which totally open and close the panel).

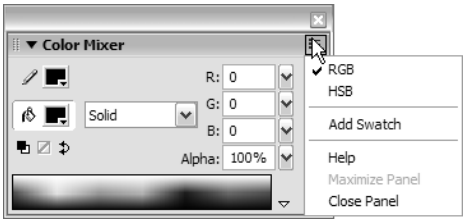
In many cases, sliders limit the maximum and minimum values you can manually enter. Their values are *not* always the absolute ranges; you can usually go further if you make the same changes through ActionScript, allowing you to *overdrive* a value for some interesting effects. This occurs particularly for the Sound and Color objects, where you can create phase reversed or distorted sound (through volume overdrive or underdrive) and create some cool color filtering effects (through overdriven color transitions).

Flash has common controls that are spread across the interface. These each work in a standard way, so if you understand how an instance of one works, you will probably understand it the next time you see the same thing for a different control. It works both ways however, because some controls are a little subtle, and if you miss them the first time, you will probably always miss them. It's a good thing there are reference books!

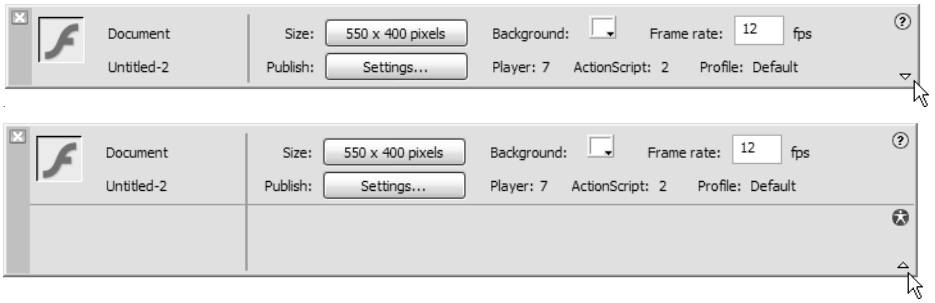
Text labels next to a text field, check box, or other input device can be clicked to activate the input area. For example, clicking the text here will toggle the check box in the same way clicking the check box directly would. This is useful for those with high resolution screens, or if you use laptops with those inaccurate and small mouse touchpad areas—accuracy is no longer a problem!



Tooltips are little bits of help text that show up if you hover over a control or button long enough. When you do, a yellow text rectangle will appear. If you don't see tooltips, select Edit > Preferences > General tab and check Show Tooltips.



Panel menu icon For standard panels, the menu icon is . The menu is accessed through a different icon on the Timeline .

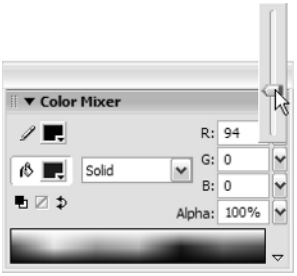


Some panels have an arrow icon at the bottom right, which makes the panel show its maximum or minimum information or resizes between a minimum open (with scrollbars) and maximum open (no scrollbars required—panel opens to show all content) size. It is particularly used on the Property inspector, the Color Mixer, and the Components panel.

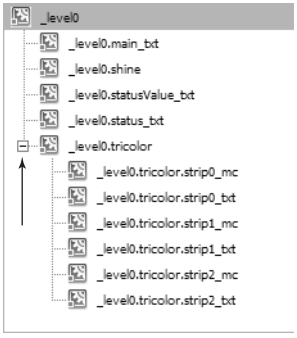
1.10 Interface Objects *(continued)*

Nonstandard User Interface Elements

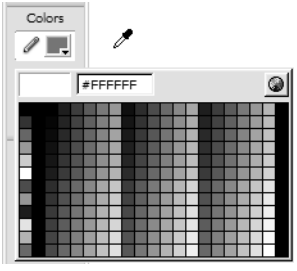
As with most applications, Flash uses many standard user interface elements, including check boxes, radio and standard buttons, text input fields, and drop-down menus. Assuming all Flash designers are familiar with common interface elements, this section lists the nonstandard ones, or those that sometimes appear with a nonstandard appearance.



Slider or "value spinner" Where a slider is next to a value, the value will update as you move the slider up or down the scale. Where the value controls something else (such as the thickness of a line), the "something else" will not update until the slider closes. You can also type a value directly into the value field and bypass the slider altogether (this is useful for entering exact values).



Tree Flash content is internally structured in a hierarchy rather like the file structure of your hard drive (which contains folders within folders and paths such as `c:/myFolder/myfile`). Rather than folders, Flash uses *timelines* to form its hierarchy, and instead of files, you have content on the Timeline. Whenever Flash needs to show you this structure, it uses a hierarchical tree with collapsible branches. To open a branch, click the + at the bottom of a branch, and to close it, click the –.



Color picker The Tools panel, Property inspector, and Color Mixer panel make extensive use of color pickers. You will know when a color picker is available when you see a little down-arrow at the bottom right of a color. Clicking such a color will cause the Color Mixer to appear and the cursor to change into the Eyedropper tool. By default, the swatch that appears will consist of the web-safe palette. You can also select a color by entering its HTML hexadecimal number (# followed by a six-digit hexadecimal number) or by using the eyedropper to select a color (either from the swatch or by clicking any pixel within the Flash application window). Clicking the icon at the top-right of the color picker allows you to use the operating system color window instead.

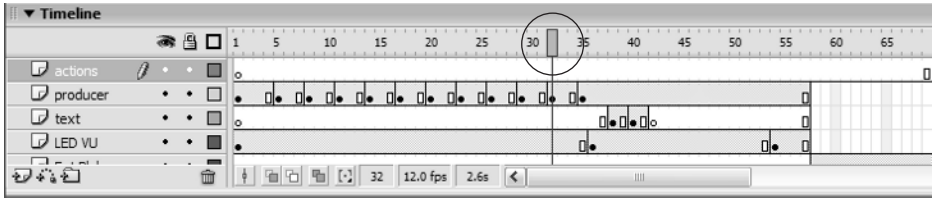
The tree is also used for other things, such as to hide detailed information until you elect to see it by opening a particular tree branch. You do this in the Keyboard Shortcuts window (Edit > Keyboard Shortcuts) and when viewing certain types of object in the debugger (such as arrays).

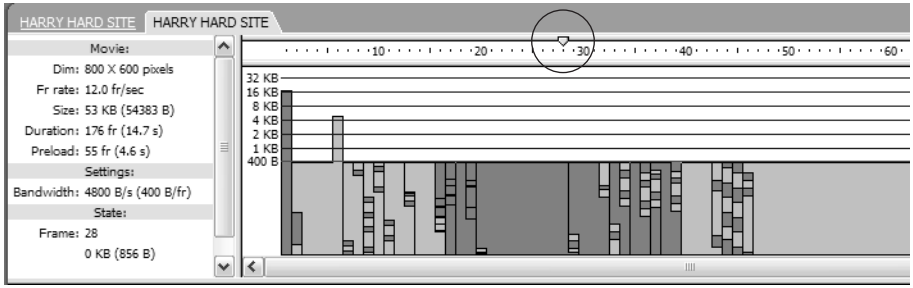
You cannot select a gradient with the color picker from the Stage (although you can from the color picker swatch); it will always return the value of the pixel currently under the eyedropper tip when you are using color picker's eyedropper on the Stage. To select a gradient from the Stage as your color, you must use the Eyedropper tool on the Tools panel.

Continues

1.10 Interface Objects *(continued)*

Unlike Adobe Photoshop, Flash doesn't allow you to zoom in or out with the color picker open to select the target color pixel accurately. In Flash, you must make sure you have zoomed into the target pixel (if you will be using the eyedropper on a Stage pixel) *before* you open the color picker.





Playheads are used only on multimedia applications, including video editing and animation applications, and they always point to the current frame being played or edited. The playhead appears in two forms, depending whether you are in the authoring (top) or test environment (bottom). In both cases, the playhead runs along a numbered track as the movie progresses (the numbers being the frame numbers). To move to a specific frame, click the number track. You can also drag the playhead by click-dragging it to a new frame.

Controller You can use the video controls provided by the Controller toolbar (make it appear via Window **>** Toolbars Controller if you don't see it). This allows you to make more complex changes to the playhead position.

MOVING BETWEEN THE AUTHORING AND EDITING ENVIRONMENT

You do not have to close the test environment to get back to the authoring environment because both exist in separate windows. You can instead minimize or reduce the size of the test environment window and leave it running. Many professional designers use a dual display and have the two separate windows on separate screens, which is especially useful when debugging.